A relaxed approach to combinatorial problems in robustness and diagnostics

F. CRITCHLEY, M. SCHYNS*, G. HAESBROECK*, C. FAUCONNIER*,
G. LU, R.A. ATKINSON and D.Q. WANG
*The Open University, University of Liège**,
University of Bristol, The University of Birmingham
and Victoria University of Wellington

Abstract: A range of procedures in both robustness and diagnostics require optimisation of a target function over all subsamples of given size. Whereas such combinatorial problems are extremely difficult to solve exactly, the global optimum is *not* required for many purposes, a "good enough" solution being good enough to fully achieve statistical objectives. Again, a relaxation strategy can be used to embed these discrete, high-dimensional problems in continuous, low-dimensional ones. Overall, nonlinear optimisation methods can be exploited to provide a general, fast algorithm to handle a wide variety of problems of this kind. Four running examples illustrate the approach. On the robustness side, minimum covariance determinant (MCD) and least trimmed squares (LTS) estimation. And, on the diagnostic side, detection of multiple multivariate outliers and global diagnostic use of the likelihood displacement function. This last is developed here as a complement to Cook's (1986) local analysis. Appropriate convergence of each branch of the algorithm is guaranteed for any target function whose relaxed form is "gravitational", such functions being introduced here as natural generalisations of (increasing transformations of) concave functions.

Keywords: combinatorial optimisation, concave functions, diagnostics, gravitational functions, nonlinear optimisation, robustness.

1 Introduction

Many optimisation problems arising naturally in statistics are combinatorial by definition and correspondingly extremely difficult to solve exactly. We focus on two such problem classes – one arising in robustness, the other in diagnostics – whose evident equivalence provides a certain unity between these two areas of statistical methodology. A key feature here is that finding the global optimum is *not* essential, a "good enough" solution being good enough to fully achieve statistical objectives.

In robustness, a class of estimators are – or can be – defined in terms of optimisation of a specified target functional over all subsamples of given size, lead examples including minimum covariance determinant (MCD) and least trimmed squares (LTS) estimation. Equally, a general problem arising in diagnostics is to identify subsamples of given size whose deletion maximally changes a statistic of interest as measured by an appropriate target function, lead examples being detection of multiple multivariate outliers and global diagnostic use of the likelihood displacement function, this last being developed here as a complement to Cook's (1986) local analysis. These problem classes are reviewed in Section 2, while Section 3 describes the lead examples used.

Section 4 reviews the relaxation strategy proposed in Critchley et al. (2004) as a means of embedding such discrete, high-dimensional optimisation problems in continuous, low-dimensional ones. Focusing without loss on minimisation problems, this strategy succeeds in smoothly reformulating any problem whose relaxed target function is (an increasing function of) a concave function. We go on to show that it also succeeds for the wider class of "gravitational" functions, introduced here. This gives sufficient generality to cover many statistical problems, as illustrated by our running examples. The relaxed target functions for each example are given in Section 5.

Section 6 describes our implementation of this relaxation strategy in which constrained nonlinear optimisation methods are exploited to provide a general, fast procedure to handle a wide variety of problems of this kind. The corresponding algorithm is illustrated and tested on the running examples in Section 7.

The paper finishes with a short discussion.

2 Two combinatorial problem classes

Throughout, $\{z_i \in \mathbb{R}^d : i \in N\}$ with $N = \{1, ..., n\}$ denotes a random sample of $n > 1$ distinct cases from an unknown distribution F. In multivariate contexts where all the variates are on the same footing, we put $d = k$ and $z_i = x_i$. In the usual notation for (generalised) linear models, we put $d = k + 1$ and $z_i^T = (x_i^T, y_i)$.

Throughout, H and M denote complementary subsets of N containing respectively $h > 0$ and $m > 0$ indices, so that $h + m = n$. *Holding onto* the cases labelled by H is exactly the same thing as *missing out* those labelled by M. Accordingly, we use $\widehat{F}_H = \widehat{F}_{-M}$ to denote the empirical distribution function assigning equal weight $h^{-1} = (n - m)^{-1}$ to the cases whose index is in H – equivalently, *not* in M.

It is convenient to have a notation for the collection of $\binom{n}{a} = \binom{n}{n-a}$ subsets of N containing exactly $0 < a < n$ indices. Putting $\mathbb{N}_a = \{\varnothing \subset A \subset N : |A| = a\}$, we follow Critchley et al. (2004) and focus on the following – entirely equivalent – combinatorial problem classes for a given scalar function $t[\cdot]$:

Problem 1. *(Combinatorial optimisation problem) ($\mathcal{D}$) Optimise $t[\widehat{F}_{-M}]$ over $M \in \mathbb{N}_m$ ($\mathcal{R}$) Optimise $t[\widehat{F}_H]$ over $H \in \mathbb{N}_h$*

The ($\mathcal{D}$) form is natural in diagnostics, the ($\mathcal{R}$) form being equally natural in robustness. The next section outlines some well-known examples, used for illustration throughout.

3 Examples: instances of Problem 1

3.1 In robustness

1. Minimum Covariance Determinant (MCD) estimator:

   The Minimum Covariance Determinant (MCD) estimator (Rousseeuw, 1985) consists of determining a subsample H, of given size h, from a multivariate sample with minimal generalised variance. It being convenient to work with the logarithm of this quantity, the MCD version of Problem 1 minimises $t = t_{MCD}$ given by:

$$t_{MCD}[\widehat{F}_H] = \log \det(\mathrm{cov}[\widehat{F}_H]). \tag{3.1}$$

2. Least Trimmed Squares (LTS) estimator:

Consider the linear model $y_i = x_i^T\beta + \varepsilon_i \quad (i \in N)$ where the $\{\varepsilon_i\}$ are independently distributed as $N(0, \sigma^2)$ and $\beta \in \mathbb{R}^k$. For given h (greater than k, to avoid exact fits), the Least Trimmed Squares estimator for β (Rousseeuw and Leroy, 1987) is defined by

$$\widehat{\beta}_{LTS} = \underset{\beta}{\operatorname{argmin}}\, S(\beta)$$

where $S(\beta) = \sum_{i=1}^{h} r^2_{(i)}(\beta)$ in which $r^2_{(1)}(\beta) \leq r^2_{(2)}(\beta) \leq \ldots \leq r^2_{(n)}(\beta)$ are the *ordered* squares of the β-residuals $\{r_i(\beta)\}_{i\in N}$, given by $r_i(\beta) = y_i - x_i^T\beta$. This definition being in terms of minimising a function of β, finding a corresponding function of $\widehat{F}_H$ to optimise is not immediate. We proceed as follows.

First, for any $\beta \in \mathbb{R}^k$, let $\{H(\beta), M(\beta)\}$ be a partition of N with $|H(\beta)| = h$ and $|M(\beta)| = m$ having the property that $\forall\, i \in H(\beta),\ \forall\, j \in M(\beta),\ r_i^2(\beta) \leq r_j^2(\beta)$. Such a partition is unique (w.p.1), while $S(\beta)$ may now be re-written as $\sum_{i\in H(\beta)} r_i^2(\beta)$. Next, let $\beta[\widehat{F}_H] = \operatorname{argmin}_{\beta\in\mathbb{R}^k} \sum_{i\in H} r_i^2(\beta)$ denote the least squares estimate for the subsample indexed by H. We abbreviate this as $\widehat{\beta}_H$, denoting by

$$R[H] = \sum\nolimits_{i\in H} r_i^2(\widehat{\beta}_H)$$

the corresponding residual sum of squares. Finally, let $\widehat{H} = \operatorname{argmin}_{H\in\mathbb{N}_h} R[H]$ be a subsample achieving the smallest residual sum of squares over all subsamples of h cases, $\widehat{H}$ being unique (w.p.1). Then, for any $\beta \in \mathbb{R}^k$, $S(\beta) \geq R[H(\beta)] \geq R[\widehat{H}]$. But, by minimality, $\widehat{H} = H(\widehat{\beta}_{\widehat{H}})$, implying $R[\widehat{H}] = S(\widehat{\beta}_{\widehat{H}})$. Thus, $S(\widehat{\beta}_{LTS}) = S(\widehat{\beta}_{\widehat{H}})$ so that, the minimum of $S(\cdot)$ being assumed unique, $\widehat{\beta}_{LTS} = \widehat{\beta}_{\widehat{H}}$ can be found via minimisation of $t = t_{LTS}$ given by:

$$t_{LTS}[\widehat{F}_H] = R[H]. \tag{3.2}$$

3.2 In diagnostics

1. Detection of multiple multivariate outliers:

Atkinson (1986) introduced a general two-stage strategy to overcome masking problems in a variety of multiple outlier detection contexts, Critchley et al. (2001) confirming that this strategy is both effective and fast in the linear model. Here, we consider its use in the important problem of detecting multiple outliers in multivariate data,

retaining the exploratory spirit of identifying potential outliers, if any, via a graphical display of a suitable diagnostic. It is assumed that at least a majority of cases follow a common pattern, a maximum number $m \leq \lfloor n/2 \rfloor$ of potential outliers being specified. In practice, it can be insightful to use a range of values of m.

The overall idea is that *outliers inflate dispersion*, measured here by the total variance (other measures of dispersion could of course be employed, use of t_{MCD} illustrating the unity between robustness and diagnostics noted at the outset). Stage I seeks a subsample M of m cases whose deletion maximally decreases dispersion. Thus, the corresponding version of Problem 1 minimises $t = t_{\text{trace}}$ given by

$$t_{\text{trace}}[\widehat{F}_{-M}] = 100 \times \frac{\text{trace}(\text{cov}[\widehat{F}_{-M}])}{\text{trace}(\text{cov}[\widehat{F}])}, \tag{3.3}$$

the optimal subset being denoted $\widehat{M}$.

At stage II, each case $i \in \widehat{M}$ is added back, by itself, to the retained cases (those labelled by $\widehat{H} = \widehat{M}^C$), computing each time the resulting percentage change in dispersion:

$$\delta_i = 100 \times \frac{\text{trace}(\text{cov}[\widehat{F}_{\widehat{H}\cup\{i\}}]) - \text{trace}(\text{cov}[\widehat{F}_{\widehat{H}}])}{\text{trace}(\text{cov}[\widehat{F}_{\widehat{H}}])},$$

so that $\delta_i \geq 0 \Leftrightarrow \frac{h}{h+1} \left\| x_i - \text{mean}[\widehat{F}_{\widehat{H}}] \right\|^2 \geq \text{trace}(\text{cov}[\widehat{F}_{\widehat{H}}])$. Overall, the idea is that a plot of the $\{\delta_i\}_{i\in\widehat{M}}$ will reveal as potential outliers those cases, if any, with relatively large positive values of δ_i.

2. Global diagnostic use of the likelihood displacement function:

Suppose here that the data $\{z_i : i \in N\}$ independently follow a parametric statistical model, z_i having log-likelihood $l_i(\theta)$, say, and let $\widehat{\theta}$ maximise the overall log-likelihood $l(\theta) = \sum_{i\in N} l_i(\theta)$. Complementary to the local analysis of Cook (1986), a generic diagnostic problem is to find, for given m, that subset M of m cases with the greatest effect on likelihood inference in the sense of maximising the likelihood displacement function. Thus, the corresponding version of Problem 1 maximises $t = t_{LD}$ given by

$$t_{LD}[\widehat{F}_{-M}] = 2\left\{ l(\widehat{\theta}) - l(\widehat{\theta}_{-M}) \right\}, \tag{3.4}$$

where $\widehat{\theta}_{-M}$ maximises $\sum_{i\in N\setminus M} l_i(\theta)$.

The Atkinson (1986) analysis can be adapted to unmask "likelihood outliers" thus: stage I maximises $t_{LD}[\cdot]$, while stage II plots $\{\delta_i\}_{i\in\widehat{M}}$ where

$$\delta_i = 2\left\{l_{\widehat{H}}(\widehat{\theta}_{\widehat{H}}) - l_{\widehat{H}}(\widehat{\theta}_{\widehat{H}\cup\{i\}})\right\} \geq 0$$

in which $l_{\widehat{H}}(\cdot)$ is the log-likelihood for the retained subsample $\widehat{H}$, maximised at $\widehat{\theta}_{\widehat{H}}$, while $\widehat{\theta}_{\widehat{H}\cup\{i\}}$ maximises the log-likelihood for $\widehat{H}\cup\{i\}$.

4 A relaxation strategy

4.1 Smooth embedding of combinatorial problems

The formulation of Problem 1 shows that the problems of interest here are *combinatorial*. Unfortunately, such problems are often extremely difficult to solve exactly. In particular, complete enumeration of all feasible solutions rapidly becomes impractical with increasing problem size. More efficient – but still exact – approaches are available, such as the branch and bound (or branch and cut) algorithm. However, in the particular case of the MCD estimator, extensive work by Agulló (personal communication) has shown that the branch and bound algorithm is only really operational if $n \leq 50$ and $k \leq 5$.

Instead, in this paper, we implement an approach suggested in Critchley et al. (2004). The idea here is to *relax* the problem, placing it in the world of nonlinear optimisation whose tools we then exploit. That is, the discrete, high-dimensional Problem 1 is embedded in a smooth, low-dimensional one, as follows. This is a specific instance of convex relaxation, which dates back at least as far as Birkhoff's theorem on permutation matrices as extreme points of the doubly stochastic matrices.

First, we use probability vectors to label weighted empirical distributions. For each $p = (p_i)$ in the set $\mathbb{P}^n$ of all probability n-vectors, $\widehat{F}(p)$ puts probability p_i on case z_i. In particular, $p_\circ = (n^{-1})$ labels the usual empirical distribution $\widehat{F}$. Thus, the set $\mathbb{V}^n_{-m} \equiv \mathbb{V}^n_h$ comprising the $\binom{n}{m} = \binom{n}{h}$ distinct permutations of $h^{-1}(0^T_m, 1^T_h)^T$ labels the distributions $\{\widehat{F}_{-M} : M \in \mathbb{N}_m\} \equiv \{\widehat{F}_H : H \in \mathbb{N}_h\} = \{\widehat{F}(v) : v \in \mathbb{V}^n_{-m}\}$ over which an optimum is sought. We refer to members of $\mathbb{V}^n_{-m} \equiv \mathbb{V}^n_h$ as (indexing) *h-subsets*, since they put equal weight h^{-1} on each of h indices and zero weight on the others.

Next, we embed $\mathbb{V}^n_{-m} \equiv \mathbb{V}^n_h$ in its convex hull $\mathbb{P}^n_{-m} \equiv \mathbb{P}^n_h = \{p \in I\!P^n : p_i \leq h^{-1} \ \ \forall\ i\}$,

noting that, dually, $\mathbb{V}_h^n$ is the set of all vertices (extreme points) of $\mathbb{P}_h^n$. For any $p \in \mathbb{P}_h^n$,

$$N_0(p) = \{i \in N : p_i = 0\}\,,\ N_*(p) = \left\{i \in N : 0 < p_i < h^{-1}\right\} \text{ and } N_1(p) = \left\{i \in N : p_i = h^{-1}\right\} \tag{4.1}$$

are possibly empty, disjoint sets covering N with sizes, $n_0(p)$, $n_*(p)$ and $n_1(p)$ say, summing to n (note that $n_*(p) = 1$ is impossible). Thus, p is a relative interior point of $\mathbb{P}_h^n$ if $n_*(p) = n$ and a relative boundary point otherwise, being a vertex if and only if $n_*(p) = 0$. A relative boundary point p belongs to the exposed face $\mathbb{F}(p)$ of $\mathbb{P}_h^n$ comprising the convex hull of the $n_v(p) = \binom{n_*(p)}{h-n_1(p)} = \binom{n_*(p)}{n-h-n_0(p)}$ vertices $v \in \mathbb{V}_h^n$ with $N_0(v) = N_0(p)$ and $N_1(v) = N_1(p)$, whose dimension $n_v(p) - 1$ is zero if and only if p is a vertex.

Finally, we replace the target function $t[\cdot]$ by its smooth version $t(p) = t[\widehat{F}(p)]$. This strategy results in the following smooth reformulation of Problem 1:

Problem 2. *(Smooth reformulation of Problem 1) Optimise $t(p)$ over $p \in \mathbb{P}_{-m}^n \equiv \mathbb{P}_h^n$.*

Focusing now without loss on smooth minimisation problems, it follows at once that any (increasing function of a) concave function $t(\cdot)$ attains its minimum over the feasible region $\mathbb{P}_{-m}^n \equiv \mathbb{P}_h^n$ of Problem 2 at a member of the feasible region $\mathbb{V}_{-m}^n \equiv \mathbb{V}_h^n$ of Problem 1. In fact, this relaxation strategy succeeds for a wider class of "gravitational" functions, defined next.

4.2 Gravitational functions

We begin by defining gravitational functions on a general convex set $\mathbb{P}$. Recall that a direction d ($\|d\| = 1$) from $p \in \mathbb{P}$ is called feasible if $p + \delta d \in \mathbb{P}$ for all small enough $\delta > 0$.

We call a smooth function $t(\cdot) : \mathbb{P} \to \mathbb{R}$ *gravitational* if it has the property that when, in any given direction, you start off downhill, you keep going downhill. That is, if for each point $p \in \mathbb{P}$ and for each feasible direction d from p:

$$d^T t'(p) \leq 0 \Rightarrow d^T t'(p + \delta d) \leq 0,\ (\delta > 0, p + \delta d \in \mathbb{P})\,, \tag{4.2}$$

where $t'(\cdot)$ denotes the gradient vector. Subsuming smoothness, every concave function is, therefore, gravitational, having the stronger property that:

$$d^T t'(p) \leq 0 \Rightarrow d^T t'(p + \delta d) \leq d^T t'(p) \leq 0,\ (\delta > 0, p + \delta d \in \mathbb{P})\,,$$

while every increasing function of a gravitational function is gravitational. We note in passing that, generalising a familiar result for concave functions, p is a global maximum of a gravitational function $t(\cdot)$ if and only if $d^T t'(p) \leq 0$ for every feasible direction d from p.

We focus now on gravitational functions $t(\cdot)$ defined on the convex set $\mathbb{P} = \mathbb{P}_h^n$.

The linear constraint $p^T 1_n = 1$ means that any movement within $\mathbb{P}_h^n$ is in a *centred* direction d satisfying $d = C_n d$, where $C_n = (I_n - n^{-1} 1_n 1_n^T)$. Thus, instead of a gradient vector $t'(\cdot)$ computed without regard to this constraint, we may use the unique centred gradient $t^c(\cdot) = C_n t'(\cdot)$, noting that $d^T t^c(p) = d^T t'(p)$. This is a special case of a projected gradient, widely used in constrained optimisation (see, for example, Bazaraa et al., 1993).

Now, if $\pm d$ are both feasible from a local minimum p of $t(\cdot)$, $d^T t^c(p) = 0$. Thus, if p is a global minimum and $t(\cdot)$ is gravitational, $t(\cdot)$ must be constant over all $p \pm \delta d$ in $\mathbb{P}_h^n$. Accordingly, we have:

Lemma 1. *Let $t(\cdot)$ be a gravitational function on $\mathbb{P}_h^n$. Then: (a) if $t(\cdot)$ is global minimised at a relative interior point $p \in \mathbb{P}_h^n$, $t(\cdot)$ is constant on $\mathbb{P}_h^n$. (b) if $t(\cdot)$ is global minimised at a relative boundary point $p \in \mathbb{P}_h^n$, $t(\cdot)$ is constant on $\mathbb{F}(p)$.*

Lemma 1 gives at once:

Proposition 2. *A gravitational function attains its minimum over $\mathbb{P}_h^n$ at a vertex.*

In many statistical problems, the constancy described in Lemma 1 for any non-vertex $p \in \mathbb{P}_h^n$ has probability zero. In such cases, a gravitational function is minimised over $\mathbb{P}_h^n$ *only* at a vertex (w.p.1).

5 Examples: relaxed versions of the target functions

We give here the relaxed versions of the example target functions.

5.1 In robustness

The relaxed target functions $t_{MCD}(\cdot)$ and $t_{LTS}(.)$ below are both concave, ensuring that each attains its minimum at a vertex.

1. Minimum Covariance Determinant estimator:

   The MCD target function (3.1) relaxes to

   $$t_{MCD}(p) = \log\det(\widehat{\Sigma}(p)) \tag{5.1}$$

   where $\widehat{\Sigma}(p) = \sum_{i \in N} p_i \left(x_i - \overline{x}(p)\right) \left(x_i - \overline{x}(p)\right)^T$, in which $\overline{x}(p) = \sum_{i \in N} p_i x_i$.

2. Least Trimmed Squares estimator:

   The LTS target function (3.2) relaxes to

$$t_{LTS}(p) = \sum_{i \in N} p_i r_i^2(\hat{\beta}(p)) \tag{5.2}$$

   where $\hat{\beta}(p) = \text{argmin}_{\beta \in \mathbb{R}^k} \sum_{i \in N} p_i r_i^2(\beta)$.

5.2 In diagnostics

The relaxed target function $t_{\text{trace}}(\cdot)$ below is a concave quadratic, whose minimum is therefore attained at a vertex. Maximisation of $t_{LD}(\cdot)$ is equivalent to minimisation of $-t_{LD}(\cdot)$, whose properties are noted in the Appendix along with gravitational examples.

1. Detection of multiple multivariate outliers:

   The multivariate outlier detection target function (3.3) relaxes to

$$t_{\text{trace}}(p) = 100 \times \frac{\text{trace}(\widehat{\Sigma}(p))}{\text{trace}(\widehat{\Sigma}(p_\circ))}. \tag{5.3}$$

2. Global diagnostic use of the likelihood displacement function:

   Here, we relax by putting probability weight p_i on $l_i(\cdot)$, the log-likelihood for z_i, rather than on z_i itself. We write $l(\theta; p) = \sum_{i \in N} p_i l_i(\theta)$ so that, in particular, $l(\theta) = nl(\theta; p_\circ)$. Thus, the unperturbed MLE $\widehat{\theta} = \widehat{\theta}(p_\circ)$, where $\widehat{\theta}(p) = \arg\max l(\cdot; p)$. Overall, the likelihood displacement target function (3.4) relaxes to:

$$t_{LD}(p) = 2\left\{ l(\widehat{\theta}) - l(\widehat{\theta}(p)) \right\}. \tag{5.4}$$

6 Optimisation procedure

6.1 Context

Switching to the world of continuous optimisation is not a solution in itself, some of its subclasses being much easier to handle than others. Unconstrained problems are easier than constrained problems (indeed, a classical way to deal with constrained problems is via a sequence of related unconstrained problems). Linear problems, with or without constraints, can be solved with great efficiency. Focusing now without loss on minimisation problems,

convex functions are relatively straightforward. In the much more challenging nonconvex case, among other possibilities, we may resort to heuristics such as simulated annealing and genetic algorithms. An additional difficulty here is the possibility of multiple local optima, minimisation of a concave function being a prime example. Convergence to the global optimum is rarely guaranteed in the nonconvex case, even if techniques can be applied to increase the probability of this. Due to this complex diversity, there is no universally applicable optimisation tool. A complete survey of the field is beyond the scope of this paper, but details on many nonlinear optimisation and related approaches can be found in, for example, Pardalos and Rosen (1987), Bazaraa et al. (1993), Horst and Tuy (2003), Conn et al. (2000) and Sartenaer (2003).

For the examples of interest here, the continuous formulation of Problem 2 belongs to a difficult category of minimisation problems: nonconvex – often gravitational, if not concave – functions under convex constraints (here, bound and linear constraints), the minima only being encountered when constraints are active. We may also want to preserve some properties of the underlying estimators or methods, such as affine equivariance, which is not guaranteed by all optimisation methods. However, the goal of this paper is not to develop a new algorithm capable of solving arbitrary constrained nonlinear problems. Our objective is much more modest:

> to assess whether relaxation provides a suitable, general approach to a range of important statistical problems which can be relaxed into the form of Problem 2, the strategy being to produce a *single*, fast algorithm for problems of this type.

We emphasise again that in many statistical problems "good enough" is good enough, in the sense that it is not necessary to have an algorithm which guarantees convergence to the global optimum. In particular, this is true with the running examples here, in each of which it is sufficient that the retained subset $\widehat{H}$ itself contain no gross outliers. Accordingly, in this paper, we use only basic nonlinear optimisation techniques – notably, the feasible direction method based on projected gradients. More advanced optimisation techniques could well be used to advantage, especially when pursuing the quite different strategy of developing a suite of separate algorithms, each designed to exploit properties particular to a specific type of target function $t(\cdot)$. For example, concerning $t_{\text{trace}}(\cdot)$, Danninger and Bomze (1993, Theorem 2) have developed a global optimality condition for use when minimising a concave

quadratic. Indeed, it could well be of interest to explore the use of global optimisation methods (see, for example, Pardalos and Rosen (1987) and Horst and Tuy (2003)) in any context where the user is happy to trade speed off against optimality. More widely still, we have the challenge of bringing the most appropriate operational research techniques to bear on important statistical problems. But alternatives such as this – including algorithmic comparison on specific target functions – we leave for further developments.

We describe next the descent algorithm used for the relaxed Problem 2.

6.2 Starting points and descent to a vertex

6.2.1 Starting points

For nonconvex problems such as those considered here, the choice of starting points is crucial. Indeed, in the gravitational case, as soon as the algorithm starts descending a valley, there is no turning back. The vertex where the iterative descent procedure ends up is completely determined by the point from which it starts.

Traditionally, random h-subsets (vertices of $\mathbb{P}_h^n$) are chosen to initialise combinatorial algorithms. Using many such starting points allows investigation of many different valleys. For the examples considered here, simulations have shown that drawing 100 vertices at random yields procedures that are both fast and effective. One can also depart from carefully selected points, such as $p_\circ = (n^{-1})$ which does not favour any of the cases. (Note, however, that starting from $p_\circ$ is not an option in likelihood displacement contexts since, as noted in the Appendix, $t^c_{LD}(p_\circ) = 0$ under regularity.)

6.2.2 Direction of descent

Our approach uses local information to optimally descend, in an iterative manner, from a given starting point to a vertex.

In general terms, a steepest descent technique departs from a point in the opposite direction to the gradient vector. Here, the linear constraint $p^T 1_n = 1$ restricts the search space to centred vectors so that we use, instead, the centred gradient. Thus, if p is the current position in the search space, the next iterate is $p+\delta d(p)$, where the optimal direction is given by $d(p) = -\frac{t^c(p)}{\|t^c(p)\|}$, while $\delta > 0$ represents the size of the move, discussed next.

6.2.3 Size of move

If the target function is gravitational, choosing δ as large as possible will yield the lowest value for the target function in the direction $d(p)$. Accordingly, we increase δ until at least one of the variables (elements of p) reaches a boundary. The variables reaching the boundary are fixed at that value until the end of the optimisation procedure. Operationally, this means setting to zero the corresponding elements of $t^c(p)$, a second form of projection. At each step the number of free variables decreases by at least one, so that our descent algorithm needs at most n steps to reach a vertex.

6.3 Swapping strategies to arrive at a candidate local minimum

The above descent strategy is guaranteed to converge to a vertex, but this need not be a "candidate local minimum" in the sense defined next. We describe here subsequent swapping steps which *are* guaranteed to lead to such a vertex.

6.3.1 Characterisation of a candidate local minimum

We say that a vertex $v \in \mathbb{V}_h^n$ is a candidate local minimum for a smooth target function $t(\cdot)$ if every feasible direction d from it is uphill (that is, satisfies $d^T t^c(v) \geq 0$). This is clearly necessary for v to be a local minimum of $t(\cdot)$.

The algorithm uses the following necessary and sufficient condition for a vertex to be a candidate local minimum in Problem 2. In the notation of (4.1), it is straightforward to show that:

Proposition 3. *Let $t(p)$ be a smooth function. A vertex v is a candidate local minimum of $t(p)$ over $\mathbb{P}_h^n$ if and only if every neighbouring vertex is in an uphill direction. That is, if and only if*

$$\min_{i \in N_0(v)} t_i^c(v) \geq \max_{i \in N_1(v)} t_i^c(v). \tag{6.1}$$

6.3.2 Locally-proposed 1-swaps

When reaching a vertex, v say, the algorithm always checks whether it is a candidate local minimum or not. If it is, it stops. If not, there is a neighbouring vertex in a strictly downhill

direction. For $i_0 \in N_0(v)$ corresponding to the minimum and $i_1 \in N_1(v)$ to the maximum in (6.1), $d = (e_{i_0} - e_{i_1})/\sqrt{2}$ is such a direction, e_i denoting the i^{th} unit vector. Using (4.2) again, the maximal move $\delta = \sqrt{2}/h$ in this direction is optimal, this move simply *swapping* the values of the two elements $v_{i_0} = 0$ and $v_{i_1} = h^{-1}$ of v. The process can be applied iteratively, convergence to a candidate local minimum being assured since the function is strictly decreasing at each step and there are a finite number of vertices to jump to. This part of the algorithm, called *locally-proposed 1-swap improvement*, can usually be performed very quickly using updating formulae derived from the objective function.

The process of swapping is well-known in combinatorial problems, the Feasible Solution Algorithm (Hawkins, 1994 and Hawkins and Olive, 1999) for the MCD or LTS estimators being a lead example in robust statistics. When the starting point is already a vertex, this iterative procedure, being fast and ending at a candidate local minimum, can be used as an algorithm in its own right.

6.3.3 Locally-proposed l-swaps

The above discussion focuses on 1-swaps, meaning that only one vertex element of each type is changed. We could think of interchanging $l > 1$ values.

Let v be a vertex which is not a candidate local minimum. In some cases, the second lowest value of the centred gradient in $N_0(v)$ is also smaller than the second largest value in $N_1(v)$, so that swapping this *pair* of elements strictly decreases the target function. In general, let $l_{\max} \leq \min\{h, m\}$ be the largest value of l such that the l^{th} lowest value of the centred gradient in $N_0(v)$ is smaller than the l^{th} largest value in $N_1(v)$. Then, for any l between 1 and $l_{\max}$, the corresponding l-swap strictly decreases the target function. In the present algorithm, we have implemented both 1-swaps and $l_{\max}$-swaps.

6.4 Summary of the algorithm

This general, fast minimisation algorithm can be summarised as follows:

- Step 1: Generation of starting – including vertex – points p^0.
- Step 2: Projected gradient descent from each p^0 until reaching a vertex v.
- Step 3: 1-swap or $l_{\max}$-swap descent from all starting vertices v^0, and from all solutions v obtained at step 2, until the candidate local minimum criterion (6.1) is met.

- Step 4: Return all candidate local minima found and, in particular, the best of them.

We emphasise that each branch of this algorithm converges to a candidate local minimum for *any* smooth gravitational target function (in particular, for any increasing function of a concave function), and that it preserves other properties, such as affine equivariance.

6.5 An adapted algorithm

It is not always straightforward to establish that a particular target function $t(\cdot)$ we wish to minimise is gravitational, while it is always of interest to explore how widely an algorithm can be applied. Such considerations motivate adapting the above algorithm for use with target functions that may not be gravitational. One simple way to do this is as follows.

The above algorithm relies on gravitationality only in as much as the maximal move in a descent direction brings the maximal decrease. However, an effective minimisation procedure does not require each move to have such a strong property. In particular, it is not necessary for the target function to be gravitational for the maximal move to strictly decrease it.

Recall now that the underlying problem is to minimise $t(\cdot)$ over the vertex set $\mathbb{V}^n_{-m} \equiv \mathbb{V}^n_h$, noting that the above algorithm involves two types of move between such points: moves $v^0 \to v$ in step 2 when projected gradient descent starts from a vertex, and the locally-proposed l-swaps comprising step 3. One simple way, then, to adapt this algorithm is to regard any such between-vertex move as a *proposed* move: if it strictly decreases the target function, it is accepted; otherwise, we stop where we are. Apart from these additional checks (unnecessary when $t(\cdot)$ is known to be gravitational), the algorithm is unchanged.

Each branch of this adapted algorithm converges rapidly to a vertex that is either a candidate local minimum or one from which the locally-proposed l-swap does not lead to a strict improvement. We call these *l-terminal vertices*. Step 4 returns all such vertices and, in particular, the best of them. Other properties, such as affine equivariance, are again preserved.

7 Examples: illustration and tests of the algorithm

This section, organised as follows, illustrates how the above unifying algorithm works and tests its performance on a range of statistical problems.

Section 7.1 uses least trimmed squares (LTS) in simple linear regression, illustrating how iterations proceed from different starting points. Sections 7.2 and 7.3 then demonstrate the algorithm's effectiveness in the multiple multivariate outlier detection and global likelihood displacement problems respectively. Two examples of this latter problem are given, the first (Section 7.3.1) testing the algorithm's global performance, and the second (Section 7.3.2) its adapted form. For brevity, we simply note here that its performance in both LTS and MCD problems has also been tested using the same collection of test data sets employed by Rousseeuw and Van Driessen (1999 and 2006), the optimal solution being obtained in all cases. Finally, Section 7.4 demonstrates the algorithm's good performance in a large scale problem, via LTS analysis of the well-known Boston housing data set.

7.1 Illustration: LTS for simple linear regression

The data here comprise a random sample of 45 cases from the normal simple linear regression $y_i = \alpha + \beta x_i + \varepsilon_i$ with $\alpha = 1$, $\beta = 1.5$ and $\sigma = 2$, together with 5 clustered, high leverage, large residual outliers. The resulting data set is shown in panel (a) of Figure 1 together with the, badly biased, least squares fit (which is also the fit for the initial point $p_\circ = (n^{-1}))$.

The LTS fit corresponds to a weighted least squares fit in which an optimal subset of h cases out of n have equal weight. Here we use $h = 26$, corresponding to the highest possible breakdown point. Starting from a given initial point p^0, the algorithm uses a constrained steepest descent strategy which improves the LTS criterion at each iteration, while updating the probability vector containing the weights to be used for the next fit.

The symbols used in the plots reflect the weight attributed to the corresponding case: if $p_i = 0$, a small, empty triangle is used; if $0 < p_i < 1/h$, an empty square is plotted; finally, if $p_i = 1/h$, a black dot is represented (interpreting h as n in panel (a), as is appropriate there).

This iterative process is illustrated in panels (b) and (c) of Figure 1, starting from $p_\circ$. After five iterations of steepest descent, the algorithm reaches the trial fit and weights shown in panel (b). At this stage, all five clustered outliers have been excluded – one at each iteration. This results in a very different fit compared to that at $p_\circ$ (panel (a)). Indeed, it is essentially the same as the final fit obtained, shown in panel (c), illustrating that placing little or no weight on the outliers is, indeed, "good enough".

Panels (d) to (f) of Figure 1 again refer to the first, fifth and last iteration of the algorithm,

starting now from a randomly chosen vertex. The final solution reached (panel (f)) is identical to that starting from $p_\circ$. This provides further evidence that the LTS target function has indeed been minimised, only the h 'most collinear' cases being retained. The very close proximity of the fits in panels (e) and (f) illustrates again that placing little or no weight on the outliers is, indeed, "good enough".

7.2 Multiple multivariate outlier detection

For visual clarity, the two-stage multiple multivariate outlier detection procedure described above is illustrated here on the bivariate data set shown in panel (a) of Figure 2, comparable results being obtained in higher dimensions. These data comprise 100 independent cases, 80 from the standard bivariate normal distribution, relative to which the other 20 are shift outliers.

Stage I consists of the minimisation of the scalar dispersion measure (5.3) over $\mathbb{P}^n_{-m}$ with $m = 50$. As expected, the optimal subset $\widehat{M}$ (whose cases are plotted with a small, empty triangle in panel (b) of Figure 2) omits all 20 outliers, the retained cases $\widehat{H} = \widehat{M}^C$ (plotted as black dots) being compactly placed amongst the majority cluster.

At stage II, the δ_i measures ($i \in \widehat{M}$) distinguish potential outliers – those with relatively large, positive δ_i values (plotted again with a small, empty triangle in panel (c) of Figure 2) – from non-discordant cases (plotted there with a black cross).

Panel (d) summarises the analysis.

7.3 Global use of the likelihood displacement function

7.3.1 Estimation of the mean in exponential samples

As shown in the Appendix, $-t_{LD}(\cdot)$ is gravitational – but not, in general, concave – for inference for an exponential mean.

Supposing, as holds with probability 1 for a sample from any continuous distribution, that the $\{x_i\}_{i \in N}$ are distinct, the global performance of the algorithm can be checked *analytically*. For, as is intuitive, it follows that $t_{LD}(\cdot)$ has exactly two local maxima over $\mathbb{P}^n_{-m} = \mathbb{P}^n_h$, at vertices $v_{\min}$ and $v_{\max}$ of $\mathbb{V}^n_{-m} = \mathbb{V}^n_h$, these putting equal weight h^{-1} on the members of $\widehat{H}_{\min} :=$ {the h smallest observed values} and $\widehat{H}_{\max} :=$ {the h largest observed values} respectively, the former being the global maximum.

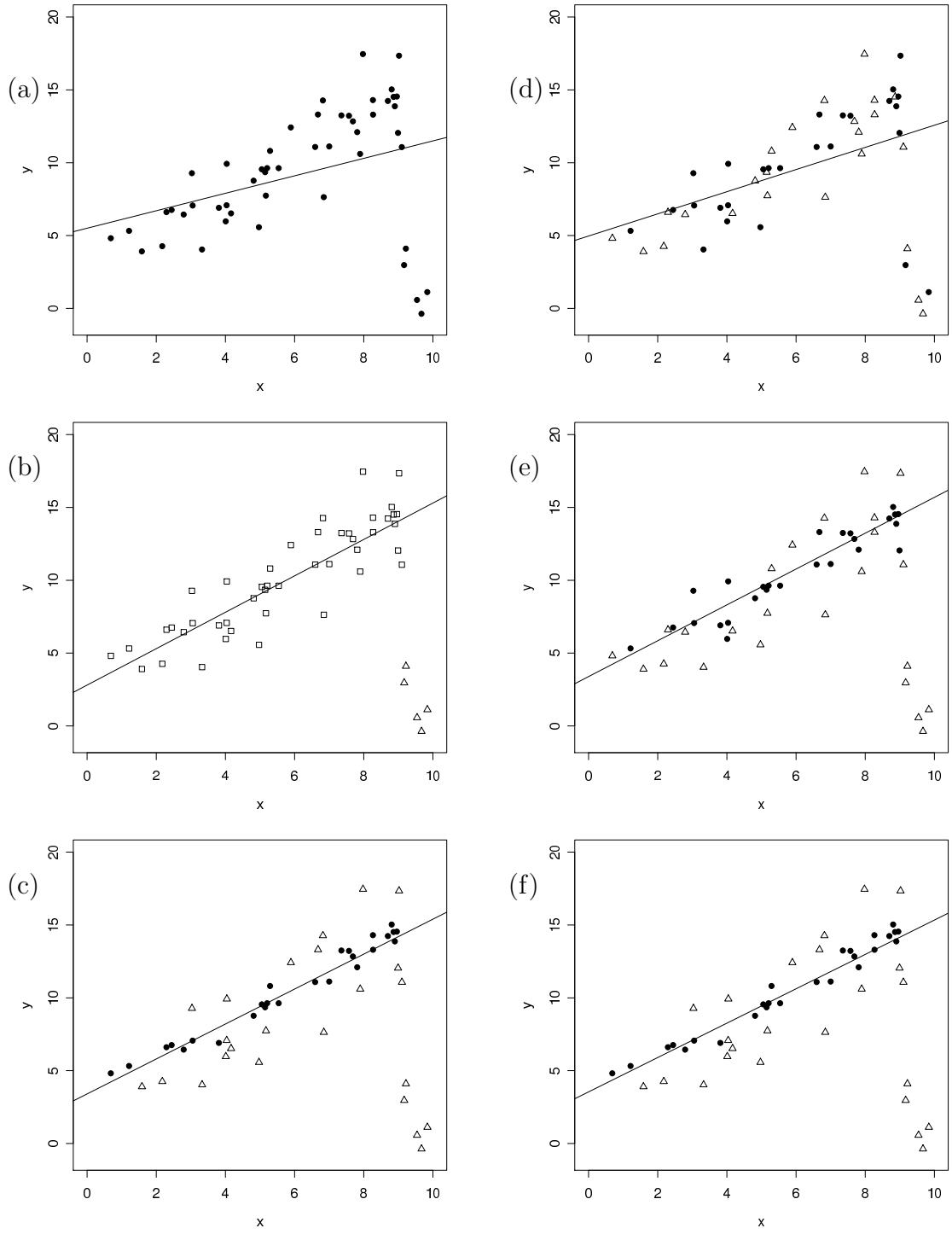


Figure 1: Performance of the algorithm for computing the LTS estimator on a data set containing a cluster of high leverage, large residual cases, starting from $p_\circ = (n^{-1})$ (lefthand column) and from a randomly chosen vertex (righthand column). Panels (a) and (d) show the initial fit, panels (b) and (e) the fit after five iterations of steepest descent, and panels (c) and (f) the final fit. Plot symbols reflect the weight currently assigned to cases, as detailed in Section 7.1.

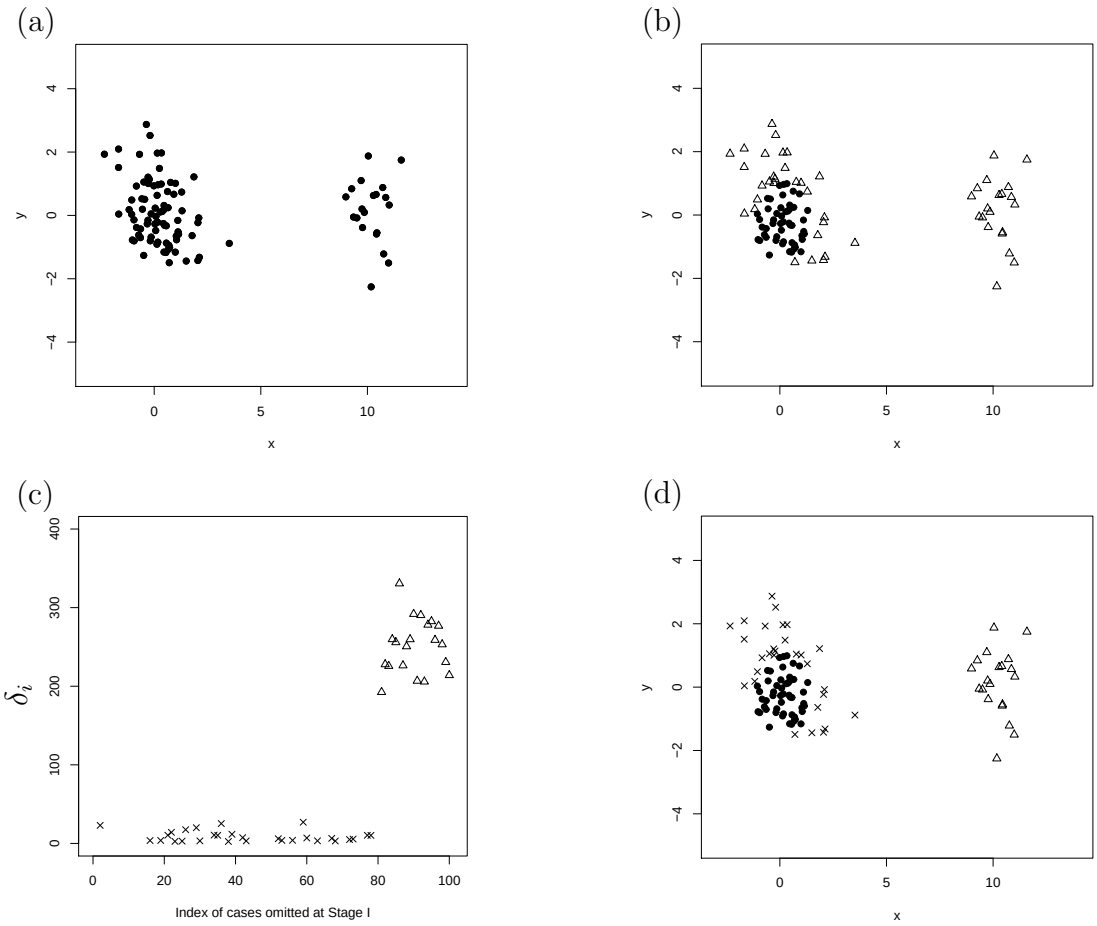


Figure 2: Performance of the algorithm for detecting outliers using the two-stage procedure introduced by Atkinson (1986). Panel (a) plots the original data, panel (b) identifies the cases defining $\widehat{M}$, while panel (c) gives the percentage change of dispersion when adding a single case from $\widehat{M}$ to the retained cases, distinguishing potential outliers from non-discordant cases. Panel (d) summarises the results. Plot symbols are detailed in Section 7.2.

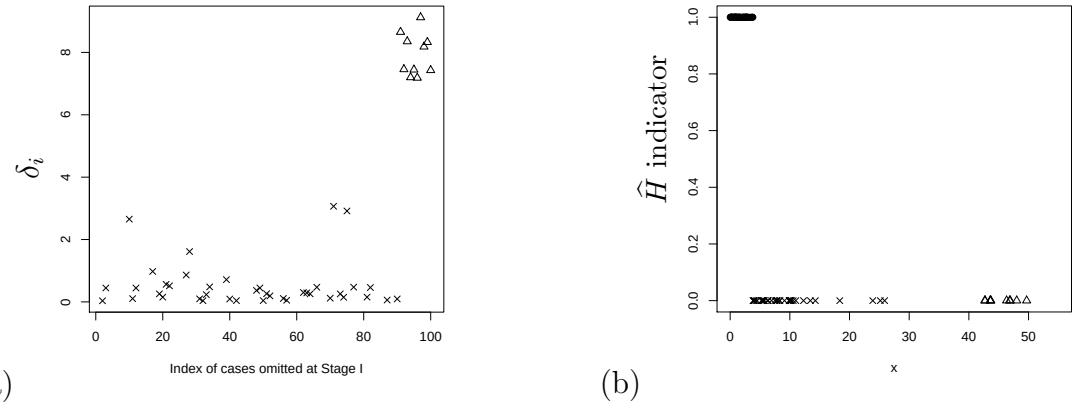
Indeed, we can *prove* the algorithm always works in this case, in the following sense. It follows from the expression for $-t^c_{LD}(\cdot)$ given in the Appendix that:

> for any starting point p^0 with $\overline{x}(p^0) < \overline{x}$, the algorithm converges to $v_{\min}$,
> while:
> for any starting point p^0 with $\overline{x}(p^0) > \overline{x}$, the algorithm converges to $v_{\max}$,

one of these inequalities, defining the zones of attraction of $v_{\min}$ and $v_{\max}$, holding w.p.1 for any randomly chosen p^0. This analysis also confirms the value of, as we do, using multiple random starting vertices and returning *all* candidate local optima found, both local optima

here being of potential interest.

We finish with a worked example. Using 90 cases drawn from an exponential distribution with mean 5 together with 10 clear outliers, Figure 3 illustrates how the two-stage procedure of Atkinson with $m = n/2$ performs when applied to the likelihood displacement function. In panel (a), the δ_i measures of the observations deleted at the first stage are represented and clearly separate the points into two groups – those with small δ_i values and the others – panel (b) summarising the whole procedure. The same plot symbols are used as in Section 7.2 above: black dots correspond to the cases labeled by $\widehat{H}$, black crosses denote the non-discordant cases in $\widehat{M}$, while the small, empty triangles characterise the potential outliers. Note however that, in order to distinguish the different types of points in panel (b), the symbols are plotted with respect to 1 or 0 (vertically) according to whether they belong, or not, to $\widehat{H}$.



(a) (b)

Figure 3: Performance of the algorithm for detecting cases influential in estimating an exponential mean using the likelihood displacement function.

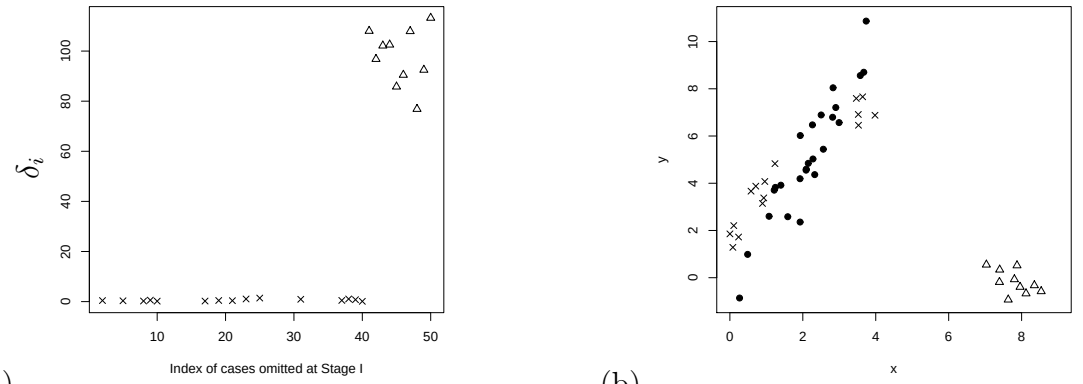
7.3.2 Multiple linear regression

Considering again the multiple regression setting $y_i = \beta^T x_i + \varepsilon_i$ where the $\{\varepsilon_i\}$ are independently distributed as $N(0, \sigma^2)$ and $\beta \in I\!R^k$, with σ known ($\sigma = 1$, say, without loss), the likelihood displacement function takes the form

$$t_{LD}(p) = 2\left\{l(\hat{\beta}) - l(\hat{\beta}(p))\right\}$$

where $l(\beta) = -\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta^T x_i)^2$, $\hat{\beta}(p) = (X^T P X)^{-1} X^T P y$ in which $P = \mathrm{diag}(p)$ and $X = (x_i^T)$. The centred gradient is derived in Appendix A.4, but it is not clear whether the target

function is gravitational or not. However, the adapted algorithm suggested in Section 6.5 can still be applied. Focussing again for visual clarity on the simple regression case, the two-stage procedure of Atkinson with $m = n/2$ returns then the results of Figure 4, where a cluster of 10 high leverage, large residual, outliers have been added to a sample of 40 cases from the model, the same three plot symbols as above being used. The likelihood displacement here being proportional to p-generalised Cook's distance, introduced in Critchley et al. (2001), this successful analysis confirms computational results from that paper, obtained here with a much simpler algorithm.



(a) (b)

Figure 4: Performance of the algorithm for detecting influential points in linear regression using the likelihood displacement function.

7.4 LTS on Boston housing data

To illustrate that the algorithm can also deal with a large data set, it was applied in a regression context to the Boston housing data of Harrison and Rubinfeld (1978) and its results compared with those obtained by the improved Feasible Solution Algorithm of Hawkins and Olive (1999). As these two authors suggest, the binary predictor variable indicating adjacency to the Charles river was omitted, resulting in a data set consisting of $n = 506$ observations in 13 dimensions (12 independent and one dependent variables). Minimising the LTS objective function with $h = 260$, the improved FSA algorithm reaches a final value of 236.7. Our relaxed algorithm based on the smooth objective function (5.2) yields a final minimised value of 232.07, lower than improved FSA. One has to note that both algorithms are outperformed by the original FSA algorithm adapted for the LTS function (Hawkins 1993),

since Hawkins and Olive (1999) reports that the final value attained is then 222.0. However, in order to achieve this lower value, the computing time needs to be greatly increased.

8 Discussion

Gravitational functions subsume (increasing functions of) concave functions. In this paper, the relaxation strategy for combinatorial problems proposed by Critchley et al. (2004) has been implemented in a single algorithm capable of handling any problem leading to minimisation of such a function over the relevant convex hull (indeed, an adapted algorithm allows the gravitational condition itself to be relaxed). This gives sufficient generality to cover a wide range of important statistical problems, lead examples from robustness and diagnostics being used for illustration here.

Although these smooth reformulations belong to the very difficult class of constrained, nonconvex minimisation problems, the algorithm presented is both general and fast. This is due to its using only basic tools – notably, projected gradients and swaps – with the added advantage of preserving desirable properties, such as affine equivariance. Its performance has been illustrated and tested, with encouraging results.

Whereas we have emphasised that in many statistical problems it is not necessary to have an algorithm which guarantees convergence to a global optimum, alternative forms of the present algorithm can be explored. In particular, (a) other swapping strategies and (b) suitably adapted forms of the alternative starting points advocated by Rousseeuw and van Driessen (1999), in an MCD context, to maximise robustness. Again, more advanced optimisation techniques could be considered, especially in connection with the quite different strategy of developing a suite of separate algorithms, each designed to exploit properties particular to a specific type of target function. This constitutes a large body of future work.

Appendix: properties of $-t_{LD}(\cdot)$

A.1 For all models, $-t_{LD}$ is concave throughout a neighbourhood of $p = p_\circ$:
We have at once from (5.4) that $t_{LD}(p) \geq t_{LD}(p_\circ) = 0$ while, under regularity, $t^c_{LD}(p_\circ) = 0$. Thus:

$-t_{LD}(\cdot)$ is concave (hence, gravitational) throughout a neighbourhood of $p = p_\circ$, *whatever* the underlying model.

This concavity extends to all of $\mathbb{P}_h^n$ for k-variate known dispersion normal samples ($k \geq 1$) with $\theta = \mu$. In general, whether or not $-t_{LD}(\cdot)$ is gravitational on all of $\mathbb{P}_h^n$ depends on the particular log-likelihoods $\{l_i(\cdot)\}_{i\in N}$ adopted. We establish next a general expression for the centred gradient vector $-t_{LD}^c(p)$, computable directly from them, and exploited by the algorithm described in Section 6.

A.2 The centred gradient vector $-t_{LD}^c(p)$:

Denoting the i^{th} score vector by $s_i(\theta) = \partial l_i(\theta)/\partial\theta$, differentiating (5.4) with respect to p and centring yields:

$$-t_{LD}^c(p) = 2C_n \sum_{i\in N} \partial l_i(\widehat{\theta}(p))/\partial p = 2C_n D(p) \sum_{i\in N} s_i(\widehat{\theta}(p)) = 2C_n D(p) S(p) 1_n$$

where $D(p) = \partial(\widehat{\theta}(p))^T/\partial p$ and $S(p)$ has general column $s_i(\widehat{\theta}(p))$. Now, $\widehat{\theta}(p)$ is assumed to uniquely solve the normal equations $S(p)p = 0$. Differentiating these with respect to p and centring, we have:

$$C_n D(p) H(p) + C_n S^T(p) = 0$$

where $H(p) = \sum_{i\in N} p_i H_i(p)$, $H_i(p) = \partial^2 l_i(\theta)/\partial\theta\partial\theta^T|_{\theta=\widehat{\theta}(p)}$ being the i^{th} Hessian evaluated at $\widehat{\theta}(p)$. Substituting for $C_n D(p)$, we have the desired general expression:

$$-t_{LD}^c(p) = 2C_n S^T(p) \left[-H(p)\right]^{-1} S(p) 1_n,$$

$H(p)$ being assumed nonsingular.

A.3 Exponential samples:

We give now an example where, although it needs not be concave throughout $\mathbb{P}_h^n$, $-t_{LD}(\cdot)$ is always gravitational there.

Let $\{x_i\}_{i\in N}$ be a sample from the exponential distribution with mean $\theta > 0$, so that $\widehat{\theta}(p) = \overline{x}(p)$. It follows that

$$-t_{LD}^c(p) = \frac{2n\left(\overline{x} - \overline{x}(p)\right)}{\left(\overline{x}(p)\right)^2} C_n x$$

where x has general element x_i. Thus, for any feasible direction d from p,

$$d^T\left(-t_{LD}^c(p)\right) \leq 0 \Leftrightarrow \left(d^T x\right)\left(\overline{x} - \overline{x}(p)\right) \leq 0,$$

in which case, for all $\delta > 0$ with $p + \delta d$ in $\mathbb{P}_h^n$,

$$d^T\left(-t_{LD}^c(p+\delta d)\right) = \frac{2n\left\{\left(d^T x\right)\left(\overline{x} - \overline{x}(p)\right) - \delta\left(d^T x\right)^2\right\}}{\left(\overline{x}(p+\delta d)\right)^2} \leq 0,$$

so that $-t_{LD}(\cdot)$ is gravitational. However, $-t_{LD}(\cdot)$ need not be concave on $\mathbb{P}_h^n$, since the doubly-centred second derivative matrix

$$t_{LD}^{cc}(p) = \frac{2n\left(2\overline{x} - \overline{x}(p)\right)}{\left(\overline{x}(p)\right)^3} C_n x x^T C_n$$

has a negative eigenvalue whenever $\overline{x}(p) > 2\overline{x}$.

A.4 Multiple regression:

When the log-likelihood in the likelihood displacement function $t_{LD}(p)$ takes the form

$$l(\beta) = -\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta^T x_i)^2,$$

as in Section 7.3.2, we have

$$-t_{LD}^c(p) = 2C_n E(p) X (X^T P X)^{-1} X^T e(p)$$

where $e(p) = y - X\hat{\beta}(p)$ and $E = \text{diag}(e(p))$. It is unclear whether or not $-t_{LD}(\cdot)$ is gravitational in this case.

9 References

Atkinson, A.C. (1986), Masking unmasked, *Biometrika*, **73**, 533–541.

Bazaraa, M.S., Sherali, H.D. and Shetty, C.M. (1993), *Nonlinear programming: theory and algorithms*, Wiley, New York, 2nd edition.

Conn, A.R., Gould N.I.M. and Toint P.L. (2000), *Trust-Region Methods*, MPS-SIAM Series on Optimization.

Cook, R.D. (1986), Assessment of local influence (with discussion), *J.R. Statist. Soc. B*, **48**, 133–169.

Critchley, F., Atkinson, R.A., Lu, G. and Biazi, E. (2001), Influence analysis based on the case sensitivity function, *J.R. Statist. Soc. B*, **63**, 307–323.

Critchley, F., Schyns, M., Haesbroeck, G., Kinns, D., Atkinson, R.A. and Lu, G. (2004), The case sensitivity function approach to diagnostics and robust computation: a relaxation strategy, pp.113-125 in Jaromir Antoch (ed.), *COMPSTAT 2004: Proceedings in Computational Statistics*, Physica-Verlag, Heidelberg.

Danninger, G. and Bomze, I. (1993), Using copositivity for global optimality criteria in concave quadratic programming, *Mathematical Programming*, **62**, 575–580.

Harrison, D. and Rubinfeld, D.L. (1978), Hedonic prices and the demand for clean air, *journal of Environmental Economics Management*, **5**, 81–102.

Hawkins, D.M. (1993), The Feasible Solution Algorithm for least trimmed squares regression, *Computational Statistics and Data Analysis*, **17**, 185–196.

Hawkins, D.M. (1994), The Feasible Solution Algorithm for the Minimum Covariance Determinant Estimator, *Computational Statistics and Data Analysis*, **17**, 197–210.

Hawkins, D.M. and Olive, D.J. (1999), Improved feasible solution algorithms for high breakdown estimators, *Computational Statistics and Data Analysis*, **30**, 1–11.

Horst, R. and Tuy, H (2003), *Global optimization. Deterministic approaches*, Spinger, 3rd edition.

Pardalos, P.M. and Rosen, J.B. (1987), *Constrained Global Optimization: Algorithms and Applications*, Lecture Notes in Computer Science, Springer-Verlag, New York.

Rousseeuw, P.J. (1985), Multivariate estimation with high breakdown point, pp.283–297 in W. Grossmann, G. Pflug, I. Vincze, and W. Wertz (eds.), *Mathematical Statistics and Applications, Vol. B*, Reidel, Dordrecht.

Rousseeuw, P.J. and Leroy, A.M. (1987), *Robust Regression and Outlier Detection*, Wiley, New York.

Rousseeuw, P.J. and Van Driessen, K. (1999), A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, **41**, 212–223.

Rousseeuw, P.J. and Van Driessen, K. (2006), Computing LTS Regression for Large Data Sets, *Data Mining and Knowledge Discovery*, **12**, 29–45.

Sartenaer, A. (2003), Some recent developments in nonlinear optimization algorithms, *ESAIM: proceedings*, **13**, December, 41–64.