Nonparametric Density Deconvolution by Weighted Kernel Estimators

Martin L. Hazelton[*]
Massey University

Berwin A. Turlach
National University of Singapore & University of Western Australia

July 19, 2008

Abstract

Nonparametric density estimation in the presence of measurement error is considered. The usual kernel deconvolution estimator seeks to account for the contamination in the data by employing a modified kernel. In this paper a new approach based on a weighted kernel density estimator is proposed. Theoretical motivation is provided by the existence of a weight vector that perfectly counteracts the bias in density estimation without generating an excessive increase in variance. In practice a data driven method of weight selection is required. Our strategy is to minimize the discrepancy between a standard kernel estimate from the contaminated data on the one hand, and the convolution of the weighted deconvolution estimate with the measurement error density on the other hand. We consider a direct implementation of this approach, in which the weights are optimized subject to sum and non-negativity constraints, and a regularized version in which the objective function includes a ridge-type penalty. Numerical tests suggest that the weighted kernel estimation can lead to tangible improvements in performance over the usual kernel deconvolution estimator. Furthermore, weighted kernel estimates are free from the problem of negative estimation in the tails that can occur when using modified kernels. The weighted kernel approach generalizes to the case of

[*] *Address for correspondence*: Institute of Information Sciences and Technology, Massey University, Private Bag 11222, Palmerston North, New Zealand.

multivariate deconvolution density estimation in a very straightforward manner.

Key words: Density estimation, Errors in variables, Integrated square error, Measurement error, Weights.

1 Introduction

In this article we consider nonparametric estimation of the density of a random variable when we observe only a random sample that has been contaminated by additive measurement error. This problem is of interest in its own right, with applications in a wide range of fields. Examples include inference for the distribution of fluorometric data (e.g. Mendelsohn and Rice, 1982), density estimation from self reported food consumption data (e.g. Stefanski and Carroll, 1990), inference for geological age distributions (e.g. Sircombe and Hazelton, 2004), and estimation of the distribution of gene expression data (e.g. van de Wiel and Kim, 2007). Density estimation from contaminated data also has direct connections with nonparametric regression calibration and associated problems. See Carroll et al. (2006).

Our problem of interest may be stated in more detail as follows. We observe a univariate random sample $Y_1, \ldots, Y_n$ from a density g, where

$$Y_i = X_i + Z_i \qquad (i = 1, \ldots, n). \tag{1}$$

Here $X_1, \ldots, X_n$ are independent and identically distributed with unknown continuous density f, and the measurement errors $Z_1, \ldots, Z_n$ form a random sample from the continuous density η which we assume to be known. Our goal is to obtain a nonparametric estimate of f from the observed sample. According to our model, the densities f, g and η are related by the convolution equation

$$g(y) = f * \eta(x) = \int f(x)\eta(y-x)\,dx \tag{2}$$

and so estimation of f is a deconvolution problem.

The most common approach to density deconvolution has been kernel based, as we now describe. The standard kernel density estimator constructed from $Y_1, \ldots, Y_n$ is

$$\breve{g}(y) = \breve{g}(y;h) = \frac{1}{n}\sum_{i=1}^{n} K_h(y - Y_i) \tag{3}$$

where $K_h(y) = K(y/h)/h$, and K is a kernel function satisfying $\int K(y)\,dy = 1$, $\mu_2 = \mu_2(K) = \int K(y)y^2\,dy < \infty$, $K(y) \geq 0$ and $K(y) = K(-y)$. The parameter h is called the bandwidth, and controls the smoothness of the estimator. See Wand and Jones (1995) or Simonoff (1996) for an overview. The estimator $\breve{g}$ targets g rather than the desired density f, but it can be adapted by modifying the kernel function. The required deconvoluting estimator $\breve{f}$ is related to $\breve{g}$ and the error density η by $\breve{g} = \breve{f} * \eta$, so that $\psi_{\breve{f}} = \psi_{\breve{g}}/\psi_\eta$ where ψ_a denotes the characteristic function of a density a. If follows from (3) that

$$\breve{f}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h^Z(x - Y_i) \tag{4}$$

where $K_h^Z(u) = h^{-1}K^Z(u/h; h)$ and

$$K^Z(u; h) = \frac{1}{2\pi}\int e^{-itu}\frac{\psi_K(t)}{\psi_\eta(t/h)}\,dt.$$

It is acknowledged that the estimator $\breve{f}$ is due Stefanski and Carroll, although these authors' seminal paper (Stefanski and Carroll, 1990) appeared in the literature later than some other articles referring to this methodology. Asymptotic results for $\breve{f}$ have been derived by Carroll and Hall (1988), Devroye (1989), Stefanski (1990), Fan (1991a, 1991b, 1992) and Van Es and Uh (2005). Finite sample performance has been investigated through simulation studies (see Liu and Taylor, 1990, and Stefanski and Carroll, 1990, for example) and through exact calculations of the mean integrated squared error (MISE) performance criterion in some special cases Wand (1998).

The performance of $\breve{f}$ is strongly influenced by the choice of bandwidth, h. A number of authors have proposed data driven techniques for selecting this smoothing parameter in the setting of density deconvolution. Stefanski and Carroll (1990) and Hesse (1999) considered cross-validation procedures for choosing h. Delaigle and Gijbels (2004*a*) suggested a bootstrap approach to the problem. Delaigle and Gijbels (2004*b*) developed a plug-in bandwidth selector.

While kernel estimation has been the most popular approach to density deconvolution, a number of alternatives have been considered. Mendelsohn and Rice (1982) and Koo and Park (1996) discussed spline based procedures. Wavelet methods were investigated by a number of researchers, including Walter (1999), Pensky and Vidakovic (1999) and Pensky (2002). Eggermont and LaRiccia (1997) proposed a smoothed EM algorithm for computing a type of maximum likelihood estimate of f. Efromovich (1997) developed techniques based on inverse Fourier transforms, and more recently Hall and Qiu (2005) examined a discrete Fourier series approach to density deconvolution.

One of the attractions of a kernel approach to density deconvolution is the familiarity of the resulting estimator. The estimator $\breve{f}$ retains the simple structure of the standard kernel density estimator for uncontaminated data. However, in comparison to this standard estimator, $\breve{f}$ has some shortcomings. First, $\breve{f}$ may take negative values since K^Z can do so. Second, while the extension of the kernel deconvolution estimator to multivariate data is straightforward in principle (see Masry 1991, 2003, for example), there can be significant difficulties in practice. If the multivariate errors have a non-diagonal covariance matrix, or if the multivariate kernel is not a product kernel, then the theoretical analysis of this type of deconvolution estimator is very challenging and the practical computation is time consuming. Furthermore, at present there are no good data driven techniques for choosing the bandwidth matrix for this estimator.

In this article we propose a new adaptation of (3) for deconvolution density estimation. Specifically, we consider weighted kernel estimators of the form

$$\hat{f}_{\boldsymbol{w}}(x) = \hat{f}_{\boldsymbol{w}}(x; h) = \frac{1}{n} \sum_{i=1}^{n} w_i K_h(x - Y_i), \tag{5}$$

where $\boldsymbol{w} = (w_1, \ldots, w_n)^{\mathsf{T}}$ is a vector of non-negative weights that may be constrained by $\bar{w} = \frac{1}{n} \sum_{i=1}^{n} w_i = 1$ so as to ensure that $\int \hat{f}_{\boldsymbol{w}}(x)\, dx = 1$. The use of weights in kernel density estimation is not new, but previously this idea has been used for bias reduction in the context of uncontaminated data. See Jones et al. (1995) and Hall and Turlach (1999). However, weighting can also be a highly effective approach to density deconvolution. One immediate attraction of $\hat{f}_{\boldsymbol{w}}$ as opposed to $\breve{f}$ is that the former will never take negative values. As we show later, $\hat{f}_{\boldsymbol{w}}$ also has good finite sample performance and can be adapted to deal with multivariate data in a simple manner.

In the next section we consider the choice of weight vector $\boldsymbol{w}$ in theory. We show that if the optimal weighting scheme was known, then $\hat{f}_{\boldsymbol{w}}$ would have MISE of asymptotic order $n^{-4/5}$, the usual rate in density estimation for uncontaminated data. This observation may be regarded as motivation for considering weighting as an approach to kernel density deconvolution. In section 3 we examine data driven schemes for choosing $\boldsymbol{w}$. Our basic idea is to choose the weights so as to minimize the discrepancy between $\hat{f}_{\boldsymbol{w}} * \eta$ and $\breve{g}$. We employ a direct approach in which this discrepancy is minimized subject to sum and non-negativity constraints on the weights, and a regularized version in which the objective function is formed by addition at ridge-type penalty term. In section 4 the performance of our deconvolution estimator is analyzed using simulated and real data sets. Results are compared with those obtained using the classical estimator $\breve{f}$. The multivariate version of our deconvolution estimator is covered in section 5. Concluding remarks are given in section 6.

2 Weighted Deconvolution Estimators in Theory

Consider the weight vector $\boldsymbol{w}_0 = (w_{01}, \ldots, w_{0n})^\mathsf{T}$ defined by

$$w_{0i} = \frac{f(Y_i)}{g(Y_i)} \qquad (i = 1, \ldots, n). \tag{6}$$

With this choice of weights we have

$$\begin{aligned} \mathrm{E}[\hat{f}_{\boldsymbol{w}_0}(x)] &= \mathrm{E}\left[\frac{f(Y)K_h(x-Y)}{g(Y)}\right] \\ &= \int \frac{f(y)}{g(y)} K_h(x-y) g(y)\, dy \\ &= f * K_h(x). \end{aligned}$$

This is precisely the same mean as for a standard kernel estimator constructed from uncontaminated data. Standard asymptotic expansions show that $\mathrm{bias}\{\hat{f}_{\boldsymbol{w}_0}(x)\} \approx h^2 \mu_2 f''(x)/2$ as $n \to \infty$ and $h \to 0$, under suitable regularity conditions. Furthermore, the integrated squared bias (ISB) may be approximated by $h^4 \mu_2^2(K) R(f'')$ where $R(a) = \int a(y)^2\, dy$ for any squared integrable function a.

Turning to the variance,

$$\mathrm{var}\{\hat{f}_{\boldsymbol{w}_0}(x)\} = \frac{1}{n}\left[\int \frac{f^2(y)}{g(y)} K_h^2(x-y)\, dy - \{f * K_h(x)\}^2\right]$$

assuming that the support of f is a subset of the support of g (as is the case for all common measurement error models). The integrated variance (IV) may be approximated by

$$\mathrm{IV}\{\hat{f}_{\boldsymbol{w}_0}\} \approx \frac{1}{nh} R(K) \int \frac{f(y)^2}{g(y)}\, dy.$$

The corresponding result for a standard kernel estimation built from an uncontaminated random sample of size n is $\mathrm{IV} \approx R(K)/(nh)$, the same asymptotic order as $\mathrm{IV}\{\hat{f}_{\boldsymbol{w}_0}\}$. However, as one might expect, the coefficient of $1/(nh)$ is larger for the deconvolution estimator since $\int f(y)^2/g(y)\, dy > 1$.

Summing ISB and IV, it follows that $\mathrm{MISE}\{\hat{f}_{\boldsymbol{w}_0}\}$ is of rate $n^{-4/5}$ when h is chosen to be of optimal order, $h \sim n^{-1/5}$. This is the rate usually associated with density estimation for uncontaminated data. This rate will not be achievable in practice because $\boldsymbol{w}_0$ is a function of the unknown target density. Indeed, it is well known that the optimal rates for nonparametric density deconvolution are very slow. For example, Carroll and Hall (1988) showed that if f has two bounded derivatives then this rate can be no better than $(\log n)^{-1}$ when the measurement error is normally distributed. Nonetheless, the existence in theory of a set of weights with such desirable properties as $\boldsymbol{w}_0$ provides motivation for a weighted kernel approach to nonparametric density deconvolution.

3 Implementation of the Weighted Kernel Estimator

In this section we look at issues regarding the practical implementation of $\hat{f}_{\hat{\boldsymbol{w}}}$.

3.1 A Discrepancy Criterion

If the estimator $\hat{f}_{\hat{\boldsymbol{w}}}$ is good then one would expect $\hat{f}_{\hat{\boldsymbol{w}}} * \eta \approx g$. While g is unknown, we do have a natural estimator $\breve{g}$. This suggests that we might search for a vector of positive weights solving the linear system

$$\hat{f}_{\hat{\boldsymbol{w}}} * \eta(y) = \breve{g}(y),$$

possibly subject to the constraint $\bar{w} = 1$. In general this equation will not have a single solution that holds for all y in the support of g. One approach would be to seek a solution that holds for some particular set of y values. However, our preferred approach is to minimize a global measure of the discrepancy between $\hat{f}_{\hat{\boldsymbol{w}}} * \eta$ and $\breve{g}$. Specifically, we look for weights to minimize the criterion

$$Q = Q(\boldsymbol{w}) = \int \{\hat{f}_{\hat{\boldsymbol{w}}} * \eta(y) - \breve{g}(y)\}^2 \, dy. \tag{7}$$

The case of normally distributed measurement error is by far the most important in practice, and is also a case in which Q can be evaluated without recourse to numerical integration techniques. When $\eta = \phi_\sigma$ (a normal density with zero mean and variance σ^2) and the kernel is also normal, $K_h = \phi_h$, then Q has the simple expression

$$Q(\boldsymbol{w}) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left[w_i w_j \phi_{\sqrt{2}\lambda}(Y_i - Y_j) + \phi_{\sqrt{2}h}(Y_i - Y_j) - 2 w_i \phi_\omega(Y_i - Y_j) \right] \tag{8}$$

where $\lambda^2 = h^2 + \sigma^2$ and $\omega^2 = 2h^2 + \sigma^2 = \lambda^2 + h^2$.

Some degree of theoretical support for this approach to weight selection is provided by Theorem 1. If $\hat{\boldsymbol{w}}$ denotes the minimizer of Q, then the estimator $\hat{f}_{\hat{\boldsymbol{w}}}$ is consistent for f.

Theorem 1:
Assume that

(i) The kernel K is a bounded continuous density function with finite second moment.

(ii) The bandwidth $h = h_n$ is a non-random sequence satisfying $h \to 0$ and $nh \to \infty$ as $n \to \infty$.

(iii) The densities f and η are continuous and bounded on the real line, and $\int f^2(x)/g(x)\, dx < \infty$ where $g = f * \eta$.

Then $\hat{f}_{\hat{\boldsymbol{w}}}(x) \xrightarrow{P} f(x)$ for any given x.

The proof of this result is given in appendix A.

Looking forward to section 5, we note that while this theorem is stated (and proved) in the context of univariate data, the result also applies to the weighted kernel estimator for multivariate data.

3.2 Bandwidth Selection

The choice of bandwidth is typically critical in terms of performance when implementing kernel smoothers. For the classic deconvolution estimator, described by (4), it has proved necessary to develop specially tailored methods of bandwidth selection. However, we found over a range of numerical experiments that our deconvolution estimator operates well using standard methods of bandwidth selection for density estimation from uncontaminated data. This is intuitive, since the weights are chosen with respect to an error criterion Q defined in terms of estimates of g rather than of f. We wish $\hat{f}_{\hat{\boldsymbol{w}}} * \eta$ to be as close as possible to g, and hence implement $\breve{g}$ using a bandwidth calibrated for estimation of g. This bandwidth is then inherited by $\hat{f}_{\hat{\boldsymbol{w}}}$.

All the results for $\hat{f}_{\hat{\boldsymbol{w}}}$ given in Section 4 were obtained using the plug-in bandwidth selector due to Sheather and Jones (1991).

3.3 Optimization and Regularization

Optimizing $Q(\boldsymbol{w})$ in (8) under the constraints that the weights are non-negative and sum to n leads to the following quadratic program:

$$\begin{aligned} &\underset{\boldsymbol{w}}{\text{minimise}} && \frac{1}{2}\boldsymbol{w}^\mathsf{T}\mathbf{Q}\boldsymbol{w} - \boldsymbol{b}^\mathsf{T}\boldsymbol{w} && \text{(9a)} \\ &\text{subject to} && \sum_{i=1}^{n} w_i = n && \text{(9b)} \\ & && 0 \le w_i, \qquad i = 1, \ldots, n && \text{(9c)} \end{aligned}$$

where $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)^\mathsf{T}$, $\mathbf{Q}$ is an $n \times n$ matrix with $(i, j)^{\text{th}}$ entry being $\frac{1}{n^2} K_h * K_h * \eta * \eta(Y_i - Y_j)$ and $\boldsymbol{b} \in \mathbb{R}^n$ with i^{th} component being $\frac{1}{n^2}\sum_{j=1}^{n} K_h * K_k * \eta(Y_i - Y_j)$. In the case of Gaussian kernels and measurement error, the corresponding elements of $\mathbf{Q}$ and $\boldsymbol{b}$ are $\frac{1}{n^2}\phi_{\sqrt{2}\lambda}(Y_i - Y_j)$ and $\frac{1}{n^2}\sum_{j=1}^{n}\phi_{\omega}(Y_i - Y_j)$ respectively.

This is a quadratic programming problem with a single equality constraints and box constraints on the parameters. This kind of quadratic program appears in many practical applications, e.g. they arise for two-category classifying support vector machines (SVM), essentially as the dual problem to the original problem. See, among others, Cristianini and Shawe-Taylor (2000, Chapter 6), Schölkopf and Smola (2002, Chapter 7), Karatzoglou et al. (2006) and Moguerza and Muñoz (2006).

The SVM community has devoted a lot of research effort into solving such quadratic programs efficiently. The methods used predominantly in this field are sequential minimal optimization (Schölkopf and Smola, 2002, Chapter 12) and interior point algorithms. However it should be noted here that interior point algorithms, by their very nature, do not calculate the exact solution of (9) but find an approximate solution. Recently, there have been proposals in the numerical literature to solve such quadratic programs using gradient projection methods; see Serafini et al. (2005) and Dai and Fletcher (2006).

Here we propose a new algorithm for solving (9) based on compute-complete-solution-path ideas (Osborne et al., 2000; Efron et al., 2004). Details of the algorithm are given in appendix B. This algorithm can calculate the exact solution of (9) in a fast and efficient manner for moderate sample sizes. In our extensive numeric simulations we used the new algorithm for solving (9) as well as the interior point algorithm implemented by Karatzoglou et al. (2004). However, we noticed that for all practical purposes there is little difference between the solution determined by our "exact" algorithm and the solution determined by the interior point algorithm.

Theoretically the matrix $\mathbf{Q}$ in (9) is positive definite, at least as long as there are no repeated observations, whence the quadratic criterion is strictly convex and the problem has a unique solution. However, while the matrix $\mathbf{Q}$ is positive definite in theory, in finite precision arithmetic $\mathbf{Q}$ is typically singular. A helpful trick in such situations, which can be utilized in the implementation provided by Karatzoglou et al. (2004), is to employ an incomplete Cholesky decomposition of $\mathbf{Q}$. That is, an $n \times m$ matrix $\mathbf{Z}$ is determined, such that $\mathbf{Q} \approx \mathbf{Z}\mathbf{Z}^\mathsf{T}$. Within the interior point algorithm the calculations can be rearranged such that the calculations involve $m \times m$ matrices and not $n \times n$ matrices which can lead to substantial time savings.

In results not reported here, we observed that the interior point algorithm using an incomplete Cholesky factorization was fastest in calculating the solution of (9), being able to calculate solutions even for sample sizes of $n = 7,500$ in about 10 seconds. In terms of execution speed, this implementation was followed by the interior point algorithm using $\mathbf{Q}$ directly and our proposed algorithm. For all practical purposes, there was no difference in the solutions calculated between these three approaches. However, these results also demonstrated that, due to the numerical singularity of $\mathbf{Q}$, problem (9) is an ill-posed problem and the solution to this problem is too variable to be of practical use. Thus, some form of regularization is needed. Here, we propose to use a ridge type penalization by adding a multiple of $\frac{1}{2}\boldsymbol{w}^\mathsf{T}\boldsymbol{w}$ to the objective function in (9). This leads to the final version of our proposed method, namely using weights as determined by the solution of the following quadratic program, where $\mathbf{I}$ is the $n \times n$ identity matrix:

$$\underset{\boldsymbol{w}}{\text{minimise}} \quad \tfrac{1}{2}\boldsymbol{w}^\mathsf{T}(\mathbf{Q} + \tfrac{\gamma}{n}\mathbf{I})\boldsymbol{w} - \boldsymbol{b}^\mathsf{T}\boldsymbol{w} \tag{10a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} w_i = n \tag{10b}$$

$$0 \le w_i, \qquad i = 1, \dots, n \tag{10c}$$

If $\gamma > 0$, then the matrix $\mathbf{Q} + \gamma n^{-1}\mathbf{I}$ is theoretically and numerically of full rank. Hence, the variant of the interior point algorithm that uses an incomplete Cholesky factorization looses its advantage in terms of memory use and speed of execution for this problem.

3.4 Selecting the Ridge Penalty Constraint

By changing from problem (9) to problem (10) we have introduced a ridge penalty parameter γ. In practice this should be chosen based on the data. We propose that γ be selected using maximization of a five-fold likelihood cross-validation criterion. The idea here is to randomly partition the data into five blocks of equal size (give or take one if n is not a multiple of 5). We write $j \sim i$ if and only if i and j are elements of the same partition. We then compute a log-likelihood for the elements of each block using a weighted density estimate constructed from all the other data and aggregate the results. This produces cross-validation criterion

$$CV(\gamma) = CV(\hat{\boldsymbol{w}}(\gamma)) = \sum_{i=1}^{n} \log\{\hat{f}_{\hat{\boldsymbol{w}}}^{\not\sim i} * \eta(Y_i)\}. \tag{11}$$

Here $\hat{f}_{\hat{\boldsymbol{w}}}^{\not\sim i}$ denotes the weighted kernel estimator from (5) constructed only using those data which are in a block different to that of data point i. For each of the five density estimates $\hat{f}_{\hat{\boldsymbol{w}}}^{\not\sim i}$ corresponding to the five blocks, separate weights are computed by solving problem (10) based on the given value of γ. We select γ by maximizing $CV(\gamma)$ for $\gamma > 0$.

Our choice of five-fold cross-validation (as opposed to r-fold for some value $r \neq 5$) is pragmatic. The larger the value of r, the larger the computational burden required to compute $CV(\gamma)$ (since it requires that the quadratic program be solved r times). We also considered ten-fold cross-validation, but found no evidence of improved performance in the results. All the numerical work presented in the next section use five-fold cross-validation for selecting γ when implementing the ridge-penalized version of our weighted kernel estimator.

4 Numerical Results

In this section we discuss the results from a simulation study, and also look in detail at applications of our deconvolution density estimator.

4.1 Simulation Study

The performance of three kernel deconvolution density estimators are considered in this study. The first is our deconvolution estimator implemented using the optimal weight vector, $\boldsymbol{w}_0$. We refer to this as the 'oracle' estimator because it makes use of information that is not available in practice. The next estimator is our weighted kernel density estimator (WKDE) implemented using the ridge type penalty, with the ridge coefficient γ selected by cross-validation. (We also obtained results for the weighted kernel estimator without ridge penalization, but these were generally worse than for the ridged version and so are not presented here.) The final estimator is the 'classical' estimator from (4).

We implement this using the plug-in bandwidth selector developed by Delaigle and Gijbels (2004*b*), which outperformed other methods of bandwidth selection in the simulation study reported in that paper.

We present results involving four target densities and three levels of measurement error. The four target densities are normal mixtures. The first is standard normal, $N(0,1)$; the second is the kurtotic density $\frac{2}{3}N(0,1)+\frac{1}{3}N(0,0.2^2)$; the third is the symmetric bimodal density $\frac{1}{2}N(-5/2,1)+\frac{1}{2}N(5/2,1)$; and the fourth is the asymmetric, bimodal density $\frac{2}{5}N(5,5)+\frac{3}{5}N(13,13)$. These densities are displayed by the solid lines in Figure 1. They mimic those used in the simulation study of Delaigle and Gijbels (2004*b*), although we have chosen to focus solely on normal mixtures for computational convenience. For each problem the measurement error was normally distributed with zero mean. The measurement error variance was set at low, moderate and high levels corresponding to 'noise to signal' variance ratios, $\mathrm{var}(Z)/\mathrm{var}(X)$, of 0.1, 0.25 and 0.5 respectively. The densities for the contaminated data are displayed by the variety of broken lines in Figure 1.

[Figure 1 about here.]

Deconvolution density estimates were computed for samples of size $n=100$ and $n=250$. For each combination of target density, noise-to-signal variance ratio, and sample size, 500 data sets were generated. The integrated squared error (ISE) was computed for each density estimate. The ISE results are displayed using box plots on the log-scale in Figures 2–5.

[Figure 2 about here.]

[Figure 3 about here.]

[Figure 4 about here.]

[Figure 5 about here.]

The theoretical potential of weighted kernel estimators for deconvolution is demonstrated by the strong performance of the oracle estimator. Our weighted estimator implemented with weight vector $\boldsymbol{w}_0$ consistently outperforms all other methods, in some cases by a wide margin.

As one would expect, WKDE does not perform as well as the oracle weighted estimator in practice. Nonetheless, based on Figures 2–5 it seems that the performance of WKDE shows a tangible improvement over the classical kernel deconvolution estimator for target densities 2 and 3. The results for densities 1 and 4 are reasonably comparable between the WKDE and classical kernel approaches.

An alternative way of comparing the performance of the WKDE and classical methods is to consider the proportion of data sets for which the former returns smaller integrated squared error than the latter. The results are displayed in this format in Table 1. The pairwise comparisons confirm that WKDE is clearly preferable to the classical method for densities 2 and 3, while the results for density 4 and in particular density 1 are far more balanced.

[Table 1 about here.]

The numerical results were obtained using R (R Development Core Team, 2007) running on a 3GHz PC (Intel® Core$^{\mathrm{TM}}$2 Duo CPU E6850) with 4GB RAM and 8GB swap memory operating under Linux. In terms of computational cost, the WKDE method using the interior point algorithm takes on average 0.34 seconds for a sample size of $n = 250$. In further tests, we found that for $n = 2,500$ the computational time for WKDE is about 150 seconds. This demonstrates that our weighted kernel estimator is feasible for samples of this size. For larger samples it should be possible to keep the computing time in check by using a pre-binned version of the data, although we have not explored this matter in detail.

4.2 Data Analysis

We now turn to a real data example involving measurements on systolic blood pressure (SBP, in mmHg) taken from the Framingham study Kannel et al. (1986). In this study a cohort of men were examined on a number of occasions over an eight year period. We look at the SBP measurements obtained at the third clinic visit for 285 men aged 56 years and over. A pair of readings on SBP were taken at this visit for each subject. We use the set of first SBP measurement for each person as our contaminated data. We employ a normal measurement error model, for which the variance of the error terms can then be estimated using the paired data, as described by Carroll et al. (2006). We obtained a standard deviation of $\hat{\sigma}_Z = 12.8$ which we consider to be fixed for the remainder of our analysis.

Density estimates are displayed in Figure 6. The solid line is a standard kernel estimator, ignoring measurement error, and constructed with Sheather-Jones plug-in bandwidth $h = 7.2$. The dashed line is the WKDE estimate obtained using the same bandwidth, and derived using a ridge parameter of $\gamma = 20.0$. The dotted line is the classical estimate, with deconvolution plug-in bandwidth $h = 4.5$. The WKDE and classical methods produce reasonably similar estimates over much of the range of the data. The most noticeable differences are that WKDE produces a more peaked mode at the centre of the distribution, and that the classical method leads to an ugly negative bump around 85 mmHg. The negative values for the classical kernel deconvolution density could be reset to zero (and the density renormalized), but this would leave a sharp discontinuity in the derivative of $\breve{f}$ at about 92 mmHg. Furthermore, the artificial bump at about 65 mmHg would remain, a consequence of the shape of the classical deconvolution kernel function.

[Figure 6 about here.]

5 Deconvolution of Multivariate Densities

5.1 Weighted Multivariate Kernel Estimators

The weighted kernel estimation approach to density deconvolution can be extended to contaminated multivariate data in a straightforward fashion. The model remains as in (1), but with X, Y and Z now interpreted as p-variate random vectors. The measurement error distribution will typically be assumed to be p-variate normal with zero mean vector and covariance matrix Σ. This matrix need not be diagonal, so we can represent settings in which there is dependence in the measurement error across variables.

In the multivariate setting, the weights for our estimator $\hat{f}_{\hat{\boldsymbol{w}}}$ may continue to be chosen by minimization of the criterion Q with an additional ridge penalty, where the integral in (7) should be interpreted as p-dimensional. When the measurement error is multivariate normal, as described above, and the kernel is multivariate normal with covariance matrix H, equation (7) can be evaluated directly to give

$$Q(\boldsymbol{w}) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left[w_i w_j \phi_{2\Lambda}(Y_i - Y_j) + \phi_{2H}(Y_i - Y_j) - 2 w_i \phi_{\Omega}(Y_i - Y_j) \right] \qquad (12)$$

where ϕ_Σ here denotes a multivariate normal density with zero mean vector and covariance matrix Σ. The matrices Λ and Ω are defined by $\Lambda = \Sigma + H$ and $\Omega = \Sigma + 2H$.

5.2 Data Analysis

Here we consider an extension of the analysis of the Framingham blood pressure data introduced earlier. We examine bivariate data, comprising SBP measurements for each subject at the first examination on their second and third clinic visits respectively. We assume that the measurement errors in the two SBP measurements are independent, and normally distributed with zero mean. The measurement error variances can be estimated using the methodology discussed before, giving $\hat{\Sigma} = \text{diag}(158.8,\ 163.5)$. The bandwidth matrix for the contaminated data was

$$H = \begin{bmatrix} 66.2 & 50.9 \\ 50.9 & 71.1 \end{bmatrix}$$

using the plug-in method of Duong and Hazelton (2003).

The standard density estimate from the uncontaminated data is displayed as a contour plot in the left hand panel of Figure 7. Using weights obtained through WKDE implemented with ridge penalty parameter $\gamma = 0.40$, the right hand panel shows the deconvoluted density estimate. This has a much tighter distribution about the 45° line than the standard density estimate. Deconvolution indicates less variation in personal blood pressure between the two clinic visits than is suggested by the raw data.

[Figure 7 about here.]

6 Conclusions

In this paper we have proposed a new approach to density deconvolution, based on the use of weighted kernel estimators. Our methodology has a number of advantages over the classical kernel technique for deconvolution. First, our weighted kernel estimator avoids the spurious wiggles and regions of negative density that arise from the shape of the effective kernels employed in the classical method. Second, results from our simulation study indicate that weighted kernel estimation can provide tangible improvements in performance over the classical estimators for moderate sample sizes. Third, our approach generalizes very simply to multivariate setting, even when the measurement error is correlated across variables. Implementation of the classical method in such circumstances can be challenging because of the complexity of the integrals required to compute the effective deconvolution kernels.

It is natural to compute the weights for our kernel estimator so as to minimize some measure of discrepancy between the standard kernel estimate from the contaminated data on the one hand, and the convolution of the weighted deconvolution estimate with the measurement error density on the other hand. Our preference is to use an integrated squared difference between these densities but there are many alternatives that seem reasonable, at first sight at least. For example, one could seek to minimize the integrated absolute difference between the densities, or seek a perfect match between the densities on some finite set of values. There remains scope for further research in this matter.

Acknowledgement

The authors thank Professor Irene Gijbels for her helpful comments, and Dr Aurore Delaigle for supplying computer code for bandwidth selection for classical kernel deconvolution density estimators. The authors also acknowledge the comments made by two anonymous referees, which have led to very significant improvements in the paper.

References

Carroll, R. and Hall, P. (1988), 'Optimal rates of convergence for deconvolving a density', *Journal of the American Statistical Association* **83**, 1184–1186.

Carroll, R. J., Ruppert, D., Stefanski, L. A. and Crainiceanu, C. (2006), *Measurement Error in Nonlinear Models*, 2 edn, Chapman & Hall/CRC, Boca Raton, FL 33431.

Cristianini, N. and Shawe-Taylor, J. (2000), *An Introduction to Support Vector Machines*, Cambridge University Press, Cambridge.

Dai, Y. H. and Fletcher, R. (2006), 'New algorithms for singly linearly constrained quadratic programs subject to lower and upper bounds', *Mathematical Programming* **106**(3), 403–421.

Delaigle, A. and Gijbels, I. (2004*a*), 'Bootstrap bandwidth selection in kernel density estimation from a contaminated sample', *Annals of the Institute of Statistical Mathematics* **56**(1), 19–47.

Delaigle, A. and Gijbels, I. (2004*b*), 'Practical bandwidth selection in deconvolution kernel density estimation', *Computational Statistics & Data Analysis* **45**(2), 249–267.

Devroye, L. (1989), 'Consistent deconvolution in density estimation', *Canadian Journal of Statistics* **17**, 235–239.

Duong, T. and Hazelton, M. L. (2003), 'Plug-in bandwidth matrices for bivariate kernel density estimation', *Journal of Nonparametric Statistics* **15**(1), 17–30.

Efromovich, S. (1997), 'Density estimation for the case of supersmooth measurement error', *Journal of the American Statistical Association* **92**, 526–535.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004), 'Least angle regression (with discussion)', *Annals of Statistics* **32**(2), 407–499.

Eggermont, P. and LaRiccia, V. (1997), 'Nonlinearly smoothed EM density estimation with automated smoothing parameter selection for nonparametric deconvolution problems', *Journal of the American Statistical Association* **92**, 1451–1458.

Fan, J. (1991*a*), 'Global behavior of deconvolution kernel estimates', *Statistica Sinica* **1**(2), 541–551.

Fan, J. (1991*b*), 'On the optimal rates of convergence for nonparametric deconvolution problems', *The Annals of Statistics* **19**(3), 1257–1272.

Fan, J. (1992), 'Deconvolution with supersmooth distributions', *The Canadian Journal of Statistics. La Revue Canadienne de Statistique* **20**(2), 155–169.

Fletcher, R. (1987), *Practical Methods of Optimization*, 2 edn, John Wiley & Sons, New York.

Gill, P. E., Murray, W. and Wright, M. H. (1981), *Practical Optimization*, Academic Press, New York.

Goldfarb, D. and Idnani, A. (1983), 'A numerically stable dual method for solving strictly convex quadratic programs', *Mathematical Programming* **27**, 1–33.

Hall, P. and Qiu, P. (2005), 'Discrete-transform approach to deconvolution problems', *Biometrika* **92**, 135–148.

Hall, P. and Turlach, B. (1999), 'Reducing bias in curve estimation by use of weights', *Computational Statistics & Data Analysis* **30**, 67–86.

Hesse, C. H. (1999), 'Data-driven deconvolution', *Journal of Nonparametric Statistics* **10**(4), 343–373.

Jones, M., Linton, O. and Nielson, J. (1995), 'A simple bias reduction method for density estimation', *Biometrika* **82**, 327–338.

Kannel, W., Neaton, J., Wentworth, D., Thomas, H., Stamler, J., Hulley, S. and Kjelsberg, M. (1986), 'Overall and coronary heart disease mortality rates in relation to major risk factors in 325,348 men screened for MRFIT', *American Heart Journal* **112**, 825–836.

Karatzoglou, A., Meyer, D. and Hornik, K. (2006), 'Support vector machines in R', *Journal of Statistical Software* **15**(9), 1–28.
URL: *http://www.jstatsoft.org/v15/i09/*

Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A. (2004), 'kernlab – an S4 package for kernel methods in R', *Journal of Statistical Software* **11**(9), 1–20.
URL: *http://www.jstatsoft.org/v11/i09/*

Koo, J.-Y. and Park, B. U. (1996), '*B*-spline deconvolution based on the EM algorithm', *Journal of Statistical Computation and Simulation* **54**(4), 275–288.

Liu, M. and Taylor, R. (1990), 'Simulations and computations of nonparametric estimates for the deconvolution problem', *Journal of Statistical Computation and Simulation* **35**, 145–167.

Masry, E. (1991), 'Multivariate probability density deconvolution for stationary random processes', *IEEE Transactions on Information Theory* **37**, 1105–1115.

Masry, E. (2003), 'Deconvolving multivariate kernel density estimates from contaminated associated observations', *IEEE Transactions on Information Theory* **49**, 2941–2952.

Mendelsohn, J. and Rice, R. (1982), 'Deconvolution of microfluorometric histograms with *B* splines', *Journal of the American Statistical Association* **77**, 748–753.

Moguerza, J. M. and Muñoz, A. (2006), 'Support vector machines with applications (with disussion)', *Statistical Science* **21**(3), 322–362.

Osborne, M. R., Presnell, B. and Turlach, B. A. (2000), 'A new approach to variable selection in least squares problems', *IMA Journal of Numerical Analysis* **20**(3), 389–403.

Pensky, M. (2002), 'Density deconvolution based on wavelets with bounded supports', *Statistics & Probability Letters* **56**(3), 261–269.

Pensky, M. and Vidakovic, B. (1999), 'Adaptive wavelet estimator for nonparametric density deconvolution', *The Annals of Statistics* **27**(6), 2033–2053.

R Development Core Team (2007), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. **URL:** *http://www.R-project.org*

Rosset, S. and Zhu, J. (2007), 'Piecewise linear regularized solution paths', *Annals of Statistics* **35**(3), 1012–1030.

Schölkopf, B. and Smola, A. J. (2002), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA.

Serafini, T., Zanghirati, G. and Zanni, L. (2005), 'Gradient projection methods for quadratic programs and applications in training support vector machines', *Optimization Methods and Software* **20**(2–3), 353–378.

Sheather, S. J. and Jones, M. C. (1991), 'A reliable data-based bandwidth selection method for kernel density estimation', *Journal of the Royal Statistical Society series B* **53**, 683–690.

Simonoff, J. (1996), *Smoothing Methods in Statistics*, Springer, New York.

Sircombe, K. and Hazelton, M. (2004), 'Comparison of detrital zircon age distributions by kernel functional estimation', *Sedimentary Geology* **171**(1–4), 91–111.

Stefanski, L. (1990), 'Rates of convergence of some estimators in a class of deconvolution problem', *Statistics and Probability Letters* **9**, 229–235.

Stefanski, L. and Carroll, R. J. (1990), 'Deconvoluting kernel density estimators', *Statistics* **21**(2), 169–184.

Turlach, B. A. and Weingessel, A. (2007), *quadprog: Functions to solve Quadratic Programming Problems.* S original by Berwin A. Turlach, R port by Andreas Weingessel; R package version 1.4-11.

van de Wiel, M. A. and Kim, K. I. (2007), 'Estimating the false disovery rate using nonparametric deconvolution', *Biometrics* **63**, 806–815.

Van Es, B. and Uh, H.-W. (2005), 'Aymptotic normality of kernel-type deconvolution estimators', *Scandinavian Journal of Statistics* **32**, 467–483.

Walter, G. G. (1999), 'Density estimation in the presence of noise', *Statistics & Probability Letters* **41**(3), 237–246. Special issue in memory of V. Susarla.

Wand, M. (1998), 'Finite sample performance of deconvolving density estimators', *Statistics and Probability Letters* **37**, 131–139.

Wand, M. and Jones, M. (1995), *Kernel Smoothing*, Chapman & Hall, London.

A Proof of Theorem 1

By the triangle inequality,

$$\int \{g(y) - \hat{f}_{\hat{\boldsymbol{w}}} * \eta(y)\}^2 \, dy \leq \int \{\breve{g}(y) - g(y)\}^2 \, dy + \int \{\breve{g}(y) - \hat{f}_{\hat{\boldsymbol{w}}} * \eta(y)\}^2 \, dy.$$

The first term on the right hand side tends to zero in probability as $n \to \infty$ by standard results for kernel density estimators. Turning to the second term on the right hand side,

$$\begin{aligned} \int \{\breve{g}(y) - \hat{f}_{\hat{\boldsymbol{w}}} * \eta(y)\}^2 \, dy &\leq \int \{\breve{g}(y) - \hat{f}_{\boldsymbol{w}_0} * \eta(y)\}^2 \, dy \\ &\leq \int \{\breve{g}(y) - g(y)\}^2 \, dy + \int \{g(y) - \hat{f}_{\boldsymbol{w}_0} * \eta(y)\}^2 \, dy \\ &\xrightarrow{P} 0 \text{ as } n \to \infty. \end{aligned}$$

It follows from the assumed continuity of the relevant functions that $\hat{f}_{\hat{\boldsymbol{w}}} * \eta(y) \to g(y)$ for any y. Then $\psi_{\hat{f}} \to \psi_g / \psi_\eta = \psi_f$ as $n \to \infty$, and the result follows by application of continuity and inversion theorems for characteristic functions.

The Theorem holds whether or not the constraint $\bar{w} = 1$ is applied. This is because the population mean weight for $\boldsymbol{w}_0$ is unity, since $\mathrm{E}[f(Y)/g(Y)] = 1$.

B Exact algorithm

Note that problem (9) and (10) are essentially of the same structure accept that one uses $\mathbf{Q}$ and the other $\mathbf{Q} + \frac{\gamma}{n}\mathbf{I}$ in the quadratic term. Thus, the algorithm described here can be used to solve either problem but is described using the notation of problem (9).

The algorithm developed here determines the exact solution for (9) using compute-complete-solution-path ideas. For this end, the equality constraint (9b) is replaced by the constraint $\sum_{i=1}^{n} w_i = t$. After applying this change, it is easy to realize that the solution $\boldsymbol{w} = \boldsymbol{w}(t)$ of the modified problem is piecewise linear and continuous in t; see

Rosset and Zhu (2007). Thus, we just have to run a homotopy algorithm from $t = 0$ to $t = n$ to find the solution of (9).

To develop this algorithm, note that the KKT-conditions for the solution of

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimise}} \quad & \frac{1}{2}\boldsymbol{w}^{\mathsf{T}}\mathbf{Q}\boldsymbol{w} - \boldsymbol{b}^{\mathsf{T}}\boldsymbol{w} && (13a) \\ \text{subject to} \quad & \sum_{i=1}^{n} w_i = t && (13b) \\ & 0 \leq w_i, \qquad i = 1, \ldots, n && (13c) \end{aligned}$$

are that $\boldsymbol{\kappa} \in \mathbb{R}^n$ and $\nu \in \mathbb{R}$ exist with

$$\mathbf{Q}\boldsymbol{w} - \boldsymbol{b} - \boldsymbol{\kappa} - \nu\mathbf{1} = \mathbf{0} \tag{14}$$

and

$$\kappa_i \geq 0, \quad w_i \geq 0, \quad w_i\kappa_i = 0 \qquad i = 1, \ldots, n \tag{15}$$

$$\nu\left(\sum_{i=1}^{n} w_i - t\right) = 0 \tag{16}$$

and $\mathbf{1}$ and $\mathbf{0}$ denote vectors of ones and zeros, respectively.

To fix notation, for a set of indices, $\Omega \subseteq \{1, 2, \ldots, n\}$, with $|\Omega| = r$, we denote by $\mathbf{Q}_\Omega$ the $r \times r$ submatrix of $\mathbf{Q}$ indexed by Ω. Likewise, for any vector $\boldsymbol{x}$, the subvector indexed by Ω and Ω^c are denoted $\boldsymbol{x}_\Omega$ and $\boldsymbol{x}_{\Omega^c}$, respectively. $\mathbf{P}_\Omega$ denotes the permutation matrix such that $\boldsymbol{x} = \mathbf{P}_\Omega \left(\begin{smallmatrix} \boldsymbol{x}_\Omega \\ \boldsymbol{x}_{\Omega^c} \end{smallmatrix} \right)$.

The proof that the solution of (13) is piecewise linear and continuous as function of t, with breakpoints at $0 = t_0 < t_1 < t_2 < \ldots$, proceeds by construction. We will describe how the complete solution path can be calculated. Essentially, starting at $t_0 = 0$ the break points are calculated iteratively and the solution at each break point is determined. If the solution is desired for a value of t that is not a break point, then that solution can be calculated by simple linear interpolation of the solutions that correspond to the two break points that bracket t. In our case, we are interested in running the algorithm up to $t = n$ and stop at the corresponding solution.

First, note that if $\Omega = \{i : w_i > 0\}$, then the complementary conditions (15) imply that $\boldsymbol{\kappa}_\Omega = \mathbf{0}$; whence, using (14), it follows that

$$|\nu| = \|(\mathbf{Q}\boldsymbol{w})_\Omega - \boldsymbol{b}_\Omega\|_\infty$$

Thus, for $t_0 = 0$ we initialize the algorithm as follows:

$$\boldsymbol{w}^0 = \mathbf{0}, \quad \Omega = \{i : b_i = \max_j b_j\}, \quad \nu^0 = -\|\boldsymbol{b}_\Omega\|_\infty, \quad \boldsymbol{\kappa}^0 = -\boldsymbol{b} - \nu^0 \mathbf{1}$$

Clearly, this choice satisfy (14)–(16).

Now assume we are at break point t_s, then we proceed as follows:

1. Calculate $\boldsymbol{\delta}_\Omega = \mathbf{Q}_\Omega^{-1}\mathbf{1}$, $\boldsymbol{\delta} = P_\Omega \left(\begin{smallmatrix}\boldsymbol{\delta}_\Omega \\ \mathbf{0}\end{smallmatrix}\right)$. and $\tilde{\boldsymbol{\delta}} = \mathbf{Q}\boldsymbol{\delta}$.

2. For $\tau > 0$, parameterize the future path as:

$$\begin{aligned} \boldsymbol{w}^\tau &= \boldsymbol{w}^{t_s} + \tau\boldsymbol{\delta} \\ \nu^\tau &= \nu^{t_s} + \tau \\ \boldsymbol{\kappa}^\tau &= \boldsymbol{\kappa}^{t_s} - \tau(\mathbf{1} - \tilde{\boldsymbol{\delta}}) \end{aligned}$$

   Note that by construction $\tilde{\boldsymbol{\delta}}_\Omega = \mathbf{1}$. Hence, clearly, if $\boldsymbol{w}^{t_s}$, ν^{t_s} and $\boldsymbol{\kappa}^{t_s}$ satisfy (14)–(16) so do $\boldsymbol{w}^\tau$, ν^τ and $\boldsymbol{\kappa}^\tau$ for $\tau > 0$, τ small, and we can proceed along this path.

3. (a) If $\Omega^c = \emptyset$ and all entries in $\boldsymbol{\delta}$ are non-negative, then there will be no further breakpoints and we can continue to move along this path until the desired solution is found.

   (b) Otherwise we move along this path until one of the following two events happen:

   - An entry of $\boldsymbol{\kappa}^\tau_{\Omega^c}$ becomes zero, this can only happen if the corresponding entry in $\mathbf{1} - \tilde{\boldsymbol{\delta}}_{\Omega^c}$ is positive.
     That is, for all $i \in \Omega^c$ such that $1 - \tilde{\delta}_i > 0$ we have to calculate $\tau_i = \frac{\kappa_i^{t_s}}{1-\tilde{\delta}_i}$.
     Set τ^1 to the minimum of these values ($\tau^1 = \infty$ if no such τ_i exist) and i_0^1 to the corresponding index i.
   - An entry of $\boldsymbol{w}^\tau_\Omega$ becomes zero, this can only happen if the corresponding entry in $\boldsymbol{\delta}_\Omega$ is negative.
     That is, for all $i \in \Omega$ such that $\tilde{\delta}_i < 0$ we have to calculate $\tau_i = -\frac{w_i}{\delta_i}$. Set τ^2 to the minimum of these values ($\tau^2 = \infty$ if no such τ_i exist) and i_0^2 to the corresponding index i.

   Set $\tau^0 = \min\{\tau^1, \tau^2\}$. If $\tau^0 = \tau^1$, then $\Omega = \Omega \cup \{i_0^1\}$, otherwise $\Omega = \Omega \setminus \{i_0^2\}$. Determine $\boldsymbol{w}^{t_{s+1}} = \boldsymbol{w}^{\tau_0}$, $\nu^{t_{s+1}} = \nu^{\tau_0}$, $\boldsymbol{\kappa}^{t_{s+1}} = \boldsymbol{\kappa}^{\tau_0}$ and $t_{s+1} = \sum_{i=1}^n w_i^{t_{s+1}}$. Iterate by returning to step 1.

As mentioned in the main text, the matrix $\mathbf{Q}$ in the quadratic program considered in this problem is typically numerically singular despite being theoretically positive definite.

For this reason the quadratic program (9) cannot be solved by an algorithm such as the one proposed by Goldfarb and Idnani (1983), implemented in the R package `quadprog` (Turlach and Weingessel, 2007), which first calculates the unconstrained solution and then iteratively enforces violated constraints. Other commonly used general quadratic programming solver require an initial step in which a feasible starting point is determined; see, among others, Fletcher (1987) and Gill et al. (1981).

This algorithm avoids both these problems by starting at $\boldsymbol{w} = \mathbf{0}$ and operating only on submatrices $\mathbf{Q}_\Omega$ of $\mathbf{Q}$. Though, in our numerical experiments we observed that, in particular with sample sizes above $n = 500$, occasionally the algorithm fails due to numerical singularity of $\mathbf{Q}_\Omega$ if no regularization is used.

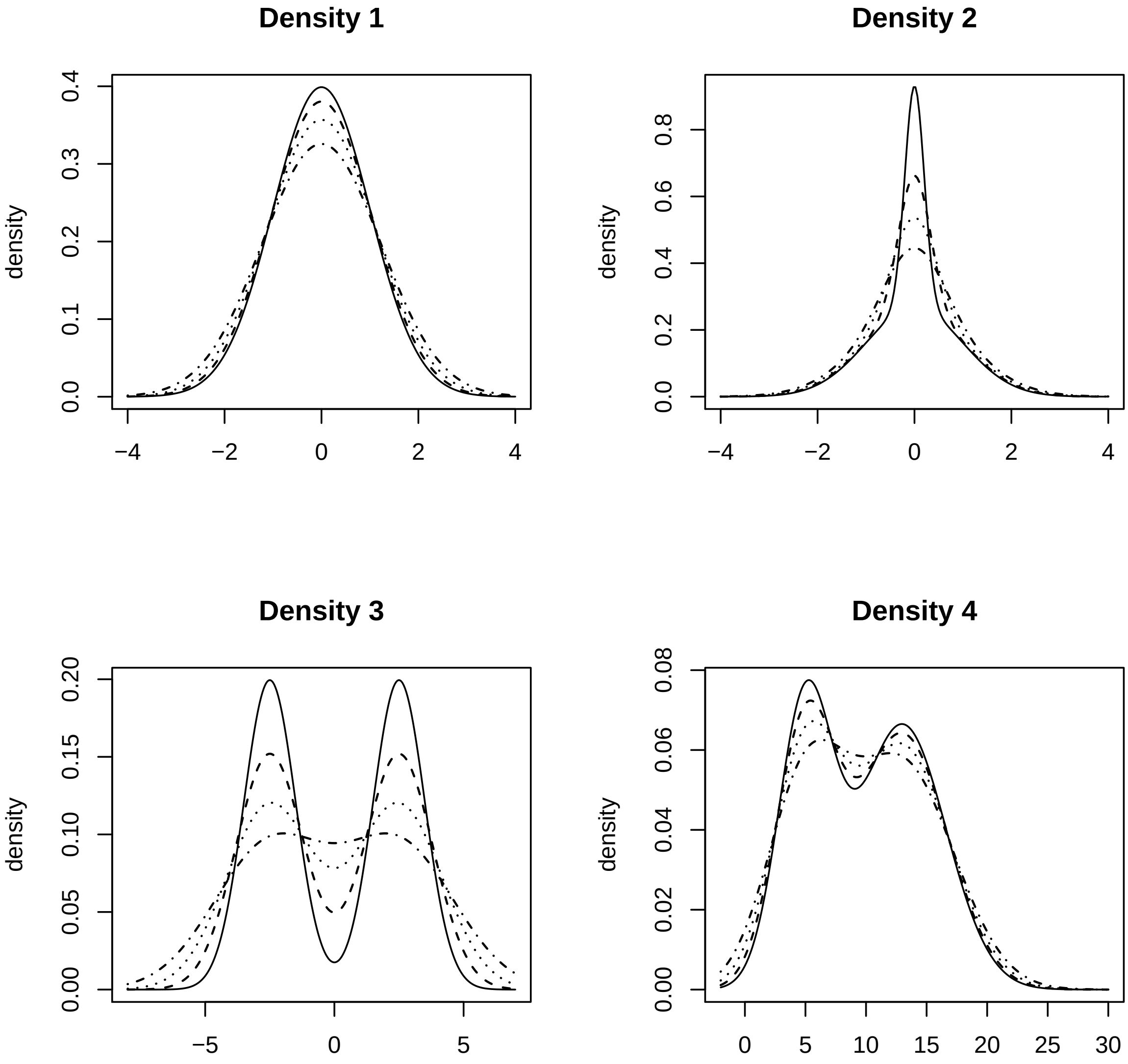

Figure 1: Target densities, f (solid lines) and densities of the contaminated data, g with 'noise to signal' variance ratios, $\text{var}(Z)/\text{var}(X)$, set at 0.1 (dashed line), 0.25 (dotted lines) and 0.5 (dot-dash line).

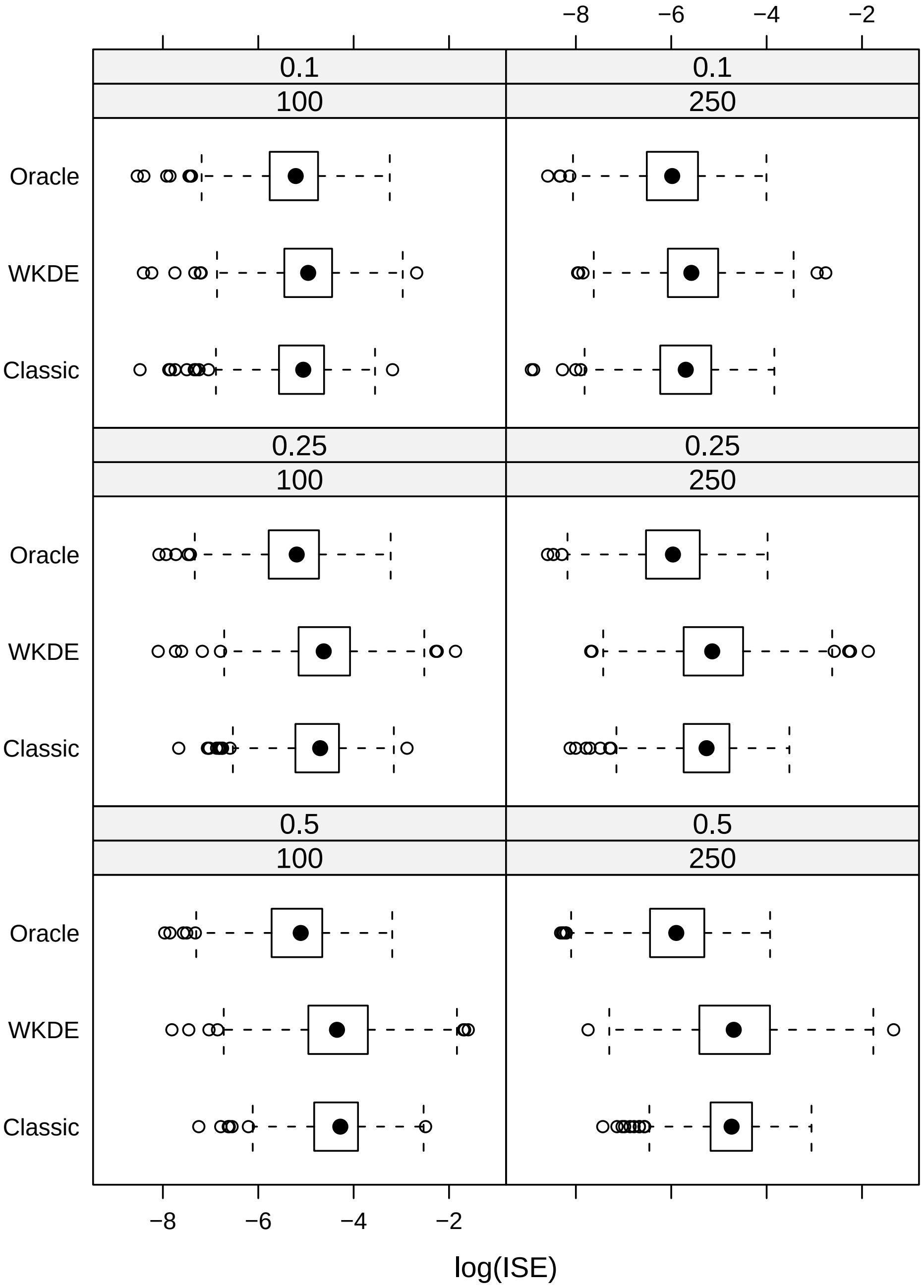

Figure 2: Box plots of integrated squared error for deconvolution estimators for target density 1, categorized by sample size (100 and 250) and noise-to-signal ratio (0.1, 0.25 and 0.5).

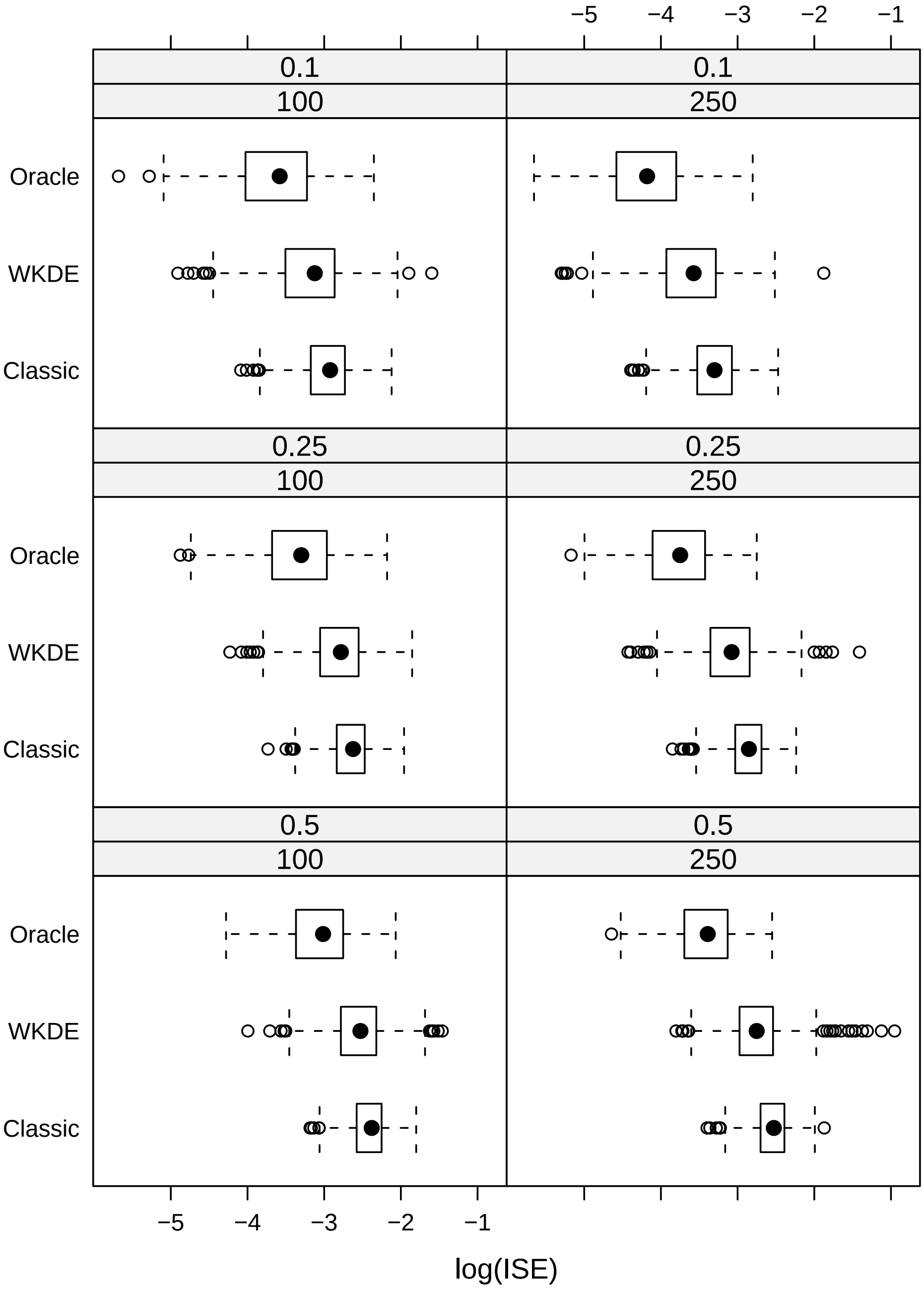

Figure 3: Box plots of integrated squared error for deconvolution estimators for target density 2, categorized by sample size (100 and 250) and noise-to-signal ratio (0.1, 0.25 and 0.5).

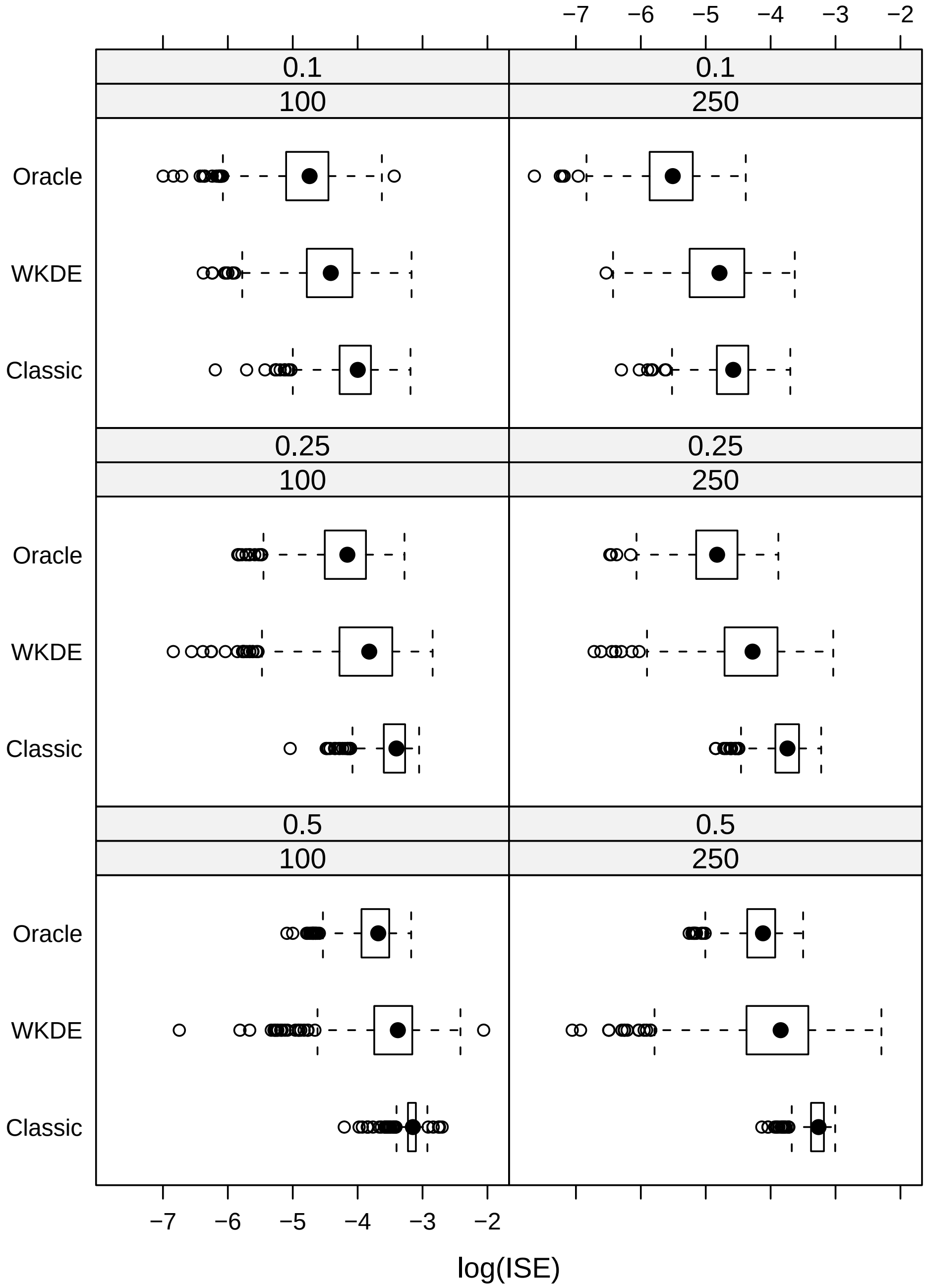

Figure 4: Box plots of integrated squared error for deconvolution estimators for target density 3, categorized by sample size (100 and 250) and noise-to-signal ratio (0.1, 0.25 and 0.5).

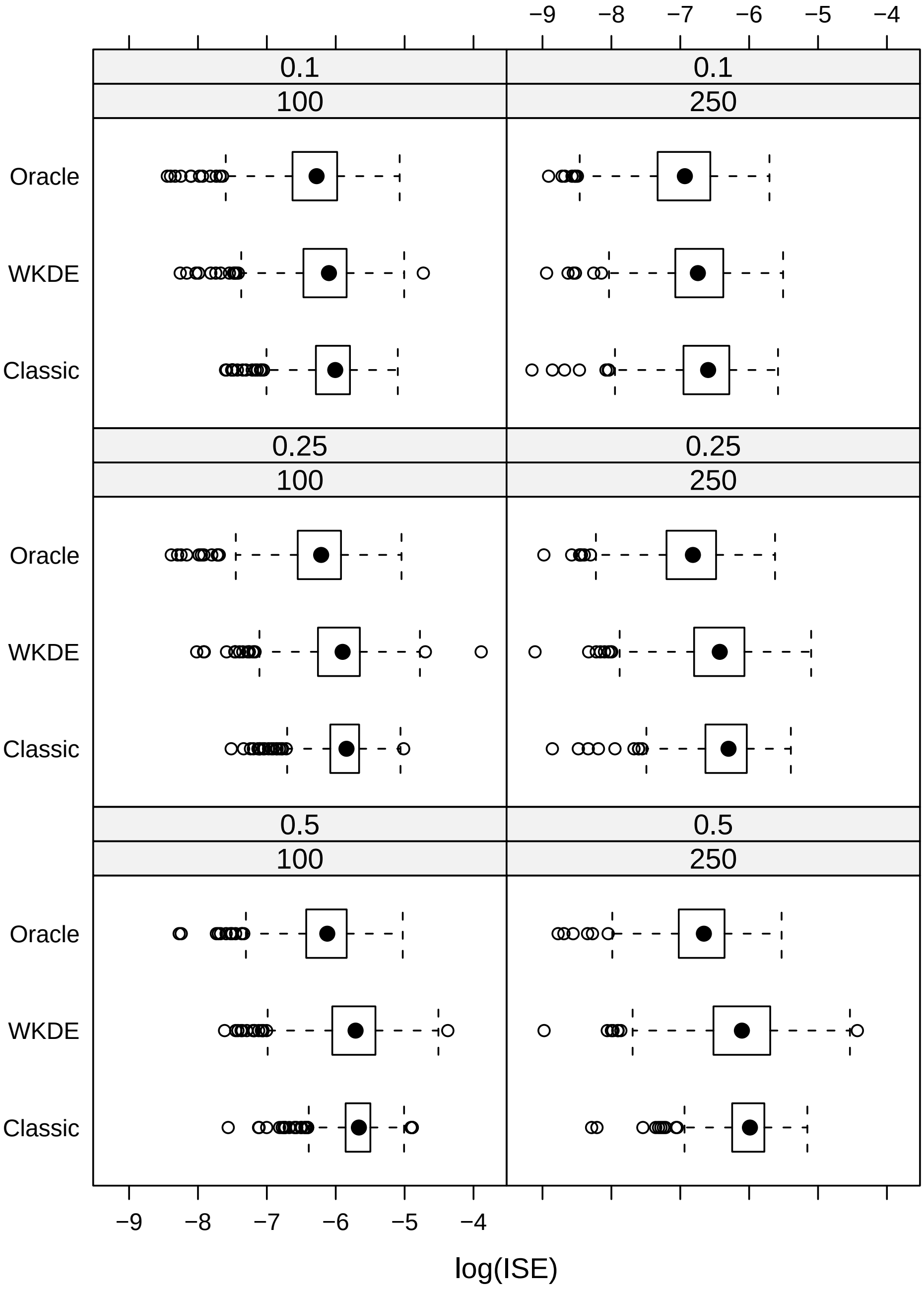

Figure 5: Box plots of integrated squared error for deconvolution estimators for target density 4, categorized by sample size (100 and 250) and noise-to-signal ratio (0.1, 0.25 and 0.5).

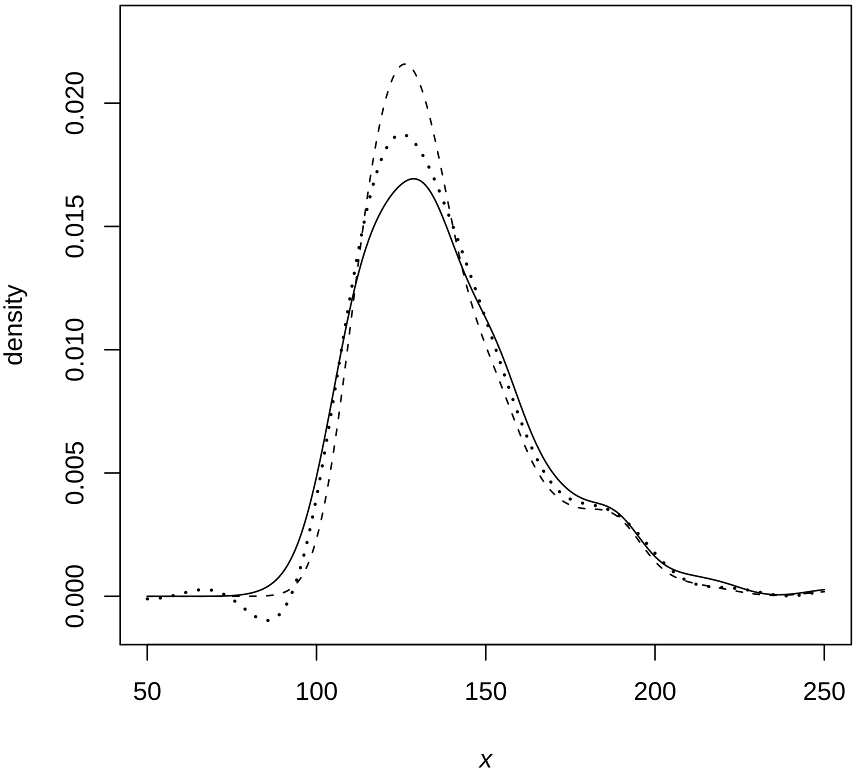

Figure 6: Density estimates for systolic blood pressure data for $n = 285$ men. The solid line is the density estimate ignoring measurement error. The dashed line is the weighted kernel deconvolution estimate. The dotted line is the classical deconvolution estimate.

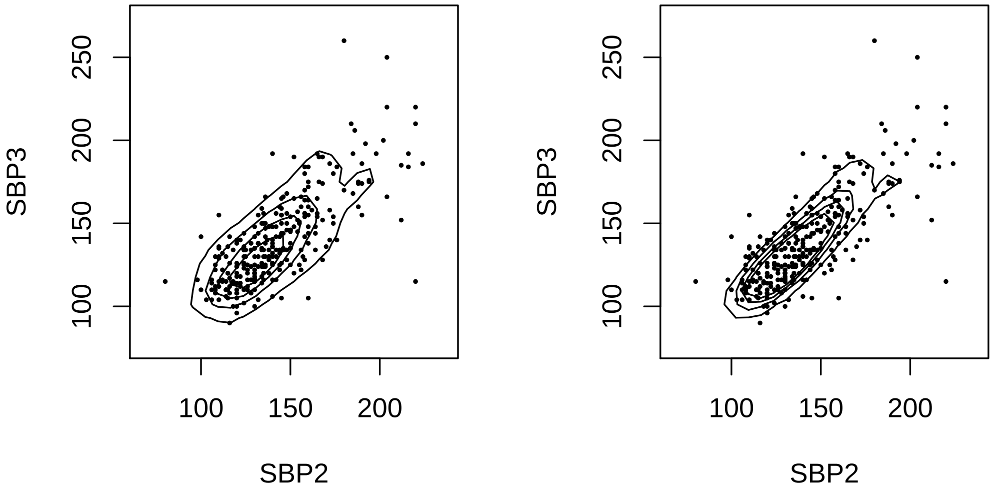

Figure 7: Contour plots of density estimates for bivariate blood pressure data from $n = 285$ men, comprising measurements taken at two separate clinic visits. The left hand panel displays the standard kernel density estimate from the contaminated data. The right hand panel displays the weighted kernel deconvolution density estimate. The contaminated data are displayed as a scatter plot in both panels.

$\mathrm{var}(Z)/\mathrm{var}(X)$	n	Density 1	Density 2	Density 3	Density 4
0.1	100	0.498	0.890	0.912	0.778
0.1	250	0.460	0.934	0.698	0.758
0.25	100	0.526	0.862	0.868	0.690
0.25	250	0.456	0.894	0.888	0.694
0.5	100	0.552	0.826	0.818	0.600
0.5	250	0.508	0.866	0.920	0.636

Table 1: Pairwise comparison of the classical and WKDE methods. The tabulated values show the proportion of simulated data sets for which WKDE returned a lower integrated squared error than the classical method.