Density estimation in linear time

Satyaki Mahalanabis and **Daniel Štefankovič**
Department of Computer Science
University of Rochester
Rochester, NY 14627
{smahalan,stefanko}@cs.rochester.edu

Abstract

We consider the problem of choosing a density estimate from a set of densities $\mathcal{F}$, minimizing the L_1-distance to an unknown distribution. Devroye and Lugosi [DL01] analyze two algorithms for the problem: Scheffé tournament winner and minimum distance estimate. The Scheffé tournament estimate requires fewer computations than the minimum distance estimate, but has strictly weaker guarantees than the latter.

We focus on the computational aspect of density estimation. We present two algorithms, both with the same guarantee as the minimum distance estimate. The first one, a modification of the minimum distance estimate, uses the same number (quadratic in $|\mathcal{F}|$) of computations as the Scheffé tournament. The second one, called "efficient minimum loss-weight estimate," uses only a linear number of computations, assuming that $\mathcal{F}$ is preprocessed. We then apply our algorithms to bandwidth selection for kernel estimates and bin-width selection for histogram estimates, yielding efficient procedures for these problems.

We also give examples showing that the guarantees of the algorithms cannot be improved and explore randomized algorithms for density estimation.

1 Introduction

We study the following density estimation problem considered in [DL96, DL01, DGL02]. There is an unknown distribution g and we are given n (not necessarily independent) samples which define empirical distribution h. Given a finite class $\mathcal{F}$ of densities, our objective is to output $f \in \mathcal{F}$ such that the error $\|f - g\|_1$ is minimized. The use of the L_1-norm is well justified because it has many useful properties, for example, scale invariance and the fact that approximate identification of a distribution in the L_1-norm gives an estimate for the probability of every event.

The following two parameters influence the error of a possible estimate: the distance of g from $\mathcal{F}$ and the empirical error. The first parameter is required since we have no control over $\mathcal{F}$, and hence we cannot select a density which is better than the "optimal" density in $\mathcal{F}$, that is, the one closest to g in L_1-norm. It is not obvious how to define the second parameter—the error of h with respect to g. We follow the definition of [DL01], which is inspired by [Yat85] (see Section 1.1 for a precise definition).

Devroye and Lugosi [DL01] analyze two algorithms in this setting: Scheffé tournament winner and minimum distance estimate. The minimum distance estimate, defined by Yatracos [Yat85], is a special case of the minimum distance principle, formalized by Wolfowitz in [Wol57]. It is a general density estimation tool which has been applied, for example, by [DL96, DL97] to the bandwidth selection problem for kernels and by [DL04, DL01] to bin-width selection for histograms. The minimum distance estimate also finds application in hypothesis testing [DGL02].

The Scheffé tournament winner algorithm requires fewer computations than the minimum distance estimate, but it has strictly weaker guarantees (in terms of the two parameters mentioned above) than the latter. Our main contribution are two procedures for selecting an estimate from $\mathcal{F}$, both of which have the same guarantees as the minimum distance estimate, but are computationally more efficient. The first has a quadratic (in $|\mathcal{F}|$) cost, matching the cost of the Scheffé tournament winner algorithm. The second one is even faster, using *linearly* many (in $|\mathcal{F}|$) computations (after preprocessing $\mathcal{F}$).

We also apply our estimation procedures to the problem of bandwidth selection for kernels and to that of bin-width selection for histograms, following [DL01, DL96, DL97, DL04]. We show that in each of these applications "efficient minimum loss-weight estimate" is faster than our "modified minimum distance estimate," which in turn is faster than the minimum distance estimate.

Now we outline the rest of the paper. In Section 1.1 we give the required definitions and introduce the notion of a test-function (a variant of Scheffé set). Then, in Section 1.2, we restate the previous density estimation

algorithms (Scheffé tournament winner and the minimum distance estimate) using test-functions. Next, in Section 2, we present our algorithms. In Section 3 we discuss two widely studied nonparametric estimation problems where the computational cost of efficient minimum loss-weight estimate (including preprocessing) is much smaller than that of both the modified minimum distance and the minimum distance estimates. In Section 4 we explore randomized density estimation algorithms. In the final Section 5, we give examples showing tightness of the theorems stated in the previous sections.

Throughout this paper we focus on the case when $\mathcal{F}$ is finite, in order to compare the computational costs of our estimates to previous ones. However our results generalize in a straightforward way to infinite classes as well if we ignore computational complexity.

1.1 Definitions and Notations

Throughout the paper g will be the unknown distribution. We will use h to denote the empirical distribution, which given samples $X_1, X_2, \ldots, X_n$, is defined for each set $A \subseteq \Omega$ as

$$h(A) = \frac{1}{n} \sum_{i=1}^{n} 1_{[X_i \in A]}$$

Let $\mathcal{F}$ be a set of densities. We will assume that $\mathcal{F}$ is finite. Let $\mathrm{d}_1(g, \mathcal{F})$ be the L_1-distance of g from $\mathcal{F}$, that is, $\min_{f \in \mathcal{F}} \|f - g\|_1$.

Given two functions f_i, f_j on Ω (in this context, densities) we define a *test-function* $T_{ij} : \Omega \to \{-1, 0, 1\}$ to be the function $T_{ij}(x) = \mathrm{sgn}(f_i(x) - f_j(x))$. Note that $T_{ij} = -T_{ji}$. We also define $\mathcal{T}_{\mathcal{F}}$ to be the set of all test-functions for $\mathcal{F}$, that is,

$$\mathcal{T}_{\mathcal{F}} = \{\, T_{ij} \;;\; f_i, f_j \in \mathcal{F} \,\}.$$

Let $\cdot$ be the inner product for the functions on Ω, defined for any 2 functions f, f' as $f \cdot f' = \int f\, f'$. Note that

$$(f_i - f_j) \cdot T_{ij} = \|f_i - f_j\|_1.$$

We use the inner product of the empirical distribution h with the test-functions to choose an estimate, which is a density from $\mathcal{F}$.

In this paper we only consider algorithms which make their decisions purely on inner products of the test-functions with h and members of $\mathcal{F}$. It is reasonable to assume that the computation of the inner product will take significant time. Hence we measure the *computational cost* of an algorithm is by the number of inner products used.

We say that f_i *wins* against f_j if

$$(f_i - h) \cdot T_{ij} < (f_j - h) \cdot T_{ji}. \tag{1}$$

Note that either f_i wins against f_j, or f_j wins against f_i, or there is a draw (that is, there is equality in (1)). We will say that f_i *loses* to f_j if

$$(f_i - h) \cdot T_{ij} \geq (f_j - h) \cdot T_{ji}.$$

The algorithms choose an estimate $f \in \mathcal{F}$ using the empirical distribution h. The L_1-distance of the estimates from the unknown distribution g will depend on the following measure of distance between the empirical and the unknown distribution:

$$\Delta := \max_{T \in \mathcal{T}_{\mathcal{F}}} (g - h) \cdot T. \tag{2}$$

Now we discuss how test-functions can be viewed as a reformulation of Scheffé sets, defined by Devroye and Lugosi [DL01] (inspired by [Sch47] and implicit in [Yat85]), as follows. The Scheffé set of densities f_i, f_j is

$$A_{ij} = \{x \;;\; f_i(x) > f_j(x)\}.$$

Devroye and Lugosi say that f_i wins against f_j if

$$\left| \int_{A_{ij}} f_i - h(A_{ij}) \right| < \left| \int_{A_{ij}} f_j - h(A_{ij}) \right|. \tag{3}$$

The advantage of using Scheffé sets is that for a concrete set $\mathcal{F}$ of densities one can immediately use the theory of Vapnik-Chervonenkis dimension [VČ71] for the family of Scheffé sets of $\mathcal{F}$ (this family is called the *Yatracos class* of $\mathcal{F}$), to obtain a bound on the empirical error.

If h, f_i, f_j are non-negative and integrate to 1 then the condition (1) is *equivalent* to (3) (to see this recall that $T_{ij} = -T_{ji}$, and add $(f_i - h) \cdot \mathbf{1} = (h - f_j) \cdot \mathbf{1}$ to (1), where $\mathbf{1}$ is the constant one function on Ω). Thus, in our algorithms the test-functions can be replaced by Scheffé sets and VC dimension arguments can be applied.

We chose to use test-functions for two reasons: first, they allow us to give succinct proofs of our theorems (especially Theorem 8), and second, they immediately extend to the case when the members of $\mathcal{F}$ do not correspond to distributions (cf, e. g., Exercise 6.2, in [DL01]).

Remark 1 Note that our value of Δ, defined in terms of $\mathcal{T}_{\mathcal{F}}$, is at most twice the Δ used in [DL01], which is defined in terms of Scheffé sets.

1.2 Previous Estimates

In this section we restate the two algorithms for density estimation from Chapter 6 of [DL01]) using test-functions. The first algorithm requires less computation but has worse guarantees than the second algorithm.

Algorithm 1 - SCHEFFÉ TOURNAMENT WINNER.
Output $f \in \mathcal{F}$ with the most wins (tie broken arbitrarily).

Theorem 2 ([DL01], Theorem 6.2) *Let $f_1 \in \mathcal{F}$ be the density output by Algorithm 1. Then*

$$\|f_1 - g\|_1 \leq 9\, \mathrm{d}_1(g, \mathcal{F}) + 8\Delta.$$

The number of inner products used by Algorithm 1 is $\Theta(|\mathcal{F}|^2)$.

Algorithm 2 - MINIMUM DISTANCE ESTIMATE.
Output $f \in \mathcal{F}$ that minimizes

$$\max \{\, |(f - h) \cdot T_{ij}| \;;\; f_i, f_j \in \mathcal{F} \,\}. \tag{4}$$

Theorem 3 ([DL01], Theorem 6.3) *Let f_1 be the density output by Algorithm 2. Then*

$$\|f_1 - g\|_1 \leq 3\,\mathrm{d}_1(g, \mathcal{F}) + 2\Delta.$$

The number of inner products used by Algorithm 2 is $\Theta(|\mathcal{F}|^3)$.

Let us point out that Theorems 6.2 and 6.3 in [DL01] require that each $f \in \mathcal{F}$ corresponds to a distribution, that is, $\int f = 1$. Since we use test-functions in the algorithms instead of Scheffé set based comparisons, the assumption $\int f = 1$ is not actually needed in the proofs of Theorems 6.2 and 6.3 (we skip the proof), and is not used in the proofs of Theorems 4, 8.

2 Our estimators

2.1 A variant of the minimum distance estimate

The following modified minimum distance estimate uses only $O(|\mathcal{F}|^2)$ computations as compared to $O(|\mathcal{F}|^3)$ computations used by Algorithm 2 (equation (5) takes minimum of $O(|\mathcal{F}|)$ terms, whereas equation (4) takes minimum of $O(|\mathcal{F}|^2)$ terms), but as we show in Theorem 4, it gives us the same guarantee as the minimum distance estimate.

Algorithm 3 - MODIFIED MINIMUM DISTANCE ESTIMATE.
Output $f_i \in \mathcal{F}$ that minimizes

$$\max\{\, |(f_i - h) \cdot T_{ij}| \;;\; f_j \in \mathcal{F} \,\}. \qquad (5)$$

Theorem 4 *Let $f_1 \in \mathcal{F}$ be the density output by Algorithm 3. Then*

$$\|f_1 - g\|_1 \leq 3\,\mathrm{d}_1(g, \mathcal{F}) + 2\Delta.$$

The number of inner products used by Algorithm 3 is $\Theta(|\mathcal{F}|^2)$.

Proof :
Let $f_1 \in \mathcal{F}$ be the function output by Algorithm 3. Let $f_2 = \mathrm{argmin}_{f\in\mathcal{F}}\|f - g\|_1$. By the triangle inequality we have

$$\|f_1 - g\|_1 \leq \|f_1 - f_2\|_1 + \|f_2 - g\|_1. \qquad (6)$$

We bound $\|f_1 - f_2\|_1$ as follows:

$$\begin{aligned} \|f_1 - f_2\|_1 &= (f_1 - f_2) \cdot T_{12} \\ &\leq |(f_1 - h) \cdot T_{12}| + |(f_2 - h) \cdot T_{12}| \\ &\leq |(f_1 - h) \cdot T_{12}| + \max_{f_j \in \mathcal{F}} |(f_2 - h) \cdot T_{2,j}| \end{aligned}$$

where in the last inequality we used the fact that $T_{12} = -T_{21}$.

By the criteria of selecting f_1 we have $|(f_1 - h) \cdot T_{12}| \leq \max_{f_j\in\mathcal{F}} |(f_2 - h) \cdot T_{2,j}|$ (since otherwise f_2 would be selected). Hence

$$\begin{aligned} \|f_1 - f_2\|_1 &\leq 2 \max_{f_j \in \mathcal{F}} |(f_2 - h) \cdot T_{2,j}| \\ &\leq 2 \max_{f_j \in \mathcal{F}} |(f_2 - g) \cdot T_{2,j}| \\ &\quad + 2 \max_{f_j \in \mathcal{F}} |(g - h) \cdot T_{2,j}| \\ &\leq 2\|(f_2 - g)\|_1 + 2 \max_{T \in \mathcal{T}_{\mathcal{F}}} |(g - h) \cdot T| \\ &= 2\|f_2 - g\|_1 + 2\Delta. \end{aligned}$$

Combining the last inequality with (6) we obtain

$$\|f_1 - g\|_1 \leq 3\|f_2 - g\|_1 + 2\Delta.$$

■

Remark 5 Note that one can modify the Lemma to only require that g and h be "close" with respect to the test functions for the "best" function in the class, that is, only $|(g - h) \cdot T_{2,j}|$ need to be small (where f_2 is $\mathrm{argmin}_{f\in\mathcal{F}}\|f - g\|_1$).

One can ask whether the observation in Remark 5 can lead to improved density estimation algorithms for concrete sets of densities. The bounds on Δ (which is given by (2)) are often based on the VC-dimension of the Yatracos class of $\mathcal{F}$. Recall that the Yatracos class Y is the set of $A_{ij} = \{x\,;\, f_i(x) > f_j(x)\}$ for all $f_i, f_j \in \mathcal{F}$. Remark 5 implies that instead of the Yatracos class it is enough to consider the set $Y_i = \{A_{ij}\,;\, f_j \in \mathcal{F}\}$ for $f_i \in \mathcal{F}$. Is it possible that the VC-dimension of each set Y_i is smaller the VC-dimension of the Yatracos class Y? The following (artificial) example shows that this can, indeed, be the case. Let $\Omega = \{0, \ldots, n\}$. For each $(n+1)$-bit binary string $a_0, a_1, \ldots, a_n$, let us consider the distribution

$$P(k) = \frac{1}{4n}(1 + (1/2 - a_0)(1/2 - a_k))2^{-\sum_{j=1}^{n} a_j 2^j},$$

for $k \in \{1, \ldots, n\}$ (with $P(0)$ chosen to make P into a distribution). For this family of 2^{n+1} distributions the VC-dimension of the Yatracos class is n, whereas each Y_i has VC-dimension 1 (since a pair of distributions f_i, f_j has a non-trivial set A_{ij} if and only if their binary strings differ only in the first bit).

2.2 An even more efficient estimator - minimum loss-weight

In this section we present an estimator which, after preprocessing $\mathcal{F}$, uses only $O(|\mathcal{F}|)$ inner products to obtain a density estimate. The guarantees of the estimate are the same as for Algorithms 2 and 3.

The algorithm uses the following quantity to choose the estimate:

$$\begin{aligned} \text{loss-weight}(f) &= \max\{\, \|f - f'\|_1 \;;\; f \text{ loses} \\ &\qquad \text{to } f' \in \mathcal{F} \,\}. \end{aligned}$$

Intuitively a good estimate should have small loss-weight (ideally the loss-weight of the estimate would be $-\infty = \max\{\}$, that is, the estimate would not lose at all). Thus the following algorithm would be a natural candidate for a good density estimator (and, indeed, it has a guarantee matching Algorithms 2 and 3), but, unfortunately, we do not know how to implement it using $O(|\mathcal{F}|)$ inner products.

Algorithm 4a - MINIMUM LOSS-WEIGHT ESTIMATE.
Output $f \in \mathcal{F}$ that minimizes loss-weight(f).

The next algorithm, seems less natural than algorithm 4a, but its condition can be implemented using only $O(|\mathcal{F}|)$ inner products.

Algorithm 4b - EFFICIENT MINIMUM LOSS-WEIGHT ESTIMATE.
Output $f \in \mathcal{F}$ such that for every f' to which f loses we have

$$\|f - f'\|_1 \leq \text{loss-weight}(f'). \quad (7)$$

Before we delve into the proof of (8) let us see how Algorithm 4b can be made to use $|\mathcal{F}|-1$ inner products. We preprocess $\mathcal{F}$ by computing L_1-distances between all pairs of densities in $\mathcal{F}$ and store the distances in an list sorted in decreasing order. When the algorithm is presented with the empirical distribution h, all it needs to do is perform comparison between select pairs of densities. The advantage is that we preprocess $\mathcal{F}$ only once and, for each new empirical distribution we only compute inner products necessary for the comparisons.

We will compute the estimate as follows.

input : family of densities $\mathcal{F}$, list L of all pairs $\{f_i, f_j\}$ sorted in decreasing order by $\|f_i - f_j\|_1$, oracle for computing inner products $h \cdot T_{ij}$.
output : $f \in \mathcal{F}$ such that: $(\forall f')$ f loses to f' $\implies$ $\|f - f'\|_1 \leq \text{loss-weight}(f')$.

1 $S \leftarrow \mathcal{F}$
2 **repeat**
3 pick the first edge $\{f_i, f_j\}$ in L
4 **if** f_i loses to f_j **then** $f' \leftarrow f_i$ **else** $f' \leftarrow f_j$ **fi**
5 remove f' from S
6 remove pairs containing f' from L
7 **until** $|S| = 1$
8 output the density in S

Detailed version of algorithm 4b - using $O(|\mathcal{F}|)$ inner products.

Note that while Algorithm 4b uses only $O(|\mathcal{F}|)$ inner products its running time is actually $\Theta(|\mathcal{F}|^2)$, since it traverses a list of length $\Theta(|\mathcal{F}|^2)$. Are we cheating? There are two answers to this question: practical and theoretical. As we will see in applications the inner products dominate the computation, justifying our focus on just the inner products (of which there are linearly many). Theoretically, if we are willing to spend exponential time for the preprocessing, we can build the complete decision tree corresponding to Algorithm 4b and obtain a linear-time density selection procedure. We find the following question interesting: Is it possible to achieve linear running time using only polynomial-time preprocessing?

Question 6 (Tournament Revelation Problem) *We are given a weighted undirected complete graph on n vertices. Assume that the edge-weights are distinct. We preprocess the weighted graph and then play the following game with an adversary until only one vertex remains: we report the edge with the largest weight and the adversary chooses one of the endpoints of the edge and removes it from the graph (together with all the adjacent edges).*

Our goal is to make the computational cost during the game linear-time (in n) in the worst-case (over the adversary's moves). Is it possible to achieve this goal with polynomial-time preprocessing?

We now show that the detailed version of algorithm 4b outputs f satisfying the required condition.

Lemma 7 *The estimate f output by the detailed version of algorithm 4b satisfies (7) for every f' to which f loses.*

Proof :
We show, using induction, that the following invariant is always satisfied on line 2. For any $f \in S$ and any $f' \in \mathcal{F} \setminus S$ we have that if f loses to f' then $\|f - f'\|_1 \leq \text{loss-weight}(f')$. Initially, $\mathcal{F} \setminus S$ is empty and the invariant is trivially true. For the inductive step, let f' be the density most recently removed from S. To prove the induction step we only need to show that for every $f \in S$ we have that if f loses to f' then $\|f - f'\|_1 \leq \text{loss-weight}(f')$. Let W be the L_1-distance between two densities in $S \cup \{f'\}$. Then loss-weight$(f') \geq W$ (since f' lost), and $\|f - f'\|_1 \leq W$ (by the definition of W). ■

Theorem 8 *Let $f_1 \in \mathcal{F}$ be the density output by Algorithm 4a (or Algorithm 4b). Then*

$$\|f_1 - g\|_1 \leq 3\,\mathrm{d}_1(g, \mathcal{F}) + 2\Delta. \quad (8)$$

Assume that we are given L_1-distances between every pair in $\mathcal{F}$. The number of inner products used by Algorithm 4b is $\Theta(|\mathcal{F}|)$.

Proof of Theorem 8:
Let $f_4 = g$. Let f_2 be the function $f \in \mathcal{F}$ minimizing $\|g - f\|_1$. We can reformulate our goal (8) as follows:

$$(f_1 - f_4) \cdot T_{14} \leq 2\Delta + 3(f_2 - f_4) \cdot T_{24}. \quad (9)$$

Let $f_3 \in \mathcal{F}$ be the function $f' \in \mathcal{F}$ such that f_2 loses against f' and $\|f_2 - f'\|_1$ is maximal (there must be at least one function to which f_2 loses, otherwise the algorithm would pick f_2 and we would be done). Note that $f_1, f_2, f_3 \in \mathcal{F}$, but f_4 does need to be in $\mathcal{F}$.

We know that f_2 loses against f_3, that is, we have (see (1))

$$2h \cdot T_{23} \leq f_2 \cdot T_{23} + f_3 \cdot T_{23}, \quad (10)$$

and, since f_1 satisfied (7), we also have

$$(f_1 - f_2) \cdot T_{12} \leq (f_2 - f_3) \cdot T_{23}. \quad (11)$$

By (2) we have

$$2(f_4 - h) \cdot T_{23} \leq 2\Delta. \quad (12)$$

Adding (10), (11), and (12) we obtain

$$2(f_2 - f_4) \cdot T_{23} + (f_2 - f_1) \cdot T_{12} + 2\Delta \geq 0. \quad (13)$$

Note that for any i, j, k, ℓ we have:

$$(f_i - f_j) \cdot (T_{ij} - T_{k\ell}) \geq 0, \quad (14)$$

since if $f_i(x) > f_j(x)$ then $T_{ij} - T_{k\ell} \geq 0$, if $f_i(x) < f_j(x)$ then $T_{ij} - T_{k\ell} \leq 0$, and if $f_i(x) = f_j(x)$ then the contribution of that x is zero. By applying (14) four times we obtain

$$(f_2 - f_4) \cdot (3T_{24} - 2T_{23} - T_{14}) + (f_1 - f_2) \cdot (T_{12} - T_{14}) \geq 0. \quad (15)$$

Finally, adding (13) and (15) yields (9). ■

Remark 9 Note that Remark 5 also applies to Algorithms 4a and 4b, since (12) is the only inequality in which Δ is used.

Lemma 10 *If the condition* (7) *of Algorithm 4b is relaxed to*

$$\|f - f'\|_1 \leq C \cdot \text{loss-weight}(f'), \tag{16}$$

for some $C \geq 1$*, an analogue of Theorem 8 with* (8) *replaced by*

$$\|f_1 - g\|_1 \leq (1 + 2C)\, \mathrm{d}_1(g, \mathcal{F}) + 2C\Delta \tag{17}$$

holds.

Proof :
The proof is almost identical to the proof of Theorem 8. Let $f_4 = g$. Let f_2 be the function $f \in \mathcal{F}$ minimizing $\|g - f\|_1$. We can reformulate our goal (17) as follows:

$$(f_1 - f_4) \cdot T_{14} \leq 2C\Delta + (1 + 2C)(f_2 - f_4) \cdot T_{24}. \tag{18}$$

Let $f_3 \in \mathcal{F}$ be the function $f' \in \mathcal{F}$ such that f_2 loses against f' and $\|f_2 - f'\|_1$ is maximal (there must be at least one function to which f_2 loses, otherwise the algorithm would pick f_2 and we would be done). Note that $f_1, f_2, f_3 \in \mathcal{F}$, but f_4 does need to be in $\mathcal{F}$.

Equations (10) and (12) from proof of Theorem 8 are satisfied here as well. Since f_1 satisfies (16), we also have

$$(f_1 - f_2) \cdot T_{12} \leq C(f_2 - f_3) \cdot T_{23}. \tag{19}$$

Adding (10) multiplied by C, (19), and (12) multiplied by C we obtain

$$2C(f_2 - f_4) \cdot T_{23} + (f_2 - f_1) \cdot T_{12} + 2C\Delta \geq 0. \tag{20}$$

By applying (14) four times we obtain

$$\begin{aligned}(f_2 - f_4) \cdot ((1 + 2C)T_{24} - 2CT_{23} - T_{14}) + \\ (f_1 - f_2) \cdot (T_{12} - T_{14}) \geq 0.\end{aligned} \tag{21}$$

Finally, adding (20) and (21) yields (18). ■

Lemma 10 allows us to run Algorithm 4b with distances between the densities computed approximately with relative error $(1 \pm \varepsilon)$ and obtain analogue of Theorem 8.

Corollary 11 *Assume that we are given approximate* L_1*-distances between every pair in* $\mathcal{F}$ *with relative error* $(1 \pm \varepsilon)$*. Let* $f_1 \in \mathcal{F}$ *be the density output by Algorithm 4a (or Algorithm 4b), where the algorithm uses the approximate distances (instead of the true distances). Then*

$$\|f_1 - g\|_1 \leq \frac{3 + \varepsilon}{1 - \varepsilon}\, \mathrm{d}_1(g, \mathcal{F}) + \frac{2 + 2\varepsilon}{1 - \varepsilon}\Delta. \tag{22}$$

The number of inner products used by Algorithm 4b is $\Theta(|\mathcal{F}|)$*.*

Proof :
Let $D(f, f')$ be the approximate L_1-distance between f and f' given to the algorithm (for every pair $f, f' \in \mathcal{F}$). Let

$$\begin{aligned}\text{loss-weight}'(f) \;=\; &\max\{\, D(f, f') \;;\; f \text{ loses} \\ &\text{to } f' \in \mathcal{F} \,\}.\end{aligned}$$

The proof of Lemma 7 yields that the estimate f output by the detailed version of algorithm 4b satisfies the following inequality

$$D(f, f') \leq \text{loss-weight}'(f').$$

for every f' to which f loses. Now using the fact that $D(f, f')$ is an $(1 \pm \varepsilon)$ approximation of $\|f - f'\|_1$ we obtain that the estimate f output by algorithm 4b satisfies the following

$$\|f - f'\|_1 \leq \frac{1 + \varepsilon}{1 - \varepsilon} \cdot \text{loss-weight}(f').$$

for every f' to which f loses. ■

3 Applications

We now describe two nonparametric density estimation problems where our estimates can be used to obtain efficient algorithms. The first of these problems is that of selecting the optimal smoothing factor for kernel estimates (Section 3.1) while the second one is that of finding an optimal bin-width for 1-dimensional histograms (Section 3.3).

3.1 Bandwidth selection for kernel estimates

We are going to show that our estimates give fast algorithms for the bandwidth selection problem for uniform kernels on $\mathbb{R}$.

Given n i.i.d samples $x_1, \ldots, x_n \in \mathbb{R}$ drawn from an unknown distribution g the kernel estimate for g is the density

$$f_{n,s}(x) = \frac{1}{ns} \sum_{i=1}^{n} K\left(\frac{x - x_i}{s}\right)$$

where K, the kernel, is a function (usually nonnegative) with $\int K = 1$ and $\int |K| < \infty$, and $s > 0$ is called the smoothing factor. For us K will be the uniform distribution on $[-1, 1]$.

Given $x_1, \ldots, x_n$ the bandwidth selection problem is to select an $s^* > 0$ such that $\|f_{n,s^*} - g\|_1$ is close to $\inf_{s>0} \|f_{n,s} - g\|_1$ [DL01, DL96, DL97]. The data splitting approach to bandwidth selection uses $n - m$ ($n \gg m > 0$) samples $x_1, \ldots, x_{n-m}$ to define the kernel estimate $f_{n-m,s}$ and remaining m samples $x_{n-m+1}, \ldots, x_n$ as a test set which defines an empirical measure h. Devroye and Lugosi ([DL96]) use the minimum distance estimate to give an algorithm for selecting s^*. Given $n > 0$ samples, they select s from an interval $[a_n, b_n]$ (where, e. g., $a_n = e^{-n}, b_n = e^n$). They discretize $[a_n, b_n]$ by defining $s_1 = a_n, s_2 = a_n(1 + \delta_n), \ldots, s_i = a_n(1 + \delta_n)^{i-1}, \ldots, s_N = a_n(1 + \delta_n)^{N-1}$ where $N = \lfloor \ln(b_n/a_n)/\ln(1+\delta_n) \rfloor$ and $\delta_n > 0$ is a parameter. They now select s^* to be s_i such that f_{n-m,s_i} is the minimum distance estimate for $\{f_{n-m,s_i} \;;\; 1 \leq i \leq N\}$ and measure h. Their main theorem is the following.

Theorem 12 *([DL96]) Let* K *be nonnegative, Lipschitz and nonzero only in* $[-1, 1]$*. Let* a_n, b_n *be such that* $na_n \to 0$*,* $b_n \to \infty$ *as* $n \to \infty$*. Assume that* $\delta_n = \frac{c}{\sqrt{n}}$

and that $\ln \frac{b_n}{a_n} \leq c' n^a$ *where* $c, c', a > 0$ *are constants. If*

$$\frac{m}{n} \to 0 \text{ and } \frac{m}{n^{4/5} \ln n} \to \infty \text{ as } n \to \infty,$$

then the estimate f_{n-m,s^*} *satisfies*

$$\sup_g \limsup_{n\to\infty} \frac{E[\|f_{n-m,s^*} - g\|_1]}{\inf_{s>0} E[\|f_{n,s} - g\|_1]} \leq 3. \tag{23}$$

Observation 13 *For* a_n, b_n, δ_n, a *as in Theorem 12,* $N = \Theta(n^{1/2+a})$.

We can replace minimum distance with the minimum loss-weight estimate (Algorithm 4b) in this setting. Simply define $\hat{s}$ to be s_i $(1 \leq i \leq N)$ such that f_{n-m,s_i} is the efficient minimum loss-weight estimate for $\{f_{n-m,s_i} ; 1 \leq i \leq N\}$ and measure h. This requires the computation of L_1 distances between all $O(N^2)$ pairs of densities. Assume however that the kernel K is such that we are able to compute approximate estimates $D_{i,j}$, $1 \leq i, j \leq N$ such that with probability at least $1 - \delta$,

$$\forall\ i, j,\ (1-\varepsilon)D_{ij} \leq \|f_{n-m,s_i} - f_{n-m,s_j}\|_1 \leq (1+\varepsilon)D_{ij} \tag{24}$$

We can now define the approximate minimum loss-weight estimate $\hat{s}'$ in the same way we defined $\hat{s}$. In other words, $\hat{s}'$ is s_i such that Algorithm 4b outputs f_{n-m,s_i} for the class $\{f_{n-m,s_i} ; 1 \leq i \leq N\}$ and the measure h, except that it uses D_{ij} instead of $\|f_{n-m,s_i} - f_{n-m,s_j}\|_1$ for each i, j. The following theorem is the analogue of Theorem 12 for both $\hat{s}$ and $\hat{s}'$.

Theorem 14 *Let* $K, a_n, b_n, \delta_n, a > 0, m$ *be as in Theorem 12. Then* $\hat{s}$ *satisfies*

$$\sup_g \limsup_{n\to\infty} \frac{E[\|f_{n-m,\hat{s}} - g\|_1]}{\inf_{s>0} E[\|f_{n,s} - g\|_1]} \leq 3. \tag{25}$$

Morever, if

$$\varepsilon \to 0 \text{ and } \frac{\delta}{n^{-2/5}} \to 0 \text{ as } n \to \infty$$

then $\hat{s}'$ *satisfies*

$$\sup_g \limsup_{n\to\infty} \frac{E[\|f_{n-m,\hat{s}'} - g\|_1]}{\inf_{s>0} E[\|f_{n,s} - g\|_1]} \leq 3. \tag{26}$$

The proof of Theorem 14 is identical to that of Theorem 12, except that the use of Theorem 3 needs to replaced by Theorem 8 for (25), and by Corollary 11 for (26).

Finally we state a lemma which shows, using ideas from [Ind06] and [LHC07], that it is indeed possible to efficiently compute approximate estimates D_{ij} satisfying (24) (with confidence δ) when the kernel K is the uniform distribution on $[-1, 1]$.

Lemma 15 *Let the kernel* K *be the uniform distribution on* $[-1, 1]$. *Let* $\varepsilon, \delta \in (0, 1)$. *Then there is a randomized algorithm which in time* $O((1/\varepsilon)^2(nN + N^2)\log(nN/\delta))$ *computes* D_{ij} *for* $i, j \in [N]$ *such that with probability* $\geq 1 - \delta$ *we have that for all* $i, j \in [N]$

$$(1-\varepsilon)D_{ij} \leq \|f_{n-m,s_i} - f_{n-m,s_j}\|_1 \leq (1+\varepsilon)D_{ij}.$$

Proof :
Follows immediately from Lemma 17. ■

Let us analyze the time required for computing $\hat{s}'$ for the uniform kernel. Let T_{ij} denote the test function for f_{n-m,s_i}, f_{n-m,s_j}. If we sort $x_1, \ldots, x_{n-m}$ (using $O(n \log n)$ time) in the preprocessing step then computing the inner product $f_{n-m,s_i} \cdot T_{ij}$ for any i, j requires only $O(n)$ time. Computing T_{ij} at any point in $\mathbb{R}$ takes $O(\log n)$ time (using a single binary search). Hence computing the inner product $h \cdot T_{ij}$ can be done in $O(m \log n)$ time.

So the preprocessing time

$$O((1/\varepsilon)^2(nN + N^2)\log(nN/\delta) + n \log n)$$

dominates the running time of the rest of the procedure, which is

$$O((n + m \log n)N).$$

Choosing $\varepsilon = 1/\log n$ and $\delta = 1/\sqrt{n}$ yields a running time of $O((nN + N^2)\text{polylog}(n))$. In contrast, modified minimum distance requires $N^2(m \log n + n)$ time while the minimum distance estimate requires $N^3(m \log n + n)$ time, both of which are much slower since in Theorem 12, $m = \Omega(n^{4/5})$.

3.2 Efficient approximation of L_1-distances using projections.

Our main tool will be the following result of [LHC07] (for related work see also [Ind06]).

Lemma 16 (Lemma 8 of [LHC07]) *Let* $v_1, \ldots, v_N \in \mathbb{R}^M$. *Let* $\varepsilon, \delta \in (0, 1)$. *Let*

$$d \geq 11(2 \log N - \log \delta)/\varepsilon^2$$

be an integer. Let R *be an* $d \times M$ *matrix whose entries are i.i.d. from the Cauchy distribution* $C(0, 1)$. *Let* $w_i = Rv_i$ *for* $i \in [N]$. *Let* D_{ij} *be the geometric mean of the coordinates of* $|w_i - w_j|$. *With probability* $\geq 1 - \delta$ *(over the choice of the entries in* R*) we have for all pairs* $i, j \in [N]$

$$(1-\varepsilon)D_{ij} \leq \|v_i - v_j\|_1 \leq (1+\varepsilon)D_{ij}. \tag{27}$$

As an immediate consequence of Lemma 16 we obtain an efficient algorithm for approximating all pairwise L_1-distances between N densities each of which is a mixture of n uniform distributions on intervals.

Lemma 17 *Let* n *and* N *be positive integers. Let* $\varepsilon, \delta \in (0, 1)$. *For each* $i \in [N]$ *let* f_i *be a mixture of* n *uniform densities on intervals (*f_i *is given by a set of* n *mixture coefficients* $\alpha_{i,1}, \ldots, \alpha_{i,n}$ *and* n *disjoint intervals* $[a_{i,1}, b_{i,1}), \ldots, [a_{i,n}, b_{i,n})$*). There is a randomized algorithm which in time* $O((1/\varepsilon)^2(nN+N^2)\log(nN/\delta))$ *computes* D_{ij} *(for* $i, j \in [N]$*) such that with probability* $\geq 1 - \delta$ *we have that for all* $i, j \in [N]$

$$(1-\varepsilon)D_{ij} \leq \|f_i - f_j\|_1 \leq (1+\varepsilon)D_{ij}. \tag{28}$$

Proof :
Let $S = s_0 < s_1 < \cdots < s_M$ be the sequence obtained by sorting the set

$$\{a_{i,j}\,;\, i \in [N], j \in [n]\} \cup \{b_{i,j}\,;\, i \in [N], j \in [n]\}.$$

Note that $M < 2Nn$. Let $v_i \in \mathbb{R}^M$ be the vector whose j-th coordinate is the measure of $[s_{j-1}, s_j)$ under f_i. We have $\|f_i - f_j\|_1 = \|v_i - v_j\|_1$ for all $i, j \in [N]$. Now we will apply Lemma 16 to $v_1, \ldots, v_N$.

Let $d = \lceil 11(2\log 2nN - \log\delta)/\varepsilon^2 \rceil$. Let R be an $d \times M$ matrix whose entries are i.i.d. from the Cauchy distribution $C(0,1)$. We can compute R in time $O(dM)$. Suppose that we computed $w_i = Rv_i$ for $i \in [N]$. Then we can compute D_{ij}, the coordinate mean of $|w_i - w_j|$ for all $i, j \in [N]$ in time $O(N^2 d)$. The equation (27) and the fact that $\|f_i - f_j\|_1 = \|v_i - v_j\|_1$ implies (28). It remains to show how to compute $w_i = Rv_i$ efficiently.

The j-th coordinate of v_i is the measure of $[s_{j-1}, s_j)$ under f_i which is $(s_j - s_{j-1})$ times the density of f_i on the interval $[s_{j-1}, s_j)$ (the density of f_i is constant on this interval). Let R' be obtained from matrix R by multiplying j-th column by $(s_j - s_{j-1})$ for $j \in [M]$. We can obtain R' from R in time $O(dM)$. Let R'' be the matrix with $R''_{ij} = R'_{i1} + R'_{i2} + \cdots + R'_{ij}$ (again we can compute R'' from R' in time $O(dM)$). We have

$$(Rv_i)_k = \sum_{j=1}^{n} \frac{\alpha_{ij}}{b_{ij} - a_{ij}} \left(R''_{k,r(b_{ij})} - R''_{k,r(a_{ij})-1} \right). \quad (29)$$

Using equation (29) we can compute all v_i in time $O(nNd)$. ■

Remark 18 In a forthcoming paper [MŠ08] we generalize Lemma 17 to piecewise polynomial densities. For each $i \in [N]$, let density f_i be specified by n disjoint intervals

$$[a_{i,1}, b_{i,1}), \ldots [a_{i,n}, b_{i,n}),$$

and in interval $[a_{i,j}, b_{i,j})$ for each $j \in [n]$ by coefficients $\alpha^{(0)}_{i,j}, \alpha^{(1)}_{i,j}, \ldots, \alpha^{(d)}_{i,j}$ such that

$$(\forall x \in [a_{i,j}, b_{i,j}))\ f(x) = \alpha^{(0)}_{i,j} + \alpha^{(1)}_{i,j} x + \ldots + \alpha^{(d)}_{i,j} x^d.$$

Theorem 5.1 of [MŠ08] states that there is a randomized algorithm which takes $O(N(N+n)(\frac{d}{\varepsilon})^3 \log \frac{N}{\delta})$ time and outputs $D_{ij},\ 1 \le i < j \le N$ such that with probability at least $1 - \delta$, for each $1 \le i < j \le N$

$$(1-\varepsilon)D_{ij} \le \|f_i - f_j\|_1 \le (1+\varepsilon)D_{ij}.$$

3.3 Bin-width selection for histogram estimates

Here we show how the efficient minimum loss-weight estimate yields a fast algorithm for finding the optimal bin-width of 1-dimensional histograms. The set of densities arising in this problem will be such that for any subset of them it will be trivial to determine the pair whose L_1-distance is maximal.

Given a bin-width $s > 0$, define A_t for each integer t to be the interval $[ts, (t+1)s)$. Given n sample points $x_1, \ldots, x_n \in \mathbb{R}$ drawn from a distribution g, a regular histogram estimate $f_{n,s}$ is defined as the density such that for each t and each $x \in A_t$

$$f_{n,s}(x) = \frac{|\{x_i\ ;\ x_i \in A_t\}|}{ns}. \quad (30)$$

Devroye and Lugosi [DL01, DG85] consider the problem of finding L_1-optimal histogram estimates. As in the case of kernel estimates, they use the first $n - m$ sample points $x_1, \ldots, x_{n-m}$ to define the histogram estimate $f_{n-m,s}$, and the remaining points $x_{n-m+1}, \ldots, x_n$ to define the empirical distribution h. Now, given a set Θ to choose from, s^* is defined to be the bin-width such that f_{n-m,s^*} is the minimum distance estimate for $\{f_{n-m,s}\ ;\ s \in \Theta\}$ and h. If each width in Θ is 2^k for some integer k, Devroye and Lugosi [DL01] prove the following about s^*.

Theorem 19 *([DL01], Theorem 10.3 and Lemma 10.5) If $\Theta \subseteq \{2^i\ ;\ i \in \mathbb{Z}\}$ then for all n and m, with $0 < m \le n/2$,*

$$E\left[\|f_{n-m,s^*} - g\|_1\right] \le 3 \inf_{s \in \Theta} E\left[\|f_{n,s} - g\|_1\right] \left(1 + \frac{2m}{n-m} + 8\sqrt{\frac{m}{n}}\right) + 8\sqrt{\frac{\log(2(m+1)n^2)}{m}} + \frac{3}{n}.$$

Once again, like kernel estimates, we can simply use efficient minimum loss-weight instead of minimum distance. Now, define $\hat{s}$ to be such that $f_{n-m,\hat{s}}$ is the efficient minimum loss-weight estimate (Algorithm 4b) for $\{f_{n-m,s}\ ;\ s \in \Theta\}$ and h.

We state below the analogue of Theorem 19 for the efficient minimum loss-weight estimate. The proof is the same, except, one uses Theorem 8 instead of Theorem 3.

Theorem 20 *If Θ is as in Theorem 19 then for all n and m with $0 < m \le n/2$,*

$$E\left[\|f_{n-m,\hat{s}} - g\|_1\right] \le 3 \inf_{s \in \Theta} E\left[\|f_{n,s} - g\|_1\right] \left(1 + \frac{2m}{n-m} + 8\sqrt{\frac{m}{n}}\right) + 8\sqrt{\frac{\log(2(m+1)n^2)}{m}} + \frac{3}{n}.$$

Let us now consider the computational cost. For each n, lets say we choose Θ to be $\{2^i\ ;\ -N \le i \le N\}$ (where, e. g., $N = n$ is a possible choice) so that we have $2N + 1$ densities to select from. Define $s_i = 2^{-N+i}$ for each $0 \le i \le 2N$. The following lemma shows that we need not actually pre-compute pairwise L_1-distances in the preprocessing step of Algorithm 4b.

Lemma 21 *For any $i \le k \le \ell \le j$,*

$$\|f_{n,s_\ell} - f_{n,s_k}\|_1 \le \|f_{n,s_j} - f_{n,s_i}\|_1.$$

Proof :
We first prove that for any n and $i < j$,

$$\|f_{n,s_j} - f_{n,s_{i+1}}\|_1 \le \|f_{n,s_j} - f_{n,s_i}\|_1, \quad (31)$$

and

$$\|f_{n,s_{j-1}} - f_{n,s_i}\|_1 \le \|f_{n,s_j} - f_{n,s_i}\|_1. \qquad (32)$$

In order to prove (31), consider any bin

$$A_t = [ts_{i+1}, (t+1)s_{i+1}) = [2ts_i, 2(t+1)s_i).$$

Denote the density of f_{n,s_j} in this bin by μ, and that of f_{n,s_i} in $[2ts_i, (2t+1)s_i), [(2t+1)s_i, (2t+2)s_i)$ respectively by μ_1, μ_2. Clearly the density of $f_{n,s_{i+1}}$ in A_t is $\frac{\mu_1+\mu_2}{2}$. However,

$$\begin{aligned}\int_{A_t} |f_{n,s_j} - f_{n,s_i}| &= s_i(|\mu-\mu_1| + |\mu-\mu_2|) \\ &\ge 2s_i\Big|\mu - \frac{\mu_1+\mu_2}{2}\Big| \\ &= \int_{A_t} |f_{n,s_j} - f_{n,s_{i+1}}|.\end{aligned}$$

Thus

$$\|f_{n,s_j} - f_{n,s_i}\|_1 = \sum_t \int_{A_t} |f_{n,s_j} - f_{n,s_i}| \ge$$

$$\sum_t \int_{A_t} |f_{n,s_j} - f_{n,s_{i+1}}| = \|f_{n,s_j} - f_{n,s_{i+1}}\|_1.$$

The proof of (32) is similar. The lemma now follows by induction. ■

So in each iteration of Algorithm 4b, the pair of densities that are picked for comparison simply correspond to the smallest and the largest bin-widths remaining to be considered. In other words, if s_i and s_j are respectively the minimum and the maximum width remaining, f_{n-m,s_i} is compared against f_{n-m,s_j}.

Now let T_{ij} denote, as usual, the test function for f_{n-m,s_i}, f_{n-m,s_j}. Now we analyze the time needed to compute $f_{n-m,s_i} \cdot T_{ij}$ and $h \cdot T_{ij}$. We first preprocess $x_1, \ldots, x_{n-m}$ by sorting them ($O(n \log n)$ time). For any x the value of $T_{ij}(x)$ can be computed in time $O(\log n)$ (using binary search on $x_1, \ldots, x_{n-m}$) and hence $h \cdot T_{ij}$ can be computed in $O(m \log n)$ time. We can compute $f_{n-m,s_i} \cdot T_{ij}$ in $O(n)$ time (using one pass over the array $x_1, \ldots, x_{n-m}$).

Hence the efficient minimum loss-weight estimate requires only $O(N(n+m\log n)+n\log n)$ computations in total. In contrast, modified minimum distance requires $O(N^2(n+m\log n)+n\log n)$ and minimum distance requires $O(N^3(n+m\log n)+n\log n)$, making efficient minimum loss-weight the fastest of the three.

4 Randomized algorithm and mixtures

In this section we explore the following question: can constant 3 be improved if we allow randomized algorithms? Let f be the output of a randomized algorithm (f is a random variable with values in $\mathcal{F}$). We would like to bound the expected error $\mathrm{E}\big[\|f-g\|_1\big]$, where the expectation is taken only with respect to coin tosses made by the algorithm (and *not* with respect to the distribution of the samples).

If instead of randomization we consider algorithms which output mixtures of densities in $\mathcal{F}$ we obtain a related problem. Indeed, let α be the distribution on $\mathcal{F}$ produced by a randomized algorithm, and let $r = \sum_{s\in\mathcal{F}} \alpha_s s$ be the corresponding mixture. Then, by triangle inequality, we have

$$\|r-g\|_1 \le \mathrm{E}\big[\|f-g\|_1\big].$$

Hence the model in which the output is allowed to be a mixture of densities in $\mathcal{F}$ is "easier" than the model in which the density selection algorithm is randomized.

We consider here only the special case in which $\mathcal{F}$ has only two densities f_1, f_2, and give an randomized algorithm with a better guarantee than is possible for deterministic algorithms. Later, in Section 5, we give a matching lower bound in the mixture model.

To simplify the exposition we will, without loss of generality, assume that $\|f_1 - f_2\|_1 > 0$. Thus for any h we have $(f_1-h)\cdot T_{12} + (h-f_2)\cdot T_{12} = \|f_1-f_2\|_1 > 0$.

Algorithm 5 - RANDOMIZED ESTIMATE.
Let

$$r = \frac{|(f_1-h)\cdot T_{12}|}{|(f_2-h)\cdot T_{12}|}.$$

With probability $1/(r+1)$ output f_1, otherwise output f_2.

(By convention, if $|(f_2-h)\cdot T_{12}| = 0$ then we take $r=\infty$ and output f_2 with probability 1).

Theorem 22 *Let $\mathcal{F} = \{f_1, f_2\}$. Let $f \in \mathcal{F}$ be the density output by Algorithm 5. Then*

$$\mathrm{E}\Big[\|f-g\|_1\Big] \le 2\,\mathrm{d}_1(g,\mathcal{F}) + \Delta,$$

where the expectation is taken only with respect to the coin tosses made by the algorithm.

Proof :
Without loss of generality assume that

$$f_2 = \mathrm{argmin}_{f\in\mathcal{F}} \|f-g\|_1.$$

First we bound the error of f_1 and later use it to bound the error of f. We have, by triangle inequality,

$$\|f_1-g\|_1 \le \|f_1-f_2\|_1 + \|f_2-g\|_1.$$

We can bound $\|f_1-f_2\|_1$ as follows

$$\begin{aligned}\|f_1-f_2\|_1 &= (f_1-f_2)\cdot T_{12} \\ &\le |(f_1-h)\cdot T_{12}| + |(f_2-h)\cdot T_{12}| \\ &= (r+1)|(f_2-h)\cdot T_{12}| \\ \le\ (r+1)|(f_2-g)\cdot T_{12}| &+ (r+1)|(g-h)\cdot T_{12}|.\end{aligned}$$

Thus,

$$\|f_1-g\|_1 \le (r+2)\|f_2-g\|_1 + (r+1)\Delta. \qquad (33)$$

Hence

$$\begin{aligned}\mathrm{E}\Big[\|f-g\|_1\Big] &= \frac{1}{r+1}\|f_1-g\|_1 + \frac{r}{r+1}\|f_2-g\|_1 \\ &\le 2\|f_2-g\|_1 + \Delta\end{aligned}$$

where in the last inequality we used (33). ■

5 Lower bound examples

In this section we construct an example showing that deterministic density selection algorithms based on test-functions cannot improve on the constant 3, that is, Theorems 2, 3, 4, 8 are tight. For algorithms that output mixtures (and hence randomized algorithms) the example yields a lower bound of 2, matching the constant in Theorem 22.

Lemma 23 *For every $\varepsilon' > 0$ there exist distributions f_1, f_2, and $g = h$ such that*

$$\|f_1 - g\|_1 \geq (3 - \varepsilon')\|f_2 - g\|_1,$$

and $f_1 \cdot T_{12} = -f_2 \cdot T_{12}$ and $h \cdot T_{12} = 0$.

Before we prove Lemma 23 let us see how it is applied. Consider the behavior of the algorithm on empirical distribution h for $\mathcal{F} = \{f_1, f_2\}$ and $\mathcal{F}' = \{f_1', f_2'\}$, where $f_1' = f_2$ and $f_2' = f_1$. Note that $T_{12}' = T_{21} = -T_{12}$ and hence

$$f_1' \cdot T_{12}' = -f_2' \cdot T_{12}' = f_1 \cdot T_{12} = -f_2 \cdot T_{12}.$$

Moreover, we have $h \cdot T_{12} = h \cdot T_{12}' = 0$. Note that all the test-functions have the same value for $\mathcal{F}$ and $\mathcal{F}'$. Hence a test-function based algorithm either outputs f_1 and f_1', or it outputs f_2 and $f_2' = f_1$. In both cases it outputs f_1 for one of the inputs and hence we obtain the following consequence.

Corollary 24 *For any $\varepsilon > 0$ and any deterministic test-function based algorithm there exist an input $\mathcal{F}$ and $h = g$ such that the output f_1 of the algorithm satisfies $\|f_1 - g\|_1 \geq (3 - \varepsilon)\mathrm{d}_1(g, \mathcal{F})$.*

Proof of Lemma 23:
Consider the following probability space consisting of of 4 atomic events A_1, A_2, A_3, A_4:

	A_1	A_2	A_3	A_4
f_1	0	$1/4+\varepsilon$	1/2	$1/4-\varepsilon$
f_2	$1/2+\varepsilon$	$1/4-\varepsilon$	0	1/4
$g=h$	1/2	1/2	0	0
T_{12}	-1	1	1	-1

Note that we have $f_1 \cdot T_{12} = -f_2 \cdot T_{12} = \frac{1}{2} + 2\varepsilon$, and $\|f_1 - g\|_1 = \frac{3}{2} - 2\varepsilon, \|f_2 - g\|_1 = \frac{1}{2} + 2\varepsilon$. The ratio $\|f_1 - g\|_1 / \|f_2 - g\|_1$ gets arbitrarily close to 3 as ε goes to zero. ■

Consider f_1 and f_2 from the proof of Lemma 23. Let $f = \alpha f_1 + (1-\alpha) f_2$ where $\alpha \geq 1/2$. For $0 < \varepsilon < 1/4$ we have $\|f - g\|_1 = 1/2 + \alpha - 2\varepsilon\alpha \geq 1 - 2\varepsilon$. By symmetry, for one of $\mathcal{F} = \{f_1, f_2\}$ and $\mathcal{F}' = \{f_1', f_2'\}$ (with $f_1' = f_2$ and $f_2' = f_1$), the algorithm outputs $\alpha f_1 + (1-\alpha) f_2$ with $\alpha \geq 1/2$, and hence we obtain the following.

Corollary 25 *For any $\varepsilon > 0$ and any deterministic test-function based algorithm which outputs a mixture there exist an input $\mathcal{F}$ and $h = g$ such that the output f of the algorithm satisfies $\|f - g\|_1 \geq (2 - \varepsilon)\mathrm{d}_1(g, \mathcal{F})$.*

Thus for two distributions the correct constant is 2 for randomized algorithms using test-functions. For larger families of distributions we do not know what the value of the constant is (we only know that it is from the interval $[2, 3]$).

Question 26 *What is the correct constant for deterministic test-function based algorithm which output a mixture? What is the correct constant for randomized test-function based algorithms?*

Next we construct an example showing that 9 is the right constant for Algorithm 1.

Lemma 27 *For every $\varepsilon' > 0$ there exist probability distributions $f_1, f_2, f_3 = f_3'$ and g such that*

$$\|f_1 - g\|_1 \geq (9 - \varepsilon')\|f_2 - g\|_1,$$

yet the Algorithm 1, for $\mathcal{F} = \{f_1, f_2, f_3, f_3'\}$, even when given the true distribution (that is, $h = g$) outputs f_1.

Proof :
Consider the following probability space with 6 events $A_1, \ldots, A_6$ and f_1, f_2 and g with the probabilities given by the following two tables:

	A_1	A_2	A_3
$g=h$	$2/3-21\varepsilon$	$1/9-2\varepsilon$	9ε
f_1	0	18ε	$2/3-12\varepsilon$
f_2	$2/3-30\varepsilon$	0	0
f_3	$2/3-21\varepsilon$	9ε	9ε
T_{12}	-1	1	1
T_{13}	-1	1	1
T_{23}	-1	-1	-1

	A_4	A_5	A_6
$g=h$	0	$2/9+14\varepsilon$	0
f_1	$2/9-13\varepsilon$	9ε	$1/9-2\varepsilon$
f_2	0	$2/9+14\varepsilon$	$1/9+16\varepsilon$
f_3	$2/9-4\varepsilon$	0	$1/9+7\varepsilon$
T_{12}	1	-1	-1
T_{13}	-1	1	-1
T_{23}	-1	1	1

Note that we have

$$\begin{aligned} f_1 \cdot T_{12} &= 7/9 - 14\varepsilon, & h \cdot T_{12} &= -7/9 + 14\varepsilon, \\ f_2 \cdot T_{12} &= -1, & f_1 \cdot T_{13} &= 1/3 + 30x, \\ h \cdot T_{13} &= -1/3 + 42x, & f_3 \cdot T_{13} &= -1 + 36x, \\ f_2 \cdot T_{23} &= -1/3 + 60x, & h \cdot T_{23} &= -5/9 + 28x, \\ f_3 \cdot T_{23} &= -7/9 + 14x. \end{aligned}$$

Hence f_1 wins over f_3, f_3 wins over f_2, and f_2 wins over f_1. Since $f_3 = f_3'$ we have that f_1 is the tournament winner. Finally, we have $\|f_1 - g\|_1 = 2 - 72\varepsilon$ and $\|f_2 - g\|_1 = 2/9 + 32\varepsilon$. As $\varepsilon \to 0$ the ratio $\|f_1 - g\|_1 / \|f_2 - g\|_1$ gets arbitrarily close to 9. ■

References

[DG85] Luc Devroye and László Györfi. *Nonparametric density estimation: the L_1 view*. Wiley series in probability and mathematical statistics. John Wiley & Sons, New York, 1985.

[DGL02] Luc Devroye, László Györfi, and Gábor Lugosi. A note on robust hypothesis testing. *IEEE Transactions on Information Theory*, 48(7):2111–2114, 2002.

[DL96] Luc Devroye and Gábor Lugosi. A universally acceptable smoothing factor for kernel density estimates. *Ann. Statist.*, 24(6):2499–2512, 1996.

[DL97] Luc Devroye and Gábor Lugosi. Nonasymptotic universal smoothing factors, kernel complexity and Yatracos classes. *Ann. Statist.*, 25(6):2626–2637, 1997.

[DL01] Luc Devroye and Gábor Lugosi. *Combinatorial methods in density estimation*. Springer Series in Statistics. Springer-Verlag, New York, 2001.

[DL04] Luc Devroye and Gábor Lugosi. Bin width selection in multivariate histograms by the combinatorial method. *Test*, 13:1–17, 2004.

[Ind06] Piotr Indyk. Stable distributions, pseudorandom generators, embeddings, and data stream computation. *Journal of the ACM*, 53(3):307–323, 2006.

[LHC07] Ping Li, Trevor J. Hastie, and Kenneth W. Church. Nonlinear estimators and tail bounds for dimension reduction. *Journal of Machine Learning Research*, 8:2497–2532, 2007.

[MŠ08] Satyaki Mahalanabis and Daniel Štefankovič. Approximating l_1-distances between mixture distributions using random projections. *arXiv.org*, `http://arxiv.org/abs/0804.1170`, April 2008.

[Sch47] Henry Scheffé. A useful convergence theorem for probability distributions. *Ann. Math. Statistics*, 18:434–438, 1947.

[VČ71] Vladimir N. Vapnik and Alexey J. Červonenkis. The uniform convergence of frequencies of the appearance of events to their probabilities. *Teor. Verojatnost. i Primenen.*, 16:264–279, 1971.

[Wol57] Jacob Wolfowitz. The minimum distance method. *The Annals of Mathematical Statistics*, 28:75–88, 1957.

[Yat85] Yannis G. Yatracos. Rates of convergence of minimum distance estimators and Kolmogorov's entropy. *Ann. Statist.*, 13(2):768–774, 1985.