Technical Report

Demixing empirical distribution functions

Alexander Munteanu,
Max Wornowizki

02/2014

technische universität
dortmund

SFB 876 Verfügbarkeit von
Information durch Analyse unter
Ressourcenbeschränkung

Part of the work on this technical report has been supported by Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center SFB 876 "Providing Information by Resource-Constrained Analysis", projects C3 and C4.

Speaker: Prof. Dr. Katharina Morik
Address: TU Dortmund University
Joseph-von-Fraunhofer-Str. 23
D-44227 Dortmund
Web: http://sfb876.tu-dortmund.de

Abstract

We consider the two-sample homogeneity problem where the information contained in two samples is used to test the equality of the underlying distributions. For instance, in cases where one sample stems from a simulation procedure modelling the data generating process of the other sample consisting of observed data, a mere rejection of the null hypothesis is unsatisfactory. Instead, the data analyst would like to know how the simulation can be improved while changing it as little as possible. Based on the popular Kolmogorov-Smirnov test and a general nonparametric mixture model, we propose an algorithm which determines an appropriate correction distribution function describing how the simulation procedure can be corrected. It is constructed in such a way that complementing the simulation sample by a given proportion of observations sampled from the correction distribution does not lead to a rejection of the null hypothesis of equal distributions when the modified and the observed sample are compared. We prove our algorithm to run in linear time and evaluate it on simulated and real spectrometry data showing that it leads to intuitive results. We illustrate its practical performance considering runtime as well as accuracy in a real world scenario.

1 Introduction

Consider the following scenario: We observe the sample $x_1, \ldots, x_{n_1} \in \mathbb{R}$ stemming from an unknown continuous distribution F. The underlying data generating process is modelled by a simulation procedure represented by the distribution G. To evaluate the quality of the simulation, consider n_2 simulated observations $y_1, \ldots, y_{n_2}$ drawn independently from G. If the simulation procedure works well, G resembles F and thus the samples are similar.

A standard nonparametric approach to test the equality of F and G is the two-sample Kolmogorov-Smirnov test. We denote the empirical distribution functions of the samples by F_e and G_e, respectively, and set $N = \frac{n_1 \cdot n_2}{n_1 + n_2}$. Setting $M = \mathbb{R}$ the null hypothesis H_0 is rejected if the test statistic

$$D_M(F_e, G_e) = \sqrt{N} \sup_{x \in M} |F_e(x) - G_e(x)|$$

exceeds an appropriately chosen critical value K_α.

In order to consider the procedure from a different perspective, we define for all $x \in M$ an upper boundary function U setting $U(x) = \min(1, F_e(x) + \frac{K_\alpha}{\sqrt{N}})$ for all $x \in M$, and in analogy define a lower boundary function L by $L(x) = \max(0, F_e(x) - \frac{K_\alpha}{\sqrt{N}})$. With these definitions the Kolmogorov-Smirnov test does not reject H_0 if and only if G_e is an element of the set

$$B = \{f : \mathbb{R} \to [0,1] | \forall x \in M : L(x) \leq f(x) \leq U(x)\}$$

called the confidence band.

In practice, a mere rejection of H_0 is not satisfying, because the data analyst would like to improve the simulation in case of rejection. Thus, the regions of undersampling respectively oversampling, i.e., the regions where G_e violates L or U, are of great interest.

In the present work, we introduce a nonparametric mixture model linking G and F by a distribution H, which represents all discrepancies between F and G. Thus, H provides valuable information on appropriate modifications to G. We propose an algorithm determining an empirical version of H and the mixing proportion in the mixture model under reasonable conditions.
There already exist various semi- and nonparametric suggestions on mixture models in the literature focussing on the estimation of the densities and the number of components in a mixture model of a given sample. Since these authors consider only single samples, they have to work in a multidimensional setting. As shown by Hall et al. [5], the quantities in a nonparametric mixture model with two components are not identifiable for one- and two-dimensional problems. The methods often rely on adjusted versions of the EM algorithm [9] or a Newton method [13] and sometimes make use of data transformations [6]. There also exist several nonparametric approaches to problems involving multiple samples and finite mixture models, as for example proposed by Kolossiatis et al. [7]. However, to the authors' knowledge, there is no literature addressing the problem sketched above in the context of mixture models. We close this gap by proposing a correction of one sample to resemble another sample based on the corresponding empirical distribution functions.

The remainder of this report is structured as follows: In Section 2 we propose a nonparametric mixture model related to the two-sample problem. We introduce several desirable properties of the model parameters and formulate two optimisation problems allowing to identify them. In Section 2 we present an algorithm to solve the problems introduced in the second section and provide intuitive explanations of the main ideas of each step of our algorithm. The proofs of correctness and linear runtime are conducted in Section 4. In Section 5 our procedure is applied to real and simulated data and the results are illustrated. Section 6 concludes with a summary and an outlook on possible future work.

2 Problem Definition

In this section a nonparametric mixture model is introduced. It links the distributions F and G by a third distribution H, which reflects all discrepancies between F and G. In order to be able to assess H, the model is transferred to an empirical equivalent. Thereafter, several constraints on parameters of the empirical model are motivated allowing to identify them properly.

To model the problem described in the introduction, we work with the fairly general two-component mixture model

$$F = \tilde{s} \cdot G + (1 - \tilde{s}) \cdot H \quad ,$$

where the so-called mixture proportion or shrinkage factor $\tilde{s}$ measures the degree of agreement of F and G while the distribution H represents all dissimilarities between F and G. Since F is fully described by G, $\tilde{s}$ and H, we are interested in identifying $\tilde{s}$ and H, because these quantities contain all relevant information for an appropriate modification of G.

It is clear that the choice $\tilde{s} = 0$ and $H = F$ solves the above equation. However, this solution is not of interest in our setting, because the data analyst wants to correct and not to discard the current simulation, which is often based on expert knowledge. This may give more insight into the data generating process itself and is thus preferable. In the other extreme case, $\tilde{s} = 1$, the simulation is correct and H is irrelevant. However, for any $\tilde{s} \in (0, 1)$ the corresponding H is unique and demixing F, that is estimating $\tilde{s}$ and H, provides useful information for improving the simulation.

Since the distributions F and G are not available in practice, we replace the corresponding distribution functions by the standard empirical estimators F_e and G_e. Combining the concept of distance used in the Kolmogorov-Smirnov test with the above mixture model, we propose to identify a shrinkage factor $s \in [0, 1]$ and a correction function $\mathcal{H}$ such that the function

$$\mathcal{F} = s \cdot G_e + (1 - s) \cdot \mathcal{H} \tag{1}$$

lies in the confidence band B and thus the Kolmogorov-Smirnov test would not reject H_0 if the distribution functions G_e and $\mathcal{F}$ were compared. Since $\mathcal{H}$ is a substitute for H, it should be a distribution function and hence lie in the set

$$\mathcal{M} = \left\{ f : \mathbb{R} \to [0, 1] \mid f \text{ monotone}, \lim_{x \to -\infty} f(x) = 0 \right\}.$$

Obviously, neither s nor $\mathcal{H}$ are unique in this situation. Hence, in the following we set some additional constraints and describe the problem in more detail allowing us to determine reasonable solutions.

Since we work with empirical distribution functions, all derived quantities are characterized by their values on the joint sample $x_1, \ldots, x_{n_1}, y_1, \ldots, y_{n_2}$. Therefore, instead of considering all functions $\mathcal{H} \in \mathcal{M}$, we restrict ourselves to those, which may be discontinuous only on $Z = \{z_1, \ldots, z_{n_1+n_2}\}$ consisting of the ordered joint sample. We denote this set of functions by $\mathcal{M}^* \subset \mathcal{M}$. This restriction is not very strong since the sample sizes in simulations are often quite large and we also avoid the inclusion of additional observation values, which would lead to a higher computational effort.

Motivated by the fact that the data analyst is interested in making as small changes as possible concerning the current simulation, we can make the mixture proportion s in the model identifiable by choosing s maximally such that the mixture $\mathcal{F}$ fits the observed data. This directly implies a minimal weight $1 - s$ for the correction function $\mathcal{H}$. We thus formulate **Problem 1**:

$$\begin{aligned} \max_{s \in [0,1]} : \quad & s \\ s.t. : \quad & \exists \mathcal{H} \in \mathcal{M}^* : \; s \cdot G_e + (1 - s) \cdot \mathcal{H} \in B \end{aligned}$$

Note that for $s^* = \frac{K_\alpha}{\sqrt{N}}$ and $\mathcal{H}^* = \frac{1}{1-s^*} \cdot L$ the property $s^* \cdot G_e + (1 - s^*) \cdot \mathcal{H}^* \in B$ holds. Thus, the optimal value of s, called s_{opt} in the following, is always greater than 0. Hence, the simulated data is always properly included in the mixture.
After Problem 1 is solved, the resulting mixture $\mathcal{F} = s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}$ lies in B. Since this does not imply the property $\lim_{x \to \infty} \mathcal{F}(x) = 1$, the function $\mathcal{H}$ could be an

improper distribution function. Therefore, there might exist several choices of $\mathcal{H}$ solving Problem 1 given s_{opt}. However, the pointwise minimal function $\mathcal{H}_{min} \in \mathcal{M}^*$ satisfying $\mathcal{F}_{min} = s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{min} \in B$ is unique. To find a reasonable distribution function $\mathcal{H}$, we propose to construct $\mathcal{H}$ by enlarging $\mathcal{H}_{min}$ in an adequate way, so that the final mixture is a proper empirical distribution function lying in B.

Before we formulate this enlargement as an optimisation problem, we want to point out that, quite intuitively, $\mathcal{H}_{min}$ should not be enlarged for small $z \in Z$. In particular, if $\mathcal{F}_{min}$ intersects the upper boundary U, adding mass before the maximal value $z \in Z$ where $\mathcal{F}_{min}(z)$ equals $U(z)$, $z_{meq} = \max_{z \in Z} \{z | \mathcal{F}_{min}(z) = U(z)\}$, leads to violations of U in z_{meq}. Note that in case of such an intersection, the global Kolmogorov-Smirnov distance on $M = \mathbb{R}$ between the final mixture and F_e will be the radius of the confidence band, regardless of the enlargement of $\mathcal{H}_{min}$. However, on subsets of $\mathbb{R}$ the distance measure can be improved if $\mathcal{H}_{min}$ is enlarged appropriately. Hence, we propose to identify z_{norm}, the smallest value after z_{meq} such that adding mass after z_{norm} decreases the Kolmogorov-Smirnov distance restricted to the set $M_{norm} = \{z \in Z | z \geq z_{norm}\}$. We then add the probability mass in such a way that the minimal distance $D_{\{z \geq z_{norm}\}}$ is attained. If there is no intersection between $\mathcal{F}_{min}(z)$ and $U(z)$, we set $z_{meq} = \min\{Z\}$ and proceed in the same way. Using the notations introduced above, finding a suitable distribution function $\mathcal{H}$ for a given value of s_{opt} can be formalised in **Problem 2**:

$$\begin{aligned} \min_{\mathcal{H} \in \mathcal{M}^*} : \quad & D_{M_{norm}}(\mathcal{F}, F_e) \\ s.t. : \quad & \mathcal{F} = s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H} \\ & \mathcal{F} \in B \\ & \mathcal{H} \geq \mathcal{H}_{min} \\ & \lim_{x \to \infty} \mathcal{H}(x) = 1 \end{aligned}$$

A solution to Problem 2 is called $\mathcal{H}_{opt}$. The corresponding final mixture is denoted by

$$\mathcal{F}_{opt} = s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{opt}. \tag{2}$$

Note that, even with these constraints, the solution to the problem of identifying $\mathcal{F} \in B$ may not be unique. Although the shrinkage factor s_{opt} is unique by its maximality property, there may be several optimal enlargements of $\mathcal{H}_{min}$ equally appropriate in the sense of the restricted Kolmogorov-Smirnov distance.

3 The Algorithm

In this section we propose an algorithm solving Problems 1 and 2 introduced in Section 2. At first, the main procedure is described. All subsequent subroutines called within the main algorithm are explained in more detail hereafter. Pseudocode is provided in order to illustrate the algorithms.

Algorithm 1 is our main procedure to solve Problems 1 and 2. It requires two sample vectors $x \in \mathbb{R}^{n_1}, y \in \mathbb{R}^{n_2}$ and a significance level α. At first, it calculates the empirical

distribution functions F_e and G_e of the samples and determines the critical value K_α at level α. In fact, K_α is the α-quantile of the distribution of $K = \sup_{t\in[0,1]} B(t)$, where $B(t)$ is a Brownian bridge [4]. For the typical significance levels $\alpha_1 = 0.05$ and $\alpha_2 = 0.01$ the critical values are $K_{\alpha_1} = 1.358$ and $K_{\alpha_2} = 1.628$, respectively. The values s and $\mathcal{F}$, candidates for the shrinkage factor s_{opt} and the final mixture $\mathcal{F}_{opt}$, are initialised and the lower bound for performing a binary search is set to s^*, cf. the description of Problem 1. The upper and lower boundary functions of the confidence band around F_e, U respectively L, are computed next. These steps can be considered as preprocessing and are carried out in the lines 1 and 2. The two-sample Kolmogorov-Smirnov test does not reject the null hypothesis of equal distributions if the relation $L \leq G_e \leq U$ holds. In this case the empirical distribution functions resemble each other well enough and the algorithm stops in line 4.

Algorithm 1: Demixing-Algorithm

Input : Observations $x \in \mathbb{R}^{n_1}$, $y \in \mathbb{R}^{n_2}$, significance level α
Output: Optimal shrinkage factor s_{opt},
optimal correcting function $\mathcal{H}_{opt} \in \mathcal{M}^*$

1 $Z \leftarrow (x, y)$; $K_\alpha \leftarrow K(\alpha)$; $N \leftarrow \sqrt{\frac{n_1 \cdot n_2}{n_1+n_2}}$; $l_b \leftarrow \frac{K_\alpha}{\sqrt{N}}$; $s \leftarrow 1$;
2 $F_e \leftarrow \text{EmpDistrFun}(x)$; $G_e \leftarrow \text{EmpDistrFun}(y)$; $\mathcal{F} \leftarrow G_e$;
$L \leftarrow \max\left\{0, G_e - \frac{K_\alpha}{\sqrt{N}}\right\}$;$U \leftarrow \min\left\{1, G_e + \frac{K_\alpha}{\sqrt{N}}\right\}$;
3 **if** $\forall z \in Z : L(z) \leq \mathcal{F}(z) \leq U(z)$ **then**
4 $\quad$ **return** $(s, 0)$
5 **repeat**
6 $\quad$ **if** $\exists z \in Z : \mathcal{F}(z) > U(z)$ **then**
7 $\quad\quad$ $(s, \mathcal{F}) \leftarrow$ **Shrink-Down**$(s, \mathcal{F})$;
8 $\quad$ **if** $\exists z \in Z : \mathcal{F}(z) < L(z)$ **then**
9 $\quad\quad$ $(s, \mathcal{F}) \leftarrow$ **Push-Up**$(s, \mathcal{F})$;
10 $\quad$ $(l_b, s, \mathcal{F}) \leftarrow$ **BinSearch**$(l_b, s, \mathcal{F})$;
11 **until** $\forall z \in Z : L(z) \leq \mathcal{F}(z) \leq U(z)$;
12 $\mathcal{F} \leftarrow$ **Normalise**$(\mathcal{F})$;
13 $\mathcal{H} \leftarrow (\mathcal{F} - s \cdot G_e)/(1 - s)$;
14 **return** $(s, \mathcal{H})$;

If the test rejects the null hypothesis, the algorithm carries out certain steps to determine an optimal mixture within the confidence band. To solve Problem 1, the following operations are applied iteratively in the main loop in lines 5 to 11: a candidate $\mathcal{F}$ lying above the upper boundary somewhere has to be shrunk, that is, multiplied by a factor from the interval $(0, 1)$, in order to correct the violation of U. This problem is addressed in line 7 in the so called *Shrink-Down* algorithm. On the contrary, a candidate falling below the lower boundary L must receive additional probability mass in appropriate regions. This is taken into account in line 9 by calling the *Push-Up* algorithm. The two operations are applied, whenever necessary, in the presented order. However, since they have opposite effects, some data situations require multiple executions of the Shrink-Down and the

Push-Up step. Iteration of these steps generates a decreasing sequence of upper bounds to s_{opt}. The well-known binary search technique embedded in the demixing algorithm in line 10 takes another approach by bounding s_{opt} from below and above. It is connected with the Shrink-Down and Push-Up step by using the current shrinkage factor s learned from them as an upper bound to s_{opt}. In return, the binary search updates s and $\mathcal{F}$, which are then passed to the Shrink-Down and Push-Up steps. The lower bound for the optimal shrinkage factor, l_b, is updated by the binary search itself.
Once the main loop is terminated, the optimal shrinkage factor s_{opt} and the corresponding minimal correction function $\mathcal{H}_{min}$ introduced on page 4 are determined and thus Problem 1 is solved. The normalisation step in line 12 takes care of Problem 2 returning an optimal mixture $\mathcal{F}_{opt}$. This allows to identify a reasonable correction function $\mathcal{H}_{opt}$ in line 13 by rearranging equation (2), which is returned afterwards together with the optimal shrinkage factor s_{opt}.

In the remainder of this section we describe the subroutines of the main algorithm in detail.

3.1 The Shrink-Down algorithm

This procedure is applied whenever a candidate $\mathcal{F}$ exceeds the upper boundary U at some point. Following the mixture model (1), it is intuitive to solve this problem by computing the maximal shrinkage value $s_d \in (0, 1)$ such that $s_d \cdot \mathcal{F}$ does not violate U any more. In other words, $\mathcal{F}$ is shrunk down. The maximal shrinkage factor to achieve this is $s_d = \min_{z \in Z} \left\{ \frac{U(z)}{\mathcal{F}(z)} \right\}$, where we set $\frac{a}{0} = \infty$ for every $a > 0$. The Shrink-Down subroutine presented in Algorithm 2 calculates this factor in line 1. Then, the total shrinkage and the candidate function $\mathcal{F}$ are updated accordingly and are eventually returned.

Algorithm 2: Shrink-Down

Input : Current values of $\mathcal{F}$ and s
Output: Updated values $\mathcal{F}$ and s

1 $s_d \leftarrow \min_{z \in Z} \left\{ \frac{U(z)}{\mathcal{F}(z)} \right\}$;
2 $s \leftarrow s_d \cdot s$;
3 $\mathcal{F} \leftarrow s_d \cdot \mathcal{F}$;
4 **return** $(s, \mathcal{F})$;

3.2 Push-Up algorithm

The Push-Up step presented in Algorithm 3 is carried out whenever the current candidate $\mathcal{F}$ violates the lower boundary L. In order to increase the values of the mixture in the problematic regions, probability mass must be added there. Note that $\mathcal{F}$ may lie below the lower boundary of the confidence band before the smallest value $z \in Z$ where $\mathcal{F}(z)$ equals $U(z)$, called $z_{eq} = \min_{z \in Z} \{z | U(z) = \mathcal{F}(z)\}$, as well as after that point. However, these two cases have a crucial difference. Adding probability mass before z_{eq} leads to a

new violation of the upper boundary U in z_{eq}, while adding mass after z_{eq} does not imply this problem. In order to distinguish between these cases, the algorithm first identifies z_{eq} in line 1. If the mixture candidate $\mathcal{F}$ equals G_e by initialisation, z_{eq} exists because $z_{eq} = \max(Z)$ holds due to $\mathcal{F}(\max(Z)) = G_e(\max(Z)) = 1 = U(\max(Z))$. As we will argue later in Lemma 4, z_{eq} is also well defined after modifications of $\mathcal{F}$.

If there are violations of L before z_{eq}, a shrinkage is necessary. Thus, keeping in mind Problem 1, the maximal shrinkage factor s_u must be identified, so that the residuals to L before z_{eq} do not exceed the residual to U in z_{eq} after shrinking. Otherwise, adding appropriate probability mass will cause a violation of U in z_{eq}. More formally, the shrinkage factor

$$\begin{aligned} s_u \quad = \max_{s \in [0,1]} \quad & \{s \mid \forall z < z_{eq} : \\ & L(z) - s \cdot \mathcal{F}(z) \leq U(z_{eq}) - s \cdot \mathcal{F}(z_{eq})\} \end{aligned}$$

must be determined. Basic arithmetic transformations of the constraint yield $s_u = \min_{z < z_{eq}} \left\{ \frac{\mathcal{F}(z_{eq}) - L(z)}{\mathcal{F}(z_{eq}) - \mathcal{F}(z)} \right\}$. After s_u is determined in line 3, the shrinkage factor s as well as $\mathcal{F}$ are updated.
In order to shift the current candidate $\mathcal{F}$ appropriately, first the positive residuals to L denoted by $d(z) = \max\{0, L(z) - \mathcal{F}(z)\}$ are computed for all $z \in Z$. These are the minimal amounts which must be added to $\mathcal{F}$ so that the lower boundary L is no longer violated. The residuals d are added to the current correction term $\mathcal{F} - s \cdot G_e$ and the sum is minimally monotonised, cf. line 7. The result, denoted by $\mathcal{H}$, is added to $s \cdot G_e$ yielding the new candidate mixture $\mathcal{F}$.

Algorithm 3: Push-Up

Input : Current values of $\mathcal{F}$ and s
Output: Updated values $\mathcal{F}$ and s

1 $z_{eq} \leftarrow \min_{z \in Z} \{z \mid U(z) = \mathcal{F}(z)\}$;
2 **if** $\exists z < z_{eq} : \mathcal{F}(z) < L(z)$ **then**
3 $\quad s_u \leftarrow \min_{z < z_{eq}} \left\{ \frac{\mathcal{F}(z_{eq}) - L(z)}{\mathcal{F}(z_{eq}) - \mathcal{F}(z)} \right\}$;
4 $\quad s \leftarrow s_u \cdot s$;
5 $\quad \mathcal{F} \leftarrow s_u \cdot \mathcal{F}$;
6 $\forall z \in Z : d(z) \leftarrow \max\{0, L(z) - \mathcal{F}(z)\}$;
7 $\forall z \in Z : \mathcal{H}(z) \leftarrow \max_{z' \leq z} \{\mathcal{F}(z') - s \cdot G_e(z') + d(z')\}$;
8 $\mathcal{F} \leftarrow s \cdot G_e + \mathcal{H}$;
9 **return** $(s, \mathcal{F})$;

3.3 Binary search algorithm

The binary search step presented in Algorithm 4 is called at the end of every iteration in the main loop. Its input consists of l_b and u_b, the current lower respectively upper

bound for s_{opt}. While l_b is derived from previous binary search steps, u_b is set to the current value of s. The algorithm computes the average of the given bounds in line 1. Using this candidate, the minimum monotone step function $\mathcal{H}_b$ is computed such that $\mathcal{F}_b = s_b \cdot G_e + \mathcal{H}_b \geq L$ holds, cf. lines 2 and 3. This is done in analogy to the corresponding lines in the Push-Up step.

If $\mathcal{F}_b$ violates the upper boundary U, then, by minimality of $\mathcal{H}_b$, no monotone step function for the shrinkage factor s_b can exist such that the corresponding mixture lies within the confidence band B. Therefore, as implied by the monotonicity property proved in Lemma 1 below, it holds that $s > s_b > s_{opt}$. Thus, in this case the algorithm updates s to s_b as a new upper bound for s_{opt} and sets the current mixture candidate to $\mathcal{F}_b$ in lines 6 and 7. Otherwise, again by Lemma 1, the relation $s_{opt} \geq s_b > l_b$ must hold, since there exists a monotone step function for the shrinkage factor s_b leading to a mixture in B. Thus, s_b is a better lower bound to s_{opt} so that l_b is updated by s_b, while all other quantities are kept.

Algorithm 4: BinSearch

Input : current lower and upper bounds to s_{opt}, l_b respectively u_b
Output: Updated values $\mathcal{F}$, s and l_b

1 $s_b \leftarrow (l_b + u_b)/2$;
2 $\forall z \in Z : d(z) \leftarrow \max\{0, L(z) - s_b \cdot G_e(z)\}$;
3 $\forall z \in Z : \mathcal{H}_b(z) \leftarrow \max_{z \leq z'}\{d_u(z')\}$;
4 $\mathcal{F}_b \leftarrow s_b \cdot G_e(z) + \mathcal{H}_b$;
5 **if** $\exists z \in Z : \mathcal{F}_b(z) > U(z)$ **then**
6 $\quad s \leftarrow s_b$;
7 $\quad \mathcal{F} \leftarrow \mathcal{F}_b$;
8 **else**
9 $\quad l_b \leftarrow s_b$;
10 **return** $(l_b, s, \mathcal{F})$;

3.4 Normalisation step

As we will show in Theorem 6 below, Problem 1 is solved when the loop of Algorithm 1 (lines 5 to 11) stops. At this point, the current value of s is the optimal shrinkage factor s_{opt}, while the current mixture is $\mathcal{F} = s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{min}$ and lies within the confidence band. However, as pointed out in the description of Problem 2, $\mathcal{F}$ may not be a proper distribution function since $\lim_{x \to \infty} \mathcal{F}(x) < 1$ may hold. This deficiency is corrected by the normalisation step presented in Algorithm 5.

To check whether $\mathcal{F}$ must be enlarged, the algorithm computes z_{meq}, the maximal value $z \in Z$ where $\mathcal{F}(z)$ equals $U(z)$. When there is no intersection of $\mathcal{F}$ and U, the algorithms sets $z_{meq} = \min(Z)$. If $z_{meq} = \max(Z)$ is satisfied, the property $\mathcal{F}(\max(Z)) = \mathcal{F}(z_{meq}) = U(z_{meq}) = U(\max(Z)) = 1$ holds, so no further corrections are necessary and $\mathcal{F}$ is returned.

Algorithm 5: Normalise

Input : Current value of $\mathcal{F}$
Output: Final mixture $\mathcal{F}$

1 **if** $\forall z \in Z : \mathcal{F}(z) < U(z)$ **then**
2 $\quad z_{meq} \leftarrow \min(Z)$;
3 **else**
4 $\quad z_{meq} \leftarrow \max_{z \in Z}\{z \,|\, U(z) = \mathcal{F}(z)\}$;
5 **if** $z_{meq} \neq \max(Z)$ **then**
6 $\quad z_{norm} \leftarrow \min_{z > z_{meq}} \{z \,|\, \mathcal{F}(z) < F_e(z)\}$;
7 $\quad \forall z \geq z_{norm} : d(z) \leftarrow \min\{F_e(z) - \mathcal{F}(z), 1 - \mathcal{F}(\max(Z))\}$;
8 $\quad$ **if** $\max_{z \geq z_{norm}} \{-d(z)\} \geq 1 - \mathcal{F}(\max(Z))$ **then**
9 $\quad\quad \tilde{z} \leftarrow \max_{z \geq z_{norm}} \{z \,|\, -d(z) \geq 1 - \mathcal{F}(\max(Z))\}$;
10 $\quad\quad z_{norm} \leftarrow \min_{z > \tilde{z}}\{z \,|\, d(z) > 0\}$;
11 $\quad \forall z \geq z_{norm} : \mathcal{H}_{norm}(z) \leftarrow \max\left\{0, \left(\max_{z_{norm} \leq z' \leq z} \{d(z')\} + \min_{z'' \geq z} \{d(z'')\}\right)/2\right\}$;
12 $\quad \forall z < z_{norm} : \mathcal{H}_{norm}(z) \leftarrow 0$;
13 $\quad \mathcal{F} \leftarrow \mathcal{F} + \mathcal{H}_{norm}$;
14 **return** $(\mathcal{F})$;

Otherwise, as stated in the motivation to Problem 2, adding any probability mass before z_{meq} would lead to a violation of U in z_{meq}. Since s_{opt} is already determined, such a violation cannot be repaired by further shrinking as in the Push-Up step. Thus, probability mass has to be added after z_{meq}. In fact, the region where mass should be added can be restricted even further. Therefore, we consider the smallest value $z \in Z$ such that $z > z_{meq}$ and $\mathcal{F}(z) < F_e(z)$ holds and denote it by z_{norm}. Since adding mass between z_{meq} and z_{norm} cannot reduce the Kolmogorov-Smirnov distance between F_e and $\mathcal{F}$, we focus on all $z \geq z_{norm}$ in the following.

Hence, the deviations $d(z) = F_e(z) - \mathcal{F}(z)$ are computed for all $z \geq z_{norm}$ in line 7. Deviations above the remaining mass $1 - \mathcal{F}(\max(Z))$ are decreased to this value. Hereafter, the algorithm checks whether the maximal increase of $\mathcal{F}$ above F_e, the maximum of all negative deviations $-d(z)$, is greater or equal to the maximal decrease of $\mathcal{F}$ below F_e, namely $1 - \mathcal{F}(\max(Z))$. As long as this is the case, adding probability mass will not decrease the Kolmogorov-Smirnov distance. Hence, in line 9 the algorithm determines the last position where the above property holds and updates z_{norm} to be greater than this position. This yields the set $M_{norm} = \{z \in Z | z \geq z_{norm}\}$, where a reduction of the Kolmogorov-Smirnov distance is possible. At the latest, M_{norm} is the last region where $\mathcal{F}$ lies below F_e.

In order to compute a distribution function $\mathcal{H}_{norm}$, which has to be added to the remaining region M_{norm}, the residuals d are considered on this set. Determining $\mathcal{H}_{norm}$ there may be seen as an L_∞ isotonic regression problem. Since we work in the setting of distribution functions, a monotone function should be constructed, which fits the residuals $d(z)$ best

in the sense of the L_∞-norm. Unweighted isotonic regression problems under the L_∞ norm can be efficiently solved in linear time by a simple approach, which is referred to as *Basic* by Stout [12]. This method is applied in line 11 of Algorithm 5. For each residual it computes the maximum of all previous values and the minimum of all subsequent values and determines the regression value as the average of these two quantities.

Note that a solution to the isotonic regression may be negative, but the distribution function $\mathcal{H}_{opt}$ must be nonnegative. However, as we will prove in Lemma 5, setting all of its negative values to 0 results in an optimal solution to the isotonic regression problem constraint to nonnegativity. Since no correction is applied before z_{norm}, $\mathcal{H}_{norm}$ is set to 0 before z_{norm} in line 12. Finally, $\mathcal{F}$ is updated and returned.

4 Analysis of the algorithms

In this section theoretical results for the algorithms of Section 3 are provided. Among other things, we prove a monotonicity property allowing to apply the binary search technique to Problem 1 and demonstrate that the Shrink-Down and Push-Up step always lead to upper bounds on s_{opt} proving their correctness. While the first part of this section deals with the correctness of our algorithm, the second one presents its runtime analysis.

4.1 Preliminaries

In the following paragraph we introduce the essential notations used repeatedly in our proofs. The shrinkage factor of G_e, the correction function and the mixture candidate after the k-th iteration of the main loop of Algorithm 1 (lines 5 to 11) are denoted by s_k, $\mathcal{H}_k$ and $\mathcal{F}_k = s_k \cdot G_e + \mathcal{H}_k$, respectively. In order to initialise them, we set $s_0 = 1$, $\mathcal{H}_0 = 0$ and $\mathcal{F}_0 = G_e$. Let $s_{d,k}$ denote the update of the shrinkage factor determined in the Shrink-Down step in the k-th iteration. Whenever this update is not computed, we set $s_{d,k} = 1$. The update of the shrinkage factor determined in the Push-Up step of the k-th iteration is called $s_{u,k}$ and treated in the same way.

4.2 Correctness of the algorithm

Our first result, mainly proving the correctness of the binary search step, shows that the property of lying within the confidence band is monotone in s. In other words, for any $s > s_{opt}$ a corresponding mixture violates a boundary of B, while for $s \leq s_{opt}$ it is always possible to find a mixture lying in B.

Lemma 1. $\exists \mathcal{H} \in \mathcal{M}^* : s \cdot G_e + (1-s) \cdot \mathcal{H} \in B \Leftrightarrow s \in [0, s_{opt}]$.

Proof. First we recall the definition of Problem 1 from page 3:

$$\begin{aligned} \max_{s\in[0,1]}: \quad & s \\ s.t.: \quad & \exists \mathcal{H} \in \mathcal{M}^* : \forall z \in Z : \\ & L(z) \leq \ s \cdot G_e(z) + (1-s) \cdot \mathcal{H}(z) \leq U(z) \end{aligned} \tag{3}$$

where $\mathcal{M}^*$ denotes the set of all nondecreasing, nonnegative step functions varying on Z only and converging to 0 for $x \to -\infty$.

Now we introduce an alternative characterization of s_{opt} by Problem A:

$$\begin{aligned} \max_{s\in[0,1]}: \quad & s \\ s.t.: \quad & \forall z \in Z : s \cdot G_e(z) \leq U(z) && \text{(4a)} \\ & \forall z' < z'' \in Z : \\ & L(z') - s \cdot G_e(z') \leq U(z'') - s \cdot G_e(z'') && \text{(4b)} \end{aligned}$$

Before we proceed to proving the proposition, we show the equivalence of Problem 1 and Problem A. For this sake, choose an arbitrary $s \in [0,1]$ such that (3) holds. Then for all $z \in Z$ it follows that $s \cdot G_e(z) \leq U(z) - (1-s) \cdot \mathcal{H}(z) \leq U(z)$ by nonnegativity of $(1-s) \cdot \mathcal{H}$, which proves that the inequality (4a) holds. Furthermore, choose $z' < z''$ from Z arbitrarily. Then $L(z') - s \cdot G_e(z') \leq (1-s) \cdot \mathcal{H}(z') \leq (1-s) \cdot \mathcal{H}(z'') \leq U(z'') - s \cdot G_e(z'')$ follows by monotonicity of $\mathcal{H}$. Thus (4b) is also respected.

For the other direction let $s \in [0,1]$ respect constraints (4a) and (4b). From (4a) it is clear that $s \cdot G_e(z)$ never exceeds the upper boundary. From (4b) we know that correcting any deficiency to the lower boundary L is possible without violating the upper boundary U on subsequent positions. Thus choosing $(1-s) \cdot \mathcal{H}(z) =$ $\max\left\{0, \max_{z^* \leq z}\{L(z^*) - s \cdot G_e(z^*)\}\right\}$ will result in a mixture within the confidence band. This means that (3) holds.

We now make use of the above equivalence of Problem 1 and Problem A to prove the proposition:

$$\exists \mathcal{H} \in \mathcal{M}^* : s \cdot G_e + (1-s) \cdot \mathcal{H} \in B \Leftrightarrow s \in [0, s_{opt}].$$

If $s \in (s_{opt}, 1]$ the property $s \cdot G_e + (1-s) \cdot \mathcal{H} \notin B$ immediately follows by definition of s_{opt}. So let $s \in [0, s_{opt}]$ be arbitrarily chosen and note that both constraints (4a) and (4b) are respected for s_{opt}. From this we can deduce that both conditions must also hold for s since $\forall z \in Z : s \cdot G_e(z) \leq s_{opt} \cdot G_e(z) \leq U(z)$ and furthermore for all $z', z'' \in Z$ with $z' < z''$ it follows

$$\begin{aligned} L(z') - s \cdot G_e(z') &= L(z') - s_{opt} \cdot G_e(z') - (s - s_{opt}) \cdot G_e(z') \\ &\leq U(z'') - s_{opt} \cdot G_e(z'') \underbrace{-(s - s_{opt})}_{>0} \cdot G_e(z') \\ &\leq U(z'') - s_{opt} \cdot G_e(z'') - (s - s_{opt}) \cdot G_e(z'') \\ &= U(z'') - s \cdot G_e(z''). \end{aligned}$$

Hence, $L(z') - s \cdot G_e(z') \leq U(z'') - s \cdot G_e(z'')$ holds.

Since, as argued before, constraints (4a) and (4b) are equivalent to constraint (3), there exists an $\mathcal{H}$ for which the mixture $s \cdot G_e + (1-s) \cdot \mathcal{H}$ lies in B, which completes the proof. □

In the next lemma, the correction function $\mathcal{H}_k$ computed in the k-th iteration of the main loop for the shrinkage factor s_k is considered. We prove that $\mathcal{H}_k$ is indeed the minimal function in $\mathcal{M}^*$ resolving violations of the lower boundary L. This result contributes to the correctness of our construction of $\mathcal{H}_{min}$ and is used in the following proofs.

Lemma 2. *$\mathcal{H}_k$ is the minimal function $\mathcal{H} \in M^*$ satisfying $s_k \cdot G_e + \mathcal{H} \geq L$.*

Proof. Let $\mathcal{H}_{k,min} \in \mathcal{M}^*$ be the minimal function with the property $s_k \cdot G_e + \mathcal{H}_{k,min} \geq L$. To prove the result we must thus show $\mathcal{H}_k = \mathcal{H}_{k,min}$. Now, the correction function $\mathcal{H}_k$ is either computed in the binary search step or in the Push-Up step. In the first case, the residuals between $s_k \cdot G_e$ and the lower boundary L are determined and then minimally monotonised, cf. lines 2 and 3 of Algorithm 4. This monotonisation is performed by considering the maximum of preceding values and is therefore minimal. Hence, this procedure must yield $\mathcal{H}_{k,min}$. In the remainder of this proof we thus treat the second case, namely the computation of $\mathcal{H}_k$ in the Push-Up step.

Following the lines 6 to 9 in Algorithm 3, we denote the positive deficiencies to L after the shrinking in the Push-Up step of iteration k by $d_k = \max(0, L - s_{d,k} \cdot s_{u,k} \cdot \mathcal{F}_{k-1})$. Setting $\tilde{\mathcal{F}}_k = s_{d,k} \cdot s_{u,k} \cdot \mathcal{F}_{k-1} + d_k$, the correction function $\mathcal{H}_k$ can be expressed as $\mathcal{H}_k = \operatorname{mon}\left(\tilde{\mathcal{F}}_k - s_k \cdot G_e\right)$. Thereby, $\operatorname{mon}(f)$ denotes the minimal monotone function such that $\operatorname{mon}(f) \geq f$. This monotonisation is performed analogically to the one in the binary search step by considering the maximum of preceding values. Note that the monotonising operator is itself monotone, that is, $\operatorname{mon}(f_1) \leq \operatorname{mon}(f_2)$ holds if $f_1 \leq f_2$. We show the proposition by induction:

$\mathbf{k = 1}$: By assumption, $\mathcal{H}_1$ is computed in the Push-Up step, so $s_1 = s_{d,1} \cdot s_{u,1}$ holds. In addition, $\mathcal{F}_0$ is defined by $\mathcal{F}_0 = G_e$. Hence, $d_1 = \max(0, L - s_{d,1} \cdot s_{u,1} \cdot \mathcal{F}_0) = \max(0, L - s_1 \cdot G_e) \leq \mathcal{H}_{1,min}$ must hold, since the last inequality holds by definition of $\mathcal{H}_{1,min}$. Because of $\tilde{\mathcal{F}}_1 = s_1 \cdot G_e + d_1$ we obtain $\mathcal{H}_1 = \operatorname{mon}(\tilde{\mathcal{F}}_1 - s_1 \cdot G_e) = \operatorname{mon}(d_1) \leq \operatorname{mon}(\mathcal{H}_{1,min}) = \mathcal{H}_{1,min}$, where the inequality follows by the monotonicity of the monotonising operator. Thus $\mathcal{H}_1 \leq \mathcal{H}_{1,min}$ is established. To prove the other inequality, note that $\mathcal{H}_1 \in \mathcal{M}^*$ and $\mathcal{H}_1 = \operatorname{mon}(d_1) \geq d_1$. Hence, $\mathcal{H}_{1,min} \leq \mathcal{H}_1$ follows by the definition of $\mathcal{H}_{1,min}$. Altogether, we get $\mathcal{H}_{1,min} = \mathcal{H}_1$.

$\mathbf{k-1 \Rightarrow k}$: The shrink updates $s_{d,k}$ and $s_{u,k}$ are bounded by 1 by construction and thus the inequality $s_k \leq s_{d,k} \cdot s_{u,k} \cdot s_{k-1} \leq s_{k-1}$ holds. Hence, the shrinkage factor s_k does not increase in k and therefore the corresponding minimal correction function $\mathcal{H}_{k,min}$ does not decrease in k. Thus, we get $\mathcal{H}_{k,min} \geq \mathcal{H}_{k-1,min} \geq s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1,min}$. The correctness of the $(k-1)$-th step assumed by the induction principle yields $\mathcal{H}_{k-1,min} = \mathcal{H}_{k-1}$ resulting in $\mathcal{H}_{k,min} \geq s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1}$. Rewriting $s_{d,k} \cdot s_{u,k} \cdot \mathcal{F}_{k-1}$ to $s_k \cdot G_e + s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1}$ allows to interpret d_k as the minimal function which must be added to $s_k \cdot G_e + s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1}$ so

that the lower boundary L of the confidence band is not violated any more. Together with $\mathcal{H}_{k,min} \geq s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1}$ established above this implies $s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1} + d_k \leq \mathcal{H}_{k,min}$. Since in addition d_k is by construction minimally chosen such that $\tilde{\mathcal{F}}_k = s_{d,k} \cdot s_{u,k} \cdot \mathcal{F}_{k-1} + d_k \geq L$ holds, we deduce

$$L - s_k \cdot G_e \leq \tilde{\mathcal{F}}_k - s_k \cdot G_e = s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1} + d_k \leq \mathcal{H}_{k,min}.$$

Applying the monotonising operator and exploiting its monotonicity this implies

$$\begin{aligned} L - s_k \cdot G_e &\leq \operatorname{mon}(L - s_k \cdot G_e) \\ &\leq \underbrace{\operatorname{mon}(s_{d,k} \cdot s_{u,k} \cdot \mathcal{H}_{k-1} + d_k)}_{=\mathcal{H}_k} \\ &\leq \operatorname{mon}(\mathcal{H}_{k,min}) \\ &= \mathcal{H}_{k,min}, \end{aligned} \tag{5}$$

and therefore $\mathcal{H}_k \leq \mathcal{H}_{k,min}$. To prove $\mathcal{H}_k \geq \mathcal{H}_{k,min}$, note that $\mathcal{H}_k$ is a function in $\mathcal{M}^*$. The inequalities (5) imply $L \leq s_k \cdot G_e + \mathcal{H}_k$. So, by definition of $\mathcal{H}_{k,min}$, $\mathcal{H}_k \geq \mathcal{H}_{k,min}$ follows and thus overall $\mathcal{H}_k = \mathcal{H}_{k,min}$ holds. □

The next result shows that the Shrink-Down step always leads to overall shrinkage factors not lower than s_{opt} and therefore may be used as an improved upper bound for s_{opt} in the binary search procedure.

Lemma 3. *If $s_k > s_{opt}$ then $s_{d,k+1} \cdot s_k \geq s_{opt}$.*

Proof. The proposition is trivial for $s_{d,k+1} = 1$ so in the following $s_{d,k+1} < 1$ is assumed. This means that the $(k+1)$-th Shrink-Down step is not skipped but executed. So $\mathcal{F}_k$ must lie above the upper boundary U for some values. Together with the definition of $s_{d,k+1}$, this ensures the existence of a $z_{eq} \in Z$ such that $s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) = U(z_{eq})$ holds. In the following we consider the two possible cases for the correction function $\mathcal{H}_k = \mathcal{F}_k - s_k \cdot G_e$:

$\mathcal{H}_\mathbf{k}(\mathbf{z_{eq}}) = \mathbf{0}$: In this case, using the definition of z_{eq} and $\mathcal{F}_k$, we deduce

$$\begin{aligned} U(z_{eq}) &= s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) \\ &= s_{d,k+1} \cdot (s_k \cdot G_e(z_{eq}) + \mathcal{H}_k(z_{eq})) \\ &= s_{d,k+1} \cdot s_k \cdot G_e(z_{eq}) \\ &< G_e(z_{eq}), \end{aligned}$$

where the last inequality follows since $0 < s_{d,k+1} < 1$, $0 < s_k \leq 1$ and $0 < G_e(z_{eq})$. The latter is satisfied, because otherwise $0 = G_e(z_{eq})$ and $\mathcal{H}_k(z_{eq}) = 0$ immediately imply $0 = U(z_{eq})$, which is a contradiction to the positivity of U.

The calculations above show that the function G_e lies above the upper boundary U in z_{eq} before any shrinking. However, the first Shrink-Down step would solve this problem and because of $\mathcal{H}_k(z_{eq}) = 0$ there cannot be a new violation of U in z_{eq} in subsequent steps. Hence, $k = 0$ and consequently $s_k = 1$ must hold. The proposition $s_k \cdot s_{d,k+1} = s_{d,1} \geq s_{opt}$ holds in this case, since $s_{d,1}$ is by construction the maximal shrinkage factor for avoiding violations of U before adding any correction function.

$\mathcal{H}_\mathbf{k}(\mathbf{z_{eq}}) > \mathbf{0}$: Let $\tilde{\mathcal{H}} \in \mathcal{M}^*$ be the minimal function one must add to $s_{d,k+1} \cdot s_k \cdot G_e$ in order to correct violations of the lower boundary L. Due to $s_{d,k+1} \leq 1$ we get $s_{d,k+1} \cdot s_k \cdot G_e \leq s_k \cdot G_e$ and thus $\tilde{\mathcal{H}} \geq \mathcal{H}_k$ holds by minimality of $\mathcal{H}_k$ shown in Lemma 2. This allows to prove

$$\begin{aligned} U(z_{eq}) &= s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) \\ &= s_{d,k+1} \cdot (s_k \cdot G_e(z_{eq}) + \mathcal{H}_k(z_{eq})) \\ &< s_{d,k+1} \cdot s_k \cdot G_e(z_{eq}) + \mathcal{H}_k(z_{eq}) \\ &\leq s_{d,k+1} \cdot s_k \cdot G_e(z_{eq}) + \tilde{\mathcal{H}}(z_{eq}) \quad . \end{aligned}$$

Thus, $s_{d,k+1} \cdot s_k \cdot G_e + \tilde{\mathcal{H}}$ violates the upper boundary of the confidence band and thus does not lie in B. By minimality of $\tilde{\mathcal{H}}$ Lemma 1 yields $s_{d,k+1} \cdot s_k > s_{opt}$, which completes the proof. □

The following proposition concerns the additional shrinkage performed in the Push-Up step. Similarly to Lemma 3 it states that a Push-Up step cannot lead to factors below s_{opt} and therefore the correctness of using the overall shrinkage factor to improve the upper bound on s_{opt}.

Lemma 4. *If* $s_{d,k+1} \cdot s_k > s_{opt}$ *then* $s_{u,k+1} \cdot s_{d,k+1} \cdot s_k \geq s_{opt}$.

Proof. The statement is immediately given for $s_{u,k+1} = 1$. It is also clear in case of $k = 0$ by construction of the shrink update $s_{u,1}$. So in the following let $s_{u,k+1} < 1$ and $k \geq 1$ hold. We prove the proposition by contradiction so assume

$$s_{d,k+1} \cdot s_k > s_{opt} > s_{d,k+1} \cdot s_{u,k+1} \cdot s_k \quad . \tag{6}$$

Consider the preceding candidate $\mathcal{F}_k$. $\mathcal{F}_k \notin B$ must hold, because otherwise the algorithm would have stopped after k steps. Furthermore $\mathcal{F}_k \geq L$ is guaranteed by construction of the Push-Up and binary search steps. Therefore $\mathcal{F}_k$ must violate the upper boundary U in the assumed case $k \geq 1$. Thus, a Shrink-Down step was executed before the current Push-Up step. Hence, the point

$$z_{eq} = \min_{z \in Z} \{z | s_{d,k+1} \cdot \mathcal{F}_k(z) = U(z)\}$$

is well defined, as pointed out in the description of the Push-Up step. The assumption $s_{u,k+1} < 1$ implies that a Push-Up step is carried out and $\exists z \in Z : z < z_{eq}$. By definition of z_{eq}, each $z < z_{eq}$ satisfies $s_{d,k+1} \cdot \mathcal{F}_k(z) < U(z) \leq U(z_{eq})$ and hence we deduce that

$$\forall z < z_{eq} : s_{d,k+1} \cdot \mathcal{F}_k(z) - U(z_{eq}) < 0. \tag{7}$$

Now consider the point

$$z' = \max \left\{ \operatorname*{argmax}_{z < z_{eq}} (L(z) - s_{d,k+1} \cdot s_{u,k+1} \cdot \mathcal{F}_k(z)) \right\}.$$

By the definition of

$$s_{u,k+1} = \max_{s \in [0,1]} \{s | \forall z < z_{eq} : L(z) - s \cdot s_{d,k+1} \cdot \mathcal{F}_k(z) \leq U(z_{eq}) \cdot (1 - s)\}$$

it follows that

$$L(z') - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') = U(z_{eq}) \cdot (1 - s_{u,k+1}). \tag{8}$$

Also consider $z'' = \min \left\{ \underset{z \leq z_{eq}}{\operatorname{argmax}} \mathcal{H}_k(z) \right\}$. Using the minimal property of $\mathcal{H}_k$ proved in Lemma 2, for $k \geq 1$ one can deduce $\mathcal{F}_k(z'') = L(z'')$, which implies $z'' < z_{eq}$. For all $z \leq z''$ we obtain

$$\begin{aligned} L(z) - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z) &\leq \mathcal{F}_k(z) - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z) \\ &= (1 - s_{u,k+1} \cdot s_{d,k+1}) \cdot \mathcal{F}_k(z) \\ &\leq (1 - s_{u,k+1} \cdot s_{d,k+1}) \cdot \mathcal{F}_k(z'') \\ &= \mathcal{F}_k(z'') - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z'') \\ &= L(z'') - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z''), \end{aligned}$$

where the first inequality holds because of $\mathcal{F}_k \geq L$ by construction of $\mathcal{F}_k$. Combining this result with the already mentioned fact that $z'' < z_{eq}$ holds, we get $z' \geq z''$. Together with the monotonicity of $\mathcal{H}_k$ and the definition of z'' we deduce

$$\mathcal{H}_k(z') = \mathcal{H}_k(z'') = \mathcal{H}_k(z_{eq}). \tag{9}$$

We now combine (6), (7), (8) and (9) to prove the proposition. By Lemma 3 and $s_{opt} > s^* > 0$ shown on page 3 the inequality $s_{d,k+1} \cdot s_k > 0$ holds. Thus, we can define $s_{u2} = \frac{s_{opt}}{s_{d,k+1} \cdot s_k}$ and inequality (6) implies

$$1 \geq s_{u2} > s_{u,k+1}, \tag{10}$$

which allows us to show

$$\begin{aligned} & L(z') - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') \\ = \; & L(z') - s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') + s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') \\ \overset{(8)}{=} \; & U(z_{eq}) \cdot (1 - s_{u,k+1}) + s_{u,k+1} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') \\ = \; & U(z_{eq}) - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') + s_{u,k+1} \cdot \underbrace{(s_{d,k+1} \cdot \mathcal{F}_k(z') - U(z_{eq}))}_{< 0 \text{ by (7)}} \\ \overset{(10)}{>} \; & U(z_{eq}) - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') + s_{u_2} \cdot (s_{d,k+1} \cdot \mathcal{F}_k(z') - U(z_{eq})) \\ = \; & U(z_{eq}) \cdot (1 - s_{u2}). \end{aligned}$$

So all in all we obtain

$$U(z_{eq}) \cdot (1 - s_{u2}) < L(z') - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z'). \tag{11}$$

Hence we get

$$\begin{aligned}
U(z_{eq}) &= U(z_{eq}) + s_{u2} \cdot \underbrace{(s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) - U(z_{eq}))}_{=0 \text{ by definition of } z_{eq}} \\
&= U(z_{eq}) \cdot (1 - s_{u2}) + s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) \\
&\overset{(11)}{<} L(z') - s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z') + s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) \\
&= L(z') - s_{u2} \cdot s_{d,k+1} \cdot (s_k \cdot G_e(z') + \mathcal{H}_k(z')) \\
&\quad + s_{u2} \cdot s_{d,k+1} \cdot \mathcal{F}_k(z_{eq}) \\
&= L(z') - \underbrace{s_{u2} \cdot s_{d,k+1} \cdot s_k}_{=s_{opt}} \cdot G_e(z') + s_{u2} \cdot s_{d,k+1} \cdot (\mathcal{F}_k(z_{eq}) - \mathcal{H}_k(z')) \\
&\leq \mathcal{H}_{opt}(z') + s_{u2} \cdot s_{d,k+1} \cdot (s_k \cdot G_e(z_{eq}) + \underbrace{\mathcal{H}_k(z_{eq}) - \mathcal{H}_k(z')}_{=0 \text{ by (9)}}) \\
&\leq \mathcal{H}_{opt}(z_{eq}) + s_{opt} \cdot G_e(z_{eq})
\end{aligned}$$

where we also used $s_{opt} \cdot G_e + \mathcal{H}_{opt} \geq L$, which holds by definition of $\mathcal{H}_{opt}$. Thus, the upper boundary U is violated for s_{opt}, which contradicts its definition, so that the proposition follows. □

The next result justifies the way we correct a solution to the unconstrained isotonic regression problem in line 8 of Algorithm 5. To be more precise, we show that setting its negative values to zero leads to the same L_∞-norm as in the constrained problem and therefore yields an optimal solution to the latter. Keep in mind that the unconstrained isotonic regression problem is solved by the *Basic* approach [12], which computes the maximum of all previous values and the minimum of all subsequent values for each observation and determines the regression value as the average of these two quantities.

Lemma 5. *Let $x \in \mathbb{R}^d$ be arbitrary. Denote by x_L the optimal solution of the L_∞ isotonic regression of x computed by the* Basic *approach [12] and define the new vector $x_{L0} = \max(x_L, 0)$ by component wise comparison to 0. Let x_{Lc} be an optimal solution of the L_∞ isotonic regression of x with the constraint of nonnegativity. Then x_{L0} is also an optimal solution to the constraint problem, i.e. $L_\infty(x, x_{Lc}) = L_\infty(x, x_{L0})$ holds.*

Proof. We show the statement considering the cases $\min(x) \geq 0$ and $\min(x) < 0$ consecutively. At first, assume that $\min(x) \geq 0$ holds. Then, by construction of x_L, we can deduce $x_L \geq 0$. Thus, x_{L0} is equal to x_L and, as a nonnegative vector, satisfies $L_\infty(x, x_{Lc}) \leq L_\infty(x, x_{L0})$. Since introducing constraints to a problem cannot lead to a better value of the objective function in the optimum, it must hold that $L_\infty(x, x_{Lc}) \geq L_\infty(x, x_L) = L_\infty(x, x_{L0})$. Together this yields the result restricted to the case $\min(x) \geq 0$.

We now consider the case $\min(x) < 0$. Since the negative values of x_L set to zero in x_{L0} result in a maximal deviation of $-\min(x)$ to x, we can write $L_\infty(x, x_{L0}) = \max(L_\infty(x, x_L), -\min(x))$. In addition, $\min(x) < 0$ and $x_{Lc} \geq 0$ imply $L_\infty(x, x_{Lc}) \geq$

$-\min(x)$, so that we deduce

$$\begin{aligned} L_\infty(x, x_{L0}) &= \max(L_\infty(x, x_L), -\min(x)) \\ &\leq \max(L_\infty(x, x_L), L_\infty(x, x_{Lc})) \\ &= L_\infty(x, x_{Lc}), \end{aligned}$$

where the last step follows, because a constraint problem cannot lead to a solution with a better value of the objective function compared to the corresponding unconstrained problem. Thus, $L_\infty(x, x_{L0}) \leq L_\infty(x, x_{Lc})$ holds. The converse inequality $L_\infty(x, x_{L0}) \geq L_\infty(x, x_{Lc})$ follows from the definition of x_{Lc}, since $x_{L0} \geq 0$. Both together yield the result restricted to the case $\min(x) < 0$, which completes the proof. □

Using the above results we prove the correctness of our algorithm in the following theorem.

Theorem 6. *Algorithm 1 returns s_{opt} and a corresponding solution $\mathcal{H}_{opt}$ both optimal in the sense of Problems 1 and 2, respectively.*

Proof. Lemma 1 shows that for $s > s_{opt}$ no mixture can lie within the confidence band B while for $s \leq s_{opt}$ there always exists a mixture lying in B. By the monotonicity of this property the binary search step converges to s_{opt}. Lemmas 3 and 4 allow to update the upper bound of the binary search by the values of the shrinkage factor after each Shrink-Down and Push-Up step. Hence, these steps further reduce the range of possible candidates for s_{opt} while never excluding s_{opt} and therefore the correct s_{opt} is still determined. Lemma 2 implies, that the correcting function $\mathcal{H}_k$ after termination of the main loop of Algorithm 1 is the function $\mathcal{H}_{min}$ introduced on page 4, which is required for solving Problem 2. Having found the set M_{norm} in the lines 5 to 10 of Algorithm 5, we use Lemma 5 to see that the corrected solution to the unconstrained L_∞ isotonic regression problem is an optimal solution to the constrained problem. Thus, it is a valid solution $\mathcal{H}_{opt}$, which completes the proof. □

4.3 Runtime analysis

For the runtime analysis we introduce a precision parameter ε. Note that ε never appears in our pseudo code or actual implementation. Instead, think of it as the *machine precision*, which might depend on the physical architecture, operating system or programming environment. Note that the main loop of Algorithm 1 in lines 5 to 11 runs until the mixture $\mathcal{F}$ is in the confidence band up to an additive deviation of ε. In other words, the loop stops when $\forall z \in Z$ the property $L(z) - \varepsilon \leq \mathcal{F}(z) \leq U(z) + \varepsilon$ holds. In the following theorem we prove that this condition is met after a constant number of iterations yielding an overall runtime linear in the input size and logarithmic in $\frac{1}{\varepsilon}$.

Theorem 7. *Let $\varepsilon \in (0, 1)$ be a fixed machine precision parameter. On an input of $n = n_1 + n_2$ observations, Algorithm 1 runs in time $O\left(n \cdot \log_2\left(\frac{1}{\varepsilon}\right)\right)$.*

Proof. First note that the Shrink-Down, the Push-Up, the binary search step and the normalisation step can be implemented in linear, i.e. $O(n)$ time. Particularly, the solution to the isotonic regression subproblem (line 8 in Algorithm 5) can be computed

in linear time as noted by Stout [12]. Therefore, it remains to bound the number of iterations of the loop in lines 5 to 11 of the main algorithm. The search interval for s is initialized to $[s^*, 1] \subset [0, 1]$ and halved at the end of every iteration where the binary search step is performed. The Shrink-Down and Push-Up steps can only further decrease the upper bound and consequently the size of the search interval. Therefore, after $\lceil \log_2 \left(\frac{2}{\varepsilon}\right) \rceil$ iterations the size of the interval decreases to at most $2^{-\lceil \log_2 \left(\frac{2}{\varepsilon}\right) \rceil} < \frac{\varepsilon}{2}$. So, after $\lceil \log_2 \left(\frac{2}{\varepsilon}\right) \rceil$ iterations every value between the upper and lower boundary lies within additive precision $\frac{\varepsilon}{2}$ to s_{opt}. Consider an $s \in [s_{opt} - \frac{\varepsilon}{2}, s_{opt} + \frac{\varepsilon}{2}]$ and let $\mathcal{H}_s \in \mathcal{M}^*$ be the minimal function such that $s \cdot G_e + (1 - s) \cdot \mathcal{H}_s \geq L$ holds. Using $s \geq s_{opt} - \frac{\varepsilon}{2}$ we see that $s \cdot G_e \geq \left(s_{opt} - \frac{\varepsilon}{2}\right) \cdot G_e = s_{opt} \cdot G_e - \frac{\varepsilon}{2} \cdot G_e \geq s_{opt} \cdot G_e - \frac{\varepsilon}{2}$ holds. The property $s \cdot G_e \geq s_{opt} \cdot G_e - \frac{\varepsilon}{2}$ implies $(1 - s) \cdot \mathcal{H}_s \leq (1 - s_{opt}) \cdot \mathcal{H}_{opt} + \frac{\varepsilon}{2}$ and we deduce

$$\begin{aligned} & s \cdot G_e + (1 - s) \cdot \mathcal{H}_s \\ \leq \; & \left(s_{opt} + \frac{\varepsilon}{2}\right) \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{opt} + \frac{\varepsilon}{2} \\ \leq \; & s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{opt} + \varepsilon \\ \leq \; & U + \varepsilon, \end{aligned}$$

because $s_{opt} \cdot G_e + (1 - s_{opt}) \cdot \mathcal{H}_{opt} \leq U$ holds by definition of s_{opt} and $\mathcal{H}_{opt}$. An analogous argument shows $s \cdot G_e + (1 - s) \cdot \mathcal{H}_s \geq L - \varepsilon$. Thus, the stopping criterion $L - \varepsilon \leq \mathcal{F} \leq U + \varepsilon$ is met after $\lceil \log_2 \left(\frac{2}{\varepsilon}\right) \rceil$ iterations and the result follows. □

5 Application

In this section we evaluate the algorithm by applying it to simulated and real data sets. We compare the runtime of our algorithm to alternative procedures on artificial data, investigate its capability to estimate the disagreement of the distributions in a finite Gaussian mixture case and examine its performance in case of false rejections of the null hypothesis. Furthermore, the algorithm is illustrated on spectrometry data from a biological domain.

5.1 Experimental setup

For our empirical evaluation we implemented all algorithms using the statistical software R [10], version 2.15.1-gcc4.3.5. The R-package BatchExperiments by Bischl et al. [1] was used to run the experiments in a batch and to distribute the computations to the cores of our computer. The computations were conducted on a 3.00GHz Intel Xeon E5450 machine with 15GB of available RAM running a SuSE EL 11 SP0 Linux distribution.
Our datasets consist of equally sized sample pairs generated for several sample sizes. Due to its central role in statistics we focus on the Gaussian distribution and consider the popular setting of a finite Gaussian mixture. Therefore, in each of these cases, one dataset is sampled from a standard Gaussian distribution. The other sample also consists of observations sampled from the standard Gaussian distribution to a fraction of s. The remaining observations stem from a second Gaussian distribution with mean

μ and standard deviation σ. To cover a broad range of situations, the parameters s, μ and σ are varied for each sample pair. More precisely, s is drawn from the uniform distribution on the interval $[0.6, 0.8]$, μ is generated by the normal distribution with mean 2 and standard deviation 1 and σ is drawn from the gamma distribution with shape and scale parameter both equal to 1. Thus, the expectations of s, μ and σ are 0.7, 2 and 1, respectively. Our demixing algorithms are supposed to notice that the samples do not share the same underlying distribution and recommend to add observations from a distribution with a mean near μ and a standard deviation near σ in a proportion of about (1-s) to the sample drawn from the standard Gaussian distribution. All statistical tests conducted in this simulation experiment as well as in the remainder of this section are carried out at a significance level of $\alpha = 0.05$.

Before we present the results of the simulation, the setting is illustrated for $n_1 = n_2 = 10\,000$ in Figure 1. In the upper row, kernel density estimations of the provided samples are presented. Demixing the samples using Algorithm 1 leads to the shrinkage factor $s_{opt} = 0.640$, which is a good approximation of the true mixture proportion $s = 0.623$. Inverse transform sampling [3] allows us to generate a third sample with 10 000 observations from the correction distribution characterised by $\mathcal{H}_{opt}$. Its empirical mean 3.36 and standard deviation 0.676 are also close to the sampled $\mu = 3.312$ and $\sigma = 0.747$. The corresponding kernel density estimation shown on the right in the lower row is almost symmetrical and unimodal. Hence, the correction distribution represents the deviation between the underlying distributions of the first and the second sample quite well. The final mixture distribution proposed by Algorithm 1, which is the sum of the weighted distribution of the second sample and the weighted correction, is given by the corresponding estimated density in the lower right corner. The curve resembles the one of the first sample as desired.

5.2 Runtime and performance evaluation

In order to assess the runtime of Algorithm 1 we compare its performance with two alternative approaches. The first alternative algorithm, called binary search procedure in the following, determines the optimal shrinkage factor s_{opt} relying only on the binary search. In contrast to Algorithm 1, the Shrink-Down and Push-Up steps are not conducted. Comparing this modified version to our algorithm makes it thus possible to evaluate the impact of the Shrink-Down and Push-Up step. Keep in mind that both steps are in principle not necessary to obtain a correct solution to Problem 1 and 2 but are supposed to accelerate the computation.

The second new algorithm applied is an intuitive and simple brute force approach based on the equivalence of Problem 1 and Problem A presented in Lemma 1. It initially sets $s = 1$ and checks constraints (4a) and (4b) of Problem A one by one. Whenever a violation is detected it updates s to the maximal value which ensures feasibility. Thus, a decreasing sequence of candidate values is generated, which eventually reaches s_{opt}. The computation time of this approach grows quadratically in the sample sizes, because all pairs $z', z'' \in Z$ with $z' < z''$ must be considered in order to identify the maximal shrinkage factor for constraint (4b). Keep in mind that all three algorithms lead to the

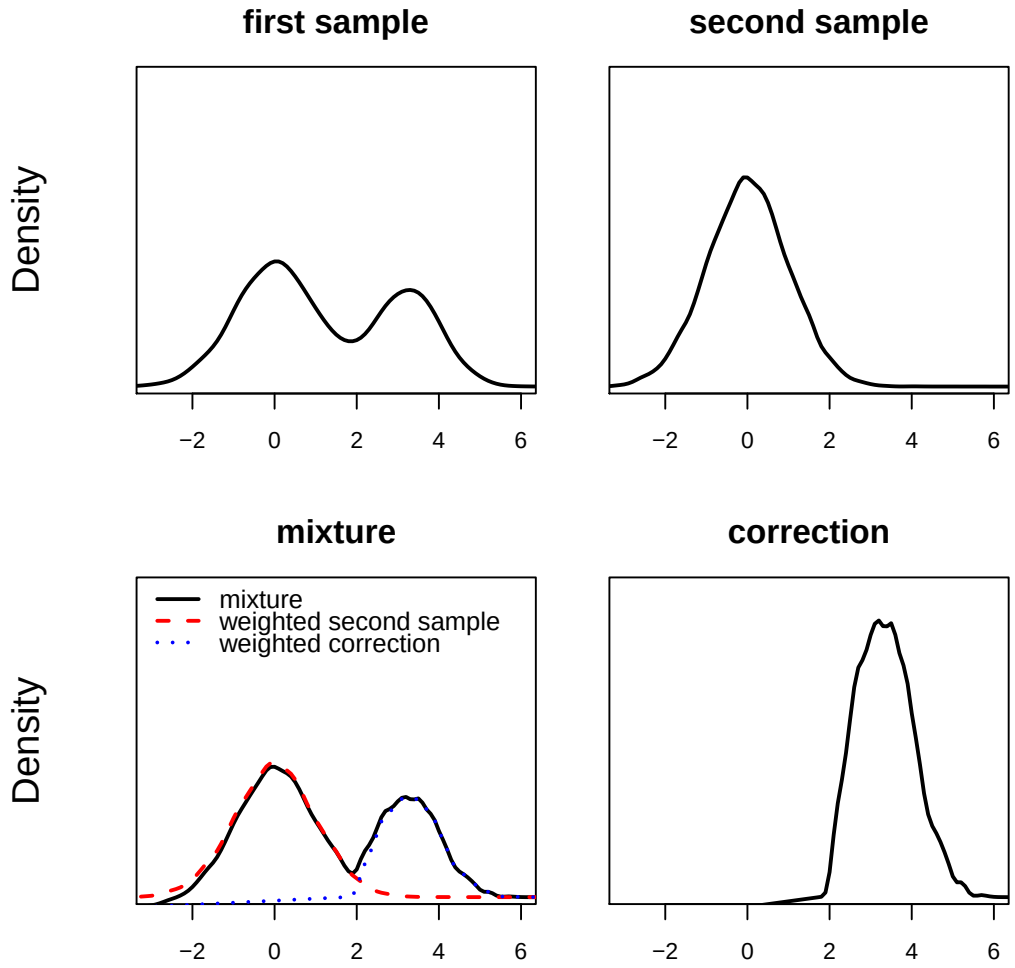

Figure 1: Kernel density estimations for two samples, the computed mixture and the correction distribution in the experimental setup.

same results but differ in the way they determine the shrinkage factor s_{opt}, which has a great impact on their computation time.

The average runtimes over 50 repetitions of the simulations described above are presented in Table 1 in seconds. We consider sample sizes $n_1 = n_2 \in \{2\,000,$
$4\,000,\ 6\,000,\ 8\,000,\ 10\,000\}$.

	2 000	4 000	6 000	8 000	10 000
Alg1	0.089	0.179	0.291	0.389	0.523
BS	0.552	1.114	1.721	2.292	2.851
BF	67.027	268.280	603.327	1074.198	1674.667

Table 1: Average runtimes for Algorithm 1 (Alg1), the binary search procedure (BS) and the intuitive brute force method (BF) for different sample sizes.

The brute force procedure is much slower than the other methods and the difference grows dramatically for larger samples due to its quadratic computation time, as opposed

to the other, linear algorithms.

Although Algorithm 1 performs better than its reduced version in terms of runtime, the computation times of these two methods are quite small and hence may be affected by background activities of the operating system and other processes running on the executing computer. To minimise this noise, we discard the brute force method and apply the remaining two approaches for higher sample sizes using the same simulation procedure as before. We now consider $n_1 = n_2 \in \{25\,000,\ 50\,000,\ 75\,000,\ 100\,000,\ 125\,000,\ 150\,000\}$ and present the average runtimes in Figure 2.

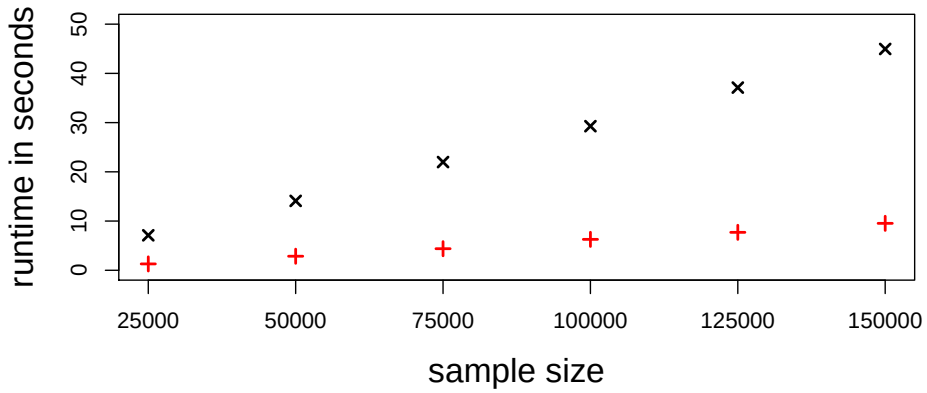

Figure 2: Average runtimes of Algorithm 1 (red) and the binary search procedure (black) for different sample sizes in seconds.

The runtime for both algorithms grows linear in the sample size and is by a factor of approximately 4.6 smaller for Algorithm 1 than for the binary search procedure. This demonstrates that the Shrink-Down and Push-Up steps lead to large savings in computation time and are therefore very valuable for large data sets.

In order to further evaluate the performance of the demixing procedure, we proceed as in the example above: for each sample pair in the above simulation we consider the shrinkage factor s_{opt} determined by Algorithm 1 as well as the empirical mean and standard deviation of a third sample of size 1 000 generated using the inverse transform sampling procedure [3] from the information contained in $\mathcal{H}_{opt}$. Averaging over the 50 repetitions for each sample size leads to the results presented in Table 2. Keep in mind that these are independent of the applied algorithm.

The results suggest that demixing leads to an overestimation of the expected mixing proportion 0.7, which decreases quite slowly as the sample size grows. This is not surprising, since by definition s_{opt} is the maximal shrinkage factor such that the corresponding mixture lies in the confidence band. Therefore, as the sample size grows, the radius of the confidence band becomes smaller and hence s_{opt} converges towards s. The estimated mean and standard deviation behave similarly by slowly approaching 2, the expected value of μ, and 1, the expected value of σ. Thus, the correction distributions proposed

	25 000	50 000	75 000	100 000	125 000	150 000
s_{opt}	0.744	0.738	0.735	0.733	0.732	0.730
Mean	2.177	2.154	2.135	2.118	2.117	2.101
SD	0.879	0.894	0.907	0.910	0.907	0.920

Table 2: Determined shrinkage factors s_{opt} and estimations of the mean and the standard deviation of the correction distribution for different sample sizes averaged over 50 repetitions.

by the methods reflect the discrepancies between the sample pairs quite well for large sample sizes.

5.3 Estimated shrinkage factors under the null hypothesis

The null hypothesis H_0 states that the analysed samples stem from the same distribution. In this situation, the Kolmogorov-Smirnov test rejects the null hypothesis by mistake in about an α-fraction of the cases, where α is the predefined significance level. In these cases, a reasonable procedure comparing the samples in the mixture framework should recognise their similarity. Thus, a shrinkage factor near 1 is desirable after a false rejection of the null hypothesis. To check the performance of our method under H_0, equally sized dataset pairs are generated for the sample sizes $n_1 = n_2 \in \{ 50,\ 100,\ 500,\ 1\,000,\ 5\,000,\ 10\,000\}$. All samples stem from the standard Gaussian distribution. Other distributions like the exponential and the t-distribution were also considered and led to comparable results. For each sample size, dataset pairs are simulated until the Kolmogorov-Smirnov test rejects in 1000 cases. These 1000 dataset pairs are passed to Algorithm 1, which determines corresponding shrinkage factors. All of them are less than 1 by construction. The results are presented via boxplots in Figure 3.

Even for small sample sizes the majority of shrinkage values are greater than 0.9. Increasing the sample size further reduces the amount of small shrinkage values. Thus, our method performs as desired: If no modifications are actually necessary, the algorithm proposes to perform none or only small modifications to the current samples.

5.4 Application to real data

Algorithm 1 is used to evaluate data from a bioinformatics application. The real data consists of so called ion mobility spectrometry (IMS) measurements which are used to detect volatile organic compounds in the air or in exhaled breath. Motivated by the need to process such measurements in real-time as they arrive one-by-one, it is a usual approach to find and annotate major peaks in the data. In this way the original information is summarised in a compressed representation. In an effort to automate and speed-up the computations, D'Addario et al. [2] propose to approximate the measurements by finite mixtures of translated probability density functions, whose parameters are estimated using a variant of the EM algorithm. The computations are performed on a set of measurements leading to a two dimensional problem, where both dimensions are

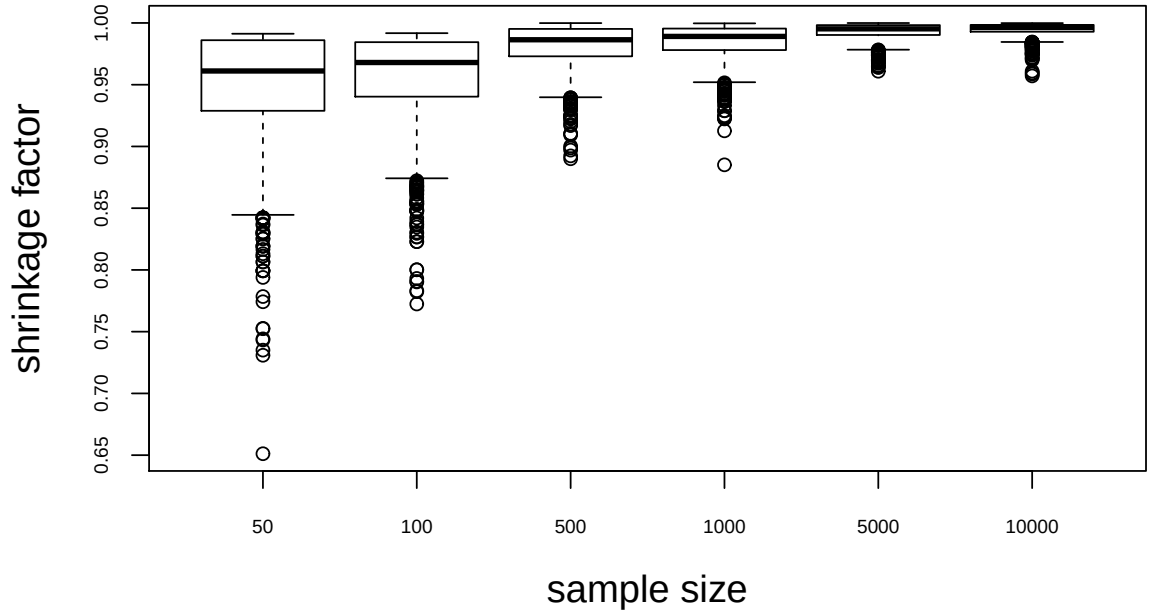

Figure 3: Shrinkage factors determined by Algorithm 1 for varying sample sizes after a false rejection of the null hypothesis $H_0 : P = Q$, where P is the standard Gaussian distribution.

modelled independently by mixtures of inverse Gaussian densities.
Focussing on one of the dimensions and conditioning on the other, we obtain 6000 spectrograms consisting of 12500 data points each, stemming from 10 minutes of IMS measurement [cf. 8]. In this data, we identify 187 groups of spectrograms belonging to the same peak models, respectively. In order to evaluate the models, we apply our algorithm at a significance level of five percent to samples of size 1000 generated from each spectrogram and the corresponding mixture model. Both of these are regarded as probability density functions up to some normalising constants.

In general our method suggests that the models fit the spectrograms quite well, since in 152 of the 187 groups the mean shrinkage factor of the spectrograms is above 0.8. In addition, we identify some interesting groups of spectrograms. The shrinkage factors of two of these are shown in Figure 4. Keep in mind that the spectrogram index represents the second dimension of the data we condition on. In both groups the model consists of a single inverse Gaussian density.

The results for group A suggest that the first half of the measurements are modelled quite well, but for increasing spectrogram indices the approximation is getting worse and worse. This shows that the model in the second dimension is not appropriate. If it consists of a single inverse Gaussian density, two components would probably lead to better approximations, since they allow to model both halves of the spectrograms with a density function, respectively. In contrast to that, the shrinkage factors for group B indicate a sufficient number of components used in the second dimension. However, the fitted density mixture seems too wide. The approximation could be substantially improved by excluding the spectrograms on the left and on the right from this group and

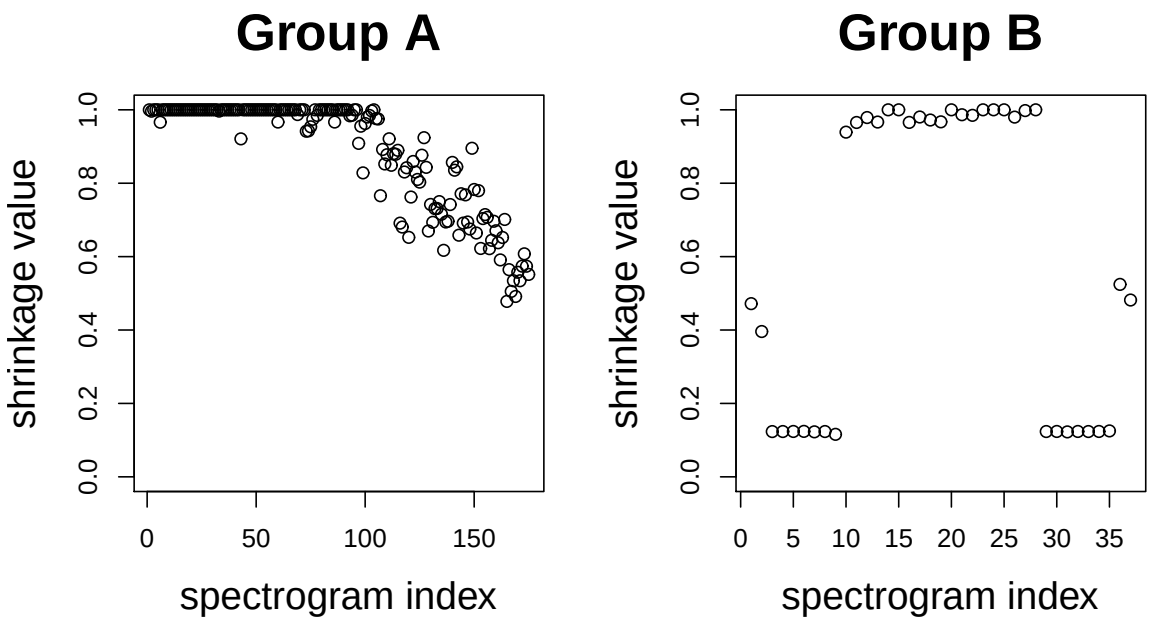

Figure 4: Shrinkage factors of two groups of spectrograms determined by Algorithm 1.

treating them by further models.

We also illustrate the method using a single spectrogram from the data set. The upper row of Figure 5 provides a kernel density estimation for the measurement 1157 and its model. Since all four plots are given on the same scale, the two peaks in the model are more narrow and differ much more in height than the ones in the original data. In addition, the peak on the left is missing. Although it looks small in this scale, it appears noteworthy when compared to the other two or examined on a larger scale. In the second row on the right a kernel estimation for the correction distribution characterised by $\mathcal{H}_{opt}$ is presented. It is based on 1000 observations generated by inverse transform sampling [3]. As expected, the correction distribution puts mass on the very right peak in order to fix the height proportions between the peaks on the right. In addition it generates the left peak missing in the model. The plot in the lower left corner shows the estimations of the modelled and the correction distribution weighted by the determined shrinkage value 0.76 and the remaining mass 0.24, respectively, as well as the kernel estimation for the final mixture, which is the sum of the weighted estimations. The proposed mixture is still a somewhat narrow, but the proportions of the peak heights as well a the small peak are better represented compared to the original model.

6 Conclusion

This technical report deals with the nonparametric two sample homogeneity problem. A widely-used tool to test the equality of the distributions corresponding to the two samples is the Kolmogorov-Smirnov test. We develop an algorithm which, in case of a rejection

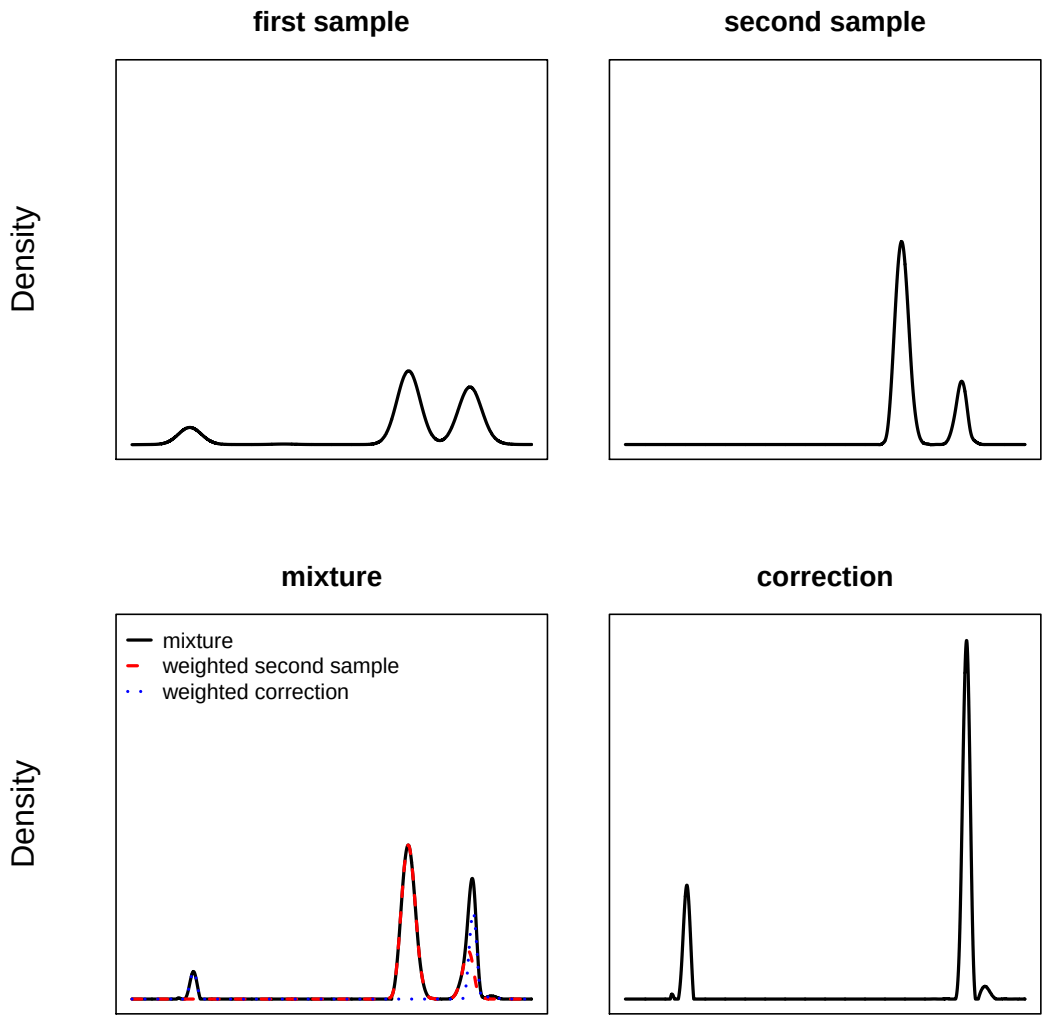

Figure 5: Kernel density estimations for a spectrogram based on the measurements, the corresponding inverse Gaussian model, the determined mixture of the model and the correction distribution are presented on the same scale.

by this test, determines how one of the samples should be modified to resemble the other. More precisely, an appropriate mixing proportion as well as an empirical distribution function are identified. Mixing the determined proportion of data generated by the corresponding correction distribution with the sample to modify, leads to a distribution which fits the other sample well in the sense of the Kolmogorov-Smirnov test. The method is especially of interest in applications, where the aim is to design a simulation to model an observed data generating process. In such a case, the information provided in the determined correction distribution may be used to improve the simulation.

The algorithm proceeds in an iterative manner applying several correction steps linked with a modified binary search technique. The constructed distribution function is shown to be optimal in a reasonable sense and the runtime of the algorithm is proved to be of

linear order. In our experience it converges in three iterations in the majority of all cases independent of the sample sizes. The algorithm proposes none or only slight corrections in cases where both datasets stem from the same distributions. The Shrink-Down and Push-Up steps applied in addition to the standard binary search algorithm lead to large savings in computation time. The correction distributions proposed in simulations as well as for a real data example are intuitive and adequate. Since the procedure is completely nonparametric, it is widely applicable and in particular not only useful in the setting of Gaussian mixtures considered in the simulations.

There are several possibilities to extend the presented ideas in future work. On the one hand, instead of focussing on distribution functions, a density based approach to the demixing problem could also be of interest, since working with densities is often even more intuitive than using distribution functions and there exists a broad literature on mixture models dealing with density estimation. On the other hand, one could use alternative test procedures for distribution functions besides the Kolmogorov-Smirnov test to construct the confidence bands. Although the Kolmogorov-Smirnov test is quite popular, related tests like the Anderson-Darling and the Cramér von Mises test detect differences between two distributions more often in certain settings [cf. 11] and could thus lead to better demixing results. In this work we focused on the Kolmogorov-Smirnov test since the simple shape of the corresponding confidence band allows for finding an efficient algorithm solving the demixing problem. The extension to analytically more sophisticated distance measures where our proofs do not carry over in a straightforward manner is a challenging as well as promising open problem for future work.

Acknowledgements: We would like to thank Prof. Dr. Roland Fried for many helpful suggestions . We thank Marianna D'Addario and Dominik Kopczynski, both members of the Bioinformatics group of Prof. Dr. Sven Rahmann, for their data from another interesting domain in cooperation with Project SFB876-TB1.

References

[1] Bischl, B., Lang, M., Mersmann, O.:, BatchExperiments: statistical experiments on batch computing clusters. R package version 1.0-968, URL `http://CRAN.R-project.org/package=BatchExperiments` (2013)

[2] D'Addario, M., Kopczynski, D., Baumbach, J. I., Rahmann, S.: A modular computational framework for automated peak extraction from ion mobility spectra. BMC Bioinform. **15**(1):25 (2014)

[3] Devroye, L.: Non-Uniform Random Variate Generation. Springer-Verlag, New York (1986)

[4] Durbin, J.: Distribution Theory for Tests Based on the Sample Distribution Function, Reg. Conf. Ser. Appl. Math., vol 9. SIAM, Philadelphia

[5] Hall, P., Neeman, A., Pakyari, R., Elmore, R.: Nonparametric inference in multivariate mixtures. Biometrika **92**(3):667–678 (2005)

[6] Hettmansperger, T. P., Thomas, H.: Almost nonparametric inference for repeated measures in mixture models. J. R. Stat. Soc., Ser. B **62**(4):811–825 (2000)

[7] Kolossiatis, M., Griffin, J. E., Steel, M. F.: On bayesian nonparametric modelling of two correlated distributions. Stat. Comput. **23**(1):1–15 (2013)

[8] Kopczynski, D., Baumbach, J. I., Rahmann, S.: Peak modeling for ion mobility spectrometry measurements. In: Proc. of the 20th Eur. Signal Process. Conf. (EUSIPCO 2012), pp 1801–1805, (2012)

[9] Pilla, R. S., Lindsay, B. G.: Alternative EM methods for nonparametric finite mixture models. Biometrika **88**(2):535–550 (2001)

[10] R Core Team:, R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, URL `http://www.R-project.org` (2013)

[11] Razali, N., Wah, Y. B.: Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. J. Stat. Model. Anal. **2**(1):21–33 (2011)

[12] Stout, Q. F.: Strict ℓ_∞ isotonic regression. J. Optim. Theory Appl. **152**(1):121–135 (2012)

[13] Wang, Y.: Maximum likelihood computation for fitting semiparametric mixture models. Stat. Comput. **20**(1):75–86 (2010)