Journal of Physics: Conference Series **699** (2016) 012016 doi:10.1088/1742-6596/699/1/012016

Exact Algorithms for Isotonic Regression and Related

Yao-Liang Yu and Eric P. Xing

Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA, 15213, USA

E-mail: {yaoliang, epxing}@cs.cmu.edu

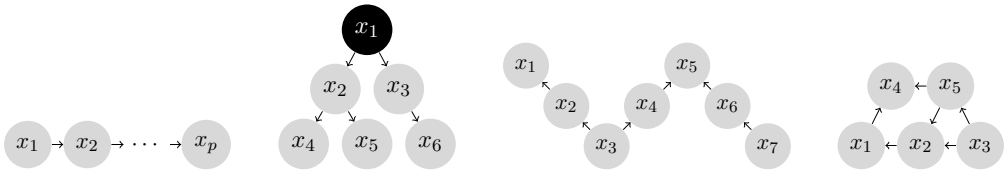
Abstract. Statistical estimation under order restrictions, also known as isotonic regression, has been extensively studied, with many important practical applications. The same order restrictions also appear *implicitly* in sparse estimation, where intuitively we should shrink variables starting from smaller ones. Inspired by the achievements in both fields, we first propose the GPAV algorithm for solving problems with order restrictions. We study its theoretical properties, present an online linear time implementation, and prove a converse theorem to pinpoint the exact correctness condition. When specialized to the proximity operator of an order restricted regularization function, GPAV recovers, as special cases, many existing algorithms, and also leads to many new extensions that even involve nonconvex functions.

1. Introduction

As (data) scientists, how do we argue global warming is not bogus? One established way is to perform statistical hypothesis testing using say the likelihood ratio: Given a collection of everyday temperatures over a certain time period, we can estimate the yearly mean temperatures under the null hypothesis that global warming is not happening (*i.e.*, constant mean), and also the mean temperatures under the alternative hypothesis that global warming is indeed happening (*i.e.*, monotonically increasing mean). Adopting an appropriate statistical temperature model, the latter would require maximizing the likelihood function subject to the *isotonic* constraint, *i.e.*, the estimated yearly mean temperatures should be monotonically increasing. This problem is widely known as isotonic regression and has been extensively studied in the 1970s, culminating in the excellent books [1, 2], with many illuminating results and applications. See §2 for some examples.

More generally, statistical estimation under shape constraints (such as monotonicity, convexity) has been an important topic in statistics since the seminal works [3, 4]. A large part of the theory focuses on studying the asymptotic distribution of estimators (parametric or nonparametric) under shape constraint, which are generally different from those obtained without shape constraint and are rather difficult to derive. An excellent summary in this direction is the recent book [5]. Computationally, how to obtain the estimator under shape constraint has also attracted lots of attention. The pool-adjacent-violators (PAV) algorithm [6], essentially the Euclidean projection onto the isotonic cone, is one of the early achievements in this field and is still widely used today.

Order constraints also implicitly appear in sparse estimation problems. If the true estimator is sparse or can be well approximated by a sparse vector, it would be beneficial to "sparsify" our estimator, effectively reducing model complexity hence potentially improving generalization. "Sparsification" is usually executed through the *proximity operator* of carefully designed sparse



(a) Chain. (b) Rooted tree. (c) "Soft" chain. (d) DAG.

Figure 1: Partial orders induced by a chain, a rooted tree, a "soft" chain, and a DAG.

regularizers, among which the ℓ_1 norm, along with its soft-shrinkage operator, is perhaps the most famous. Intuitively, if we were to shrink a subset of variables to zero, we should start with the smaller ones, since larger estimates are more likely to be genuinely nonzero. Indeed, almost all sparse regularizers that we are aware of obey this principle and *implicitly* enforce an order restriction on the output. More recent sparse regularizers, such as the k-support norm [7], OSCAR [8], and the sorted ℓ_1 norm [9] even made the order restriction explicit. Computationally, the order constraint complicates the computation of the proximity operator and is addressed case by case in many recent works [7, 9–13]. The connection to the field of isotonic regression, in particular, the PAV algorithm, is not always explored.

The main goal of this work is to provide a unified treatment of the various computational algorithms involving order constraints. Based on a surprisingly simple yet general result (Theorem 1 below) we develop in §3 the generalized pool-adjacent-violators (GPAV) algorithm, which is a strict generalization of PAV, hence making the connection to the field of isotonic regression explicit. We also prove a converse theorem that pinpoints exactly when GPAV is provably correct, which, to our best knowledge, is the first result of its kind. An online implementation is provided in §4 to bring down the time complexity from quadratic to linear. Then, in §5 we specialize the online GPAV to the problem of computing the proximity operator of functions involving order restrictions. One nice feature of online GPAV is that it can handle k-piecewise convex functions (that are generally nonconvex). In particular, it also computes the proximity operator for all 2-convex functions *exactly, globally* in *linear time*.

In this work we focus exclusively on the linear model (namely the functions g_i in Equation (1) below are univariate). However, extensions to additive models (where g_i can be multivariate functions of a specific form) can be pursued as in the work of Bacchetti [14] and Fang and Meinshausen [15].

2. Problem definition

We are interested in solving the following problem:

$$\min_{\mathbf{x}\in\mathcal{K}} \sum_{i=1}^{p} g_i(x_i), \tag{1}$$

where $g_i : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ are *univariate* functions and $\mathcal{K}$ is an isotonic *cone* induced by some partial order $\preceq$ on the set $\{1, \ldots, p\}$:

$$\mathcal{K} := \{\mathbf{x} \in \mathbb{R}^p : x_i \le x_j \text{ for all comparable pairs } i \preceq j\}. \tag{2}$$

Equivalently, since any partial order can be represented as a directed acyclic graph (DAG), we can define $\mathcal{K}$ through a DAG too. In particular, the following special case where $\mathcal{K}$ is induced by a total order (equivalently a chain graph) motivated the development historically:

$$\mathcal{K}_c := \{\mathbf{x} \in \mathbb{R}^p : x_1 \ge x_2 \ge \cdots \ge x_p\}. \tag{3}$$

There are of course many other possibilities, see Figure 1. We will address the first two cases in Figure 1 and leave the other two cases for future work.

The order constraint modeled by the isotonic cone in (2) has many practical applications. Let us mention a few for the sake of motivation.

Example 1 (Exponential Family Parameter Estimation [1, 6, 16, 17])**.** *Consider the exponential family (w.r.t. some dominating measure) $p(x|\theta) \propto \exp(T(x)\theta - f(\theta))$, where $T(x)$ is the sufficient statistic and $f(\cdot)$ is the log-partition function. Fix the unknown parameters θ. For estimation we take n_i observations for each parameter: $x_{i,j}, i = 1, \ldots, p, j = 1, \ldots, n_i$. Then the (restricted) maximum likelihood estimator is:*

$$\max_{\theta \in \mathcal{K}} \quad \sum_{i=1}^{p} n_i [\theta_i \bar{T}(\mathbf{x}_i) - f(\theta_i)], \tag{4}$$

where $\bar{T}(\mathbf{x}_i) = \frac{1}{n_i} \sum_{j=1}^{n_i} T(x_{ij})$. Surprisingly, as shown in [1], the solution depends only mildly on the log-partition function f. We refer to the excellent book of Barlow et al. [1] for many eye-opening applications under this setting.

Example 2 (Inventory Management [18–21])**.** *Consider an inventory system where a single product is produced and assembled in multiple sites and stages. Once the product is completely assembled it can be sold to the customer which has certain demands. There is a production cost and a storage cost in each site, and each site depends on its "ancestor" sites, that is, in order for site i to produce its part it will need a certain amount of parts from say site j. Obviously, this quantity cannot exceed the storage in site j, hence creating a natural order restriction. An optimal inventory policy can then be formulated and found by solving problem (1), where the functions g_i depend on the production cost, storage cost, and demand.*

Example 3 (Sparsity [7–13])**.** *Sparsity has been recognized as one of the key structures that allow statistical inference in high dimensions, and is usually forced through a penalty function such as the ℓ_1 norm. Consider the ideal orthogonal design case:*

$$\min_{\mathbf{x} \in \mathbb{R}^p} \quad \tfrac{1}{2} \|\mathbf{x} - \mathbf{y}\|^2 + f(\mathbf{x}), \tag{5}$$

i.e. *we are trying to sparsify the input vector $\mathbf{y}$. Intuitively, a bigger entry should be thresholded to zero only after the smaller entries, because the former is more likely to be genuinely nonzero than the latter. This intuition in turn puts an* implicit *order restriction on the penalty function f. Early penalty functions like the ℓ_1 norm are simply permutation-invariant, but more recent works [7, 8, 10, 13] have made the order restriction explicit. A similar setting was also considered in [9] for controlling the false discovery rate in multiple hypothesis testing problems. However, the connection to the isotonic cone is not always recognized or exploited.*

Problem (1) is very special in the sense that its objective function is separable in each coordinate, however, the isotonic cone $\mathcal{K}$ couples all coordinates. If the functions g_i are convex, (1) can be solved using for instance the iterative projected gradient algorithm, provided that we know the Euclidean projection onto $\mathcal{K}$. Instead, the GPAV algorithm we present below is based on a different idea (and will prove more useful later). First note that (1) would be easy to solve if we ignore the isotonic constraint $\mathbf{x} \in \mathcal{K}$: we simply minimize the univariate functions g_i separately (or even in parallel). Of course, the resulting *unconstrained* minimizer, denoted generically as $\mathbf{z}$ throughout, may not satisfy the isotonic constraint. The GPAV algorithm then "pools" the functions g_i in (1) so that next time the unconstrained minimizer $\mathbf{z}$ violates *strictly* less order constraints (if any). The "pooling" is intelligently performed so that the (constrained) minimum value of (1) is still maintained. Therefore after at most finitely many steps the unconstrained minimizer $\mathbf{z}$ will automatically satisfy the isotonic constraint hence be a *bona fide* minimizer.

Let us point out the convenience of working with general abstract functions g_i. Suppose we have an additional nonnegative constraint in (1), *i.e.*, $x_1 \geq x_2 \geq \cdots \geq x_p \geq 0$, which is clearly equivalent as $x_1 \geq x_2 \geq \cdots \geq x_p, x_p \geq 0$. Thus, upon redefining $g_p(x_p) = \infty$ if $x_p < 0$ (*i.e.*, restricting the effective domain of g_p to $\mathbb{R}_+$), we reduce back to (1), but this time without any *explicit* nonnegative constraint. Clearly, other interval constraints under any isotonic cone can be dealt with similarly.

3. The generalized pool-adjacent-violator algorithm

We present in this section the generalized pool-adjacent-violators (GPAV) algorithm. The presentation is kept general and abstract so that we can see the idea more clearly, without being distracted by the non-essential details. The generality will be felt when we apply the results in §5 to recover various existing results and also to uncover many new ones.

As we mentioned in the previous subsection, GPAV "pools" the functions g_i so that the unconstrained minimizer $\mathbf{z}$ will satisfy the isotonic constraint eventually. Take any unconstrained minimizer[1] $\mathbf{z}$ of the separable function $\sum_{i=1}^p g_i(x_i)$, and consider the violator set

$$\{j : z_j \leq z_i \text{ for some } i \text{ that is an immediate predecessor of } j \text{ w.r.t. the partial order } \mathcal{K}\}. \quad (6)$$

Order the violators by their z-values. We have the following result for pooling:

Theorem 1. *Let $\mathcal{K}$ be induced by a rooted tree (Figure 1b). Suppose for all i the closed function $g_i : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ is convex*[2]*. Consider any minimum violator z_j and its predecessor z_i, then there exists a constrained minimizer*

$$\mathbf{y}^\star \in \operatorname*{arg\,min}_{\mathbf{x} \in \mathcal{K}} \quad \sum_{i=1}^p g_i(x_i) \quad (7)$$

such that $y_i^\star = y_j^\star$.

Note that in a rooted tree, every node has exactly one predecessor (except the root which has none). This condition is needed as otherwise it can be tricky to pool which predecessor with the violator. Therefore, whenever the *unconstrained* minimizer $\mathbf{z}$ violates any order constraint, we obtain crucial information about the *constrained* minimizer $\mathbf{y}^\star$. Thanks to separability, the unconstrained minimizer $\mathbf{z}$ may be found cheaply (perhaps even in closed-form).

Theorem 1 is known in special cases: It was first discovered in Ayer et al. [6] where $\mathcal{K} = \mathcal{K}_c$ is induced by a chain (a special rooted tree) and each g_i is quadratic. Strömberg [22] considered the same chain order but allowed g_i to be arbitrary convex functions, see also [23, 24]. W. A. Thompson [17] considered the general rooted tree order but only with specific g_i functions. Our proof (omitted here) also differs substantially from [17]. Orders induced by an arbitrary DAG has also been considered [*e.g.* 1, 2], but the pooling strategies are very complicated, sometimes with unknown complexity.

Theorem 1 immediately suggests an efficient algorithm for solving (7), when $\mathcal{K}$ is induced by a rooted tree. In each iteration we find a minimum violator and perform the pooling, namely, set $g_i(\cdot) \leftarrow g_j(\cdot) \leftarrow \frac{1}{2}[g_i(\cdot) + g_j(\cdot)]$. Since there are at most $O(p)$ violators, and each pooling will reduce the problem size by 1, the total complexity is $O(p^2)$, assuming that finding the unconstrained minimizer $\mathbf{z}$ takes $O(p)$ time. Note that if all functions g_i are quadratic, then it is possible to use advanced data structures to improve the complexity to $O(p \log p)$ [25].

We complement Theorem 1 with a strong converse, which, to the best of our knowledge, is the first of its kind. The choices of the functions g_i below are motivated by the applications in §5.

[1] The existence of minimizers is always assumed in this work, for simplicity.
[2] In fact, we only need the weaker quasiconvexity (unimodality) here.

Theorem 2. *Fix $\lambda > \mathbf{0}$ and consider applying Theorem 1 to solve (1) with $g_i(x) := w_i(x - y_i)^2 + \lambda_i f(x)$. If it always leads to a correct minimizer for any $\mathbf{w} \in \mathbb{R}^p_+, \mathbf{y} \in \mathbb{R}^p$, $p \geq 2$, then the univariate function f must be convex.*

Thus, for nonconvex functions g_i, we cannot directly apply GPAV for solving (1), but see §5 for some exceptions.

4. Linear time algorithm for the chain graph

When the isotonic cone $\mathcal{K}$ is induced by a chain (a special rooted tree), the complexity automatically reduces from $O(p^2)$ for the general rooted tree order to $O(p)$, which is clearly the best possible. This is because in the chain case finding the minimum violator among all violators can be incrementally done in $O(1)$ time. Here we give an online view of this known fact, based on which we will extend the algorithm to piecewise convex functions in §5.

The idea is very simple: we pretend that the input functions $g_1, \ldots, g_p$ are revealed to us one by one, but each time we must solve the current *constrained* problem before the next function is revealed.

To start, we get the first function g_1 and we solve $u_1 \in \arg\min_x \; g_1(x)$ in $O(1)$ time, completing the first step and enabling the next function g_2 to be revealed. Now suppose we have found[3]

$$(u_1, \ldots, u_j) \in \underset{x_1 \leq \cdots \leq x_j}{\arg\min} \quad \sum_{i=1}^{j} g_i(x_i), \tag{8}$$

and we have kept a partition $0 = t_0 < t_1 < \ldots < t_{\ell_j} = j$ of the set $\{0, 1, \ldots, j\}$ so that for each $s = 1, \ldots, \ell_j$, $u_{t_{s-1}+1} = \cdots = u_{t_s}$. Then the next function g_{j+1} is revealed and we need to solve

$$\underset{x_1 \leq \cdots \leq x_{j+1}}{\arg\min} \quad \sum_{i=1}^{j+1} g_i(x_i). \tag{9}$$

The key is that we need *not* solve (9) from scratch. Indeed, after finding the unconstrained candidate

$$u_{j+1} \in \underset{x}{\arg\min} \quad g_{j+1}(x) \tag{10}$$

in $O(1)$ time, we realize that, augmented with our previous solution $(u_1, \ldots, u_j)$, $\mathbf{u} := (u_1, \ldots, u_{j+1})$ is in fact obtainable from applying the pooling procedures suggested in Theorem 1 to the problem (9). Of course, $\mathbf{u}$ need not be the minimizer of (9) since the (only new) order constraint $u_j \leq u_{j+1}$ may not be satisfied yet. So we continue applying Theorem 1 on top of $\mathbf{u}$ to finish the business (picking violators in any order we like). Inductively, after the p-th function is revealed, we will have $\mathbf{u}$ that is a minimizer of (7).

The partition list $0 = t_0 < t_1 < \ldots < t_{\ell_j} = j$ is maintained so that we can resume and complete the j-th round in say $O(\gamma_{j+1})$ time, instead of the brute-force $O(j)$ time, where γ_{j+1} is the number of poolings we performed in step j. Indeed, all terms fall into the interval $[t_{s-1}+1, t_s]$ have been pulled together hence require only one representative. More crucially, we always have the inequality

$$\ell_{j+1} \leq \ell_j - \gamma_{j+1} + 1, \text{ and } \gamma_{j+1} \leq \ell_j. \tag{11}$$

[3] If not for notational clarity, we should put a superscript on u_j, reflecting its possible change in later iterations than j.

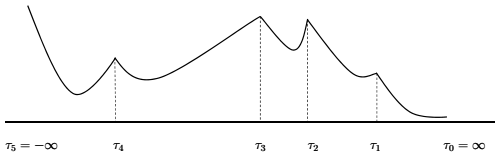


Figure 2: An example k-convex function.

Basically, this means that if we spend a lot of time (large γ_{j+1}) in pulling components in the j-th iteration, then we would have few components (small ℓ_{j+1}) to work with in the $(j+1)$-th iteration, which in turn bounds the time spent in the $(j+1)$-th iteration. Simply put, previous hard work eventually pays out in the future, which is the essence of amortized complexity analysis [26]. After the last function g_p is handled, we can count the total time consumed, using the bound in (11):

$$O\Big(\sum_{i=1}^{p} \gamma_{j+1}\Big) \le O\Big(\sum_{i=1}^{p} (\ell_j - \ell_{j+1} + 1)\Big) = O(p). \tag{12}$$

Perhaps more importantly, we observe that the above online update not only solves (7) in linear time, it actually does much more: it simultaneously solves all the subproblems (8) for all $j = 1, \dots, p$ in linear time. Let us create a vector $\mathsf{r} \in \mathbb{R}^p$ to record the minimum value in (8). Similarly, we create a vector $\mathsf{l} \in \mathbb{R}^p$ that records in linear time the minimum values in the similar subproblems:

$$(v_j, \dots, v_p) \in \operatorname*{arg\,min}_{x_j \le \cdots \le x_p} \sum_{i=j}^{p} g_i(x_i). \tag{13}$$

Then these two records will allow us to apply Theorem 1 on even piecewise convex functions g_i (that need not be convex globally). This will be detailed below.

5. Extension to piece-wise convex functions

In this section we apply the online implementation in §4 to the following problem:

$$\min_{\mathbf{x} \in \mathcal{K}_c} \sum_{i=1}^{p} w_i x_i^2 - m_i x_i + \lambda_i f(x_i), \tag{14}$$

where $f : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ is some univariate closed function, $\mathbf{w}, \boldsymbol{\lambda} \in \mathbb{R}^p_+$, and $\mathcal{K}_c$ is induced by a total order (see Figure 1a). Note that by setting $m_i = 2w_i y_i, \lambda_i \equiv \lambda$ we recover the familiar problem $\min_{\mathbf{x} \in \mathcal{K}_c} \sum_{i=1}^{p} w_i (x_i - y_i)^2 + \lambda f(x_i)$ —a constrained proximity operator of f. Setting $f \equiv 0$ recovers the isotonic regression problem in [1].

We can actually allow the function f to be nonconvex. Let us recall:

Definition 1 (k-convexity). *$f : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ is called k-convex if there exists $\infty = \tau_0 \ge \tau_1 \ge \cdots \ge \tau_k = -\infty$ such that f restricted to $[\tau_i, \tau_{i+1}]$ is convex for all i.*

Clearly, 1-convexity corresponds to the usual convexity. With $k > 1$, we can start to generate many interesting nonconvex functions, see Figure 2 for an illustration.

Example 4. *The capped absolute function [e.g. 27, 28] $|x|_\rho = \min\{|x|/\rho, 1\}$ is 2-convex (when restricted to $\mathbb{R}_+$) with $\tau_1 = \rho$. Note that $\rho|x|_\rho \to |x|$ when $\rho \to \infty$ while $|x|_\rho \to \mathbb{1}_{x \ne 0}$ when $\rho \to 0$. More generally, for any convex function f, we can define its capped cousin $f_\rho = \min\{f/\rho, 1\}$, which is 2-convex with $\tau_1 = \rho$. The capped functions are preferred in sparse estimation problems for their ability to reduce estimation bias, as compared to the convex alternatives [27, 28].*

One attractive property of k-convex functions is its computational tractability: we can chop it into convex pieces and consider each piece in turns. Indeed, if in (14) the function f is k-convex, with $\tau_1, \dots, \tau_k$ being given, we can then consider all possible partitions $0 = j_0 < j_1 < \cdots < j_k = p$. For all $s = 1, \dots, k$ and for all $j_{s-1} + 1 \le i \le j_s$, by defining

$$f_i(x_i) = \begin{cases} \lambda_i f(x_i), & \tau_{s-1} \ge x_i \ge \tau_s \\ \infty, & \text{otherwise} \end{cases}, \tag{15}$$

we can "reduce" the nonconvex problem (14) with a k-convex f into the more general problem (1) where all g_i are convex. Since the online implementation in Section 4 solves (1) in linear time, we can simply enumerate all possible partitions $0 = j_0 < j_1 < \cdots < j_k = p$, each yielding a convex problem in the form of (1). Finally, taking the minimum among these $\binom{p}{k-1}$ similar problems yields the (global) minimizer of (14). If k is a constant then we will have a polynomial time algorithm for solving the nonconvex problem (14) (where nonconvexity comes form f). Note that any function can be (uniformly) approximated by a k-convex function. Intuitively, the more nonconvex f is, i.e. a bigger k is needed for a decent approximation, the more time-consuming it is computationally, i.e., more subproblems to be considered (on the order of p^{k-1}). While it may be possible to prune out many of these subproblems, we will simply restrict to $k \le 2$ below, a choice motivated by Example 4.

Let f be a 2-convex function from now on, with $\tau_1 = \tau$, i.e., f is separately convex on $(-\infty, \tau]$ and $[\tau, \infty)$ but need not be convex on $\mathbb{R}$, see e.g. Example 4. Then, to solve our main problem in (14), we need only find the optimal split

$$x_1 \ge \cdots \ge x_j \ge \tau \ge x_{j+1} \ge \cdots \ge x_p, \tag{16}$$

among all possible positions of τ (namely j). Moreover, given the position of τ, we can convert (14) into two *unrelated* problems in the form of (1), which we already have a linear time algorithm in §4:

$$\operatorname*{arg\,min}_{x_1 \ge \cdots \ge x_j} \sum_{i=1}^{j} w_i x_i^2 - m_i x_i + f_i(x_i), \qquad \operatorname*{arg\,min}_{x_{j+1} \ge \cdots \ge x_p} \sum_{i=j+1}^{p} w_i x_i^2 - m_i x_i + h_i(x_i), \tag{17}$$

where the functions f_i and h_i are defined as:

$$f_i(x_i) = \begin{cases} \lambda_i f(x_i), & x_i \ge \tau \\ \infty, & \text{otherwise} \end{cases}, \qquad h_i(x_i) = \begin{cases} \lambda_i f(x_i), & x_i \le \tau \\ \infty, & \text{otherwise} \end{cases}. \tag{18}$$

Note that both f_i and h_i are convex thanks to the 2-convexity of f, and more importantly they do not depend on the position j. In other words, applying the online implementation in §4 we would be able to find the minimal values in (17) in linear time for all j simultaneously. These are exactly the records l and r we mentioned at the end of §4. Then the optimal position for τ in (16) is given as

$$j^\star \in \operatorname*{arg\,min}_{0 \le t \le p} \mathsf{l}_t + \mathsf{r}_{p-t}, \tag{19}$$

where $\mathsf{l}_0 = \mathsf{r}_0 = 0$. After having $j^\star$ we can just re-solve (17) with j replaced by $j^\star$. It is clear that the overall time is still linear $O(p)$, a big improvement compared to the naive $\binom{p}{1} \cdot O(p) = O(p^2)$ time (obtained by calling the online implementation p times).

The above acceleration trick can be generalized to handle a k-convex function f in (14), with the overall complexity $\binom{p}{k-1} + O(kp^{\min\{2,k-1\}} + p) = O(p^{k-1} + p)$ as opposed to the naive $\binom{p}{k-1} \cdot O(p) = O(p^k)$ time. We omit the rather technical (but less insightful) details.

Example 5. *Bogdan et al. [9] considered the following sorted ℓ_1 norm for false discovery rate (FDR) control:*

$$\|\mathbf{x}\|_{\mathsf{SLOPE}} = \sum_{i=1}^{p} \lambda_i |x|_{(i)}. \tag{20}$$

The main computation involved is again (14) with $f = |\cdot|$. A linear time algorithm in this case is provided in [9]. However, our treatment allows dealing simultaneously with any k-convex f. In particular, our algorithm covers the capped and sorted ℓ_1 norm that is known to yield less statistical bias.

6. Conclusion

In this work we explored the connection between statistical estimation under order restriction and sparse regularization in high dimensional statistical inference. We presented a general pooling principle to solve the order constrained optimization problem and proved a converse theorem to reveal the exact correctness condition. An online linear time implementation for the special chain graph was developed, and further extended to deal with piecewise convex functions. We will apply our algorithm to shrinkage parameter estimation and false discovery control problems.

Acknowledgments

The present study was financially supported by NIH Grant R01GM114311. We thank Adams Wei Yu for some helpful discussions.

References

[1] R. E. Barlow, D. J. Bartholomew, J. M. Bremner, and H. D. Brunk. *Statistical inference under order restrictions: The theory and application of isotonic regression*. John Wiley & Sons, 1972.

[2] Tim Robertson, F. T. Wright, and R. L. Dykstra. *Order Restricted Statistical Inference*. John Wiley & Sons, 1988.

[3] Ulf Grenander. "On the theory of mortality measurement: II." *Scandinavian Actuarial Journal*, vol. 39, no. 2 (1956), pp. 125–153.

[4] Herman Chernoff. "Estimation of the mode." *Annals of the Institute of Statistical Mathematics*, vol. 16, no. 1 (1964), pp. 31–41.

[5] Piet Groeneboom and Geurt Jongbloed. *Nonparametric Estimation under Shape Constraints: Estimators, Algorithms and Asymptotics*. Cambridge University Press, 2014.

[6] Miriam Ayer, H. D. Brunk, G. M. Ewing, W. T. Reid, and Edward Silverman. "An Empirical Distribution Function for Sampling with Incomplete Information." *The Annals of Mathematical Statistics*, vol. 26, no. 4 (1955), pp. 641–647.

[7] Andreas Argyriou, Rina Foygel, and Nathan Srebro. "Sparse Prediction with the k-Support Norm." In: *Proceedings of the 25th Advances in Neural Information Processing Systems (NIPS-12)*. Ed. by F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger. Curran Associates, Inc., 2012, pp. 1457–1465.

[8] Howard Bondell and Brian Reich. "Simultaneous Regression shrinkage, variable selection, and supervised clustering of predictors with oscar." *Biometrics*, vol. 64, no. 1 (2008), pp. 115–123.

[9] Małgorzata Bogdan, Ewout van den Berg, Chiara Sabatti, Weijie Su, and Emmanuel J. Candès. "SLOPE – Adaptive Variable Selection via Convex Optimization." *The Annals of Applied Statistics*, vol. 9, no. 3 (2015), pp. 1103–1140.

[10] Andrew M. McDonald, Massimiliano Pontil, and Dimitris Stamos. "Spectral k-Support Norm Regularization." In: *Proceedings of the 27th Advances in Neural Information Processing Systems (NIPS-14)*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger. Curran Associates, Inc., 2014, pp. 3644–3652.

[11] Leon Wenliang Zhong and James T. Kwok. "Efficient Sparse Modeling with Automatic Feature Grouping." In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. Ed. by Lise Getoor and Tobias Scheffer. Bellevue, Washington, USA: ACM, 2011, pp. 9–16.

[12] Jun Liu, Liang Sun, and Jieping Ye. "Projection on A Nonnegative Max-Heap." In: *Proceedings of the 24th Advances in Neural Information Processing Systems (NIPS-11)*. Ed. by J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F. Pereira, and K.Q. Weinberger. Curran Associates, Inc., 2011, pp. 487–495.

[13] Soumyadeep Chatterjee, Sheng Chen, and Arindam Banerjee. "Generalized Dantzig Selector: Application to the k-support norm." In: *Proceedings of the 27th Advances in Neural Information Processing Systems (NIPS-14)*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger. Curran Associates, Inc., 2014, pp. 1934–1942.

[14] Peter Bacchetti. "Additive Isotonic Models." *Journal of the American Statistical Association*, vol. 84, no. 405 (1989), pp. 289–294.

[15] Zhou Fang and Nicolai Meinshausen. "LASSO isotone for high-dimensional additive isotonic regression." *Journal of Computational and Graphical Statistics*, vol. 21, no. 1 (2012), pp. 72–91.

[16] R. E. Barlow and H. D. Brunk. "The isotonic regression problem and its dual." *Journal of the American Statistical Association*, vol. 67, no. 337 (1972), pp. 140–147.

[17] Jr. W. A. Thompson. "The Problem of Negative Estimates of Variance Components." *The Annals of Mathematical Statistics*, vol. 33, no. 1 (1962), pp. 273–289.

[18] Peter L. Jackson and Robin O. Roundy. "Minimizing Separable Convex Objectives on Arbitrarily Directed Trees of Variable Upper Bound Constraints." *Mathematics of Operations Research*, vol. 16, no. 3 (1991), pp. 504–533.

[19] William L. Maxwell and John A. Muckstadt. "Establishing Consistent and Realistic Reorder Intervals in Production-Distribution Systems." *Operations Research*, vol. 33, no. 6 (1985), pp. 1316–1341.

[20] Robin O. Roundy. "A 98%-Effective Lot-Sizing Rule for a Multi-Product, Multi-Stage Production/Inventory System." *Mathematics of Operations Research*, vol. 11, no. 4 (1986), pp. 699–727.

[21] Jr Arthur F. Veinott. "Least d-Majorized Network Flows with Inventory and Statistical Applications." *Management Science*, vol. 17, no. 9 (1971), pp. 547–567.

[22] Ulf Strömberg. "An algorithm for isotonic regression with arbitrary convex distance function." *Computational Statistics & Data Analysis*, vol. 11, no. 2 (1991), pp. 205–219.

[23] Michael J. Best, Nilotpal Chakravarti, and Vasant A. Ubhaya. "Minimizing Separable Convex Functions Subject to Simple Chain Constraints." *SIAM Journal on Optimization*, vol. 10, no. 3 (2000), pp. 658–672.

[24] Ravindra K. Ahuja, Dorit S. Hochbaum, and James B. Orlin. "Solving the convex cost integer dual network flow problem." *Management Science*, vol. 49, no. 7 (2003), pp. 950–964.

[25] P. M. Pardalos and G. Xue. "Algorithms for a class of isotonic regression problems." *Algorithmica*, vol. 23, no. 3 (1999), pp. 211–222.

[26] Robert Endre Tarjan. "Amortized Computational Complexity." *SIAM Journal on Algebraic Discrete Methods*, vol. 6, no. 2 (1985), pp. 306–318.

[27] Jianqing Fan. "Comments on "Wavelets in Statistics: A Review"." *Journal of the Italian Statistical Association*, vol. 6, no. 2 (1997), pp. 131–138.

[28] Tong Zhang. "Multi-stage convex relaxation for feature selection." *Bernoulli*, vol. 19, no. 5B (2013), pp. 2277–2293.