Bi-Gaussianized calibration revisited: Reducing the potential for overstating strength of evidence

Authors and affiliations:

Geoffrey Stewart Morrison [1,2,*]

Peter Vergeer [3]

Dylan Borchert [4]

[1] Forensic Data Science Laboratory, Aston University, Birmingham, UK

[2] Forensic Evaluation Ltd, Birmingham, UK

[3] Netherlands Forensic Institute, The Hague, The Netherlands

[4] Department of Mathematical Sciences, New Mexico State University, Las Cruces, New Mexico, USA

[*] e-mail: geoff-morrison@forensic-evaluation.net

ORCID:

Geoffrey Stewart Morrison 0000-0001-8608-8207

Peter Vergeer 0000-0001-8613-2079

Dylan Borchert 0009-0002-3887-7759

Author contributions:

Geoffrey Stewart Morrison: Conceptualization, Methodology, Software, Formal analysis; Writing - Original Draft; Visualization.

Peter Vergeer: Conceptualization, Software, Writing - Review & Editing.

Dylan Borchert: Conceptualization, Software, Writing - Review & Editing.

Disclaimer:

All opinions expressed in the present paper are those of the authors, and, unless explicitly stated otherwise, should not be construed as representing the policies or positions of any organizations with which the authors are associated.

Declaration of competing interest:

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding:

This work was not supported by external research funding.

Bi-Gaussianized calibration revisited: Reducing the potential for overstating strength of evidence

Abstract:

If a forensic-evaluation system is well calibrated, the likelihood ratio of the likelihood ratio is the likelihood ratio. A perfectly calibrated bi-Gaussian system is a system for which the distributions of the natural logarithms of the likelihood ratios that the system outputs in response to different-source and same-source inputs are both Gaussian, both have the same variance, and their means are at minus half a variance and plus half a variance, respectively. Morrison (2024) https://doi.org/10.1093/lpr/mgae004 introduced bi-Gaussianized calibration which monotonically (but non-linearly) maps the distribution of uncalibrated log likelihood ratios toward the Gaussian-mixture distribution of a perfectly calibrated bi-Gaussian system. Bi-Gaussianized calibration requires the determination of the variance of the perfectly calibrated bi-Gaussian system toward which to map the uncalibrated log likelihood ratios. The methods used in Morrison (2024) for determining the target variance had several shortcomings, including having a positive bias: they tended to overestimate the value of the variance. This resulted in a tendency to overstate the strength of evidence: the calibrated log likelihood ratios tended to be further from the neutral value of zero than they should be. The present paper tests methods that are expected to overcome the shortcomings of the earlier methods. The new methods include recalculation of equal-error rate using the expected value of the posterior distribution of a beta-binomial model with uniform priors. As the amount of training data decreases and/or the separation between the different-source and same-source distributions increases, the beta-binomial method increases the value of the equal-error rate, which reduces the value of the target variance. This shrinks the calibrated log-likelihood-ratio values toward the neutral

value of zero and reduces the probability of overstating the strength of evidence. Another new method compresses the empirical-cumulative-probability-density function for the training data and thereby further compresses the calibrated log-likelihood-ratio values toward the neutral value of zero.

Keywords:

Beta-binomial model; Calibration; Compression; Likelihood ratio; Shrinkage

1. Introduction

The values output by a forensic-evaluation system should be well calibrated. If they are not, they are not interpretable as likelihood ratios (Vergeer et al., 2020). If a forensic-evaluation system is well calibrated, the likelihood ratios of the likelihood-ratio values that it outputs will be the same as the likelihood-ratio values that it outputs. "That is, the likelihood ratio of the likelihood ratio is the likelihood ratio." (Birdsall, 1973, p. 18). This can be expressed formulaically as in Equation (1), in which Λ is a likelihood ratio, f is a probability-density function, and H_s and H_d are the same-source and different-source hypotheses, respectively.

$$\Lambda = \frac{f(\Lambda|H_\mathrm{s})}{f(\Lambda|H_\mathrm{d})} \quad (1)$$

A *perfectly calibrated bi-Gaussian system* is a system that has the following properties: The distribution of the natural logarithms of the likelihood ratios that the system outputs in response to different-source input pairs and the distribution of the natural logarithms of the likelihood ratios that it outputs in response to same-source input pairs are both Gaussian and both have the same standard deviation, $\ln(\Lambda_\mathrm{d}) \sim \mathcal{N}(\mu_\mathrm{d}, \sigma)$ and $\ln(\Lambda_\mathrm{s}) \sim \mathcal{N}(\mu_\mathrm{s}, \sigma)$, and the means of the different-source and same-source distributions

are $\mu_{\mathrm{d}} = -\frac{\sigma^2}{2}$ and $\mu_{\mathrm{s}} = +\frac{\sigma^2}{2}$, respectively (Peterson et al., 1954, §4.3, §4.9; Birdsall, 1973, §1.3; Good 1985, §6; van Leeuwen & Brümmer, 2013; Morrison, 2021; Morrison 2024); see Equation (2), in which $f(x|\mu,\sigma)$ is a univariate Gaussian probability-density function.[1] Figure 1 shows examples of ln(Λ) distributions and corresponding Tippett plots for perfectly calibrated bi-Gaussian systems with $\sigma \in \{1,2,3,4\}$.[2]

$$\Lambda = \frac{f\left(\ln(\Lambda)\middle|+\frac{\sigma^2}{2},\sigma\right)}{f\left(\ln(\Lambda)\middle|-\frac{\sigma^2}{2},\sigma\right)} \quad (2)$$

[1] For a perfectly calibrated bi-Gaussian system, $d' = \sigma$.

[2] Throughout the paper, we use "ln(Λ)" when referring to calibrated natural-log likelihood ratios, and we use "log(Λ)", without specifying the base for the logarithm, when referring to uncalibrated log likelihood ratios. For uncalibrated log(Λ), the base of the logarithm is not relevant since different bases simply result in different linear scaling in the logarithmic space and differences in linear scaling of uncalibrated log(Λ) have no effect on the output of calibration. When drawing Tippett plots, we scale the *x*-axis using base-ten logarithms, "$\log_{10}(\Lambda)$".

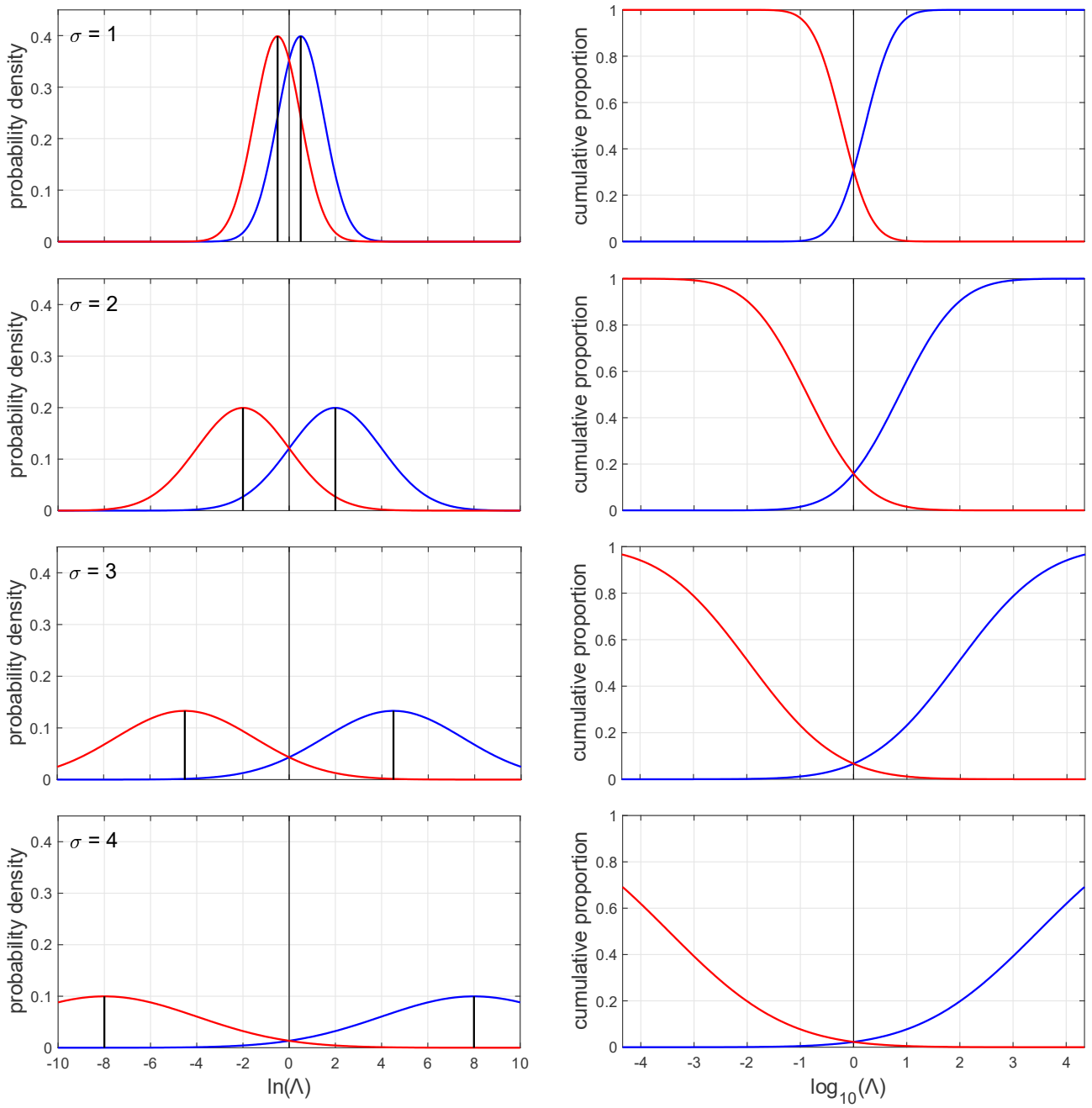


Figure 1. Left column: Different-source and same-source distributions of perfectly calibrated bi-Gaussian systems with $\sigma \in \{1,2,3,4\}$. Right column: Tippett plots corresponding to the distributions in the left column.

Morrison (2024) introduced *bi-Gaussianized calibration*, a calibration method that monotonically (but non-linearly) maps the distribution of uncalibrated log(Λ) (irrespective of same-source and different-source labels) toward the Gaussian-mixture distribution of a perfectly calibrated bi-Gaussian system. Morrison (2024)

demonstrated that bi-Gaussianized calibration can produce results that are closer to perfect calibration than the output of logistic-regression calibration (Brümmer & du Preez, 2006; González-Rodríguez et al. 2007; Morrison, 2013) and of calibration via the pool-adjacent-violators (PAV) algorithm (Ayer et al., 1955; Zadrozny & Elkan, 2002; Brümmer & du Preez, 2006; Ramos et al., 2017).

In order to perform bi-Gaussianized calibration, one must determine the particular perfectly calibrated bi-Gaussian system toward which to map the uncalibrated log(Λ), i.e., one must determine the target value for σ. For a perfectly calibrated bi-Gaussian system, there is a one-to-one mapping between its σ and its equal-error rate ($E_{=}$), and there is a one-to-one mapping between its σ and its log-likelihood-ratio cost (C_{llr}; Brümmer & du Preez, 2006). Thus, for a perfectly calibrated bi-Gaussian system, there is also a one-to-one mapping between its $E_{=}$ and its C_{llr}. Table 1 gives the $E_{=}$ values and C_{llr} values for perfectly calibrated bi-Gaussian systems with $\sigma \in \{0,1,2,3,4,5,6,\infty\}$.

Table 1. $E_{=}$ values and C_{llr} values for perfectly calibrated bi-Gaussian systems with $\sigma \in \{0,1,2,3,4,5,6,\infty\}$.

$\boldsymbol{\sigma}$	$\boldsymbol{E_{=}}$	$\boldsymbol{C_{\mathrm{llr}}}$
0	0.5	1
1	0.309	0.839
2	0.159	0.514
3	0.067	0.240
4	0.023	0.087
5	0.006	0.025
6	0.001	0.006
∞	0	0

There is an analytical solution for the $E_{=}$-to-σ mapping function: see Equation (3), in

which F^{-1} is the inverse cumulative probability function for a univariate standard Gaussian distribution ($\mu = 0,\ \sigma = 1$). See Morrison (2024) for the derivation of this equation. Figure 2a shows a plot of the function.

(3) $\sigma = -2F^{-1}(E_{=})$

In contrast, as far as we are aware, there is not an analytical solution for the C_{llr}-to-σ mapping function. Morrison (2024) used numerical integration (described in §2.5 below) to calculate the C_{llr} value corresponding to a given σ value (the same method was used to calculate the C_{llr} values in Table 1 of the present paper). Morrison (2024) calculated $\ln(C_{\text{llr}})$ values corresponding to σ^2 values equally spaced at intervals of 0.25 in the range 0 to 25, then fitted a curve to the $\ln(C_{\text{llr}})$ values versus σ^2 values, then inverted the formula for the curve so that it could be used to calculate the target σ value given a query C_{llr} value. The formula for that curve is given in Equation (4). The fitted coefficient values were $b = 17.7$ and $c = 9.33 \times 10^{-3}$.[3] Figure 2b shows a plot of this function and also a plot of the output of the "exact" method that is described in §2.5 below.

(4) $\sigma = \sqrt{-\frac{1}{c}\ln\left(\frac{1}{b}\ln(C_{\text{llr}}) + 1\right)}$

[3] If we repeat this process but extend the range of σ^2 values out to 36, the fitted coefficient values are $b = 24.6$ and $c = 6.54 \times 10^{-3}$.

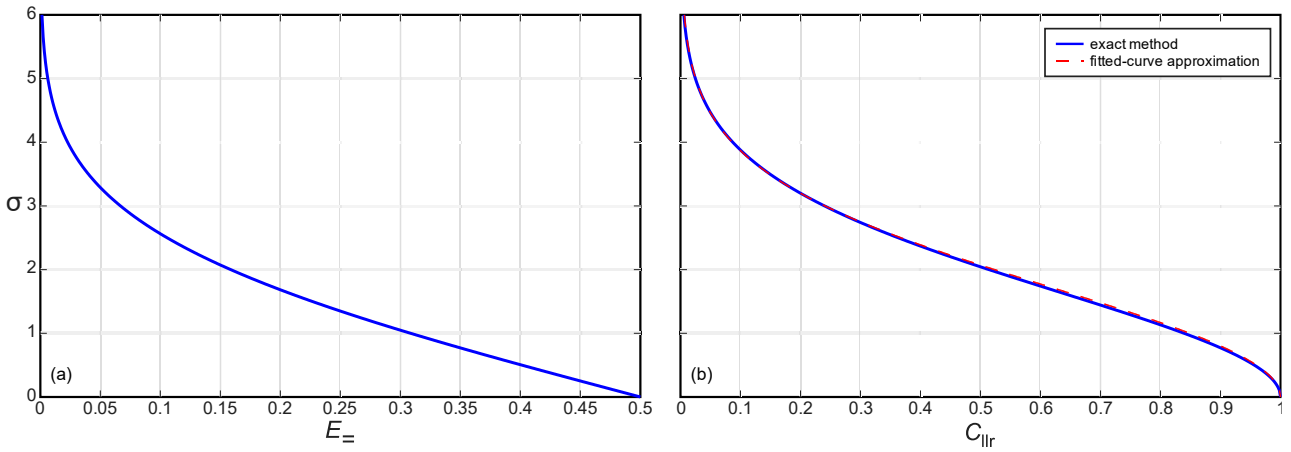


Figure 2. (a) Analytic $E_=$-to-σ mapping function for perfectly calibrated bi-Gaussian systems. (b) C_{llr}-to-σ mapping functions for perfectly calibrated bi-Gaussian systems: Exact method, described in §2.5 below, and fitted-curve approximation from Morrison (2024).

Morrison (2024) explored the use of four different methods for calculating the target σ for bi-Gaussianized calibration. Each method used either the $E_=$-to-σ mapping function or the fitted-curve C_{llr}-to-σ mapping function described above. Details of the methods are provided in Morrison (2024). Briefly:

1. **EER method:** $E_=$ was calculated by: 1. sorting all of the uncalibrated log(Λ) values from smallest to largest; 2. for each uncalibrated log(Λ) value, placing a threshold at that value and calculating the false-alarm rate and miss rate at that value; 3. finding the first value for which the false-alarm rate was less than the miss rate; 4. taking the false-alarm rate at that value; 5. taking the miss rate at the immediately preceding value; and 6. calculating the mean of that false-alarm rate and that miss rate. For an example, see Figure 4 in §2.2 below.

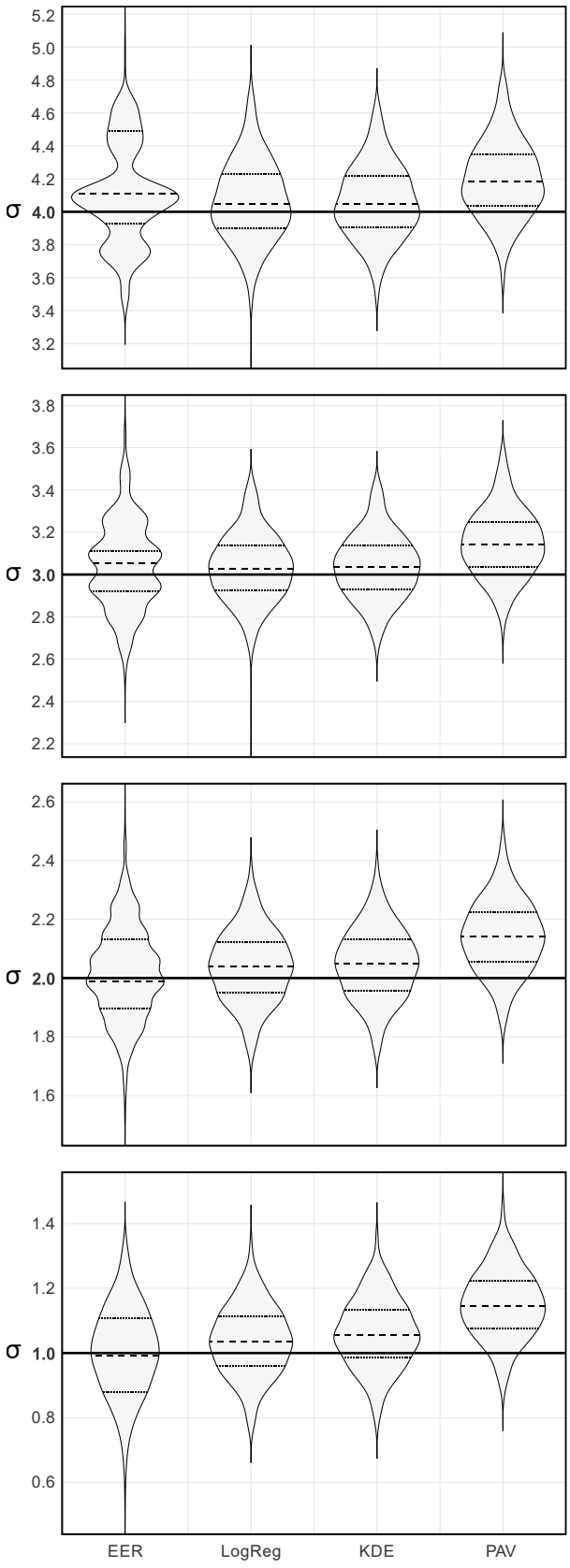
2. **LogReg method:** Logistic regression was used to calibrate the uncalibrated log(Λ) values. Logistic regression calculates values for an intercept coefficient and a slope coefficient. These are used to linearly shift and scale uncalibrated log(Λ) values. The same uncalibrated log(Λ) values that were used to train the logistic-regression model were calibrated, and the C_{llr} value for the resulting logistic-

regression-calibrated log(Λ) values was calculated.

3. **KDE method:** 1. A KDE model was fitted to the same-source uncalibrated log(Λ) values. 2. A KDE model was fitted to the different-source uncalibrated log(Λ) values. 3. For a given query value, a calibrated likelihood ratio was calculated by dividing the probability density of the same-source model evaluated at the query value by the probability density of the different-source model evaluated at the query value. The bandwidths for the KDE models were calculated using Silverman's Gaussian-approximation method (Silverman, 1986). The same uncalibrated log(Λ) values that were used to train the KDE models were calibrated, and the C_{llr} value for the resulting KDE-calibrated log(Λ) values was calculated.

4. **PAV method:** The PAV algorithm was used to calibrate the uncalibrated log(Λ) values. PAV non-linearly shifts and scales uncalibrated log(Λ) values so as to minimize C_{llr}, with the only constraint being maintenance of monotonicity. The same uncalibrated log(Λ) values that were used to train the PAV algorithm were calibrated, and the C_{llr} value for the resulting PAV-calibrated log(Λ) values was calculated.

Morrison (2024) used Monte Carlo simulation to generate synthetic data. 1000 sample sets, each consisting of 100 same-source ln(Λ) values and 4950 different-source ln(Λ) values, were generated from perfectly calibrated bi-Gaussian systems with $\sigma \in \{1,2,3,4\}$. For each sample set, the target σ value was calculated using each of the four methods. Figure 3 shows, for each method, the distributions of the calculated target σ values.



146 **Figure 3.** Violin plots of distributions of target σ values calculated using each of the methods from
147 Morrison (2024). In each panel, the solid horizontal line indicates the σ value used to generate the
148 Monte Carlo samples, $\sigma \in \{1,2,3,4\}$. Each panel is a reproduction of a figure from Morrison (2024).

Although there is an analytical solution for the $E_{=}$-to-σ mapping function, the output of the EER method can be susceptible to noise due to sampling variability. As system performance improves, $E_{=}$ values decrease, but they become dependent on the ranking of sparsely distributed uncalibrated log(Λ) values that are on the tails of both the same-source distribution and the different-source distribution; thus, the $E_{=}$ values become more susceptible to noise due to sampling variability. As can be seen in Figure 3, of the four methods, the output of the EER method tended to have the largest variability, and, as the σ used for generating the Monte Carlo samples increased, the target σ calculated using the EER method were discretized. Since the sample sets contained fewer same-source samples than different-source samples, it is sparseness in same-source samples, rather than in different-source samples, that is primarily responsible for the discretization (step changes in miss rate are visible in Figure 4 in §2.2 below).

Because C_{llr} calculates a penalty value for each and every log(Λ) value and then averages over the penalty values, methods dependent on calculation of C_{llr} were less susceptible to sampling variability than were methods based on calculation of $E_{=}$; however, because the C_{llr}-to-σ mapping function is based on a curve-fitting method, rather than an analytical solution, it is susceptible to errors due to the fitted curve being an approximation of the relationship between C_{llr} and σ.

In addition:

1. The output of the LogReg method becomes unstable (and sensitive to sampling variability) as performance improves and the degree of separation between same-source and different-source uncalibrated log(Λ) values approaches and exceeds complete separation. As can be seen in Figure 3, when the σ used for generating the Monte Carlo samples was 3 and when it was 4, the LogReg method produced a few low-value outliers for the target σ.

2. The output of the KDE method can vary based on the choice of bandwidth for each model's kernels.

3. The PAV algorithm overfits its training data; hence, its results do not generalize. As can be seen in Figure 3, of the four methods, the output of the PAV method exhibited the greatest positive bias in target σ values. This can be attributed to the PAV algorithm overfitting the sample data and, for each sample set, minimizing the C_{llr} value for that particular sample set.

As can be seen in Figure 3 (with the exception of the EER method when the σ used for generating the Monte Carlo samples was 1 or 2), all methods tended to overestimate the target σ. Overestimating the target σ is a problem because it will result in calibrated ln(Λ) that overstate the strength of evidence, i.e., the calibrated ln(Λ) values will be further from the neutral value of 0 than they should be. Vergeer et al. (2016), Morrison & Poh (2018), and Alberink et al. (2026) all expressed concern about overstating strength of evidence, particularly when, as is common in forensic applications, the amount of case-relevant training data available is small.

In the present paper, we explore alternative methods for calculating the target σ for bi-Gaussianized calibration. These methods are potentially less susceptible to the sampling variability, discretization, curve-fitting approximation, and positive bias problems described above. We also explore methods that are designed to reduce the potential for overstating strength of evidence. In particular:

1. We calculate $E_{=}$ using a method that makes use of cumulative KDEs. This is expected to be less susceptible to noise due to sampling variability.

2. We calculate C_{llr} after applying regularized logistic regression with a relatively large regularization weight. This regularization is expected to result in greater stability. It is also expected to result in logistic-regression-calibrated log(Λ) values that are closer to the neutral value of 0, which will result in larger C_{llr} values, which will result in smaller target σ values, which will result in shrinkage

of the bi-Gaussian-calibrated ln(Λ) values toward the neutral value of 0.

3. We calculate C_{llr} after applying KDE models for which the method we use for setting the bandwidths is expected to result in better fits for non-Gaussian distributions.

4. We calculate σ from C_{llr} using an "exact" method that does not depend on a curve-fitting approximation.

5. We recalculate $E_{=}$ as the expected value of the posterior distribution of a beta-binomial model with a uniform prior. This will result in larger $E_{=}$ values, which will result in smaller target σ values, which will result in shrinkage of the bi-Gaussian-calibrated ln(Λ) values toward the neutral value of 0. This method can be applied in combination with any of the other methods.

6. We apply a method that compresses the empirical-cumulative-density function that is used to calculate the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function. This imposes limits on the minimum and maximum values for the calibrated ln(Λ) and, between those limits, compresses the calibrated ln(Λ) toward the neutral value of 0. This compression is in addition to any shrinkage that may have been induced by regularized logistic regression and/or by recalculation of $E_{=}$ using a beta-binomial model. The compression method can be applied in addition to any of the other methods.

Below:

- We describe each of the new methods in detail.

- We use simulated data to assess the results of calculating target σ values using each of the first three methods (for the second and third method, we use the fourth method's C_{llr}-to-σ mapping function), both with and without the addition of each of the fifth (recalculation of $E_{=}$).

- We select the new method that we believe has the best properties (the ISJ-KDE-β method, i.e., the third method combined with the fourth and fifth method) and apply it (with and without compression, the sixth method) to two sets of real uncalibrated log(Λ) data.

2. Methodology

2.1. Code and data availability

The MATLAB code and the data used to perform the experiments described below are available at https://forensic-data-science.net/calibration-and-validation/#biGauss2026. Python and R versions of the functions for performing bi-Gaussianized calibration are also provided at that URL.

2.2. CKDE method

Rather than finding $E_{=}$ by sweeping a threshold through the empirical false-alarm and miss rates for the uncalibrated log(Λ) training data, as in the EER method used in Morrison (2024), the cumulative-kernel-density-estimate (CKDE) method sweeps a threshold through CKDEs fitted to the same-source and the different-source uncalibrated log(Λ) training data.

The method is as follows:

1. Fit a KDE model to the same-source uncalibrated log(Λ) values. The CKDE of this model is used to calculate $\pi_{\mathrm{s}\leq}(x)$: the probability of obtaining values *less than or equal to* the query value (x), see Figure 4.

2. Fit a KDE model to the different-source uncalibrated log(Λ) values. The CKDE of this model is used to calculate $\pi_{\mathrm{d}\geq}(x)$: the probability of obtaining values *greater than or equal to* the query value (x), see Figure 4.

3. Evaluate $\pi_{\mathrm{s}\leq}(x_i)$ and $\pi_{\mathrm{d}\geq}(x_i)$ at each of 10,000 query values ($i \in \{1 \ldots 10{,}000\}$) that are equally spaced between the smallest and the largest

253 uncalibrated log(Λ) values in the training data.

254 4. At each of the query values, calculate $\delta(x_i) = \pi_{\mathrm{s}\leq}(x_i) - \pi_{\mathrm{d}\geq}(x_i)$.

255 5. Find the first $\delta(x_i) > 0$.

256 6. Calculate $E_{=} = \frac{1}{4}\left(\pi_{\mathrm{d}\geq}(x_{i-1}) + \pi_{\mathrm{d}\geq}(x_i) + \pi_{\mathrm{s}\leq}(x_{i-1}) + \pi_{\mathrm{s}\leq}(x_i)\right)$.

257 For the CKDE models, we calculated the bandwidths using the Improved Sheather–
258 Jones (ISJ) method (Botev et al., 2010). We independently calculated the bandwidth
259 for the same-source model and the bandwidth for the different-source model.

260 Figure 4 plots an example of empirical false-alarm and miss rates for different-source
261 and same-source uncalibrated log(Λ) values and of corresponding CKDE $\pi_{\mathrm{d}\geq}(x_i)$
262 and $\pi_{\mathrm{s}\leq}(x_i)$. The uncalibrated log(Λ) values used in this example were taken from real
263 forensic-voice-comparison data (see §3.2 below). The $\pi_{\mathrm{s}\leq}(x_i)$ CKDE curve is
264 smoother than the empirical miss-rate curve. The jaggedness of the empirical miss-rate
265 curve is due to there only being 111 same-source uncalibrated log(Λ) values in the data.

266

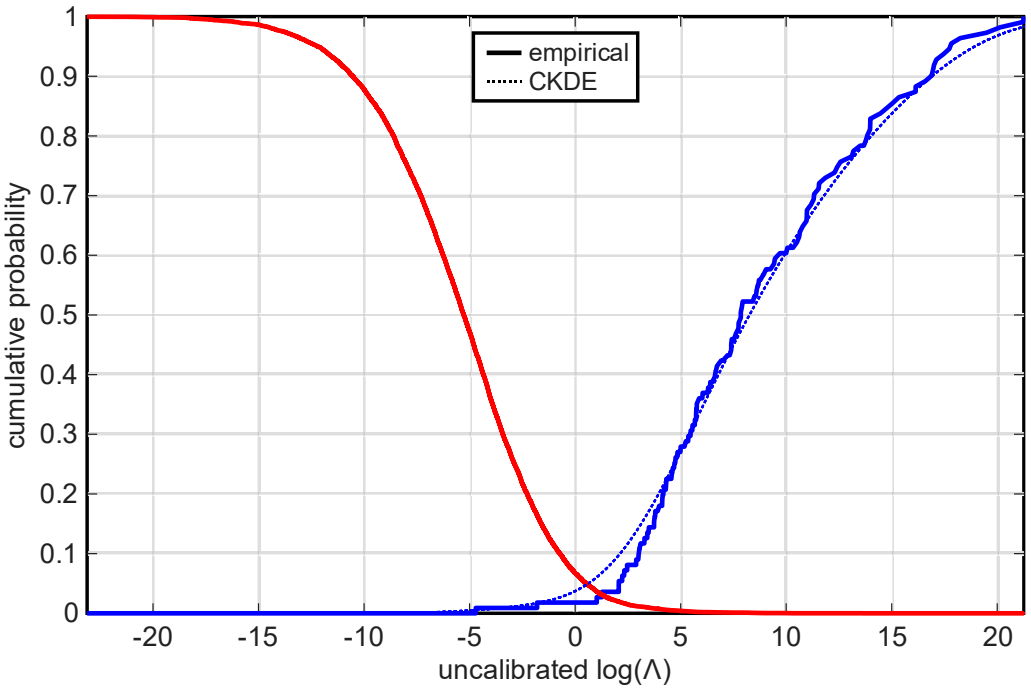


Figure 4. Example plots of empirical false-alarm and miss rates for uncalibrated different-source and same-source log(Λ) values and of cumulative kernel-density estimates $\pi_{\mathrm{d}\geq}(x_i)$ fitted to the different-source uncalibrated log(Λ) values and $\pi_{\mathrm{s}\leq}(x_i)$ fitted to the same-source uncalibrated log(Λ) values.

2.3. RegLogReg method

The RegLogReg method for bi-Gaussianized calibration applies regularized logistic regression prior to calculating C_{llr}.

It is usual to apply a small amount of regularization to address potential numerical problems when fitting a logistic regression model. Applying a large amount of regularization leads to a substantial reduction in the value of the logistic regression's slope coefficient. This results in the calibrated ln(Λ) values being shrunk toward the neutral value of 0. The effect of regularization is larger when the amount of training data is smaller, and it is larger when the degree of separation between the same-source and different-source uncalibrated log(Λ) values is greater.

We applied the regularization method described in Morrison & Poh (2018) and used a

283 regularization weight of $\kappa = 1$. This is expected to induce a moderate amount of
284 shrinkage.[4]

285 2.4. ISJ-KDE method

286 The ISJ-KDE method is the same as the KDE method used in Morrison (2024) (see §1
287 above) except that, instead of calculating the bandwidths using Silverman's Gaussian-
288 approximation method (Silverman, 1986), it calculates them using the ISJ method
289 (Botev et al., 2010). We independently calculated the bandwidth for the same-source
290 model and the bandwidth for the different-source model. Figure 6 and Figure 7 in
291 §2.9.2 and §2.9.3 below show the results of fitting independent ISJ-KDE models to
292 real same-source and different-source uncalibrated log(Λ). The ISJ-KDE method for
293 bi-Gaussianized calibration applies ISJ-KDE calibration prior to calculating C_{llr}.

294 Botev et al. (2010) demonstrated that, for synthetic data generated using multiple
295 different non-Gaussian distributions, use of the ISJ method for calculating the
296 bandwidth usually led to better KDE fits than use of a bandwidth-calculation method
297 that (like Silverman's Gaussian-approximation method) included a Gaussian-reference
298 rule.

299 2.5. Exact C_{llr}-to-σ mapping function

300 As stated in §1 above, as far as we are aware, there is not an analytical solution for the
301 C_{llr}-to-σ mapping function; however, below, we describe what we call an "exact"
302 method for calculating a target σ value given a C_{llr} value. The method is exact in
303 that it does not depend on a fitted-curve approximation.

304 Equation (5), in which $f(x|\mu,\sigma)$ is a univariate Gaussian probability-density function,
305 is the formula for calculating C_{llr} given the σ value of a perfectly calibrated bi-

[4] Morrison (2024) also applied the regularization method described in Morrison & Poh (2018) but used only a small regularization weight ($\kappa = 0.01$) to avoid potential numerical problems.

Gaussian system (see Morrison, 2024, §2.3 for the derivation of this formula).

(5) $C_{\text{llr}} = \frac{1}{\ln(2)} \int_{-\infty}^{\infty} f\left(x \middle| \frac{\sigma^2}{2}, \sigma\right) \ln(1 + e^{-x})\, dx$

As far as we are aware, there is not an analytical solution to the integral in Equation (5), but it can be solved numerically. To perform numerical integration, we use the adaptive quadrature method implemented in the "integral" function in MATLAB (Shampine, 2008). This allows us to calculate C_{llr} given the σ value of a perfectly calibrated bi-Gaussian system, but what we want is the opposite: the target σ value of a perfectly calibrated bi-Gaussian system given a query C_{llr} value.

To calculate the target σ value of a perfectly calibrated bi-Gaussian system given a query C_{llr} value, we use an iterative search algorithm. We use the "fminbnd" function in MATLAB, which implements an algorithm that is based on golden section search and parabolic interpolation (Brent, 1973; Forsythe et al., 1976). We limit the search to the range $0 \leq \sigma \leq 12$. σ values cannot be negative, and $\sigma = 12$ is far beyond any real-world value that we expect to encounter in casework. The function we attempt to minimize is $\left(\ln(C_{\text{llr candidate}}) - \ln(C_{\text{llr query}})\right)^2$, in which $C_{\text{llr query}}$ is the C_{llr} value calculated after application of RegLogReg calibration or ISJ-KDE calibration, and $C_{\text{llr candidate}}$ is a candidate C_{llr} value calculated from a candidate σ value ($\sigma_{\text{candidate}}$) using Equation (5) and numerical integration. Each iteration of the search algorithm tests a different $\sigma_{\text{candidate}}$ value. The algorithm searches for smaller and smaller values of the minimization function and stops when the value of the minimization function falls below a prespecified tolerance value. We set the tolerance value to 10^{-6}. When the algorithm stops, it returns the $\sigma_{\text{candidate}}$ value as the target σ value.

In all of the experiments described below, we used the exact C_{llr}-to-σ mapping function.

2.6. Recalculation of $E_=$ using a Bayesian beta-binomial model with a uniform prior

A false-alarm rate is a proportion. A miss rate is a proportion. The equal-error rate, the value at which the false-alarm rate and miss rate are equal is, therefore, also a proportion.

Rather than using the maximum-likelihood estimate of the proportion of items of interest in a population, i.e., dividing the count of items of interest in a sample by the total number of items in the sample, $\theta = \frac{c}{n}$, one can calculate the Bayesian estimate of the proportion of items of interest in a population as the expected value of the posterior distribution for the proportion, θ^*. The likelihood for the sample count is given by a binomial distribution, $\mathrm{Bin}(c|\theta, n)$; the conjugate prior is a beta distribution, $\mathrm{Beta}(\theta|a, b)$, in which a and b are the prior hyperparameters; and the posterior distribution is a beta distribution, $\mathrm{Beta}(\theta|c + a, n - c + b)$. Integrating out θ leads to the expected value for the posterior distribution, see Equation (6).

(6) $$\theta^* = \int_0^1 \theta \mathrm{Beta}(\theta|c + a, n - c + b)\, d\theta = \frac{c+a}{c+a+n-c+b} = \frac{c+a}{n+a+b}$$

We calculate $E_=$ using the CKDE method, or we calculate C_{llr} using the RegLogReg method or the ISJ-KDE method. If we have calculated C_{llr}, we use the exact C_{llr}-to-σ mapping function to calculate σ (see §2.5 above) then use the analytical σ-to-$E_=$ mapping function to calculate $E_=$ (the analytical σ-to-$E_=$ mapping function is given in Equation (7), in which F is the cumulative probability function for a univariate standard Gaussian distribution, $\mu = 0,\ \sigma = 1$).[5] We then calculate $c = n_{\mathbb{S}} E_=$, in which $n_{\mathbb{S}}$ is the number of sources used to generate the same-source and different-source uncalibrated log(Λ) in the training set. In doing this, we treat $n_{\mathbb{S}} E_=$ as the count

[5] Equation (7) is the inverse of Equation (3). See Morrison (2024) §2.5 for the derivation of this function. Morrison (2024) Equation 9 mistakenly used the symbol f (probability density function) where F (cumulative density function) was intended.

of items of interest in the sample (i.e., the count of errors in the sample) and $n_{\mathbb{S}}$ as the sample size. As prior hyperparameters, we use $a = b = 1$, i.e., a uniform prior. We then calculate an updated value for $E_{=}$ using Equation (8).[6]

(7) $E_{=} = F\left(-\frac{\sigma}{2}\right)$

(8) $E_{=}^{*} = \frac{c+a}{n+a+b} = \frac{n_{\mathbb{S}}E_{=}+1}{n_{\mathbb{S}}+2}$

We then use the analytical $E_{=}$-to-σ mapping function (Equation (3)) to calculate the target σ^{*} corresponding to $E_{=}^{*}$.

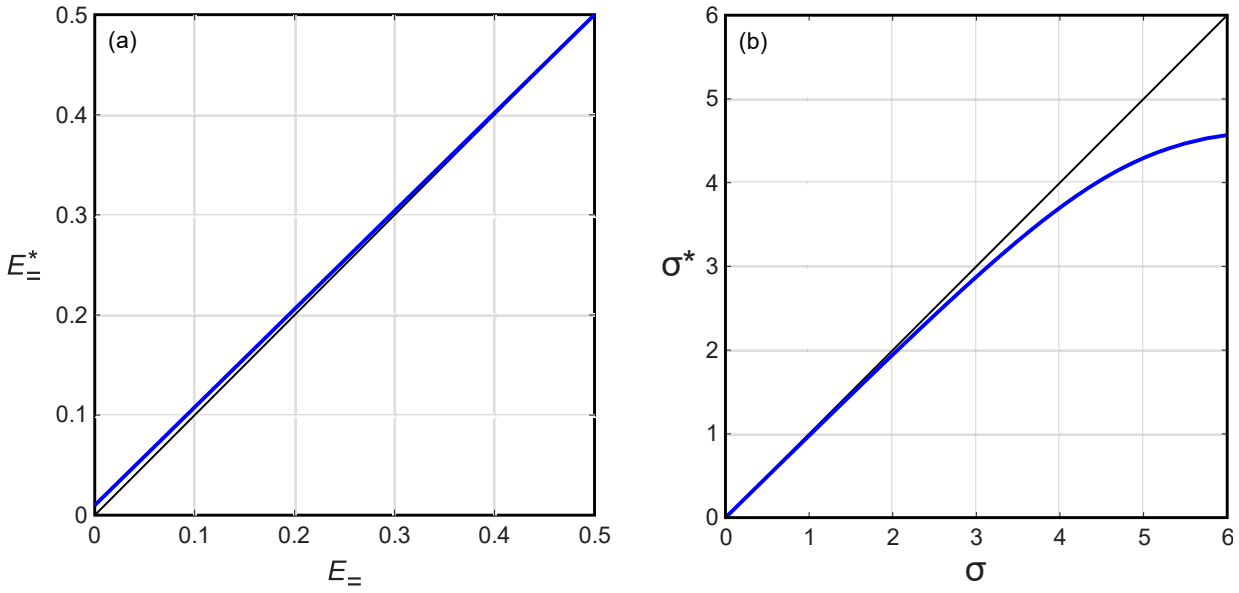
Because of the uniform prior ($a = b$), $E_{=}^{*}$ is larger than $E_{=}$. It is closer to 0.5. The smaller the value of $n_{\mathbb{S}}$ the greater the effect of the prior will be, and the smaller the value of $E_{=}$ the greater the effect of the prior will be. Because $E_{=}^{*}$ is larger than $E_{=}$, the target σ^{*} is smaller than the target σ, and the calibrated ln(Λ) values are shrunk toward the neutral value of 0. If one wanted to induce more or less shrinkage, one could use larger or smaller values for a and b (with the constraint that $a = b$). Table 2 shows, for selected values of σ, the corresponding $E_{=}$, $E_{=}^{*}$, and σ^{*} values given $n_{\mathbb{S}} = 100$ and $a = b = 1$. Figure 5 shows the $E_{=}$-to-$E_{=}^{*}$ and σ-to-σ^{*} mapping functions given $n_{\mathbb{S}} = 100$ and $a = b = 1$. The σ-to-σ^{*} mapping function asymptotes: given $n_{\mathbb{S}} = 100$ and $a = b = 1$, as $E_{=} \to 0$, $E_{=}^{*} \to \frac{1}{102}$ and $\sigma^{*} \to 4.67$.

[6] For $n_{\mathbb{S}}$ in Equation (8), one could use values other than the number of sources, e.g., one could use the harmonic mean, $\frac{2}{\frac{1}{n_{\mathrm{s}}}+\frac{1}{n_{\mathrm{d}}}}$, in which n_{s} is the number of same-source uncalibrated log(Λ) in the training set and n_{d} is the number of different-source uncalibrated log(Λ) in the training set.

371 **Table 2.** $E_{=}$, $E_{=}^{*}$, and σ^{*} values corresponding to $\sigma \in \{0,1,2,3,4,5,6,\infty\}$ given $n_{\mathbb{S}} = 100$ and
372 $a = b = 1$.

$\boldsymbol{\sigma}$	$\boldsymbol{E_{=}}$ (%)	$\boldsymbol{E_{=}^{*}}$ (%)	$\boldsymbol{\sigma^{*}}$
0	50	50	0
1	30.9	31.2	0.98
2	15.9	16.5	1.95
3	6.7	7.5	2.87
4	2.3	3.2	3.70
5	0.6	1.6	4.29
6	0.1	1.1	4.57
∞	0	1.0	4.67

373



374 **Figure 5.** (a) $E_{=}$-to-$E_{=}^{*}$ and (b) σ-to-σ^{*} mapping functions given $n_{\mathbb{S}} = 100$ and $a = b = 1$.

375

376 When we add beta-binomial recalculation of $E_{=}$ to one of the other methods, we will
377 indicate this by adding "-β" at the end of the abbreviation for that method, i.e., CKDE-

β, RegLogReg-β, and ISJ-KDE-β.

2.7. Compression of likelihood-ratio values

We developed a compression method integrated as part of bi-Gaussianized calibration. The compression method limits the minimum and maximum calibrated ln(Λ) values and, between the minimum and maximum values, compresses the calibrated ln(Λ) values toward the neutral value of 0. This is compression in addition to any shrinkage due to regularized logistic regression (§2.3 above) or due to the recalculation of $E_=$ using a beta-binomial model (§2.6 above). We use the term "compression" to distinguish this method from shrinkage due to recalculation of $E_=$ using a beta-binomial model and to distinguish it from the bounding methods described in Vergeer et al. (2016) and Alberink et al. (2026).

The compression method is based on the number of sources ($n_{\mathbb{S}}$) used to generate the uncalibrated log(Λ) values of the training data. Bi-Gaussianized calibration maps the empirical cumulative distribution (ECD) of uncalibrated log(Λ) toward the cumulative Gaussian-mixture distribution of a perfectly calibrated bi-Gaussian system (see Morrison, 2024, §2.7 for details of the algorithm). The ECD of the uncalibrated log(Λ) is constructed with equal weight given to the same-source and different-source categories. Our compression method adjusts the ECD so that its minimum value is $\frac{\gamma}{n_{\mathbb{S}}}$ and its maximum value is $1-\frac{\gamma}{n_{\mathbb{S}}}$, in which γ is a compression factor. The algorithm for the method is as follows:

1. Sort all the uncalibrated log(Λ) values, both different source and same source, from smallest to largest. If any different-source uncalibrated log(Λ) and same-source uncalibrated log(Λ) have the same value, sort the different-source log(Λ) before the same-source log(Λ).

2. Set the weight for the first uncalibrated log(Λ) value (a different-source value) to $\frac{\gamma}{n_{\mathbb{S}}}$.

3. Set the weight for every other different-source uncalibrated log(Λ) value to $\frac{\frac{1}{2}-\frac{\gamma}{n_{\mathbb{S}}}}{n_{\mathrm{d}}-1}$, in which n_{d} is the number of different-source uncalibrated log(Λ) values.

4. Set the weight for every same-source uncalibrated log(Λ) value to $\frac{\frac{1}{2}-\frac{\gamma}{n_{\mathbb{S}}}}{n_{\mathrm{s}}}$, in which n_{s} is the number of same-source uncalibrated log(Λ) values.

5. Calculate the ECD as the cumulative sum of the weights. The first ECD value is $\frac{\gamma}{n_{\mathbb{S}}}$, and the last ECD value is $\frac{\gamma}{n_{\mathbb{S}}} + (n_{\mathrm{d}} - 1)\frac{\frac{1}{2}-\frac{\gamma}{n_{\mathbb{S}}}}{n_{\mathrm{d}}-1} + n_{\mathrm{s}}\frac{\frac{1}{2}-\frac{\gamma}{n_{\mathbb{S}}}}{n_{\mathrm{s}}} = 1 - \frac{\gamma}{n_{\mathbb{S}}}$.

For selected values of σ, with $\gamma \in \{½, 1, 2\}$ and $n_{\mathbb{S}} \in \{50, 100, 200\}$, Table 3 shows the theoretical maximum values for bi-Gaussianized-calibrated likelihood ratios. The theoretical minimum values are the inverse of the maximum values. Using a larger value for γ shifts the minimum and maximum ECD values further away from 0 and 1 and, hence, results in greater compression of the minimum and maximum bi-Gaussian-calibrated ln(Λ) values toward to the neutral value of 0.

Table 3. Maximum likelihood-ratio values for $\sigma \in \{1,2,3,4\}$, with $\gamma \in \{½, 1, 2\}$ and $n_{\mathbb{S}} \in \{50,100,200\}$.

		$n_{\mathbb{S}}$		
σ	γ	**50**	**100**	**200**
4	½	11,017,502	32,781,045	88,926,187
	1	3,278,001	11,017,502	32,782,045
	2	822,651	3,278,001	11,017,502
3	½	42,670	96,669	204,330
	1	17,190	42,670	96,669
	2	6,096	17,190	42,670
2	½	450	775	1,277
	1	246	450	775
	2	123	246	450
1	½	13	17	22
	1	10	13	17
	2	7	10	13

In the experiments reported below, we use $\gamma = 1$.[7]

That the compression method results in minimum and maximum ECD values that are shifted away from 0 and 1 can be observed in Figure 9 in §3.2.2 below. Figure 10 in §3.2.2 below exemplifies the effect of the compression method on the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function.

[7] For results reported below as "without compression", the minimum and maximum ECD values were set to $\frac{1}{n_t}$ and $1 - \frac{1}{n_t}$, respectively, in which $n_t = n_d + n_s$, i.e., the total number of uncalibrated log(Λ) values (both same-source and different-source) used for training.

2.8. Simulated data

As in Morrison (2024), we used Monte Carlo simulation to generate synthetic data from perfectly calibrated bi-Gaussian systems. For each value of σ, we generated 1000 sample sets. For each sample set, we generated samples based on having 100 sources and comparing two items from each source with each other and one item from each source with one item from every other source, i.e., we generated 100 same-source ln(Λ) sample values and 4950 different-source ln(Λ) sample values.

For each sample set, we calculated the target σ value using each of CKDE, CKDE-β, RegLogReg, RegLogReg-β, ISJ-KDE, and ISJ-KDE-β.

In contrast with Morrison (2024), which generated samples for perfectly calibrated bi-Gaussian systems with $\sigma \in \{1,2,3,4\}$, for the present paper we generated samples for perfectly calibrated bi-Gaussian systems with $\sigma \in \{1,2,3,4,5,6\}$. We extended to $\sigma = 6$ because one of the real-data examples in Morrison (2024), the glass-fragment example, which we also use in the present paper (see §2.9.3 below), had (using the LogReg method) a calculated target σ just above 6.

2.9. Real data

2.9.1. Introduction

We selected the new method that we believe has the best properties (the ISJ-KDE-β method) and used it (with and without compression) to calibrate uncalibrated log(Λ) values from two real datasets. These were the same real datasets as used in Morrison (2024). One dataset consisted of uncalibrated log(Λ) values resulting from comparison of voice recordings and the other consisted of uncalibrated log(Λ) values resulting from comparison of glass fragments. The datasets are briefly described in §2.9.2 and §2.9.3 below.

2.9.2. Voice recordings

The voice dataset consisted of uncalibrated log(Λ) values produced by a forensic-voice-comparison system (Weber et al., 2022). The distributions of the same-source and different-source uncalibrated log(Λ) values are shown in Figure 6. For this dataset, the number of sources was 61 (other data had been used to train earlier parts of the forensic-voice-comparison system). The dataset consisted of 111 same-source uncalibrated log(Λ) and 9,719 different-source uncalibrated log(Λ) (some speakers contributed more that two recordings).

As can be seen in Figure 6, these data are not distributed as two Gaussians with the same variance. They are therefore expected to benefit from non-linear calibration.

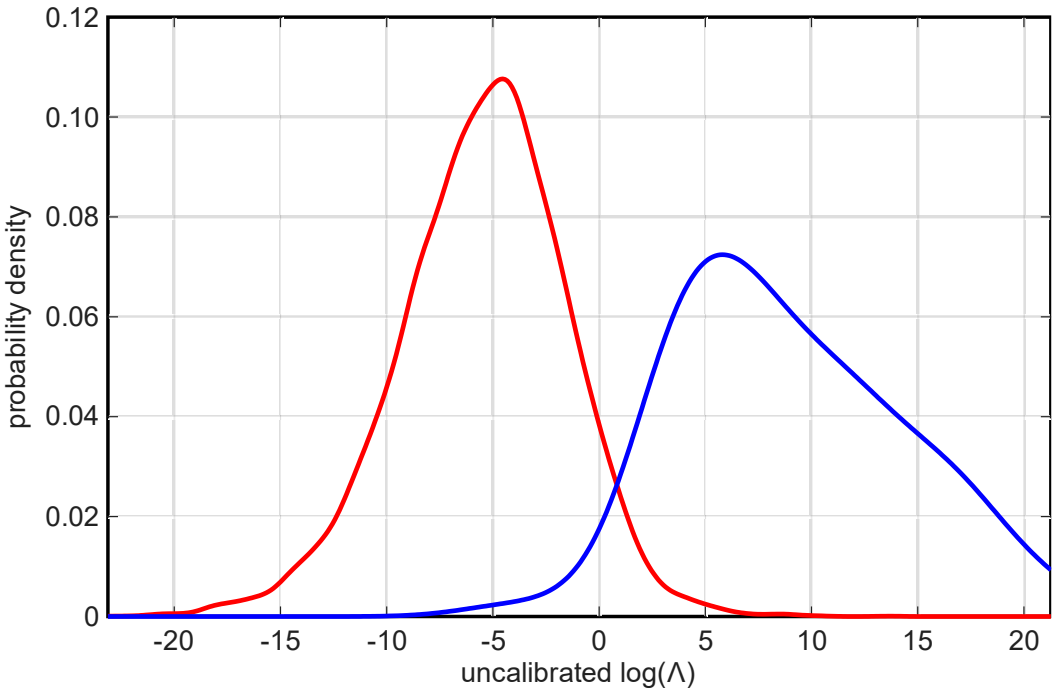


Figure 6. Kernel-density plots of the same-source and different-source uncalibrated log(Λ) values from the voice dataset.

2.9.3. Glass fragments

The glass dataset consisted of uncalibrated log(Λ) values calculated by applying the

multivariate-kernel-density method of Aitken & Lucy (2004a, 2004b) to pairs of groups of glass fragments (Vergeer et al., 2016; van Es et al., 2017). For this dataset, the number of sources was 320. The dataset consisted of 320 same-source uncalibrated log(Λ) and 51,040 different-source uncalibrated log(Λ).

The distributions of the same-source and different-source uncalibrated log(Λ) values are shown in Figure 7. As can be seen in Figure 7, these data have extreme distributions. The different-source distribution has a probability point mass at approximately −300, then has values spread between −300 and 6.5. The point mass is due to numerical issues that resulted in about 80% of the different-source uncalibrated likelihood ratios being below the precision of the software used to implement the multivariate-kernel-density method. We reset these results to the lowest finite uncalibrated likelihood-ratio value in the data. Other than for a few outliers, the same-source density is concentrated in a relatively narrow range of low positive values. Its mode is at 9.0. The different-source and same-source values overlap in the range −1.7 to 6.5. A strength of bi-Gaussianized calibration is that it can handle these extreme distributions.

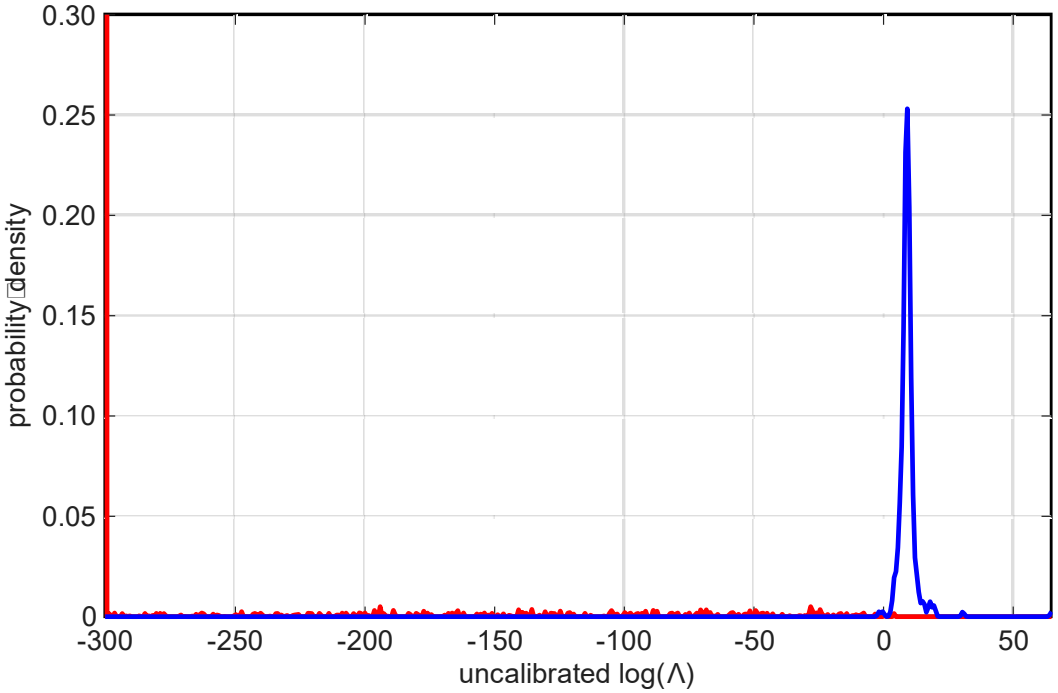
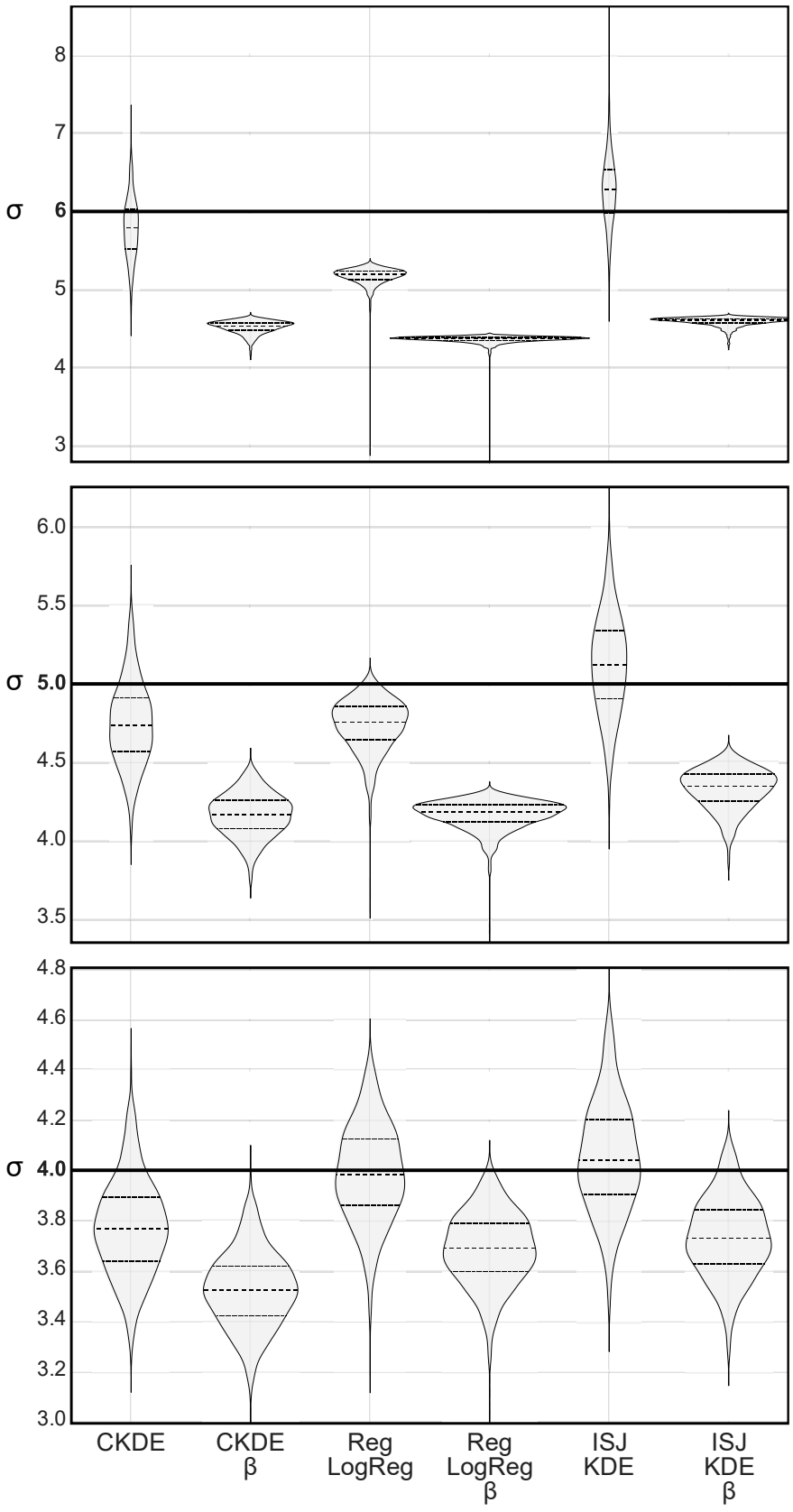


Figure 7. Kernel-density plots of the same-source and different-source uncalibrated log(Λ) values from the glass dataset.

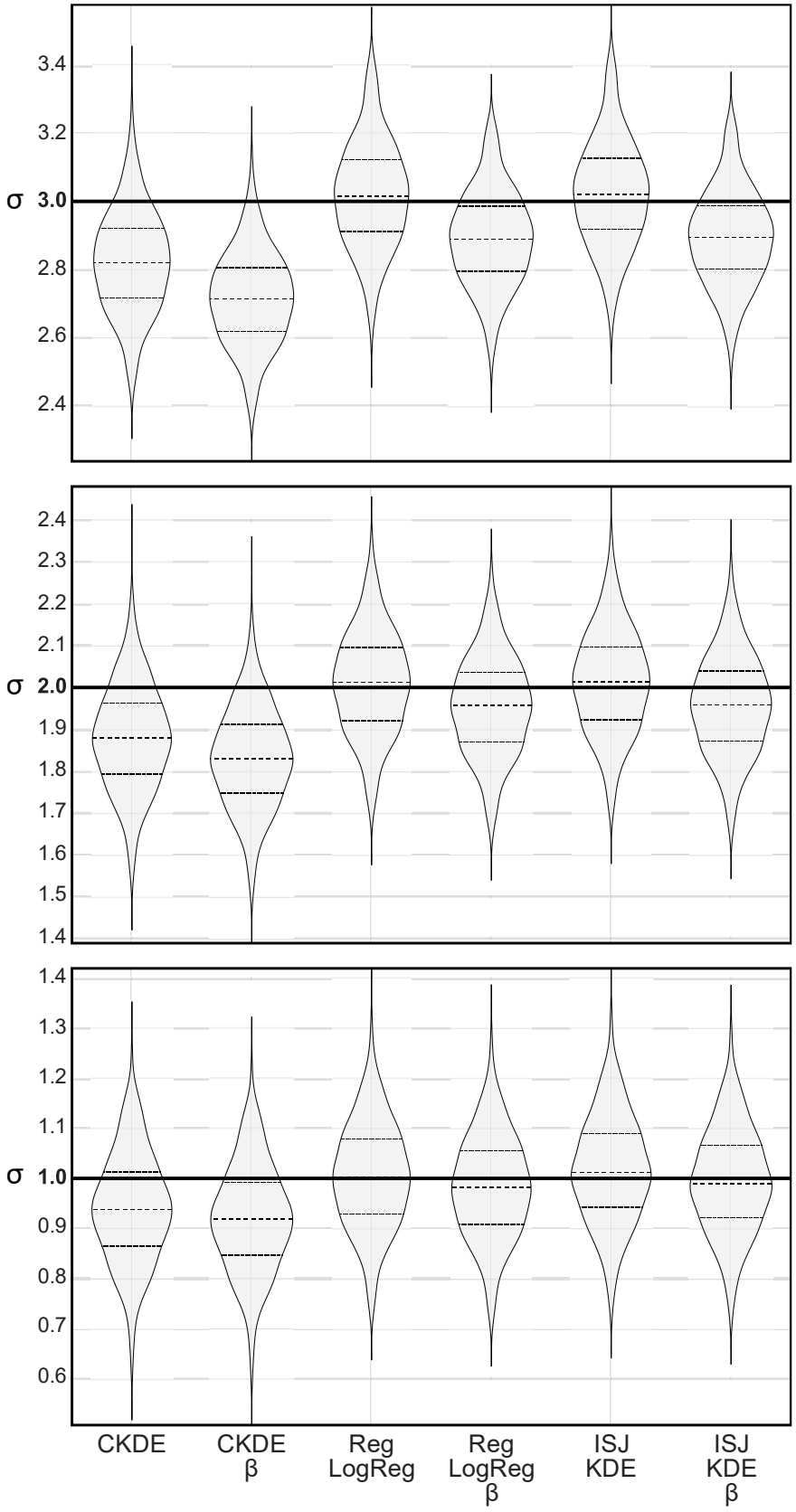
3. Results and discussion

3.1. Simulated data

Figure 8 shows, for each of CKDE, CKDE-β, RegLogReg, RegLogReg-β, ISJ-KDE, and ISJ-KDE-β, the distributions of the calculated target σ values at each of the different σ values used to generate the Monte Carlo samples, $\sigma \in \{1,2,3,4,5,6\}$.



489 **Figure 8.** Continued on next page …



490 **Figure 8.** Violin plots of distributions of target σ values calculated using each of the new methods.
491 In each panel, the solid horizontal line indicates the σ value used to generate the Monte Carlo
492 samples, $\sigma \in \{1,2,3,4,5,6\}$.

493

494 With the exception of the ISJ-KDE method applied to Monte Carlo samples generated
495 using high values for σ ($\sigma \in \{4,5,6\}$), all of the new methods appeared to be
496 successful in eliminating substantial positive bias in target σ values.

497 The CKDE method was effective in eliminating discretization of calculated values for
498 $E_=$ and, hence, discretization of calculated values for σ. Although not designed to
499 substantially reduce target σ values, the CKDE method tended to overestimate $E_=$
500 values and, hence, to underestimate target σ values (see Figure 4 in §2.2 above for an
501 example of CKDE overestimating $E_=$). This bias appeared consistently across all σ
502 values used for generating Monte Carlo samples. Later in this section, in discussing the
503 beta-binomial results, we argue that the desired pattern is one of negligible reduction
504 in target σ values when low σ values are used to generate Monte Carlo samples, with
505 the amount of reduction in the target σ value increasing as the σ value used to
506 generate the Monte Carlo samples increases. This was not the pattern observed when
507 the CKDE method was applied.

508 The relatively high regularization weight used for the RegLogReg method did not
509 result in substantial reduction in target σ values when Monte Carlo samples were
510 generated using low to high σ values ($\sigma \in \{1,2,3,4\}$), but it did substantially reduce
511 target σ values when Monte Carlo samples were generated using very high σ values
512 ($\sigma \in \{5,6\}$). Like the LogReg method, the RegLogReg and RegLogReg-β methods
513 exhibited some low-value outliers for target σ values. Compared to the LogReg
514 method, however, these outliers did not appear until the σ used for generating the
515 Monte Carlo samples was very high ($\sigma \in \{5,6\}$).

516 As indicated above, for Monte Carlo samples generated using high values for σ ($\sigma \in$
517 $\{4,5,6\}$), the ISJ-KDE method had a substantial positive bias in target σ values. This
518 was particularly prominent for Monte Carlo samples generated using very high σ
519 values ($\sigma \in \{5,6\}$). A potential contributor to this pattern of results is that, for ISJ-KDE

calibration itself or for KDE calibration itself (as opposed to bi-Gaussianized calibration), the mapping function between uncalibrated log(Λ) values and calibrated ln(Λ) values is not monotonic (see Figure 15 and Figure 16 in Appendix A). If low σ values are used for generating the Monte Carlo samples, the mapping function could be monotonic over the range of values that are to be calibrated, but, if high σ values are used for generating the Monte Carlo samples, the mapping function for regions that are on the tail of either the same-source distribution or the different-source distributions, and the mapping function for regions that are on the tails of both distributions, will likely be non-monotonic. Since we train and test using the same Monte Carlo samples, there is always a same-source kernel centred at a same-source query value, but, because data in a joint-tail region are otherwise sparce, there is unlikely to be a different-source kernel centred nearby. This will lead to calibrated ln(Λ) values for same-source query values being overestimated. Mutatis mutandis for different-source query values. This will lead to underestimated values for C_{llr} and, in turn, overestimated values for target σ.

Across the different methods, adding recalculation of $E_{=}$ using a Bayesian beta-binomial model with a uniform prior led to a consistent pattern of results:

- For the lowest value of σ used to generate the Monte Carlo samples ($\sigma = 1$), the reduction of target σ^* values was negligible.

- As the value of σ used to generate the Monte Carlo samples increased (through $\sigma \in \{2,3,4,5,6\}$), the reduction in target σ^* values increased and that increase accelerated.

- For very high values of σ used to generate the Monte Carlo samples ($\sigma \in \{5,6\}$), the reduction in target σ^* values was large.

This is the expected pattern and the desired pattern. Understanding why this is the desired pattern is, perhaps, best conceptualized in terms of $E_{=}$ values, see Table 2 in

§2.6 above and the following discussion.

A sample size of 100 is sufficiently large that it can support a claim of an error rate of 31% ($\sigma = 1$) or a claim of an error rate of 16% ($\sigma = 2$), but a sample size of 100 is not sufficiently large that it can support a claim of an error rate of 0.6% ($\sigma = 5$) or a claim of an error rate of 0.1% ($\sigma = 6$). An error rate of 0.6% corresponds to observing, on average, 6 errors in every 1000, and an error rate of 0.1% corresponds to observing, on average, 1 error in every 1000. One cannot support a claim of an error rate of 6 in 1000 or a claim of an error rate of 1 in 1000 using a sample size of 100. One would need a sample size orders of magnitude larger than 100 in order to be able to support claims of such small error rates.

Recalculation of $E_{=}$ using a Bayesian beta-binomial model with a uniform prior leaves $E_{=}$ values that are easily supported by the sample size almost unchanged, but, as σ increases and $E_{=}$ values get smaller and become increasingly difficult to support given the sample size, recalculation using a Bayesian beta-binomial model with a uniform prior increases $E_{=}^{*}$ values and, hence, decreases σ^{*} values by increasing amounts.

Taking all of the above into account, of the methods we tested, we believe that the ISJ-KDE-β method exhibits the best properties. Although the ISJ-KDE method overestimated target σ values when high and very high values of σ were used to generate the sample data ($\sigma \in \{4,5,6\}$), that positive bias was eliminated when $E_{=}$ was recalculated using a Bayesian beta-binomial model with a uniform prior. We select the ISJ-KDE-β method over CKDE and CKDE-β methods because the latter methods reduce the target σ values even when the corresponding $E_{=}$ values would be easily supported by the amount of training data. We select the ISJ-KDE-β method over RegLogReg and RegLogReg-β methods because, when Monte Carlo samples were generated using very high σ values ($\sigma \in \{5,6\}$), the latter methods exhibited low-value outliers for target σ values.

3.2. Real data

3.2.1. Introduction

Based on the results of the experiments using simulated data, we selected the ISJ-KDE-β method as our preferred method. In §3.2.2 and §3.2.3 below, we present the results of applying this method (with and without compression) to the real datasets described in §2.9 above. For comparison purposes, we also present the results of applying logistic-regression calibration (straightforward logistic-regression calibration, not the LogReg method for bi-Gaussianized calibration).[8,9]

As per the recommendation in Morrison et al. (2021), we calibrated the uncalibrated $\log(\Lambda)$ values from the real datasets using leave-one-out/leave-two-out cross-validation. For each cross-validation loop, the values of $n_{\mathbb{S}}$, n_{s}, and n_{d} were adjusted accordingly.

In §3.2.4 below, we discuss whether the likelihood-ratio values obtained are justified given the sample size.

3.2.2. Voice recordings

For the voice data, applying the ISJ-KDE-β method to all the training data resulted in a target σ value of 3.17.[10] This is somewhat smaller than the target σ of 3.44 that was obtained in Morrison (2024) using the LogReg method for bi-Gaussianized

[8] Regularized logistic regression was used, but shrinkage was not induced. The regularization weight was $\kappa = 0.01$.

[9] When we presented our work at the 2026 International Conference on Forensic Inference and Statistics (ICFIS 2026), an audience member asked how the output of the ISJ-KDE-β bi-Gaussianized-calibration method compared with the output of traditional KDE calibration. In Appendix A, we compare the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function for ISJ-KDE-β bi-Gaussianized-calibration with compression with the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function for traditional KDE calibration.

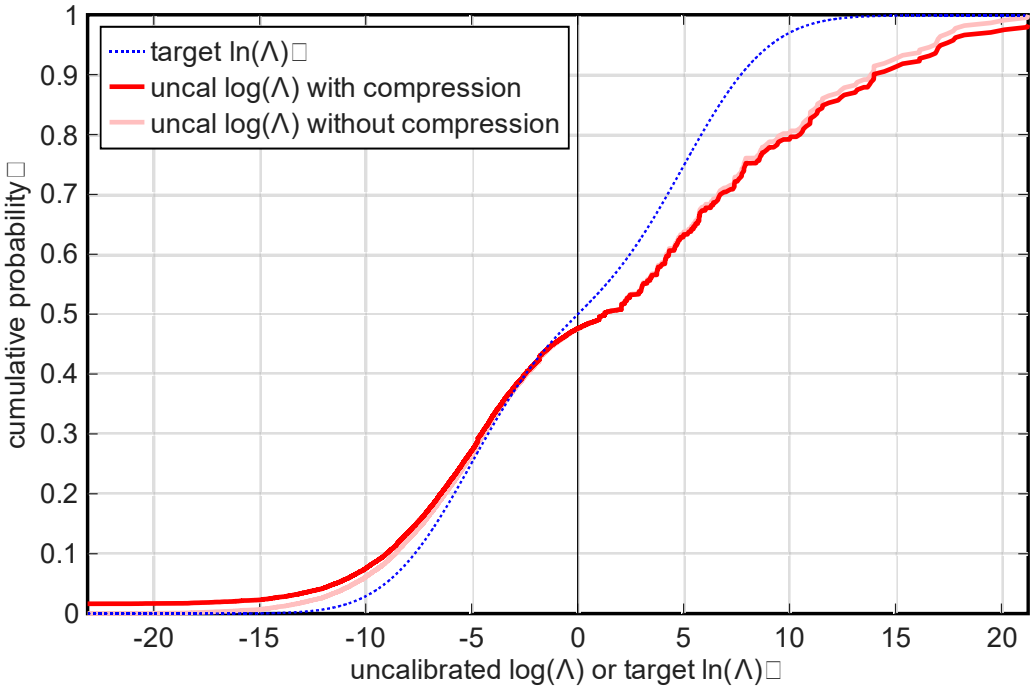
[10] The exact target σ value would have differed for different cross-validation loops, but, for simplicity (for both the voice data and the glass data), we report only the target σ value calculated using all the training data without cross validation.

591 calibration. The latter method was not designed to induce shrinkage.

592 For the voice data, Figure 9 shows the empirical cumulative distribution for the
593 uncalibrated log(Λ) values, both with and without compression, and the cumulative
594 distribution of the Gaussian-mixture model for the target perfectly calibrated bi-
595 Gaussian system with $\sigma = 3.17$. Compression resulted in the minimum and maximum
596 ECD values being at $\frac{\gamma}{n_{\mathbb{S}}} = \frac{1}{61}$ and $1 - \frac{\gamma}{n_{\mathbb{S}}} = 1 - \frac{1}{61}$, respectively. Bi-Gaussianized
597 calibration shifts each point on the uncalibrated log(Λ) curve horizontally to the
598 corresponding point on the curve for the cumulative distribution of the target Gaussian-
599 mixture model.[11] Figure 10 shows the resulting mapping functions (both with and
600 without compression) between uncalibrated log(Λ) values and calibrated ln(Λ) values.
601 The mapping functions are monotonic but non-linear. The compression method
602 compresses the calibrated ln(Λ) values toward the neutral value of 0 in a manner which
603 is sigmoidal: calibrated ln(Λ) values that are further from 0 are compressed by a larger
604 proportion than calibrated ln(Λ) values that are closer to 0.

605

[11] The algorithm for bi-Gaussianized calibration includes interpolating the empirical-cumulative-density curve obtained from the training data in order to obtain empirical-cumulative-density values for the test data. Morrison (2024) used linear interpolation. For the present paper, we use piecewise cubic Hermite interpolating polynomials (PCHIP). PCHIP fits training values exactly and preserves monotonicity, but, compared to linear interpolation, PCHIP results in a smoother mapping function.



606 **Figure 9.** Empirical cumulative distributions with and without compression for uncalibrated log(Λ)
607 values from the voice data, and the cumulative distribution of the Gaussian-mixture model for the
608 target perfectly calibrated bi-Gaussian system with $\sigma = 3.17$.

609

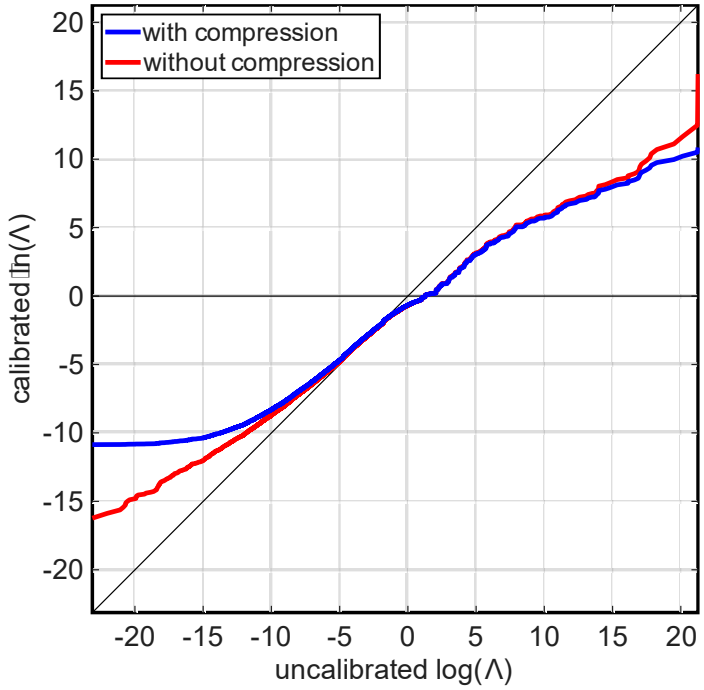


Figure 10. Mapping functions between uncalibrated log(Λ) values from the voice data and calibrated ln(Λ) values corresponding to a perfectly calibrated bi-Gaussian system with $\sigma = 3.17$. Mapping functions both with and without compression are shown.

Figure 11 shows probability-density[12] and Tippett plots for:

- the voice data after cross-validated calibration using the ISJ-KDE-β bi-Gaussianized-calibration method, both with and without compression;
- the target perfectly calibrated bi-Gaussian system with $\sigma = 3.17$; and
- the voice data after cross-validated calibration using logistic regression.

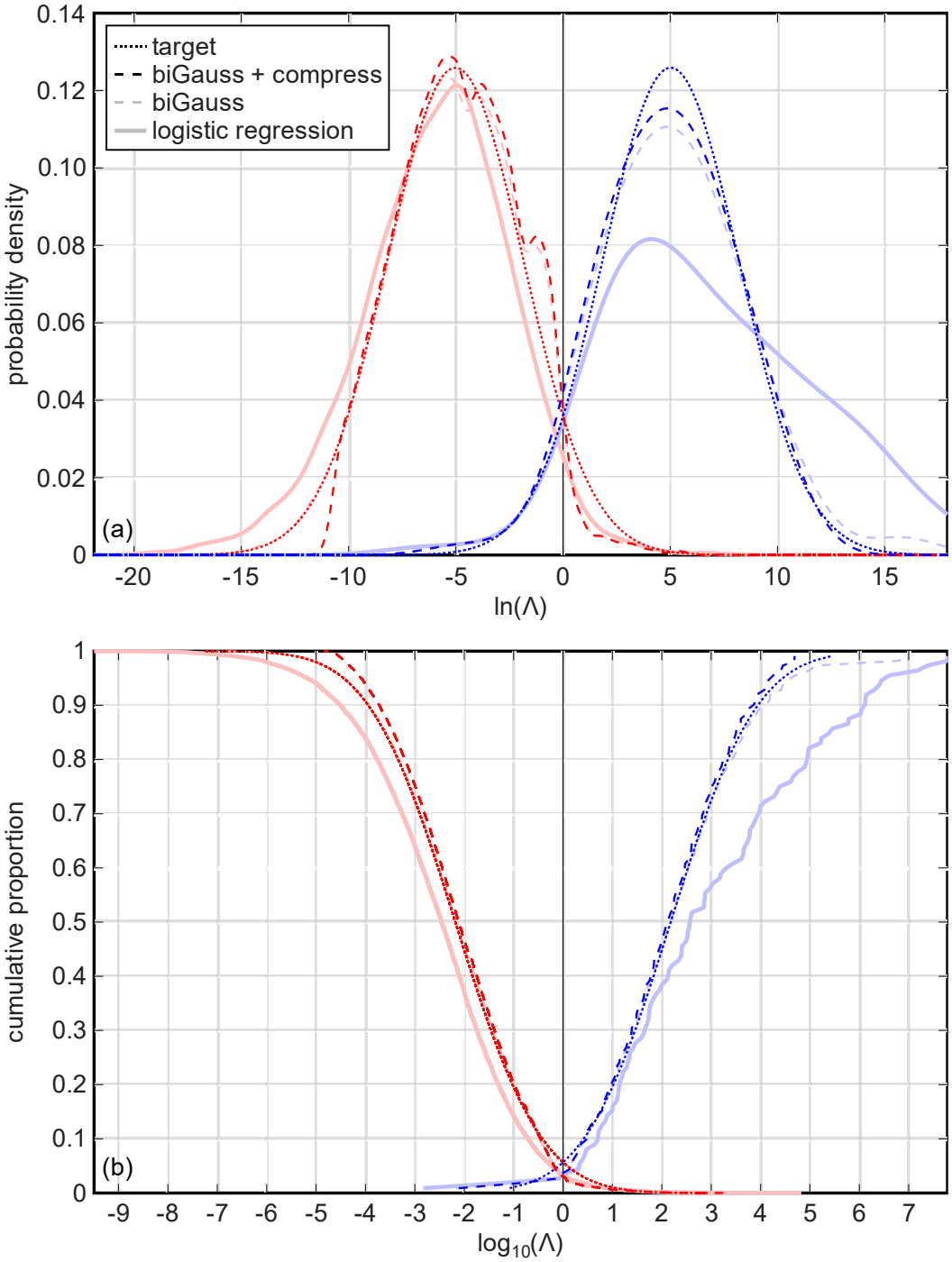
As can be seen in Figure 11, the output of ISJ-KDE-β bi-Gaussianized calibration without compression is closer to the target perfectly-calibrated bi-Gaussian system than is the output of logistic-regression calibration. The calibrated ln(Λ) values (and the calibrated $\log_{10}(\Lambda)$ values) resulting from ISJ-KDE-β bi-Gaussianized calibration are,

[12] In Figure 11a, all the probability-density functions for bi-Gaussianized calibration and logistic-regression calibration were plotted using ISJ-KDEs.

generally, less extreme than those resulting from logistic-regression calibration. Shrinkage because of application of the beta-binomial model with a uniform prior is a contributor to the latter effect, but bi-Gaussianized calibration's non-linear mapping is also a contributor. Applying the compression method resulted in further compression of calibrated ln(Λ) values (and of calibrated $\log_{10}(\Lambda)$ values) toward the neutral value of 0. This resulted in deviation from the target perfectly-calibrated bi-Gaussian system. Compression results in miscalibration of log-likelihood-ratio values, but this is miscalibration in service of reducing the probability of overstating the strength of evidence.

For the ISJ-KDE-β bi-Gaussianized-calibration method, C_{llr} was 0.177 with compression and 0.176 without compression. For logistic-regression calibration, C_{llr} was 0.172. Shrinkage because of application of the beta-binomial model and the application of compression could be causes of the slightly higher C_{llr} value for the ISJ-KDE-β method. C_{llr} is a single-value summary of performance. We argue that C_{llr} does not capture all relevant aspects of system performance, so having the lowest C_{llr} should not be the sole factor considered in deciding which system is best. We argue that there are other properties that should also be considered, including whether the likelihood ratios are well calibrated and whether the method reduces the probability of overstating the strength of evidence.[13]

[13] Morrison (2025) argues that the extent to which a system takes account of typicality is also a factor that should be considered.



643 **Figure 11.** For the voice data, probability-density plots (a) and Tippett plots (b) for the output of bi-
644 Gaussianized calibration (ISJ-KDE-β method both with and without compression), for the target
645 perfectly-calibrated bi-Gaussian system (with $\sigma = 3.17$), and for the output of logistic-regression
646 calibration. The *x*-axes for panels (a) and (b) are the same, but they are labelled using different
647 logarithmic bases.

648

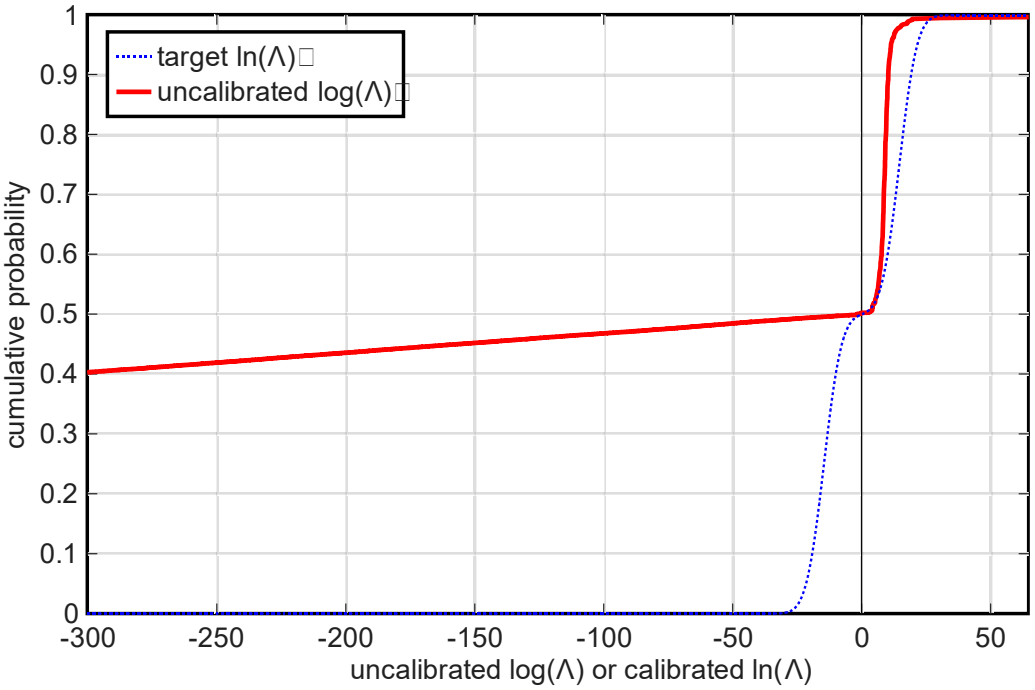
649 ### 3.2.3. Glass fragments

650 For the glass data, applying the ISJ-KDE-β method to all the training data resulted in a
651 target σ of 5.41.[14] This is substantially smaller than the target σ of 6.02 that was
652 obtained in Morrison (2024) using the LogReg method for bi-Gaussianized calibration.
653 The latter method was not designed to induce shrinkage.

654 For the glass data, Figure 12 shows the empirical cumulative distribution with
655 compression for the uncalibrated log(Λ) values, and the cumulative distribution of the
656 Gaussian-mixture model for the target perfectly calibrated bi-Gaussian system with
657 $\sigma = 5.41$. Due to the probability point mass at about −300 representing about 80% of
658 different-source uncalibrated log(Λ), the empirical cumulative probability for the
659 uncalibrated log(Λ) values starts at about 0.4. Figure 13 shows the mapping functions
660 (both with and without compression) between uncalibrated log(Λ) values and
661 calibrated ln(Λ) values. The mapping functions are monotonic but non-linear. Their
662 overall shape is sigmoidal.

663

[14] Given a sample size of 320 and a uniform prior for the beta-binomial model ($a = b = 1$), as $E_= \to 0$, $E_=^* \to \frac{1}{322}$ and $\sigma^* \to 5.47$.



664 **Figure 12.** Empirical cumulative distribution with compression for uncalibrated log(Λ) values from
665 glass data, and the cumulative distribution of the Gaussian-mixture model for the target perfectly
666 calibrated bi-Gaussian system with $\sigma = 5.41$.

667

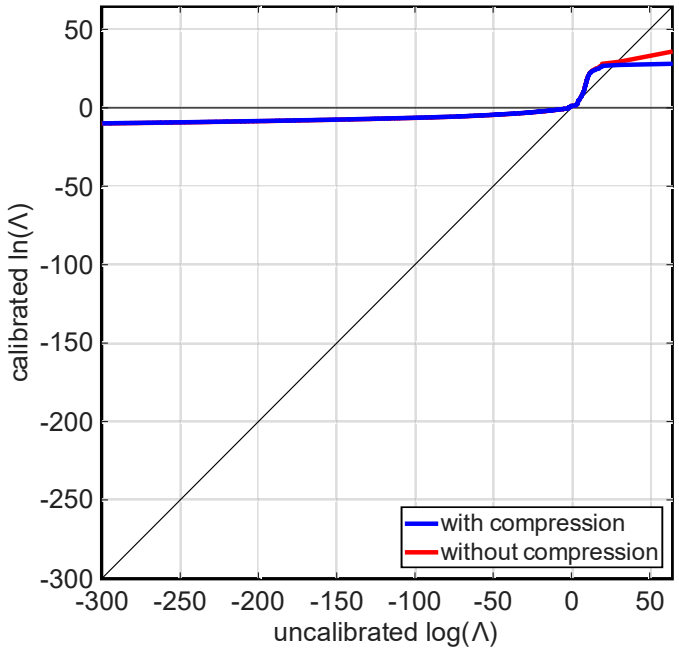


Figure 13. Mapping functions (both with and without compression) between uncalibrated log(Λ) values from glass data and calibrated ln(Λ) values corresponding to a perfectly calibrated bi-Gaussian system with $\sigma = 5.41$.

Figure 14 shows probability-density[15] and Tippett plots for:

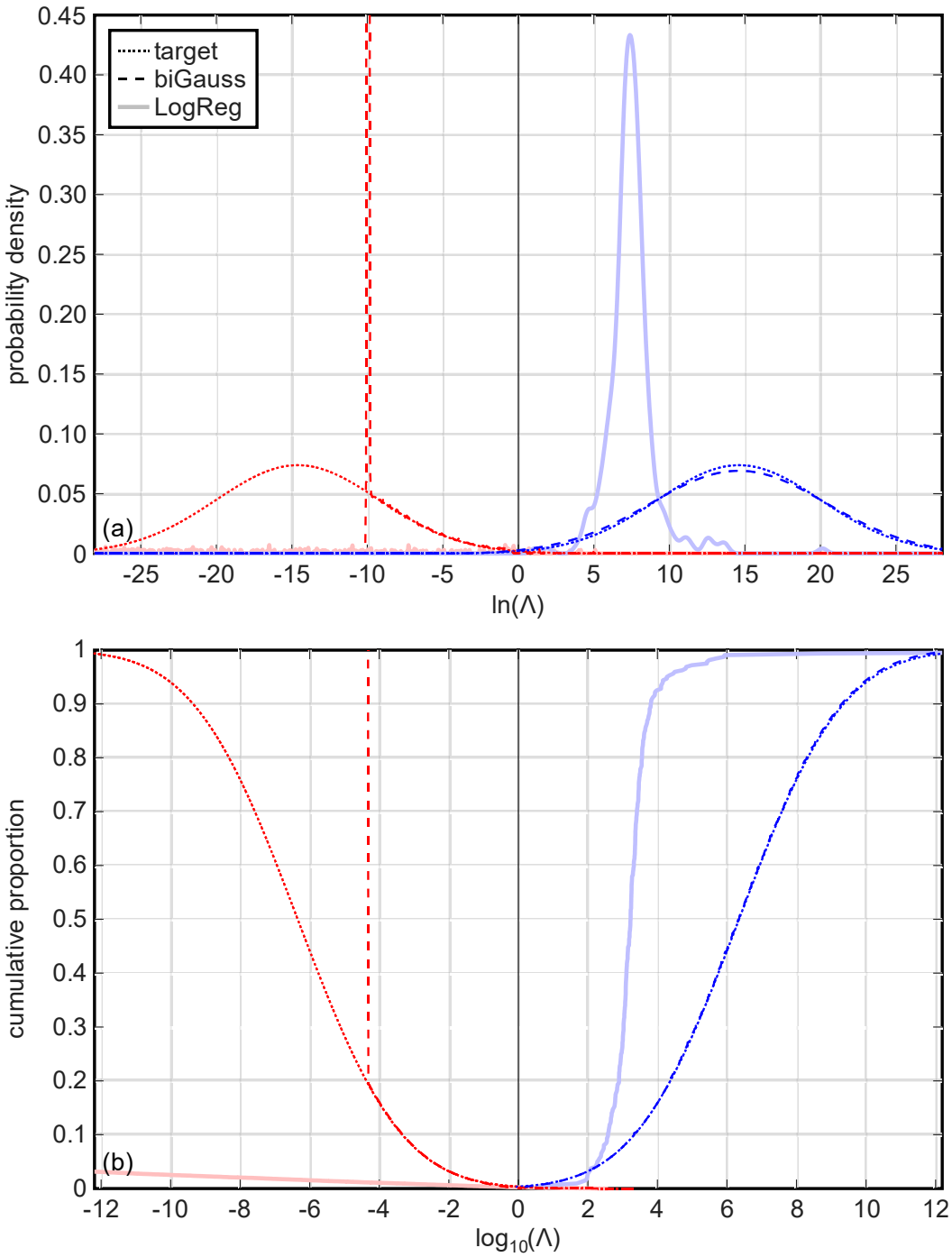
- the glass data after cross-validated calibration using the ISJ-KDE-β bi-Gaussianized-calibration method with compression;
- the target perfectly calibrated bi-Gaussian system with $\sigma = 5.41$; and
- the glass data after cross-validated calibration using logistic regression.

As can be seen in Figure 14, apart from at and beyond the probability point mass, the output of ISJ-KDE-β bi-Gaussianized calibration with compression is close to the target perfectly-calibrated bi-Gaussian system. In contrast, the output of logistic-

[15] In Figure 14a, the different-source probability-density function after bi-Gaussianized calibration was plotted using a KDE for which the bandwidth was manually selected so as to provide a reasonable visual representation of both the probability point mass and the remainer of the probability-density function. All the other probability-density functions in Figure 14a were plotted using ISJ-KDEs.

regression calibration is very far from the target perfectly-calibrated bi-Gaussian system. Using ISJ-KDE-β bi-Gaussianized calibration with compression, the calibrated likelihood ratio at the point mass was 1/21,057. If this were the value returned in the context of casework, one could report that the evidence is at least 21,057 times more likely if the different-source hypothesis were true than if the same-source hypothesis were true (or, depending on context, use a rounded version of that number).

For the ISJ-KDE-β bi-Gaussianized-calibration method, C_{llr} was 0.007 with compression and 0.007 without compression. For logistic-regression calibration, C_{llr} was 0.006.



690 **Figure 14.** For the glass data, probability-density plots (a) and Tippett plots (b) for the output of bi-
691 Gaussianized calibration (ISJ-KDE-β method with compression), for the target perfectly-calibrated
692 bi-Gaussian system (with $\sigma = 5.41$), and for the output of logistic-regression calibration. The *x*-axes
693 for panels (a) and (b) are the same, but they are labelled using different logarithmic bases.

694

695 ### 3.2.4. Are the obtained likelihood-ratio values justified given the sample size?

One may question whether likelihood ratios of the magnitudes output by ISJ-KDE-β bi-Gaussianized calibration (see Figure 11 and Figure 14) can be justified using the small sample sizes provided by the real data (61 and 320 sources, see §2.9.2 and §2.9.3 above). Parametric statistical models are based on assumptions about the distributions of the data. Stronger assumptions about distributions allow one to draw stronger inferences using less data. To be able to draw equally strong inferences using weaker assumptions, one would need to use more data. We argue that what allows us to justify the magnitudes of likelihood ratios calculated using bi-Gaussianized calibration is the assumption that calibrated ln(Λ) should be distributed as perfectly calibrated bi-Gaussian systems.

Given the assumption that calibrated ln(Λ) should be distributed as perfectly calibrated bi-Gaussian systems, the question of whether the calibrated likelihood-ratio values can be justified given the sample size can be reconceptualized as whether the sample size (the number of sources used to generate the training data, $n_{\mathbb{S}}$) is sufficient to justify the $E_{=}^{*}$ value (see §2.6 above and the discussion about desirable properties in §3.1 above). If one is satisfied that the sample size is sufficient to justify the $E_{=}^{*}$ value, and if one accepts the distributional assumption, then the minimum and maximum possible likelihood-ratio values are the most extreme values that, for the training data, occur on the tails of the distributions that are determined by that $E_{=}^{*}$ value.

If one were more skeptical as to whether the $E_{=}^{*}$ value could be supported by the sample size, one could use a stronger prior belief than that represented by a uniform prior, i.e., one could use larger prior hyperparameter values than $a = b = 1$. These would represent a stronger prior belief that $E_{=}$ is 0.5. If $E_{=}$ were 0.5, σ would be 0, and all the log-likelihood-ratio values would be 0 (all the likelihood-ratio values would be 1). If one believed, for example, that the smallest $E_{=}^{*}$ that could be supported by a

sample size of 100 is 5%, one would set $a = b = \frac{5}{0.9} = 5.\dot{5}$, see Equation (9).[16] Similarly, if one believed, for example, that the smallest $E_{=}^{*}$ that could be supported by a sample size of 200 is 2.5%, one would set $a = b = \frac{5}{0.95} \approx 5.26$.[17]

(9) $$E_{=}^{*} = \frac{n_{\mathbb{S}} E_{=} + a}{n_{\mathbb{S}} + a + b}$$

$$a = b,\ E_{=} \to 0$$

$$\Rightarrow E_{=}^{*} = \frac{a}{n_{\mathbb{S}} + 2a}$$

$$\Rightarrow a = \frac{E_{=}^{*} n_{\mathrm{S}}}{1 - 2E_{=}^{*}}$$

$$E_{=}^{*} = 0.05,\ n_{\mathbb{S}} = 100$$

$$\Rightarrow a = \frac{0.05 \times 100}{1 - 2 \times 0.05} = \frac{5}{0.9} = 5.\dot{5}$$

If one still had concerns about the magnitudes of the calibrated ln(Λ), one could further reduce them by applying compression and by increasing the value of the compression factor, γ (see §2.7 above), or one could apply bounding methods such as those proposed in Vergeer et al. (2016) and Alberink et al. (2026), see Appendix B.

4. Conclusion

Morrison (2024) introduced bi-Gaussianized calibration, a calibration method that monotonically maps the distribution of uncalibrated log likelihood ratios (log(Λ)) toward the distribution of a perfectly calibrated bi-Gaussian system. A necessary step in bi-Gaussianized calibration is to determine the σ value for the target perfectly

[16] $E_{=}^{*} = 0.05$ with $n_{\mathbb{S}} = 100$ and $n_{\mathrm{s}} + n_{\mathrm{d}} = 5050$ corresponds to $\sigma^{*} = 3.29$, $C_{\mathrm{llr}}^{*} = 0.184$, and a theoretical maximum calibrated likelihood-ratio value of 13,934,208 without compression and 192,398 with compression at $\gamma = 1$.

[17] $E_{=}^{*} = 0.025$ with $n_{\mathbb{S}} = 200$ and $n_{\mathrm{s}} + n_{\mathrm{d}} = 20{,}100$ corresponds to $\sigma^{*} = 3.92$, $C_{\mathrm{llr}}^{*} = 0.095$, and a theoretical maximum calibrated likelihood-ratio value of 4,677,981,747 without compression and 19,816,664 with compression at $\gamma = 1$.

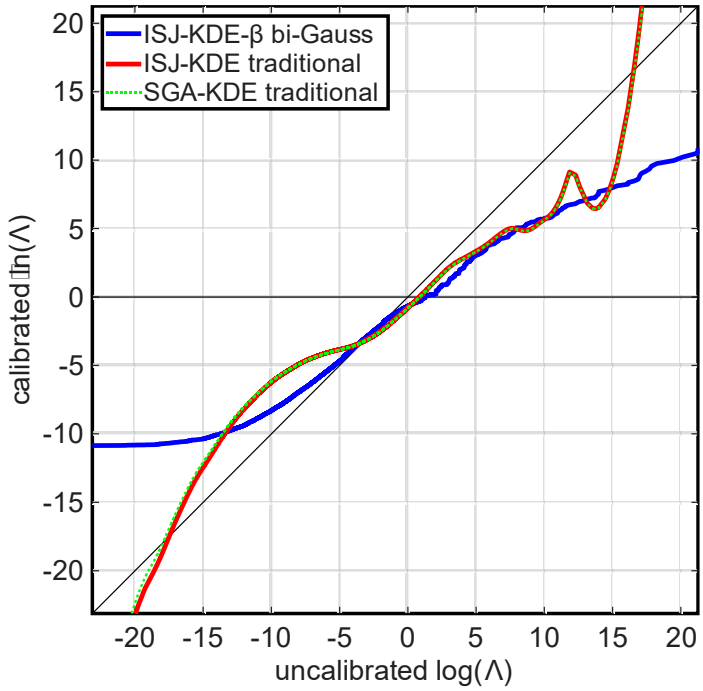
calibrated bi-Gaussian system. The methods used to determine target σ values in Morrison (2024) suffered from several weaknesses: susceptibility to sampling variability, discretization, curve-fitting approximation, and positive bias. The current paper presented new methods that overcame those problems. One of those methods used a beta-binomial model with uniform priors to recalculate $E_{=}$ (we use $E_{=}^{*}$ to represent the recalculated $E_{=}$) and then calculated the target σ^{*} for the perfectly calibrated bi-Gaussian system from $E_{=}^{*}$. As the sample size decreases, and/or as the degree of separation between the same-source and different-source distributions increases, $E_{=}^{*}$ increases relative to $E_{=}$, hence, σ^{*} decreases and the bi-Gaussian-calibrated likelihood ratios shrink toward the neutral value of 1. This method has the desirable effect of reducing the probability of overstating the strength of evidence. In addition, this method allows us to conceptualize justification of the minimum and maximum values obtained for calibrated likelihood ratios in terms of whether the sample size for the training data (the number of sources in the training data) is large enough to support the error rate $E_{=}^{*}$. In addition to shrinkage induced by the beta-binomial method, we introduced a bi-Gaussianized-calibration-integral compression method that, based on sample size, further compresses the calibrated likelihood-ratio values (including the minimum and maximum calibrated likelihood-ratio values) toward the neutral value of 1.

Appendix A: Comparison with traditional KDE calibration

In this appendix, we compare the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function for traditional KDE calibration with the uncalibrated-log(Λ)-to-calibrated-ln(Λ) mapping function for ISJ-KDE-β bi-Gaussianized calibration with compression. We used both the ISJ method and Silverman's Gaussian-approximation (SGA) method to select the bandwidths for traditional KDE calibration. Figure 15 and Figure 16 show the mapping functions for the voice data and the glass data, respectively. For both datasets, the KDE-calibration mapping functions are non-monotonic and have major deviations from the bi-Gaussianized-calibration mapping functions. In the case of the

767 glass data, using the ISJ method to select the KDE bandwidths resulted in the mapping
768 function having sudden and extreme deviations. In addition, over much of the range of
769 the input data, this KDE mapping function output calibrated values of minus or plus
770 infinity. We think that these results provide sufficient reason to not use traditional KDE
771 calibration; hence, we did not proceed to attempt to perform cross-validated calibration
772 on the real data using traditional KDE calibration.

773



774 **Figure 15.** For the forensic-voice-comparison data, mapping functions between uncalibrated log(Λ)
775 values and ln(Λ) values calibrated using ISJ-KDE-β bi-Gaussianized calibration with compression
776 and using traditional KDE calibration with bandwidth selected using the ISJ method and the SGA
777 method.

778

779

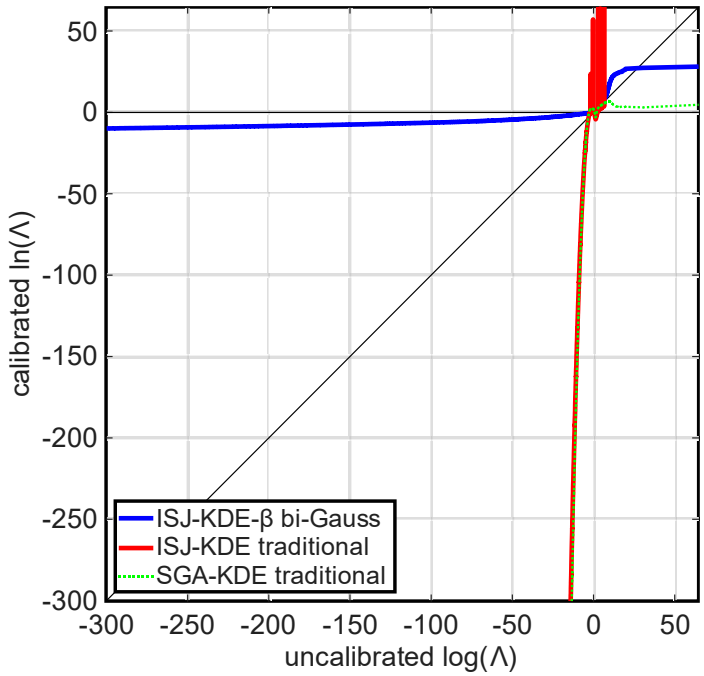


Figure 16. For the glass data, mapping functions between uncalibrated log(Λ) values and ln(Λ) values calibrated using ISJ-KDE-β bi-Gaussianized calibration with compression and using traditional KDE calibration with bandwidth selected using the ISJ method and the SGA method.

Appendix B: Comparison with bounding methods

Vergeer et al. (2016) and Alberink et al. (2026) proposed methods for bounding the minimum and maximum values of reported likelihood ratios (hereinafter we will refer to these as "bounding methods"). The bounding method proposed in Vergeer et al. (2016) is called "empirical lower and upper bounds (ELUB)". It makes use of normalized Bayes-error rate combined with the addition of misleading likelihood ratios. The bounding method proposed in Alberink et al. (2026) is not named therein. It makes use of tail probabilities and beta-binomial models with Jeffreys' reference priors or with uniform priors. For both the bounding methods, log(Λ) values that are outside the bounds are replaced with the value of the nearest bound, and log(Λ) values that are within the bounds are unchanged.

Table 4 gives the minimum and maximum calculated likelihood ratio values using ISJ-KDE-β bi-Gaussianized calibration, both with and without compression, and the

minimum and maximum values resulting from applying the bounding methods to the likelihood ratios calculated using ISJ-KDE-β bi-Gaussianized calibration without compression.

Table 4. For the voice data and the glass data, minimum and maximum likelihood-ratio values calculated using ISJ-KDE-β bi-Gaussianized calibration, both with and without compression, and the minimum and maximum values resulting from applying the Vergeer et al. (2016) and Alberink et al. (2026) bounding methods to the output of ISJ-KDE-β bi-Gaussianized calibration without compression.

Compression / Bounding method	Voice data		Glass data	
	minimum	**maximum**	**minimum**	**maximum**
bi-Gauss without compression	1/10,289,300	10,152,703	1/21,569*	3.94×10^{15}
bi-Gauss with compression	1/70,336	48,867	1/21,057*	1.55×10^{12}
Vergeer et al. (2016) ELUB	1/37	1,253	1/305	38,973
Alberink et al. (2026) Jeffreys' prior	1/47	593	1/599	27,803
Alberink et al. (2026) uniform prior	1/37	480	1/306	14,876

* value at probability point mass

For bi-Gaussianized calibration, the bounds are the minimum and maximum likelihood-ratio values obtained for the training data. Compression moves these bounds closer to $\ln(\Lambda) = 0$, but, even after compression, these bounds are much further from $\log(\Lambda) = 0$ than are the bounds resulting from applying the bounding methods.

Whereas bi-Gaussianized calibration makes parametric assumptions, the bounding methods do not. Bi-Gaussianized calibration assumes that the distribution of calibrated $\ln(\Lambda)$ values correspond to the Gaussian mixture of a perfectly-calibrated bi-Gaussian system. Based on this assumption, bi-Gaussianized calibration allows for minimum

and maximum calibrated ln(Λ) values that are much further from the neutral value of 0 than is the case for the bounding methods.

If one were concerned that the parametric assumption for bi-Gaussianized calibration leads to calibrated ln(Λ) values that are too far from the neutral value of 0, one could apply the beta-binomial method with larger values for the prior hyperparameters (see §3.2.4 above), and/or one could apply the compression method with a larger compression factor (see §2.7 above).

Unlike for the bounding methods, that do not alter log(Λ) values that are within the bounds, in addition to limiting the minimum and maximum values, applying bi-Gaussianized calibration with the beta-binomial method and/or with the compression method shrinks/compresses calibrated ln(Λ) values that are within the bounds.

References

Aitken C.G.G., Lucy D. (2004a). Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*, 53, 109–122. https://doi.org/10.1046/j.0035-9254.2003.05271.x

Aitken C.G.G., Lucy D. (2004b). Corrigendum: Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*, 53, 665–666. https://doi.org/10.1111/j.1467-9876.2004.02031.x

Alberink I., Leegwater A.J., Malmborg J., Nordgaard A., Sjerps M.J., van der Ham L. (2026). A transparent method to determine limit values for Likelihood Ratio systems likelihood ratio systems. *Law, Probability and Risk*, 25, mgaf015, https://doi.org/10.1093/lpr/mgaf015

Ayer M., Brunk H.D., Ewing G.M., Reid W.T., Silverman E. (1955). An empirical distribution function for sampling with incomplete information. *The Annals of Mathematical Statistics*, 26, 641–647. https://www.jstor.org/stable/2236377

839 Birdsall T.G. (1973). The theory of signal detectability: ROC curves and their
840 character. Technical Report No. 177. Cooley Electronics Laboratory,
841 Department of Electrical and Computer Engineering, The University of
842 Michigan. Ann Arbor, Michigan.

843 Botev Z.I., Grotowski J.F., Kroese D.P. (2010). Kernel density estimation via
844 diffusion. *Annals of Statistics*, 38, 2916–2957. https://doi.org/10.1214/10-
845 AOS799

846 Brent, R.P. (1973). *Algorithms for Minimization without Derivatives*. Englewood
847 Cliffs, NJ: Prentice-Hall.

848 Brümmer N., du Preez J. (2006). Application independent evaluation of speaker
849 detection. *Computer Speech and Language*, 20, 230–275.
850 https://doi.org/10.1016/j.csl.2005.08.001

851 Forsythe G.E., Malcolm M.A., Moler C.B. (1976). *Computer Methods for*
852 *Mathematical Computations*. Englewood Cliffs, NJ: Prentice Hall.

853 González-Rodríguez J., Rose P., Ramos D., Toledano D.T., Ortega-García, J. (2007).
854 Emulating DNA: Rigorous quantification of evidential weight in transparent and
855 testable forensic speaker recognition. *IEEE Transactions on Speech and Audio*
856 *Processing*, 15, 2104–2115. https://doi.org/10.1109/TASL.2007.902747

857 Good I.J. (1985). Weight of evidence: A brief survey. In Bernardo J.M., DeGroot
858 M.H., Lindley D.V., Smith A.F.M. (Eds.), *Bayesian Statistics 2* (pp. 249–270).
859 Elsevier.

860 Morrison G.S. (2013). Tutorial on logistic-regression calibration and fusion:
861 Converting a score to a likelihood ratio. *Australian Journal of Forensic*
862 *Sciences*, 45, 173–197. https://doi.org/10.1080/00450618.2012.733025

863 Morrison G.S. (2021). In the context of forensic casework, are there meaningful

metrics of the degree of calibration? *Forensic Science International: Synergy*, 3, 100157. https://doi.org/10.1016/j.fsisyn.2021.100157

Morrison G.S. (2024). Bi-Gaussianized calibration of likelihood ratios. *Law, Probability & Risk*, 23, mgae004. https://doi.org/10.1093/lpr/mgae004

Morrison G.S. (2025). Taking account of typicality in calculation of likelihood ratios. *Law, Probability & Risk*, 24, mgaf009. https://doi.org/10.1093/lpr/mgaf009

Morrison G.S., Enzinger E., Hughes V., Jessen M., Meuwly D., Neumann C., Planting S., Thompson W.C., van der Vloed D., Ypma R.J.F., Zhang C., Anonymous A., Anonymous B. (2021). Consensus on validation of forensic voice comparison. *Science & Justice*, 61, 229–309. https://doi.org/10.1016/j.scijus.2021.02.002

Morrison G.S., Poh N. (2018). Avoiding overstating the strength of forensic evidence: Shrunk likelihood ratios / Bayes factors. *Science & Justice*, 58, 200–218. https://doi.org/10.1016/j.scijus.2017.12.005

Peterson W.W., Birdsall T.G., Fox W.C. (1954). The theory of signal detectability. *Transactions of the IRE Professional Group on Information Theory*, 4, 171–211. https://doi.org/10.1109/TIT.1954.1057460

Ramos D., Krish R.P., Fierrez J., Meuwly D. (2017). From biometric scores to forensic likelihood ratios. In Tistarelli M., Champod C. (Eds.), *Handbook of Biometrics for Forensic Science* (pp. 305–327). Cham, Switzerland: Springer. https://doi.org/10.1007/978-3-319-50673-9_14

Shampine L.F. (2008). Vectorized adaptive quadrature in MATLAB. *Journal of Computational and Applied Mathematics*, 211, 131–140. https://doi.org/10.1016/j.cam.2006.11.021

Silverman B.W. (1986). *Density Estimation for Statistics and Data Analysis*. London,

889 UK: Chapman & Hall/CRC.

890 van Es A., Wiarda W., Hordijk M., Alberink I., Vergeer P. (2017). Implementation
891 and assessment of a likelihood ratio approach for the evaluation of LA-ICP-MS
892 evidence in forensic glass analysis. *Science & Justice*, 57, 181–192.
893 https://doi.org/10.1016/j.scijus.2017.03.002

894 van Leeuwen D.A., Brümmer N. (2013). The distribution of calibrated likelihood-
895 ratios in speaker recognition. *Proceedings of Biometric Technologies in*
896 *Forensic Science, BTFS 2013* (pp. 24–29). https://cls.ru.nl/staff/dvleeuwen/btfs-
897 2013/proceedings-btfs2013.pdf

898 Vergeer P., Alberink I., Sjerps M.J., Ypma R.J.F (2020). Why calibrating LR-systems
899 is best practice. A reaction to "The evaluation of evidence for
900 microspectrophotometry data using functional data analysis", in FSI 305.
901 *Forensic Science International*, 314, 110388.
902 https://doi.org/10.1016/j.forsciint.2020.110388

903 Vergeer P., van Es A., de Jongh A., Alberink I., Stoel R. (2016). Numerical
904 likelihood ratios outputted by LR systems are often based on extrapolation:
905 When to stop Extrapolating. *Science & Justice*, 56, 482–491.
906 https://doi.org/10.1016/j.scijus.2016.06.003

907 Weber P., Enzinger E., Labrador B., Lozano-Díez A., Ramos D., González-
908 Rodríguez J., Morrison G.S. (2022). Validation of the alpha version of the E^3
909 Forensic Speech Science System (E^3FS^3) core software tools. *Forensic Science*
910 *International: Synergy*, 4, 100223. https://doi.org/10.1016/j.fsisyn.2022.100223

911 Zadrozny B., Elkan C. (2002). Transforming classifier scores into accurate multiclass
912 probability estimates. *Proceedings of the Eighth ACM SIGKDD International*
913 *Conference on Knowledge Discovery and Data Mining* (pp. 694–699).
914 https://doi.org/10.1145/775047.775151