L_1-Statistical Procedures and Related Topics
IMS Lecture Notes – Monograph Series (1997) Volume 31

ℓ_1 computation: An interior monologue

Roger Koenker

Department of Economics, University of Illinois, Champaign, USA

Abstract: Some recent developments on the computation of least absolute error estimators are surveyed and a number of extensions to related problems are suggested. A very elementary example is used to illustrate the basic approach of "interior point" algorithms for solving linear programs. And a simple preprocessing approach for ℓ_1 type problems is described. These developments, taken together, have the effect of dramatically improving the efficiency of absolute error computations, making them comparable to least squares methods even in massive datasets.

Key words: Linear models, regression quantiles, ℓ_1 estimation, computation.

AMS subject classification: 62G05, 62J05, 68A20.

1 Why square errors?

Gauss (1823), in what can only be admired as an epitome of "proof by intimidation", defended his decision to minimize sums of *squared* errors in the following terms:

> It is by no means self evident how much loss should be assigned to a given observation error. On the contrary, the matter depends in some part on our own judgment. Clearly we cannot set the loss equal to the error itself; for if positive errors were taken as losses, negative errors would have to represent gains. The size of the loss is better represented by a function that is naturally positive. Since the number of such functions is infinite, it would seem that we should choose the simplest function having this property. That function is unarguably the square, and the principle proposed above results from its adoption.

> Laplace has also considered the problem in a similar manner, but he adopted the absolute value of the error as his measure of loss. Now if I am not mistaken, this convention is no less arbitrary than mine. Should an error of double size be considered as tolerable as a single error twice repeated, or worse? Is it better to assign only twice as much influence to a double error or more? The answers are not self-evident, and the problem cannot be resolved by mathematical proofs, but only by an arbitrary decision. Moreover, it cannot be denied that Laplace's convention violates continuity and hence resists analytic treatment, while the results that my contention leads to are distinguished by their wonderful simplicity and generality.

Despite the best efforts of such distinguished advocates as Laplace (1789), Edgeworth (1888), and Kolmogorov (1931), methods of estimation based on minimizing sums of absolute errors have languished in the shade of the edifice that Gauss built on the foundation of least squares. Why? There seem to be at least two elementary reasons. First, the "wonderful simplicity and generality" of squared error has produced an elegant statistical theory of the behavior of least squares estimators which, particularly in its finite-sample form for Gaussian cases, can only inspire awe and envy on the part of advocates of the quantile-esque methods of absolute errors. Some solace may be found in the very critical attack on least-squares based methods by the robustness movement launched by John Tukey in the 1940's. The second, and perhaps even more damaging, is the perception that absolute error estimators are "difficult to compute." To appreciate that this perception was perfectly valid at the end of the 19th century, one need only read a little of Edgėworth's (1888) own arcane description of his geometric "algorithm" to compute the bivariate least absolute error regression estimator.

With the advent of George Dantzig's simplex algorithm in the late 1940's this situation changed dramatically, and by the mid-50's there were several formulations of the ℓ_1 estimator for regression as a linear program and explicit simplex-based programs to compute it. The paper of Wagner (1959) clarified the important role of the ℓ_1 dual problem. These efforts culminated in the algorithm of Barrodale and Roberts (1974) which still serves as the ℓ_1 algorithm of choice for most statistical computing environments. Contrary to a plethora of dire warnings throughout the literature, about the difficulty of ℓ_1 computation this algorithm actually delivers least absolute error regression estimates *faster* than the corresponding least squares algorithms in many packages, including Splus and Stata, for problems of moderate size, up to a few hundred observations. However, for larger problems the Barrodale and Roberts algorithm exhibits $\mathcal{O}(n^2)$ growth in execution time, and thus quickly lives up to its slothful reputation. Portnoy (1991) provides a detailed probabilistic complexity analysis for the simplex version of the parametric quantile regression problem which sheds some light on the theoretical rationale for the observed $\mathcal{O}_p(n^2)$ behavior of the simplex approach. In Portnoy and Koenker (1997), we have shown that combining recent developments on interior point methods for solving linear programs with careful preprocessing can improve both the theoretical and practical performance of ℓ_1 regression computations to the point that they are competitive with least squares over the entire range of contemporary problem dimensions.

In this paper I will briefly review these recent developments in ℓ_1 computation and then sketch some ideas for extending these developments into a broader range of related problems in statistics.

2 Means vs. medians

The most elementary instance of our basic problem may be posed as the simple question: Which is easier to compute, the median or the mean? Surprisingly, the question is deceptively difficult. At the most naive level, it would be immediately recognized that the median has an advantage for computation "by hand", an attribute implicit in the "median-polish" algorithms suggested by Tukey for robust ANOVA. Somewhat less naively, with modern computers in mind we might contemplate computing the mean in $\mathcal{O}(n)$ elementary operations (n additions, and one division), while the median appears to require sorting n numbers, a task which requires $\mathcal{O}(n \log n)$ comparisons. Further reflection suggests, however, that the median may not actually require a full sorting of the observations; a cleverly chosen partial sorting may suffice. Considerable further reflection yields the celebrated algorithm of Floyd and Rivest (1975), which manages to compute the median in $\mathcal{O}(n)$ comparisons. At this point we require a more delicate comparison of the relative effort of additions and comparisons and the precise constants associated with the $\mathcal{O}(n)$ median algorithm. Since such delicacy seems inherently machine dependent to some degree, we will not attempt to pursue it further here, but will simply note that it is not implausible that a *sophisticated* algorithm for the median could, for n sufficiently large, outperform the computation of the mean, thus restoring the superiority of the median.

3 Simplex for median regression

Portnoy and Koenker (1997) reconsider the problem of solving the ℓ_1 regression problem,

$$\min_{b \in \Re^p} \sum_{i=1}^{n} |y_i - x_i' b| \tag{1}$$

which may be formulated as the linear program,

$$\min\{ \; e'u + e'v \;\mid\; y = Xb + u - v, (u, v) \in \Re_{+}^{2n}\}. \tag{2}$$

This problem has the dual formulation

$$\max\{y'd \;\mid\; X'd = 0, \quad d \in [-1, 1]^n\}, \tag{3}$$

or equivalently, setting $a = d + \frac{1}{2}e$,

$$\max\{y'a \;\mid\; X'a = \tfrac{1}{2}X'e, \;\; a \in [0, 1]^n\}. \tag{4}$$

The simplex approach to solving this problem may be briefly described as follows. A p-element subset of $\mathcal{N} = \{1, 2, ..., n\}$ will be denoted by h, and $X(h), y(h)$ will denote the submatrix and subvector of X, y with the corresponding rows and elements identified by h. Recognizing that solutions to (1) may be characterized as planes which pass through precisely $p = \dim(b)$ observations, or as convex combinations of such "basic" solutions, we can begin with any such solution, which we may write as,

$$b(h) = X(h)^{-1} y(h). \tag{5}$$

We may regard any such "basic" primal solution as an extreme point of the polyhedral, convex constraint set. A natural algorithmic strategy is then to move to the adjacent vertex of the constraint set in the direction of steepest descent. This transition involves two stages: the first chooses a descent direction by considering the removal of each of the current basic observations and computing the gradient in the resulting direction, then having selected the direction of steepest descent and thus an observation to be removed from the currently active "basic" set, we must find the maximal step length in the chosen direction by searching over the remaining $n - p$ available observations for a new element to introduce into the "basic" set. Each of these transitions involves an elementary "simplex pivot" matrix operation to update the current basis. The iteration continues in this manner until no direction is found at which point the current $b(h)$ can be declared optimal.

Sheynin (1973) has noted that Gauss was already aware in 1809 that minimizing absolute errors, as suggested by Boscovich and Laplace, entailed this "zero residual" property. It is therefore tempting to speculate on why it required another 150 years to develop the "wonderfully simple" idea of moving from vertex to vertex in the direction of steepest descent. One possible explanation for this involves the distinction made by Gill, Murray and Wright (1991) between iterative and direct algorithms. As they put it,

> ...we consider as *direct* a computation procedure that produces one and only one estimate of the solution, without needing to perform *a posteriori* tests to verify that the solution has been found...In contrast, an iterative method generates a sequence of trial estimates of the solution, called *iterates*. An iterative method includes several elements: an initial estimate of the solution; computable tests to verify whether or not an alleged solution is correct; and a procedure for generating the next iterate in the sequence if the current estimate fails the test.

Thus, the iterative nature of the simplex algorithm makes it rather like a voyage of exploration of the 15th century, sailing into the Atlantic believing that the world was flat, not knowing when, or even if, the voyage would end. Gaussian elimination, on the other hand, made least squares like a trip along a familiar road; at each step one knew exactly how much further effort was necessary. With the emergence of computers in the 1940's, the risk, or uncertainty, of the iterative approach was transferable to the machine, and the spirit of adventure blossomed as investigators put down their pencils and watched the tapes whir and the lights flicker.

Like the advantage of the median over the mean for hand computations, the simplex algorithm for median regression performs extremely well relative to least squares in problems of modest size. In Figure 1 we can compare performance of the Barrodale and Roberts (1974) algorithm for median regression with the standard least squares algorithm as implemented in the function `lm(y ~ x)` in Splus. For $p = 4$ and $n < 2000$, median regression *à la* Barrodale and Roberts is actually faster than the corresponding least squares computation. As the dimension of the parameter increases, the advantage of ℓ_1 over L_2 is somewhat attentuated, but even with $p = 16$, there is still an advantage up to sample sizes of about 400.

In larger problems simplex-based computations for median regression pale in comparison with speeds achievable by least squares. In Figure 2 I illustrate this comparison over problems in the range $20,000 \leq n \leq 120,000$, and we see that the time required for the modified simplex approach embodied in the Barrodale and Roberts algorithm tends to grow quadratically in n while the QR factorization approach of `lm` grows only linearly in n. By sample size $n = 120,000$ this results in computations of nearly one hour for median regression a procedure which can be carried out in 10-20 seconds by least squares. Is this differential inherent in the linear programming formulation of the ℓ_1 problem, confirming Gauss's claims, or is it simplex that is at fault?

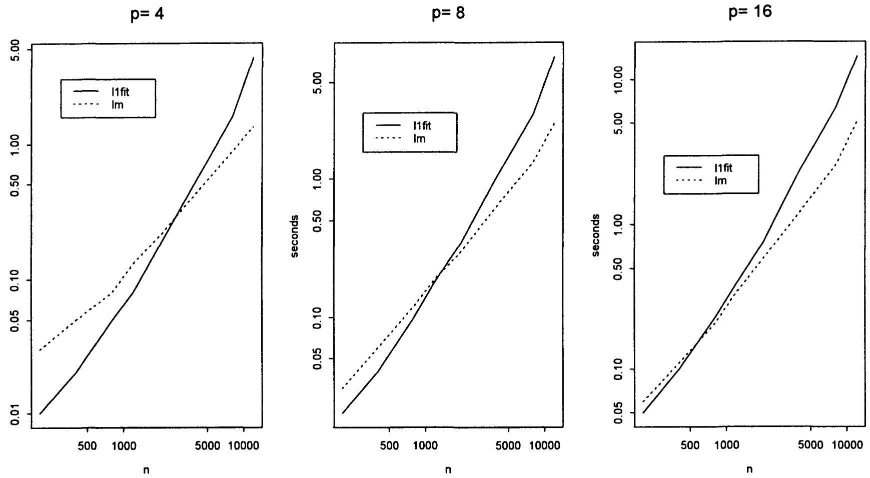

Figure 1: Timing comparison of ℓ_1 and ℓ_2 algorithms: Times are in seconds for the median of five replications for iid Gaussian data. The parametric dimension of the models is $p + 1$ with p indicated above each plot, p columns are generated randomly and an intercept parameter is appended to the resulting design. Timings were made at 8 design points in n: 200, 400, 800, 1200, 2000, 4000, 8000, 12000. The solid line represents the results for the simplex-based Barrodale and Roberts algorithm, `l1fit(x,y)` in Splus, and the dotted line represents least squares timings based on `lm(y ~ x)`.

Ironically, one of the great research challenges of numerical analysis of recent decades has been, "Why is simplex so quick?" Examples, notably that of Klee and Minty (1972), have shown that in problems of dimension, n, simplex can take as many as 2^n simplex pivots, each requiring $\mathcal{O}(n)$ effort. From this perspective $\mathcal{O}_p(n^2)$ effort for randomly generated ℓ_1 problems appears to be quite brilliant. The recent paper of Shamir (1993) surveys the extensive literature exploring this gap between theoretical worst-case behavior and practical average-case performance. The discussion of simplex in Gill, Murray and Wright (1991) is especially good on this aspect.

4 Newton to the max: An elementary example

To illustrate the shortcomings of the simplex method, or indeed of any strategy for solving linear programs which relies on an iterative path along the *exterior* of the constraint set, consider the problem depicted in Figure 3. We have a random polygon whose vertices lie on the unit circle and our objective is to find a point in the polygon that maximizes the sum of its coordinates, that is, the point furthest north-east in the figure.

Since any point in the polygon can be represented as a convex weighting of the extreme points, the problem may be formulated as

$$\max\{e'u | X'd = u, \;\; e'd = 1, \;\; d \in \Re^n_+\}, \tag{6}$$

where e denotes a (conformable) vector of ones, X is an $n \times 2$ matrix with rows representing the n vertices of the polygon and d is the vector of convex weights to be determined. Eliminating u we may rewrite (6) somewhat more simply as

$$\max\{s'd | e'd = 1, \;\; d \in \Re^n_+\}, \tag{7}$$

where $s = Xe$. This is an extremely simple linear program which serves as a convenient geometric laboratory animal for studying various approaches to solving such problems. Simplex is particularly simple in this context, because the constraint set *is* literally a simplex. If we begin at a random vertex, and move around the polygon until optimality is achieved, we pass through $O(n)$ vertices in the process. Of course, a random initial vertex is rather naive, and one could do much better with an intelligent "Phase 1" approach that found a *good* initial vertex. In effect we can think of the "interior point" approach we will now describe as a class of methods to accomplish this, rendering unnecessary further travel around the outside of the polygon.

Although prior work in the Soviet literature offered theoretical support for the idea that linear programs could be solved in polynomial time, thus avoiding the pathological exponential growth of the Klee-Minty examples, the paper of Karmarker (1984) constituted a watershed in the numerical analysis of linear programming. It offered not only a cogent argument for the polynomiality of interior point methods of solving LP's, but also provided for the first time direct evidence that interior point methods were demonstrably faster than simplex in specific, large, practical problems.

To explore several variants of interior point methods we will use our simple polygonal problem. Further details about more general LP's and applications to ℓ_1 regression, and quantile regression more generally, may be found in Portnoy and Koenker (1997). The basic approach we will describe to interior point methods for linear programming is set out in the important survey paper by Lustig, Marsden and Shanno (1994). A more detailed exposition may be found in the new monograph of Wright (1996)

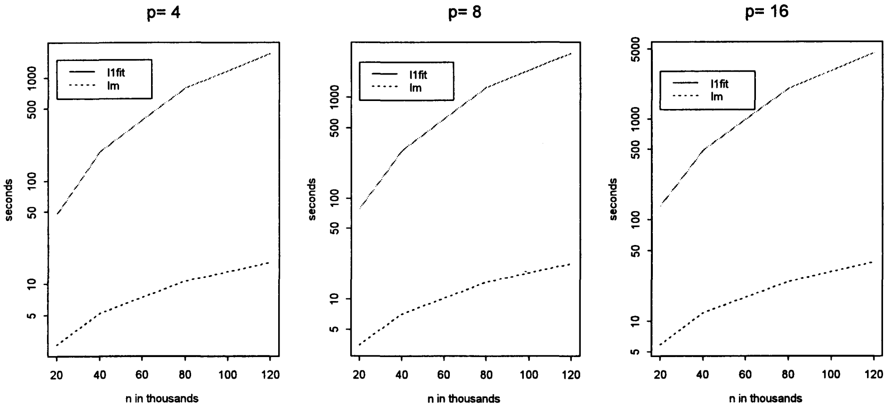

Figure 2: Timing comparison of ℓ_1 and ℓ_2 algorithms: Times are in seconds for the median of five replications for iid Gaussian data. The parametric dimension of the models is $p+1$ with p indicated above each plot, p columns are generated randomly and an intercept parameter is appended to the resulting design. Timings were made at 4 design points in n: 20,000, 40,000, 80,000, 120,000. The solid line represents the results for the simplex-based Barrodale and Roberts algorithm, `l1fit(x,y)` in Splus, and the dotted line represents least squares timings based on `lm(y` $\sim$ `x)`.

It is an amusing irony, illustrating the spasmodic progress of science, that the most fruitful practical formulation of the interior point revolution of Karmarker (1984) can be traced back to a series of Oslo working papers by Ragnar Frisch in the early 1950's. This work is summarized in Frisch (1956), and was considerably elaborated and extended in the monograph of Fiacco and McCormick (1968). The connection between Karmarker's approach and the earlier literature was developed in Gill, Murray, Saunders, Tomlin and Wright (1986). The basic idea of Frisch was to replace the linear inequality constraints of the *LP*, by what may be called a log barrier, or potential, function. Thus, in our example, we may reformulate (7) as,

$$\max\{s'd + \mu \sum_{i=1}^{n} \log d_i | e'd = 1\} \tag{8}$$

where now the barrier term $\mu \sum \log d_i$ serves as a penalty which keeps us away from the boundary of the positive orthant. By judicious choice of a sequence $\mu \longrightarrow 0$ we might hope to converge to a solution of the original problem.

The salient virtue of the log barrier formulation is that, unlike the original formulation, it yields a differentiable objective function which is consequently attackable by Newton's method. Restricting attention, for the moment, to the primal log-barrier formulation 8 and defining,

$$B(d, u) = s'd + \mu \sum \log d_i \tag{9}$$

we have $\nabla B = s + \mu D^{-1}e$ and $\nabla^2 B = -\mu D^{-2}$ where $D = \mathrm{diag}(d)$. Thus, at any initial feasible, d, we have the associated Newton subproblem

$$\max_p \{(s + \mu D^{-1}e)'p - \frac{\mu}{2} p' D^{-2} p | e'p = 0\}.$$

This problem has first order conditions

$$\begin{aligned} s + \mu D^{-1}e - \mu D^{-2}p &= ae \\ e'p &= 0 \end{aligned}$$

and multiplying through by $e'D^2$, and using the constraint, we have,

$$e'D^2 s + \mu e' D e = a e' D^2 e.$$

Thus solving for the Lagrange multiplier $\hat{a}$ we obtain the Newton direction

$$p = \mu^{-1} D^2 s + De - \hat{a}e \tag{10}$$

where $\hat{a} = (e'D^2e)^{-1}(e'D^2s + \mu e'De)$. Pursuing the iteration $d \leftarrow d + \lambda p$, thus defined, *with μ fixed* until convergence, yields the central path $d(\mu)$ which describes a yellow brick road to the solution d^* of the original problem (6). We must be careful to keep the step lengths λ small enough to maintain the interior feasibility of d. Note that the initial feasible point $d = e/n$ represents $d(\infty)$.

As emphasized by Gonzaga (1992) and others, this central path is a crucial construct for the interior point approach. Algorithms may be usefully evaluated on the basis of how well they are able to follow this path. Clearly, there is some tradeoff between staying close to the path and moving along the path, thus trying to reduce μ, iteration by iteration. Improving upon existing techniques for balancing these objectives is the subject of a vast outpouring of current research. Excellent introductions to the subject are provided in the survey paper of Margaret Wright (1992) and the recent monograph of Stephen Wright (1996).

Thus far, we have considered only the primal version of our simple polygonal problem, but it is also advantageous to consider the primal and dual forms together. The dual of (7) is very simple:

$$\min\{a | ea - z = s, \quad z \geq 0\}. \tag{11}$$

The scalar, a, is the Lagrange multiplier on the equality constraint of the primal introduced above, while z is a vector of "residuals," or slack variables in the terminology of linear programming. This formulation of the dual exposes the real triviality of the problem – we are simply looking for the maximal element of the vector $s = Xe$. This is a very special case of the linear programming formulation of finding any ordinary quantile. But the latter would require us to split z into its positive and negative parts, and would also introduce upper bounds on the variables, d, in the primal problem.

Another way to express the central path, one that nicely illuminates the symmetric roles of the primal and dual formulations of the original problem, is to solve the equations,

$$\begin{aligned} e'd &= 1 \\ ea - z &= s \\ Dz &= \mu e. \end{aligned} \tag{12}$$

That solving these equations is equivalent to solving (8) may be immediately seen by writing the first order conditions for (8) as

$$\begin{aligned} e'd &= 1 \\ ea - \mu D^{-1}e &= s, \end{aligned}$$

and then appending the definition $z = \mu D^{-1}e$. The equivalence then follows from the negative definiteness of the Hessian $\nabla^2 B$. This formulation is also useful in highlighting a crucial interpretation of the log-barrier penalty parameter, μ. For any feasible pair (z, d) we have

$$s'd = a - z'd,$$

so $z'd$ is equal to the duality gap, the discrepancy between the primal and dual objective functions at the point (z, d). At a solution, we have the complementary slackness condition $z'd = 0$, thus implying a duality gap of zero. Multiplying through by e' in the last equation of (12) , we may take $\mu = z'd/n$ as a direct measure of progress toward a solution.

Applying Newton's method to these equations yields

$$\begin{pmatrix} Z & 0 & D \\ e' & 0 & 0 \\ 0 & e & I \end{pmatrix} \begin{pmatrix} p_d \\ p_a \\ p_z \end{pmatrix} = \begin{pmatrix} \mu e - Dz \\ 0 \\ 0 \end{pmatrix},$$

where we have again presumed initial, feasible choices of d and z. Solving for p_a we have

$$\hat{p}_a = (e'Z^{-1}De)^{-1}e'Z^{-1}(Dz - \mu e)$$

which yields the primal-dual Newton direction:

$$\begin{aligned} p_d &= Z^{-1}(\mu e - Dz - Dep_a) && (13) \\ p_z &= ep_a. && (14) \end{aligned}$$

It is of obvious interest to compare this primal-dual direction with the purely primal step derived above. In order to do so, however, we need to specify an adjustment mechanism for μ.

To this end we will now describe an approach suggested by Mehrotra (1992) that has been widely implemented by developers of interior point algorithms, including the interior point algorithm for quantile regression described in Portnoy and Koenker (1997). Given an initial feasible triple (d, a, z), consider the affine-scaling Newton direction obtained by evaluating (13) at $\mu = 0$. Now compute the step lengths for the primal and dual variables respectively using

$$\lambda_d = argmax\{\lambda \in [0,1] | d + \lambda p_d \geq 0\}$$

$$\lambda_z = argmax\{\lambda \in [0,1] | z + \lambda p_z \geq 0\}.$$

But rather than precipitously taking this step, Mehrotra suggests adapting the direction somewhat to account for both the "recentering effect" introduced by the μe term in (13) and also for the nonlinearity introduced by the last of the first order conditions.

Consider first the recentering effect. If we contemplate taking a full step in the affine scaling direction we would have,

$$\hat{\mu} = (d + \lambda_d p_d)'(z + \lambda_z p_z)/n,$$

while at the current point we have,

$$\mu = d'z/n.$$

Now, if $\hat{\mu}$ is considerably smaller than μ, it means that the affine scaling direction has brought us considerably closer to the optimality condition of complementary slackness: $z'd = 0$. This suggests that the affine scaling direction is favorable, that we should reduce μ, in effect downplaying the contribution of the recentering term in the gradient. If, on the other hand, $\hat{\mu}$ isn't much different than μ, it suggests that the affine-scaling direction is unfavorable and that we should leave μ alone, taking a step which attempts to bring us back closer to the central path. Repeated Newton steps with μ fixed put us exactly on this path. These heuristics are embodied in Mehrotra's proposal to update μ by

$$\mu \leftarrow \mu(\hat{\mu}/\mu)^3.$$

To deal with the nonlinearity, Mehrotra (1992) proposed the following "predictor-corrector" approach. A full affine scaling step would entail

$$(d + p_d)'(z + p_z) = d'z + d'p_z + p_d'z + p_d'p_z.$$

The linearization implicit in the Newton step ignores the last term, in effect predicting that since it is of $\mathcal{O}(\mu^2)$ it can be neglected. But since we have already computed a preliminary direction, we might as well reintroduce this term to correct for the nonlinearity as well to accomplish the recentering. Thus, we compute the modified direction by solving

$$\begin{pmatrix} Z & 0 & D \\ e' & 0 & 0 \\ 0 & e & I \end{pmatrix} \begin{pmatrix} \delta_d \\ \delta_a \\ \delta_z \end{pmatrix} = \begin{pmatrix} \mu e - Dz - P_d p_z \\ 0 \\ 0 \end{pmatrix},$$

where $P_d = diag(p_d)$. This modified Newton direction is then subjected to the same step-length computation and a step is finally taken. It is important in more realistic problem settings that the linear algebra required to compute the solution to the modified step has already been done for the affine scaling step. This usually entails a Cholesky factorization of a matrix which happens to be scalar here, so the modified step can be computed by simply backsolving the same system of linear equations already factored to compute the affine scaling step.

In Figure 3 we provide an example intended to illustrate the advantage of the Mehrotra modified step. The solid line indicates the central path. Starting from the same initial point $d = e/n$, the dotted line represents the first affine scaling step. It is successful in the limited sense that it stays very close to the central path, but it only takes a short step toward our final destination. In contrast, the first modified step, indicated by the dashed line, takes us much further. By anticipating the curvature of the central path, it takes a step more than twice the length of the unmodified, affine-scaling step. On the second step the initial affine-scaling step is almost on target, but again somewhat short of the mark. The modified step is more accurately pointed at the desired vertex and is thus, again, able to take a longer step.

It is difficult in a single example like this to convey a sense of the overall performance of these methods. After viewing a large number of realizations of these examples myself, I come away convinced that the Mehrotra modified step consistently improves upon the affine scaling step, a finding that is completely consistent with the theory.

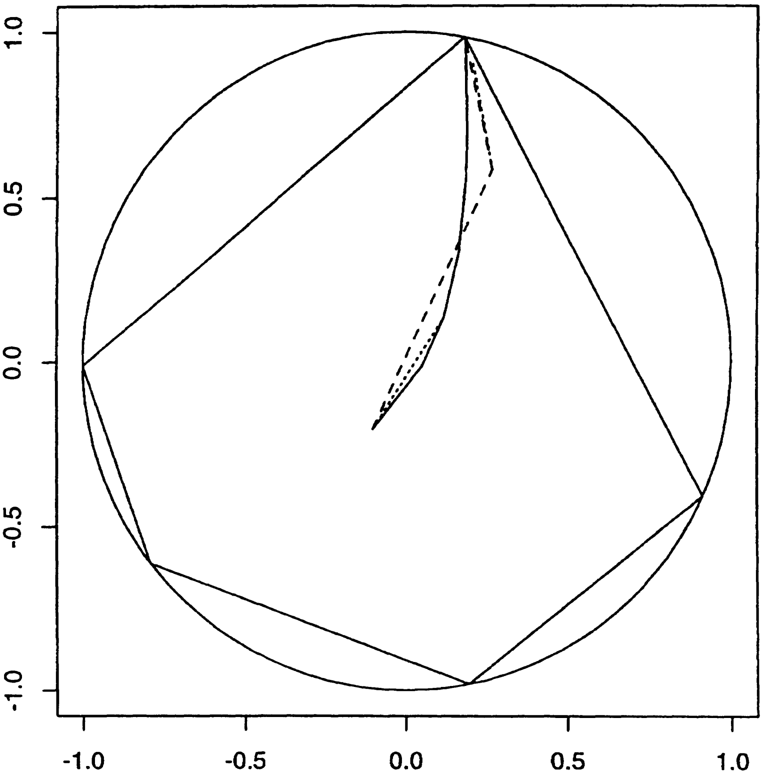

Figure 3: A simple example of interior point methods for linear programming: The figure illustrates a random pentagon of which we would like to find the most northeast vertex. The central path beginning with an equal weighting of the 5 extreme points of the polygon is shown as the solid curved line. The dotted line emanating from the this center is the first affine scaling step. The dashed line is the modified Newton direction computed according to the proposal of Mehrotra. Subsequent iterations are unfortunately obscured by the scale of the figure.

In Portnoy and Koenker (1997), we noted that recent work on the probabilistic analysis of the computational complexity of interior point methods suggests that algorithms with $\mathcal{O}_p(np^3 \log^2 n)$ operations are possible for quantile regression with n observations and p parameters. While such performance is considerably better, in large problems, than the observed $\mathcal{O}_p(n^2p^2)$ performance of simplex, it is still inferior to the $\mathcal{O}(np^2)$ complexity of least squares. In the next section I very briefly describe a preprocessing strategy for quantile regression problems that has been successful in further narrowing this computational gap.

5 Preprocessing for quantile regression

The idea of preprocessing quantile regression problems described in Portnoy and Koenker (1997) actually preceded the implementation of the interior point methods discussed above. Preprocessing rests on an extremely simple idea: if, by preliminary estimation, or some other form of statistical necromancy, we could determine the signs of a significant group of observations, we could then combine observations with positive residuals into a single "globbed" observation, and similarly glob together the negative observations, so that the original problem,

$$\min \sum_{i=1}^{n} \rho_\tau(y_i - x_i'b) \tag{15}$$

with $\rho_\tau(u) = u(\tau - I(u < 0))$ would be equivalent to,

$$\min \sum_{i \in N \setminus J_L \cup J_H}^{n} \rho_\tau(y_i - x_i'b) + \rho_\tau(y_L - x_L'b) + \rho_\tau(y_H - x_H'b) \tag{16}$$

where $N = \{1, 2, ..., n\}$, $x_K = \sum_{i \in J_K} x_i$ for $K \in \{K, L\}$ and y_L and y_H can be chosen arbitrarily small and large respectively, to ensure that the corresponding residuals on the globbed observations remain negative and positive. In this process we have reduced the problem of n original observations to $n - \sharp\{J_L, J_H\} + 2$ observations so if the cardinality of the J-sets is large we have gained substantially. Under plausible sampling assumptions we can, based on a preliminary subsample of m observations, make a prediction region for $\{x_i\beta : i = 1, 2, ..., n\}$ of width $\mathcal{O}(p/\sqrt{m})$, so assigning observations above this region to J_H and observations below this region to J_L, we would have $M = \mathcal{O}_p(np/\sqrt{m})$ observations falling inside the region. This is illustrated in Figure 4.

Minimizing the computational effort required to compute the preliminary fit based on m observations plus the effort required for the solution of the globbed problem (16) with M observations, we obtain $m^* = \mathcal{O}((np)^{2/3})$, which under our conjectured performance of the underlying interior point algorithm yields a complexity for the full problem of

$$C = \mathcal{O}_p(n^{2/3} p^3 \log^2 n) + \mathcal{O}(np^2), \tag{17}$$

where the first term comes from the solution of the two quantile regression problems and the second term arises from the computation of the confidence band.

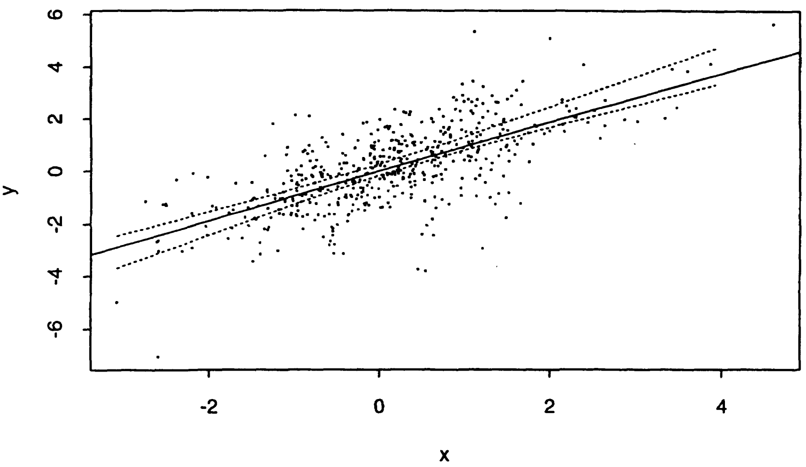

Figure 4: A bivariate example of quantile regression preprocessing: The figure illustrates a bivariate scatter plot of 500 observations with y conditionally student t on 10 degrees of freedom. The curved dotted lines describe a confidence band for the response variable based on the median regression fit for a sub-sample of 126 observations. After globbing there are only 107 observations, including the two globbed observations. All the points outside the band are collapsed into this pair of pseudo-observations. The fit to the globbed sample is indicated by the solid line; since it falls inside the band we are assured that the globs are correct and that this solution is identical to a fit of the entire original sample.

Further details are provided in Portnoy and Koenker (1997) and I will comment only briefly here on the important fact that any implementation of this preprocessing approach must verify that the solution to the globbed problem actually vindicates the predicted signs based on the confidence region. Since the simultaneous confidence region can be chosen to assure this with arbitrarily high probability, the eventuality that we may need to repeat the cycle to remedy some inaccurately predicted signs introduces another multiplicative factor which does not affect the orders in probability in the complexity computation.

The crucial consequence of the formal complexity theory and the extensive concomitant empirical testing of our implementation of the algorithm is that the computational effort required for quantile regression can be made comparable with the effort required for least squares over the full range of currently plausible problem dimensions. In the final empirical example of Portnoy and Koenker (1997), we compare timings for a typical large econometric application of quantile regression with $n = 113,547$ and $p = 6$. With the new algorithm, quantile regression estimates take about 10 seconds on a Sparc-Ultra, comparable to the least squares time of 8 seconds. Simplex solution of the same quantile regression problems requires approximately an hour on the same machine.

6 Prospects

There are many open questions posed by the rapid development of computational methods for quantile regression. I would like to touch on three topics in this brief final section. The first is applications to inference and the general problem of parametric programming viewed in the light of interior point methods. The second is applications to nonlinear quantile regression. And the third concerns nonparametric applications of quantile regression.

As I have tried to emphasize elsewhere, an important virtue of the simplex approach to ℓ_1-type computation is the direct transition to parametric programming, or sensitivity analysis. Having obtained a solution at one quantile we immediately compute an interval of optimality for this solution, at the endpoints the solution alters. We may then make a simplex pivot which takes us to an adjacent vertex of the constraint set; continuing this process traces out the entire path of solutions to the problem (15) for $\tau \in [0,1]$. Efficient computation of the quantile regression *process* is crucial for the smooth L-statistics described in Koenker and Portnoy (1990), and the corresponding dual process is central to the elegant theory of rank statistics introduced by Gutenbrunner and Jureckova (1992). Very similar computations are required to compute the penalized quantile regression spline estimators introduced in Koenker, Ng and Portnoy (1995) where the degree of smoothing (bandwidth) parameter λ plays the role of τ.

The homotopy methods of interior point algorithms also lend themselves naturally to parametric programming. In large problem it may be sufficient to compute solutions on some grid in τ or λ and we may thus avoid passing through all the intermediate vertices by tunnelling through the interior of the constraint set, passing directly from one grid point to the next. Algorithms to do this are conceptually straightforward, given the existing research, see for example Monteiro and Mehrotra (1995), but they require some careful engineering.

Non-linear quantile regression, that is quantile regression estimation like (15) with a nonlinear response function replacing the linear predictor $x_i'\beta$, are increasingly common in applications. Here too, interior point methods and the preprocessing approaches described above can play a useful role. Some ideas along this line have been already described in Koenker and Park (1996). There is, however, considerable scope for refinement.

Finally, in nonparametric applications of quantile regression there are a wide array of competing methods, all of which can profit from more efficient computational methods for large data sets. This is particularly true of the quantile smoothing spline approach of Koenker, Ng and Portnoy (1995), which offers new challenges in terms of exploiting sparsity in the interior point matrix computations. This is a topic which has received intense scrutiny in the interior point literature, and there are a number of very promising approaches already available.

We are, I believe, on the verge of overthrowing the long-standing computational disability of ℓ_1 methods. In the next century, we may hope that the young statistician looking for improved robustness, or simply for a more complete view of her data, may say of quantile regression, echoing Molly Bloom, "...yes I said yes I will Yes."

References

[1] Barrodale, I. and F.D.K. Roberts (1974). Solution of an overdetermined system of equations in the ℓ_1 norm. *Commun. ACM* **17**, 319-320.

[2] Edgeworth, F.Y. (1888). On a new method of reducing observations relating to several quantities. *Philosophical Magazine* **25**, 184-191.

[3] Fiacco, A.V. and G.P. McCormick (1968). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* New York: Wiley.

[4] Floyd, R.W. and R.L. Rivest (1975). Expected Time Bounds for Selection. *Commun. ACM* **18**, 165-173.

[5] Frisch, R. (1956). La Résolution des problèmes de programme linéaire par la méthode du potential logarithmique. *Cahiers du Séminaire d'Econometrie*, **4**, 7-20.

[6] Gauss, C.F. (1823). *Theoria Combinationis Observationum Erroribus Minimis Obnoxiae: Pars Prior.* Translated by G.W. Stewart (1995) as *Theory of the Combination of Observations Least Subject to Error.* Philadelphia: SIAM.

[7] Gill, P., W. Murray, and M. Wright (1991). *Numerical Linear Algebra and Optimization.* Redwood City: Addison-Wesley.

[8] Gill, P., W. Murray, M. Saunders, T. Tomlin, and M. Wright (1986). On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method. *Mathematical Programming* **36**, 183-209.

[9] Gonzaga, C.C. (1992). Path-following methods for linear programming. *SIAM Rev.* **34**, 167-224.

[10] Gutenbrunner, C. and J. Jureckova (1992). Regression quantile and regression rank score process in the linear model and derived statistics. *Ann. Statist.* **20**, 305-330.

[11] Karmarkar, N. (1984). A new polynomial time algorithm for linear programming. *Combinatorica* **4**, 373-395.

[12] Klee, V. and G.J. Minty. How good is the simplex algorithm? In *Inequalities*, Ed. O. Shisha. New York: Adademic Press.

[13] Koenker, R. and G. Bassett (1978). Regression quantiles, *Econometrica*, 46, 33-50.

[14] Koenker, R. and V. d'Orey (1987,1993). Computing regression quantiles, *Applied Statistics*, **36**, 383-393, and **43**, 410-414.

[15] Koenker R., P. Ng and S. Portnoy (1995). Quantile Smoothing Splines. *Biometrika* **81**, 673-80.

[16] Koenker R., and S. Portnoy (1990). L-Estimation for the linear model. *J. Am. Statist. Assoc.* **82**, 851-857.

[17] Koenker R. and B.J. Park (1996). An interior point algorithm for nonlinear quantile regression. *Journal of Econometrics* **71**, 265-283.

[18] Kolmogorov, A.S. (1931). The method of the median in the theory of errors. *Mat. Sbornik* **38**, 47-50. Reprinted in english in *Selected Works of A. N. Komogorov*, Ed. A.N. Shiryayev. Dordrecht: Kluwer.

[19] Laplace, P.-S. (1789). Sur quelques points du système du monde. Mémoires de l'Académie des Sciences de Paris. Reprinted in *Œuvres Complètes*, Vol. 11. Paris: Gauthier-Villars.

[20] Lustig, I.J., R.E. Marsden and D.F. Shanno (1992). On umplementing Mehrotra's predictor-corrector interior-point method for linear programming. *SIAM J. Optim.* **2**, 435-449.

[21] Lustig, I.J., R.E. Marsden and D.F. Shanno (1994). Interior point methods for linear programming: computational state of the art, with discussion. *ORSA Journal on Computing* **6**, 1-36.

[22] Monteiro, R.D.C., and S. Mehrotra, (1995) A general parametric analysis approach and its implications to sensitivity analysis in interior point methods. Preprint.

[23] Mehrotra, S. (1992). On the implementation of a primal-dual interior point method. *SIAM J. Optim.* **2**, 575-601.

[24] Portnoy, S. (1991). Asymptotic behavior of the number of regression quantile breakpoints. *SIAM J. Scient. Statist. Comput.* **12**, 867-883.

[25] Portnoy, S. and Koenker, R. (1997). The Gaussian Hare and the Laplacian Tortoise: Computation of Squared-error vs. Absolute-error Estimators. *Statistical Science.* Forthcoming.

[26] Shamir, R. (1993). Probabilistic analysis in linear programming. *Statistical Science* **8**, 57-64.

[27] Wagner, H.M., (1959). Linear programming techniques for regression analysis. *J. Am. Statist. Assoc.* **54**, 206-212.

[28] Wright, M.H. (1992). Interior methods for constrained optimization. *Acta Numerica*, 341-407.

[29] Wright, S.J. (1996). *Primal-Dual Interior-Point Methods.* Philadelphia: SIAM.