# Hasse Diagrams and the Generalized PAV Algorithm for Monotonic Regression in Several Explanatory Variables

Burdakov, O., Grimvall, A., Hussian, M., and Sysoev, O.

Department of Mathematics, Linköping University, SE-58183 Linköping, Sweden

## Abstract

Monotonic regression is a nonparametric method for estimation of models in which the expected value of a response variable $y$ increases or decreases in all coordinates of a vector of explanatory variables $\boldsymbol{x} = (x_1, \ldots, x_p)$. Here, we examine statistical and computational aspects of our recently proposed generalization of the pool-adjacent-violators (PAV) algorithm from one to several explanatory variables. In particular, we show how the goodness-of-fit and accuracy of obtained solutions can be enhanced by presorting observed data with respect to their level in a Hasse diagram of the partial order of the observed $\boldsymbol{x}$-vectors, and we also demonstrate how these calculations can be carried out to save computer memory and computational time. Monte Carlo simulations illustrate how rapidly the mean square difference between fitted and expected response values tends to zero, and how quickly the mean square residual approaches the true variance of the random error, as the number of observations increases up to $10^4$.

## 1. Introduction

Monotonic response patterns play a fundamental role in the modeling and analysis of a great variety of technical, biological, and economic systems. For example, it is common that the rates of biological, chemical, and physical processes are monotonic functions of factors such as temperature, pressure, and humidity. Moreover, it can be reasonable to assume that health risks are monotonic functions of a set of risk factors, and that the sale of a product will increase with decreasing price and increased advertising.

Monotonic regression (MR) is a nonparametric method used to estimate models of the form

$$y_j = f(x_{1j}, ..., x_{pj}) + \varepsilon_j, \quad j = 1, ..., n$$

where $f$ is increasing or decreasing in each of the coordinates, and $\{\varepsilon_j\}$ stands for independent error terms with mean zero. The special case when the expected response is increasing in all explanatory variables is referred to as isotonic regression (IR). Generally, MR and IR can be formulated as optimization problems in which a loss function is minimized under a set of simple monotonicity constraints (Ayer *et al.*, 1955; Barlow *et al.*, 1972; Best and Chakravarti, 1990). If $M_n = \{(x_{1j}, \ldots., x_{pj}, y_j),\ j = 1, \ldots, n\}$ denotes a set of observed data, we can, for example, determine the fitted values $z_j,\ j = 1, ..., n,$ that minimize

$$\sum_{j=1}^{n} (z_j - y_j)^2$$

under the constraints

$z_i \leq z_j$ if $x_{ki} \leq x_{kj}$ for all $k = 1, \ldots, p$

Despite its simplicity, the practical use of MR/IR has long been hampered by the lack of computational methods that can accommodate several explanatory variables. The most commonly used algorithm, known as the pool-adjacent-violators (PAV) algorithm (Barlow *et al*., 1972; Hanson *et al*., 1973), is appropriate mainly for regression in one discrete or continuous explanatory variable, or two or more discrete variables that are varied at only a few levels (Dykstra and Robertson, 1982; Bril *et al*., 1984). Other computational methods, such as algorithms based on averages of monotonic functions embracing the entire data set, can accommodate an arbitrary set of explanatory variables (Mukarjee, 1988; Mukarjee and Stern, 1994; Strand, 2003). However, the derived solutions can be rather far from optimal in the sense of least squares. Yet other techniques are restricted to specific types of isotonic regression problems (Restrepo and Bovik, 1993; Schell and Singh, 1997; Pardalos and Xue, 1999) or are computationally very time-consuming for large data sets (Maxwell and Muchstadt, 1985; Roundy, 1986; Best and Chakravarti, 1990).

In two recent reports (Burdakov *et al*., 2004; 2006), we presented a new approach to MR/IR problems in which the PAV algorithm was generalized from fully to partially ordered data. This algorithm, which we refer to as the GPAV (generalized pool-adjacent-violators) algorithm, made it feasible to handle regression problems involving thousands of observations of two or more continuous *x*-variables. Specifically, we showed that this algorithm provides close to optimal solutions in the sense of least squares and has complexity $O(n^2)$, where $n$ is the number of observations. Here, we emphasize that the GPAV algorithm treats the observations sequentially and that the order in which data are introduced may influence the obtained result. A solution can

be provided by any topological sort, i.e., any arrangement of the data that is compatible with the partial order of the observed explanatory variables, but the accuracy of that solution depends on the order in which the observations are entered. In this paper, we examine the advantages of using Hasse diagrams of partially ordered data to presort the observations. In addition, we show how the computations involved in such combinations of sorting procedures and the GPAV algorithm can be performed in a manner that simultaneously saves computer memory and makes the computational burden surmountable.

## 2. Computational methods

### The PAV and GPAV algorithms

The PAV algorithm for IR in one explanatory variable (Barlow *et al*., 1972; Hanson *et al*., 1973) assumes that the data $M_n = \{(x_i , y_i), i = 1, \ldots, n\}$ are presorted so that $x_1, \ldots, x_n$ form a nondecreasing sequence. Then the fitted response values form a nondecreasing sequence $z_i$, $i = 1, \ldots, n$, that can be represented as clusters of adjacent indices for which the associated $z$-values are identical. The PAV algorithm identifies these clusters and $z$-values in a recursive procedure in which the observations $y_i$, $i = 1, \ldots, n$, are entered into the calculations one at a time. If $z_1, \ldots, z_r$ denote the solution for $M_r$ that is optimal in the sense of least squares, we form a preliminary solution for $M_{r+1}$ by adding a new cluster consisting of the integer $r$+1 and setting $z_{r+1} = y_{r+1}$. This preliminary solution is subsequently modified into an optimal solution for $M_{r+1}$ by pooling the adjacent preliminary $z$-values that violate the monotonicity constraints; to be more precise, the same value $(y_{r+1} + \ldots + y_{r+1-k})/(k+1)$ is

assigned to each of $z_{r+1-k}$, …, $z_{r+1}$, where $k$ is the smallest integer such that $z_1$, …, $z_{r-k}$ along with new values of $z_{r+1-k}$, …, $z_{r+1}$ form a non-decreasing sequence.

The GPAV algorithm developed by Burdakov and coworkers (2004; 2006) can provide optimal or close to optimal solutions to the IR problem for an arbitrary set of explanatory variables. As in the ordinary PAV algorithm, observations are entered in the calculations one at a time, and monotonicity violators are removed by pooling the $z$-values associated with adjacent clusters of indices.

To enable a stringent definition of the GPAV algorithm, we introduce the partial order

$$\boldsymbol{x}_i \prec \boldsymbol{x}_j \Leftrightarrow x_{ki} \le x_{kj}, \text{ for all } k = 1, ..., p$$

where $\boldsymbol{x}_i = (x_{1i}, \ldots, x_{pi})$ and $\boldsymbol{x}_j = (x_{1j}, \ldots, x_{pj})$ denote vectors of explanatory variables. Furthermore, we sort observed data to ensure for all indices $i \le j$ that either $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are not compatible or $\boldsymbol{x}_i$ is dominated by $\boldsymbol{x}_j$. The resulting order is called topological.

The main idea of the GPAV algorithm is described below (for details see Burdakov *et al*., 2004; 2006). First, we introduce the notation $I = \{i_1, ..., i_m\}$ for a cluster of indices $i_1$, …, $i_m$, the symbol $\omega(I) = m$ for the number of elements in $I$, and the symbol $z(I)$ for the common value of all $z_i$, $i \in I$. Moreover, when two adjacent clusters $I_1$ and $I_2$ are joined to form a new cluster $I_1 \cup I_2$, we compute the associated $z$-value by setting

$$z(I_1 \cup I_2) = (\omega(I_1) \cdot z(I_1) + \omega(I_2) \cdot z(I_2)) / (\omega(I_1) + \omega(I_2))$$

The recursive procedure is started by forming the cluster $I = \{1\}$ and setting $z_1 = y_1$, and then the subsequent steps are defined as follows:

(i) Given that the clusters $I_1$, …, $I_q$ and their associated values $z(I_1)$, …, $z(I_q)$ are a solution for $M_r$, a preliminary solution for $M_{r+1}$ is formed by introducing the cluster $I_{q+1}$ consisting of the integer $r+1$ and setting $z(I_{q+1}) = y_{r+1}$.

(ii) The final solution for $M_{r+1}$ is obtained by sequentially joining $I_{q+1}$ with immediate predecessor clusters until the clusters for which the associated $z$-values violate the monotonicity constraints have been removed. (A cluster $I_j$ is called an immediate predecessor of $I_l$, if an $i \in I_j$ and a $k \in I_l$ exist such that $\boldsymbol{x}_i \prec \boldsymbol{x}_k$ and there is no $m$ (different from $i$ and $k$) such that $\boldsymbol{x}_i \prec \boldsymbol{x}_m \prec \boldsymbol{x}_k$.)

If a cluster has more than one immediate predecessor, the clusters violating the monotonicity are removed sequentially, starting with the cluster representing the strongest violation.

When $p = 1$, the GPAV and PAV algorithms are identical and provide (unique) solutions that are optimal in the least squares sense. When $p > 1$, different orderings of the data may give rise to different solutions to the MR/IR problem under consideration. However, after it has been determined in which order the observations should be entered in the calculations, GPAV produces a unique solution.

## General computational aspects of the MR/IR problem

A conventional mathematical formulation of the MR/IR problem involves matrices of size $n$x$n$. The partial order on the set of $\boldsymbol{x}$-vectors can be summarized in the adjacency matrix

$$P = (p_{ij})$$

where

$$p_{ij} = \begin{cases} 1, & \text{if } \boldsymbol{x}_i \prec \boldsymbol{x}_j \\ 0, & \text{otherwise} \end{cases}$$

If the $\boldsymbol{x}$-vectors have been sorted topologically to match the given partial order and none of these $\boldsymbol{x}$-vectors are identical, $P$ is upper triangular. Furthermore, it is easy to see that

$$Q = P - sgn(P^2)$$

is a binary upper triangular matrix, if the operator *sgn* replaces positive entries with ones, and also that

$$q_{ij} = \begin{cases} 1, & \text{if } \boldsymbol{x}_i \text{ is an immediate predecessor of } \boldsymbol{x}_j \\ 0, & \text{otherwise} \end{cases}$$

In other words, the matrix $Q = (q_{ij})$ summarizes all the nonredundant monotonicity constraints that will be taken into account when the MR/IR problem is solved. It can also be noted that the powers of $Q$ provide information about the size of the longest chain of elements in the partially ordered set of $\boldsymbol{x}$-vectors. If $r$ is the smallest integer for which $Q^r = 0$, then the maximum chain length is $r - 1$.

Our computational algorithms are based on two observations. First, Q is normally a sparse matrix (see the results of the simulation experiments), which implies that it can be stored in the standard compact manner by listing the row and column numbers $(i , j)$ for which $q_{ij} = 1$. Second, this list can be established recursively without storing more than one column of the $P$-matrix that, in general, has a large number of nonzero elements. If the observations are

entered one at a time, the list of row and column numbers ($i$ , $j$) for which $q_{ij}$ = 1 can be extended step by step by recursively extending the list

$$L_k = \{(i, j);\ q_{ij} = 1 \text{ and } j \le k\}$$

for $k$ = 1, …, $n$.

## Hasse diagrams

A Hasse diagram (Davey and Priestly, 2001) can provide a simple picture of a finite partially ordered set (poset) $S$. Each element of $S$ is presented by a vertex in the diagram, and two vertices $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are connected with a line that goes upward from $\boldsymbol{x}_i$ to $\boldsymbol{x}_j$, if $\boldsymbol{x}_i \prec \boldsymbol{x}_j$ and there is no $\boldsymbol{x}_k$ (different from $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$) such that $\boldsymbol{x}_i \prec \boldsymbol{x}_k \prec \boldsymbol{x}_j$ (Figure 1).

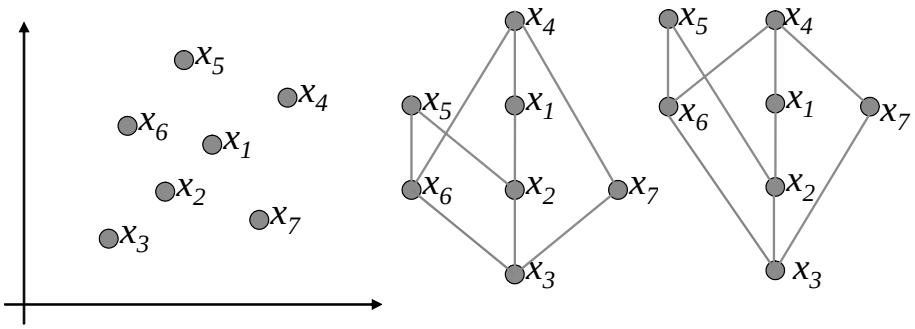


**Figure 1**. Seven points in the Euclidean space $R^2$ and two alternative Hasse diagrams of this partially ordered set.

Any Hasse diagram uniquely determines a partial order, but there are many possible diagrams for specifying a given order. Here, we (partly) standardize our Hasse diagrams by following the concept of level to assign vertical positions to the elements of a poset. In fact, we consider two different definitions of this concept.

**Definition 1**: All minimal elements are assigned level 0, and then the other elements are assigned one level higher than the maximum level of all their predecessors.

**Definition 2**: All maximal elements are assigned a preliminary maximum level $l_{max}$, and the other elements are subsequently assigned one level lower than the minimum level of all their successors. The minimum level is then set to zero by shifting down the entire Hasse diagram vertically.

The diagrams in the middle and on the right in Figure 1 are coherent with definitions 1 and 2, respectively. In both cases, the levels range from 0 to 3.

## Using Hasse diagrams to sort observed data

Our preliminary simulation experiments (Burdakov *et al*., 2004; 2006) indicated that the accuracy of the GPAV solutions can be enhanced, if the observations are entered into the calculations according to an order that takes into account the number of predecessors or successors of each $\boldsymbol{x}$-vector. In the current study, we further examined the impact of presorting observed data by comparing procedures that we refer to as GPAV-R, GPAV-H1, and GPAV-H2, where R stands for random and H for Hasse. All three of these algorithms are

started by employing a quick-sort algorithm to topologically order the observations as follows. If all $x_{1j}$, $j = 1, \ldots, n$, are different, it is sufficient to sort the $\boldsymbol{x}$-vectors with respect to their first coordinate; otherwise, remaining ties can be removed by using additional coordinates. Given a topologically ordered set of observations, we generate all non-redundant monotonicity constraints. GPAV-R involves no further reordering of observed data, whereas the GPAV-H algorithms include another quick-sorting of the $\boldsymbol{x}$-vectors with respect to their levels in a Hasse diagram. H1 and H2 refer to definitions 1 and 2, respectively.

In addition, we introduce an algorithm called GPAV-M(ixed), which is constructed by computing the optimal convex linear combination of the solutions obtained with the other three GPAV algorithms. If the data set to be analyzed is completely ordered, as for $p = 1$, all four algorithms provide identical solutions. In the more general case, when data are partially ordered, the solutions obtained with GPAV-R are at greater risk of being influenced by the initial (random) ordering of the observations than are the more standardized GPAV-H solutions.

## Simulation experiments

### Data generation

Test sets of data were generated according to the equation

$$y_i = \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i, \quad i = 1, \ldots, n$$

where the values of the explanatory variables were drawn from a bivariate normal distribution with mean zero and covariance matrix

$$\boldsymbol{C} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

The error terms ($\varepsilon$) were independent and identically distributed, and light- or heavy-tailed distributions of these error terms were generated according to normal and double- exponential distributions with mean zero and variance one. More detailed information about the different models that were utilized is presented in Table 1. The number of observations was varied between 100 and 10,000, and the number of simulated data sets for each model was 1,000 for the two smaller sample sizes ($n$ = 100 and 1,000) and 100 for the largest sample size ($n$ = 10,000).

**Table 1**. Summary of the models used in the simulation experiments

| Model | Regression coefficients | | Correlation between explanatory variables | Error term distribution |
|---|---|---|---|---|
| | $\beta_1$ | $\beta_2$ | | |
| 1 | 1 | 1 | 0 | Normal |
| 2 | 1 | 1 | 0 | Double-exponential |
| 3 | 1 | 1 | 0.9 | Normal |
| 4 | 1 | 1 | 0.9 | Double-exponential |
| 5 | 0 | 0 | 0 | Normal |
| 6 | 0 | 0 | 0 | Double-exponential |
| 7 | 0 | 0 | 0.9 | Normal |
| 8 | 0 | 0 | 0.9 | Double-exponential |

## Computational burden and memory requirements

Table 2 shows that 10,000 is far below the maximum number of observations for which it is feasible to undertake monotonic regressions on a laptop with moderate capacity. Furthermore, it is worth noting that additional explanatory variables can easily be handled, because these variables are entered in the calculations only through the partial order of the ***x***-vectors. The numerical results presented in this section were obtained by implementing the all algorithms in Visual Basic for Excel on a PC (1.5 GHz) running under Windows XP. A MATLAB implementation of the same algorithms required somewhat more CPU time. Closer examination of the computational burden showed that the identification of all nonredundant constraints was the computationally most time-consuming part of the proposed algorithms. Also, it can be noted that the CPU time was approximately proportional to the square of the sample size, which agrees with previously published theoretical results (Burdakov *et al*., 2006).

**Table 2**. Average CPU time for different parts of the GPAV approach to isotonic regression, using data samples generated according to model 1

| No. of observations | Average CPU time (s) | | | |
|---|---|---|---|---|
| | Quick-sorting observed data | Identification of non-redundant constraints | Running GPAV-R | Running GPAV-H1 and GPAV-H2 |
| 100 | 0.001 | 0.013 | 0.005 | 0.013 |
| 1,000 | 0.012 | 1.61 | 0.41 | 0.76 |
| 10,000 | 0.15 | 198 | 36 | 76 |

The array of pairs $L_k$ is the largest matrix created by the entire algorithm. Theoretically, it is possible to construct sequences of posets for which the number of such constraints grows in proportion to $n^2$. However, typical regression data are usually much more favorable from a computational point of view. Table 3 provides some examples of the number of nonredundant constraints (edges in Hasse diagrams) that are needed to define the partial order of samples from bivariate normal distributions.

**Table 3.** Average number of edges in Hasse diagrams of samples from bivariate normal distributions

| No. of observations | Average number of edges | |
|---|---|---|
| | Independent explanatory variables ($\rho = 0$) | Strongly correlated explanatory variables ($\rho = 0.9$) |
| 100 | 324 | 275 |
| 1,000 | 5,501 | 4,742 |
| 10,000 | 77,737 | 69,416 |

## Goodness-of-fit and accuracy

The application of different implementations of the GPAV algorithm to simulated data illustrates the advantages of using levels defined by Hasse diagrams to determine the order in which observations should be entered in the GPAV algorithm. In all simulations in which the expected response increased

in relation to the explanatory variables (models 1–4), this sorting reduced the mean square residual

$$MSR = \sum_i (z_i - y_i)^2 / n$$

of the obtained solutions (see Table 4). For the largest samples ($n$ = 10,000), the improvement was dramatic. For example, when data were generated according to model 1, the mean square residuals for the GPAV-R, GPAV-H1, and GPAV-H2 algorithms were 1.526, 0.906 and 0.906, respectively. The results regarding the accuracy of the fitted response function were even more convincing. While the mean square error

$$MSE = \sum_i (z_i - E(y_i \mid \boldsymbol{x}_i))^2 / n$$

for the two GPAV-H algorithms tended to zero with increasing sample size, the GPAV-R algorithm failed to produce consistent estimates of the true response surface $E(y \mid \boldsymbol{x})$.

The results in Table 4 also show that, on average, the two GPAV-H algorithms performed almost identically, and combining the two GPAV-Hasse solutions (and the GPAV-R solution) into an optimal convex linear expression had relatively little effect on the goodness-of-fit. However, some of the results obtained for large data sets indicate that there exist cases in which merging different GPAV solutions in an optimal linear combination may be worth the extra computational effort. For instance, for model 1 and sample size 10,000 it can be seen that the mean square differences between fitted and expected response values were 0.026, 0.039, and 0.040 for GPAV-M, GPAV-H1, and GPAV-H2, respectively.

Due to the flexibility (high effective dimension) of MR models, the unadjusted mean square residual can strongly underestimate the true variance of the error terms. In part, this can be attributed to the fact that $z_i - y_i = 0$ if $\boldsymbol{x}_i$ is a minimal element in the poset of $\boldsymbol{x}$-vectors, and the associated error term $\varepsilon_i$ has a large negative value. Similarly, $z_i - y_i = 0$ if $\boldsymbol{x}_i$ is a maximal element, and the associated error term $\varepsilon_i$ is large and positive. Table 4 illustrates the over-fitting to data generated by models in which there was a strong (monotonic) relationship between the response variable and the explanatory variables. Although the true variance of the error terms was equal to 1 in all investigated models of this type (models 1–4), the expected value of the mean square residual was invariably less than 1. In fact, it was only about 0.4 when the sample size was moderately large ($n$ = 100) and the explanatory variables were independent. The over-fitting was less pronounced for the largest data sets ($n$ = 10,000). This problem was also reduced when the correlation of the explanatory variables was increased, because we then approached the case of a single explanatory variable.

We have already noted that the mean square difference between fitted and expected response values decreased steadily to zero with increasing sample size ($n$). A tenfold increase in sample size made this expression about three times lower (see Table 4). Closer examination of the fitted response values indicated that this rate of convergence would not have been dramatically improved even if we had had access to an algorithm that provided exact solutions to the MR/IR problem. The over-fitting to observations representing minimal and maximal $\boldsymbol{x}$-vectors can not be avoided, which implies that the fitted values may differ substantially from the true expected responses in such points.

The simulation results presented in Table 5 show that all the GPAV algorithms performed satisfactorily for data sets generated by models in which the expected response was constant, (i.e., did not depend on the explanatory variables). This was expected, because, for such data, a solution consisting of a single cluster gives the most accurate estimates of $E(y_i \mid \boldsymbol{x}_i)$, and hence nothing can be gained by using different GPAV algorithms to optimize the formation of the clusters.

Table 6 presents additional details regarding the effective dimension of MR models. We let *D* denote the number of clusters in the GPAV solution and used the correction factor $c = 1.5$ proposed by Meyer and Woodroofe (2000) to compute the following adjusted estimates of the residual variance:

$$MSR_{Adj} = MSR\frac{n}{n-1.5D} = \sum_i (z_i - y_i)^2 /(n-1.5D)$$

As can be seen, these adjusted estimates of the residual variance are still biased.

**Table 4.** Goodness-of-fit and accuracy of the solutions obtained for different GPAV-algorithms when the data sets were generated by models with a strong relationship between the response variable and the explanatory variables ($\beta_1 = \beta_2 = 1$). Standard errors of the estimated means are given within brackets.

| Model | *n* | Mean square residual | | | | Mean square error | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | GPAV-R | GPAV-H1 | GPAV-H2 | GPAV-M | GPAV-R | GPAV-H1 | GPAV-H2 | GPAV-M |
| 1 | 100 | 0.434 (0.003) | 0.412 (0.003) | 0.411 (0.003) | 0.406 (0.003) | 0.360 (0.002) | 0.344 (0.002) | 0.343 (0.002) | 0.341 (0.002) |
| | 1,000 | 0.952 (0.002) | 0.724 (0.001) | 0.722 (0.001) | 0.715 (0.001) | 0.280 (0.001) | 0.109 (0.0003) | 0.108 (0.0003) | 0.101 (0.0003) |
| | 10,000 | 1.526 (0.003) | 0.906 (0.001) | 0.906 (0.001) | 0.893 (0.001) | 0.600 (0.002) | 0.039 (0.0003) | 0.040 (0.0003) | 0.026 (0.0002) |
| 2 | 100 | 0.456 (0.005) | 0.432 (0.005) | 0.431 (0.005) | 0.426 (0.005) | 0.344 (0.003) | 0.327 (0.003) | 0.325 (0.003) | 0.324 (0.003) |
| | 1,000 | 0.973 (0.003) | 0.739 (0.002) | 0.736 (0.002) | 0.730 (0.002) | 0.283 (0.001) | 0.104 (0.0004) | 0.103 (0.0004) | 0.097 (0.0004) |
| | 10,000 | 1.565 (0.039) | 0.905 (0.021) | 0.905 (0.021) | 0.893 (0.020) | 0.636 (0.029) | 0.037 (0.003) | 0.037 (0.003) | 0.026 (0.002) |
| 3 | 100 | 0.542 (0.003) | 0.531 (0.003) | 0.529 (0.003) | 0.525 (0.003) | 0.236 (0.002) | 0.230 (0.002) | 0.229 (0.002) | 0.228 (0.002) |
| | 1,000 | 0.874 (0.001) | 0.797 (0.001) | 0.795 (0.001) | 0.790 (0.001) | 0.110 (0.0003) | 0.064 (0.0002) | 0.064 (0.0002) | 0.060 (0.0002) |
| | 10,000 | 1.090 (0.001) | 0.929 (0.001) | 0.929 (0.001) | 0.922 (0.001) | 0.145 (0.0003) | 0.020 (0.0001) | 0.020 (0.0001) | 0.013 (0.0001) |
| 4 | 100 | 0.556 (0.005) | 0.547 (0.005) | 0.546 (0.005) | 0.542 (0.005) | 0.224 (0.002) | 0.218 (0.002) | 0.217 (0.002) | 0.216 (0.002) |
| | 1,000 | 0.875 (0.002) | 0.807 (0.002) | 0.806 (0.002) | 0.801 (0.002) | 0.101 (0.0003) | 0.062 (0.0003) | 0.061 (0.0003) | 0.057 (0.0003) |
| | 10,000 | 1.082 (0.002) | 0.927 (0.002) | 0.928 (0.002) | 0.921 (0.002) | 0.140 (0.0004) | 0.020 (0.0001) | 0.020 (0.0002) | 0.014 (0.0001) |

**Table 5.** Goodness-of-fit and accuracy of the solutions obtained for different GPAV-algorithms when the data sets were generated by models in which the response variable did not depend on the explanatory variables ($\beta_1 = \beta_2 = 0$). Standard errors of the estimated means are given within brackets.

| Model | *n* | Mean square residual | | | | Mean square error | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | GPAV-R | GPAV-H1 | GPAV-H2 | GPAV-M | GPAV-R | GPAV-H1 | GPAV-H2 | GPAV-M |
| 5 | 100 | 0.875 (0.004) | 0.857 (0.004) | 0.856 (0.004) | 0.852 (0.004) | 0.128 (0.002) | 0.146 (0.002) | 0.147 (0.002) | 0.148 (0.002) |
| | 1,000 | 0.981 (0.001) | 0.970 (0.001) | 0.970 (0.001) | 0.969 (0.001) | 0.020 (0.0003) | 0.031 (0.0003) | 0.031 (0.0003) | 0.031 (0.0003) |
| | 10,000 | 0.998 (0.001) | 0.995 (0.001) | 0.995 (0.001) | 0.995 (0.001) | 0.003 (0.0001) | 0.005 (0.0001) | 0.005 (0.0001) | 0.005 (0.0001) |
| 6 | 100 | 0.867 (0.007) | 0.851 (0.007) | 0.851 (0.007) | 0.847 (0.006) | 0.127 (0.002) | 0.143 (0.002) | 0.143 (0.002) | 0.143 (0.002) |
| | 1,000 | 0.982 (0.002) | 0.972 (0.002) | 0.972 (0.002) | 0.971 (0.002) | 0.019 (0.0003) | 0.029 (0.0004) | 0.029 (0.0004) | 0.029 (0.003) |
| | 10,000 | 0.994 (0.002) | 0.992 (0.002) | 0.992 (0.002) | 0.991 (0.002) | 0.003 (0.0001) | 0.005 (0.0001) | 0.005 (0.0001) | 0.005 (0.0001) |
| 7 | 100 | 0.928 (0.004) | 0.926 (0.004) | 0.926 (0.004) | 0.925 (0.004) | 0.075 (0.001) | 0.077 (0.001) | 0.077 (0.001) | 0.078 (0.001) |
| | 1,000 | 0.989 (0.001) | 0.987 (0.001) | 0.987 (0.001) | 0.987 (0.001) | 0.012 (0.0002) | 0.014 (0.0002) | 0.014 (0.0002) | 0.014 (0.0002) |
| | 10,000 | 0.999 (0.001) | 0.998 (0.001) | 0.998 (0.001) | 0.998 (0.001) | 0.002 (0.0001) | 0.002 (0.0001) | 0.002 (0.0001) | 0.002 (0.0001) |
| 8 | 100 | 0.921 (0.007) | 0.920 (0.007) | 0.920 (0.007) | 0.919 (0.007) | 0.072 (0.002) | 0.074 (0.002) | 0.074 (0.002) | 0.075 (0.002) |
| | 1,000 | 0.990 (0.002) | 0.988 (0.002) | 0.988 (0.002) | 0.988 (0.002) | 0.011 (0.0002) | 0.013 (0.0002) | 0.013 (0.0002) | 0.013 (0.0002) |
| | 10,000 | 0.995 (0.002) | 0.995 (0.002) | 0.995 (0.002) | 0.995 (0.002) | 0.002 (0.0001) | 0.002 (0.0001) | 0.002 (0.0001) | 0.002 (0.0001) |

**Table 6.** Average number of clusters and adjusted mean square residuals for the solutions obtained for different GPAV algorithms when the data sets were generated by bivariate regression models in which the error terms had a standard normal distribution and $\beta_1 = \beta_2 = 1$ or $\beta_1 = \beta_2 = 0$

| Model | *n* | No. of clusters | | | Adjusted mean square residual | | |
|---|---|---|---|---|---|---|---|
| | | GPAV-R | GPAV-H1 | GPAV-H2 | GPAV-R | GPAV-H1 | GPAV-H2 |
| 1 | 100 | 39 | 42 | 42 | 1.051 | 1.139 | 1.142 |
| | 1,000 | 77 | 138 | 139 | 1.077 | 0.913 | 0.914 |
| | 10,000 | 108 | 340 | 336 | 1.551 | 0.954 | 0.954 |
| 3 | 100 | 29 | 31 | 31 | 0.967 | 0.997 | 1.002 |
| | 1,000 | 65 | 96 | 96 | 0.969 | 0.931 | 0.929 |
| | 10,000 | 108 | 250 | 240 | 1.108 | 0.966 | 0.964 |
| 5 | 100 | 10 | 12 | 12 | 1.030 | 1.047 | 1.049 |
| | 1,000 | 15 | 21 | 22 | 1.003 | 1.003 | 1.003 |
| | 10,000 | 19 | 34 | 34 | 1.001 | 1.001 | 1.001 |
| 7 | 100 | 7 | 7 | 7 | 1.029 | 1.033 | 1.033 |
| | 1,000 | 10 | 12 | 11 | 1.004 | 1.005 | 1.004 |
| | 10,000 | 13 | 17 | 17 | 1.001 | 1.001 | 1.001 |

## Optimal weighting of solutions

We have already noted that the two GPAV-Hasse algorithms normally produced similar solutions and that any of these algorithms is superior to the

GPAV-R algorithm. This was further confirmed by recording the weights of the GPAV-R, GPAV-H1 and GPAV-H2 solutions in the optimal linear combination GPAV-M. The results given in Table 7 show that, on average, the GPAV-R solution was assigned a small weight, whereas the two GPAV-Hasse solutions were given approximately the same weight.

**Table 7.** Optimal weighting of the solutions obtained with the algorithms GPAV-R, GPAV-H1, and GPAV-H2. Standard errors of the estimated means are given within brackets.

| Model | No. of observations | Weights | | |
|---|---|---|---|---|
| | | GPAV-R | GPAV-H1 | GPAV-H2 |
| 1 | 100 | 0.085 (0.005) | 0.422 (0.010) | 0.494 (0.011) |
| 1 | 1,000 | 0.001 (0.000) | 0.469 (0.004) | 0.531 (0.004) |
| 1 | 10,000 | 0.000 (0.000) | 0.504 (0.005) | 0.496 (0.005) |
| 3 | 100 | 0.125 (0.006) | 0.360 (0.010) | 0.515 (0.010) |
| 3 | 1,000 | 0.001 (0.0001) | 0.466 (0.004) | 0.534 (0.004) |
| 3 | 10,000 | 0.000 (0.0000) | 0.501 (0.005) | 0.499 (0.005) |

## Discussion

We recently demonstrated that the GPAV algorithm can provide optimal or close to optimal solutions to MR/IR problems, and it also outperforms

alternative techniques, such as simple averaging (Burdakov *et al.*, 2004; 2006). Our present findings show that the calculations can be carried out with algorithms that combine a reasonable computational burden with very modest computer memory requirements. Furthermore, our simulation results demonstrate that the performance of the GPAV algorithm to a large extent depends on the order in which the observations are entered into the calculations. In particular, we found that algorithms with observations entered according to the Hasse diagram level of their $\boldsymbol{x}$-vectors outperform an algorithm (Hasse-R) in which no attempts are made to select a particular ordering among all those that are consistent with the given partial order.

Forming linear combinations of two or more solutions to an IR/MR problem can improve the goodness-of-fit and accuracy of the fitted values. This was especially apparent in some of the simulations involving large samples ($n$ = 10,000). However, in general, the the individual GPAV-Hasse solutions are almost as good as the optimal linear combination of GPAV-R, GPAV-H1 and GPAV-H2. Moreover, there are only small differences in the performance of GPAV-H1 and GPAV-H2.

When parametric or nonparametric regression models are fitted to observed data, it is customary to determine the (approximate) degrees of freedom and to use an adjusted mean square residual to estimate the variance of the error terms in the original observations (Mallows, 1973; 1995; Hastie *et al.*, 2001). Such adjustments are based on the assumption that a parametric form of the model is known or that the true response surface is smooth, thus they are not directly applicable to MR. Nonetheless, the results presented in Tables 4 and 5 provide rules of thumb regarding the magnitude of the over-fitting problem when MR is used to analyze small or moderately large data sets.

As early as the 1970s, Hanson and coworkers (1973) discussed the consistency of least squares solutions to the MR problem. These authors presented sufficient conditions for the case of a single explanatory variable, whereas the results obtained for $p > 1$ were very limited. The consistency of the GPAV estimates is even more intricate, because the obtained solutions need not be optimal in the least squares sense. Notwithstanding, our simulation results are convincing. Regardless of the sample size and error term, the mean square difference between fitted and true expected response values decreased steadily to zero when the sample size was increased and any of the GPAV-H algorithms was employed to determine the fitted response values.

Taken together, the results reported in this article demonstrate that the GPAV approach to MR in two or more variables is now ready for large-scale application.

## Acknowledgements

The authors are grateful for financial support from the Swedish Research Council and the Swedish Environmental Protection Agency.

## References

Ayer, M., Brunk, H.D., Ewing, G.M., Reid, W.T., and Silverman, E. (1955). An empirical distribution function for sampling with incomplete information. *The Annals of Mathematical Statistics*, **26**:641-647.

Barlow, R.E., Bartholomew, D.J., Bremner, J.M., and Brunk, H.D. (1972). *Statistical inference under order restrictions*. New York: Wiley.

Best, M.J. and Chakravarti, N. (1990). Active set algorithms for isotonic regression: a unifying framework. *Mathematical Programming*, **47**:425-439.

Bril, G., Dykstra, R., Pillers, C., and Robertson, T. (1984). Algorithm AS 206, isotonic regression in two independent variables. *Applied Statistics*, **33**:352-357.

Burdakov, O., Grimvall, A., and Hussian, M. (2004). A generalised PAV algorithm for monotonic regression in several variables. In: Antoch, J. (ed.) *COMPSTAT, Proceedings of the 16th Symposium in Computational Statistics held in Prague*. Heidelberg, New York: Physica-Verlag (Springer).

Burdakov O., Sysoev O., Grimvall A., and Hussian M. (2006). An $O(n^2)$ algorithm for isotonic regression. In: Di Pillo, G. and Roma, M. (eds) *Large Scale Nonlinear Optimization. Series: Nonconvex Optimization and Its Applications*, Springer-Verlag, **83**, pp. 25-33.

Davey, B.A., and Priestly, H.A. (2002*). Introduction to lattices and order*. Cambridge: Cambridge University Press.

De Simone, V., Marino, M., and Toraldo, G. (2001). In: Floudas, C.A. and Pardalos, P.M. (eds) *Encyclopedia of optimization*. Dordrecht: Kluwer Academic Publishers.

Dykstra, R. and Robertson, T. (1982). An algorithm for isotonic regression for two or more independent variables. *The Annals of Statistics*, **10**:708-716.

Hanson, D.L., Pledger, G., and Wright, F.T. (1973). On consistency in monotonic regression. *The Annals of Statistics*, **1**:401-421.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The elements of statistical learning*. New York: Springer.

Mallows, C.L. (1973). Some comments on $C_p$. *Technometrics*, **15**:661-675.

Mallows, C.L. (1995). More comments on $C_p$. *Technometrics*, **37**:362-372.

Maxwell, W.L. and Muchstadt, J.A. (1985). Establishing consistent and realistic reorder intervals in production-distribution systems. *Operations Research*, **33**: 1316-1341.

Mayer, M. and Woodroofe, M. (2000). On the degrees of freedom in shape-restricted regression. *The Annals of Statistics*, **28**, 1083-1104.

Mukarjee, H. (1988). Monotone nonparametric regression. *The Annals of Statistics*, **16**:741-750.

Mukarjee, H. and Stern, H. (1994). Feasible nonparametric estimation of multiargument monotone functions. *Journal of the American Statistical Association*, **425**:77-80.

Pardalos, P.M. and Xue, G. (1999). Algorithms for a class of isotonic regression problems. *Algorithmica*, **23**:211-222.

Restrepo, A. and Bovik, A.C. (1993). Locally monotonic regression. *IEEE Transactions on Signal Processing*, **41**:2796-2810.

Roundy, R. (1986). A 98% effective lot-sizing rule for a multiproduct multistage production/inventory system. *Mathematics of Operations Research*, **11**: 699-727.

Schell, M.J. and Singh, B. (1997). The reduced monotonic regression method. *Journal of the American Statistical Association*, **92**:128-135.

Strand, M. (2003). Comparisons of methods for monotone nonparametric multiple regression. *Communications in statistics, simulation and computations*, **32**:165-178.