Overview of My Papers on Shape-Constrained Regression

Quentin F. Stout

qstout@umich.edu

University of Michigan
Ann Arbor, MI

This is a quick overview of my papers on shape-constrained regression. I decided it was the easiest way to explain how several of them are interrelated. The papers involve monotonic (isotonic) regression, unimodal regression, and step functions. The basic idea is to have a best approximation of real-valued data, where the data is at the vertices of some directed acyclic graph (dag). A dag $G(V, E)$ with defines a partial order (poset) over the vertices $V = (v_1, \dots v_n)$, where $v_i \prec v_j$, $v_i, v_j \in V$ if and only if there is a path from v_i to v_j. The partial orderings considered are linear, tree, d-dimensional grids, points in d-dimensional space with component-wise ordering, and arbitrary orderings.

A real-valued function $\vec{z} = (z_1 \dots z_n)$ on V is isotonic if whenever $v_i \prec v_j$, then $z_i \leq z_j$, i.e., it is a weakly order-preserving map from G to $\Re$. In most areas of mathematics this is known as a monotonic function, but for some reason in this context it is usually called isotonic. By *data* $(\vec{y}, \vec{w})$ on G we mean there is a weighted value (y_i, w_i) at vertex v_i, $1 \leq i \leq n$, where y_i is an arbitrary real number and w_i, the weight, is ≥ 0. By unweighted data we mean $w_i = 1$ for all i.

For $1 \leq p \leq \infty$, or $p = 0$, given data $(\vec{y}, \vec{w})$ on dag $G(V, E)$, an L_p *isotonic regression* of the data is an isotonic function $\vec{z}$ over V that minimizes

$$
\begin{array}{ll}
\left(\sum_{i=1}^{n} w_i |y_i - z_i|^p\right)^{1/p} & 1 \leq p < \infty \\
\max_{i=1}^{n} w_i |y_i - z_i| & p = \infty \\
\sum_{i=1}^{n} w_i \cdot (y_i \neq z_i) & p = 0
\end{array}
$$

among all isotonic functions.

Fast algorithms for finding isotonic regressions depend on the dag and metric, and whether the data is weighted or not. Because of this there are numerous relevant papers, and I try to keep track of the fastest (in terms of O-notational analysis of worst case) in "Fastest Known Isotonic Regression Algorithms". However, the results in 9 have greatly changed the fastest algorithms, and I haven't yet updated the list. Unimodal functions are somewhat simpler in that they are usually only defined on linear orders and are an isotonic piece followed by an anti-isotonic piece (crudely speaking, they go up then down). Unimodal functions can also be defined on undirected trees, trying to determine an optimal root. See 5). For step functions I only consider linear orders, generating a sequence of some fixed number of steps on which they are constant.

Points in d-dimensional space with component-wise ordering are an important class of orderings, but do not directly specify a dag. This is addressed in 3), giving a representation, a *rendezvous dag*, of size $\Theta(n \log^d n)$, and a *reduced rendezvous dag* of size $\Theta(n \log^{d-1} n)$. These dags add vertices to V, but greatly reduce the worst-case number of edges. This is important because the time of many of the algorithms depends on the number of edges as well as number of vertices. The extra vertices are *Steiner vertices*. For vertices $v, w \in V$, $v \prec w$ iff there is a path of length 2 from v to w in the rendezvous dag, where the intermediate point is a Steiner vertex. In the reduced rendezvous dag the path may be longer.

When the points form a grid the rendezvous dags aren't needed because the number of grid edges are linear (for fixed d) in n. For grids, there is an important difference between $d = 2$ and $d > 2$. For L_1 and L_2 the former makes dynamic programming approaches possible that are impossible in higher dimensions. The dynamic programming for 2-d grids can be extended to arbitrary points in 2-dimensions (see 4).

Finally, some of the algorithms below, and those of others, will yield faster algorithms if advances in flow algorithms or matrix multiplication occur. In some cases these or close relatives are the primary determiners of worst-case time, especially in algorithms for arbitrary dags. For example, 8) uses the algorithm for arbitrary dags to show that isotonic L_0 regression on multidimensional data can be accomplished in $o(n^{3/2})$ time. This is based on the rendezvous dags from 3), having $\tilde{\Theta}(n)$ edges, with a recent flow algorithm (Gao-Liu-Peng) taking $\tilde{\Theta}\left(w^{\frac{3}{2}-\frac{1}{328}}\right)$ time on a graph of w edges. Until the 2021 Gao-Liu-Peng algorithm the best that was accomplished was $\tilde{\Theta}(n^{3/2})$, but by just citing it the result improved slightly. These improvements were then used in 9 to give improved algorithms for general L_p, $1 \leq p < \infty$.

Unimodal Regression

1. Stout, QF (2008), "Unimodal regression via prefix isotonic regression", *Computational Stat. and Data Analysis* 53, pp. 289–297, gives basic algorithms for unimodal regression. They utilize the computations for an isotonic regression on $1 \ldots i$ to help determine an isotonic regression on $1 \ldots i+1$. Optimal algorithms are given for weighted and unweighted L_1 and L_2, and unweighted L_∞. A much more complicated algorithm for weighted L_∞ appears in 5). The UniIsoRegression package in CRAN contains implementations of several of these algorithms.

2. Paper 5 includes algorithms for weighted and unweighted L_∞ unimodal regression on undirected trees. They are quite different in that they determine the mode (root) without first computing a sequence of prefix regressions.

Isotonic Regression

3. Stout, QF (2015), "Isotonic regression for multiple independent variables", *Algorithmica* 71, pp. 450–470. Except for 6), all of the algorithms below for multidimensional data, $d \geq 3$, depend on this to give an efficient dag for the implied ordering. In this paper algorithms are given for L_1 and L_2. L_∞ is not unique, so algorithms are given for several options, including strict L_∞ (see 7)). Most of this was originally posted on the web in 2008, see 5). It wasn't until years later that I learned that others were looking at the same problem and that the rendezvous graph is a Steiner 2-transitive-closure spanner.

4. Stout, QF (2013), "Isotonic regression via partitioning", *Algorithmica* 66, pp. 93–112. This has algorithms for L_1 isotonic regression for a variety of dags, creating the regression via a sequence of binary partitions. The technique is also applied to: approximations for L_p regressions, $1 \leq p < \infty$; exact regression for $p = 2, 3, 4, 5$; and regressions with multiple values per vertex. The UniIsoRegression package in CRAN contains implementations of L_1 and L_2 algorithms for 2-d grids.

5. Stout, QF (2018), "Weighted L_∞ isotonic regression", *J. Computer Sys. and Sci.* 91, pp. 69–81. This is a major revision of the original version that was posted on the web in 2008. Some of the material in that paper was moved to 3) since the original paper was far too long and the multidimensional results extend far beyond L_∞. L_∞ regression is not unique, and this paper considers several variants, one related to 1). See also 7). It also introduces *river regression*, a regression on rooted trees corresponding to some classification and taxonomy problems.

6. Stout, QF (2015), "L_∞ isotonic regression for linear, multidimensional, and tree orders", arXiv 1507:02226. This gives algorithms that use a new non-constructive feasibility test to determine if there is an L_∞ regression with specified error. For linear, tree, and multidimensional grids they are optimal. The paper is unusual in that the algorithms for multidimensional data in arbitrary positions replace the explicit dags in 3) with repeated sorting, taking only linear space.

7. Stout, QF (2012), "Strict L_∞ isotonic regression", *J. Optimization Theory and App.* 152, pp. 121–135. L_∞ isotonic regression is not unique, and this paper introduces a natural option, namely $\lim_{p\to\infty}$ of the unique L_p regression. If you take the errors at the vertices and sort them in decreasing order then the strict regression is the first, in lexical order, of this n-element list. This is closely related to strong L_0 regression 8), and it might have been better to call this strong L_∞ instead of strict. The version of the paper linked to here has added appendix material that did not appear in the journal version. It gives a different way of showing that a regression is the strict regression, and a faster (in practice, not O-notation) way of finding the strict regression.

8. Stout, QF (2021), "L_0 isotonic regression with secondary objectives", arXiv:2106.00279v2. L_0 isotonic regression is defined when the data is linearly ordered labels, not just real numbers. It is not unique, so this paper adds secondary criteria, such as minimizing L_2 error when the labels are real numbers. It also examines regularized L_p isotonic regression, minimizing $||\cdot||_p + \alpha||\cdot||_0$ for a fixed α, and L_0 regression on vertices in multidimensional space. The paper introduces strong L_0 regression, which applies in the general case when only the ordering of labels is used. If you take the regression error at each vertex, where the error is defined as the number of labels between the regression value and original value, and sort these in increasing order, then the strong L_0 regression is the first in lexical order of this n-element list (there may be ties). Strict L_∞ regression (7) minimizes large errors, while strong L_0 maximizes small ones.

9. Stout, QF (2021), "L_p isotonic regression using an L_0 approach", arXiv:2107.00251v2. Significant advances in maximum flow algorithms have changed the relative performance of various approaches to isotonic regression. If the transitive closure is given then the standard approach used for L_0 (Hamming distance) isotonic regression (finding anti-chains in the transitive closure of the violator dag), combined with new flow algorithms, gives an $\{0,1\}$-valued L_1 isotonic regression algorithm taking $\tilde{\Theta}(n^2)$ time on a graph of n vertices. Then partitioning is used to find an arbitrary real-valued L_1 isotonic regression in the same time (with the $\tilde{\Theta}$ hiding an addition log factor). The previous fastest was $\Theta(n^3)$. For points in d-dimensional space with coordinate-wise ordering, $d \geq 3$, L_1 regression can be found in $o\big(n^{1.5}\big)$ time, improving on the previous best of $\tilde{\Theta}(n^2 \log^d n)$. Similar results are obtained for L_p approximations, $1 < p < \infty$, and for exact L_2 regression when the values and weights are restricted.

Step Functions

10. Stout, QF (2014), "An algorithm for L_∞ approximation by step functions", arXiv 1412.2379. Given a fixed number of steps, this considers both steps in isotonic order and arbitrary steps. It also solves the k-center problem for 1-dimensional data and the variable width histogram problem. It uses bounded error envelopes instead of the unbounded ones used in most L_∞ algorithms.

11. Hardwick, JP and Stout, QF (2014), "Optimal reduced isotonic regression", arXiv 1412.2844. The problems considered include Fisher's "unrestricted maximum homogeneity" approximation and Ioannidis' optimal variable-width "serial histogram" problem (also known as "v-optimal histograms"). The algorithms also determine optimal k-means clustering of 1-dimensional data. This paper only considers L_2, though an earlier version also examined L_1.