UNIVERSITY OF
CAMBRIDGE

Data Intensive Science
S1 Coursework

Liz Tan (eszt2)
Department of Physics, University of Cambridge
Michaelmas Term 2024

Word count: 2994 words

1 Introduction

This report compares the statistical power of two methods—an extended maximum likelihood (EML) fit and a weighted fit using *sWeights*—for analysing samples from a two-dimensional probability distribution containing signal and background components. Both methods were used to estimate a decay parameter revealing bias in both approaches, with *sWeights* exhibiting greater bias.

2 The model

The model consisted of an independent signal and a background distribution:

$$P(X,Y) = f g_s(X) h_s(Y) + (1-f) g_b(X) h_b(Y) \tag{1}$$

where $P(X,Y)$ is the joint probability density function (p.d.f.), f is the signal fraction, g_s and g_b are the p.d.f.s of the signal and background in the X direction respectively and h_s and h_b are the p.d.f.s of the signal and background in the Y direction, respectively. The signal density in X is defined by a *Crystal Ball* probability distribution

$$g_s(X;\mu,\sigma,\beta,m) = N \cdot \begin{cases} e^{-Z^2/2}, & \text{for } Z > -\beta, \\ \left(\frac{m}{\beta}\right)^m e^{-\beta^2/2} \left(\frac{m}{\beta} - \beta - Z\right)^{-m}, & \text{for } Z \leq -\beta. \end{cases} \tag{2}$$

with parameters $\mu = 3, \sigma = 0.3, \beta = 1$ and $m = 1.4$. This equation is given in terms of Z, where $Z = (X - \mu)/\sigma$. The distribution is only valid for $\beta > 0$ and $m > 1$. The signal density in Y is given by an exponential decay distribution:

$$h_s(Y : \lambda) = \lambda e^{-\lambda Y} \tag{3}$$

with $\lambda = 0.3$. The background p.d.f. for X, $g_b(X)$ is a uniform distribution, and the background distribution for Y, $h_b(Y)$ is a normal distribution with mean $\mu_b = 0$ and standard deviation $\sigma_b = 2.5$. All of these distributions were appropriately truncated over the region $X \in [0, 5]$ and $Y \in [0, 10]$.

2.1 Part A: Normalisation constant of the *Crystall Ball* function

To find the normalisation constant N in Equation 2, we can perform an integral over all regions where this function is defined and set this integral equal to unity. We can form an integral in Z by performing a change of variables transformation:

$$\sigma dZ = dX \tag{4}$$

Hence we can write the integral of the *Crystal Ball* function from Equation 2 as

$$\sigma \left[\int_{Z=-\beta}^{Z=\infty} e^{-Z^2/2} dZ + \int_{Z=-\infty}^{Z=-\beta} \left(\frac{m}{\beta}\right)^m e^{-\beta^2/2} \left(\frac{m}{\beta} - \beta - Z\right)^{-m} dZ \right] = N^{-1} \tag{5}$$

The Gaussian c.d.f., Φ is defined as

$$\Phi(\beta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\beta} e^{\frac{-X^2}{2}} dX \tag{6}$$

Because $e^{X^2/2}$ is a symmetric function, the limits can be swapped and still give the same result. Hence, Equation 6 can be substituted into the first integral in Equation 5. We can evaluate the second integral from Equation 5 explicitly:

$$\left(\frac{m}{\beta}\right)^m e^{-\beta^2/2} \int_{-\infty}^{-\beta} \left(\frac{m}{\beta} - \beta - Z\right)^{-m} dZ = \left(\frac{m}{\beta}\right)^m e^{-\beta^2/2} \left[\left(\frac{m}{\beta} - \beta - Z\right)^{-m+1} \right]_{-\infty}^{-\beta} \frac{1}{m-1} \tag{7}$$

Because $m > 1$ always, the expression $(m/\beta - \beta - Z)^{-m+1} \to 0$ as $Z \to -\infty$. Thus, Equation 7 is equivalent to

$$\left(\frac{m}{\beta}\right) e^{-\beta^2/2} \frac{1}{m-1} \tag{8}$$

By substituting Equation 6 and 8 into 5, the inverse of the normalisation factor can be written as

$$N^{-1} = \sigma \left[\frac{m}{\beta(m-1)} e^{-}\beta^2/2 + \sqrt{2\pi}\Phi(\beta) \right] \tag{9}$$

2.2 Part B: Defining the p.d.f.s and normalisation

The signal and background p.d.f.s from Equation 1 were defined using `scipy.stats` (for g_s and the normalisation constant in h_b) along with explicitly defined functions. They were appropriately normalised in their truncated range by dividing by $F(\beta)-F(\alpha)$, where F is the cumulative distribution function.

To ensure that appropriate normalisation of the p.d.f.s, 1000 different parameters were randomly created. These parameters were chosen to avoid extreme values, as such cases could introduce numerical instability. For example, large μ and σ values in g_s could result in the majority of the unnormalised distribution falling outside the valid range of the function. This would cause the normalisation factor to become very small, leading to potential numerical instability or overflow when dividing by it. With these random parameters, the p.d.f.s were checked to be normalised over their respective ranges using the `scipy.integrate.quad` function. All of the tests passed, with the integrals equal to unity within a 1×10^{-5} tolerance, demonstrating successful normalisation.

2.3 Part C: Plotting the distributions

Figure 1 shows the marginal distributions in X and Y. The joint probability function is shown in Figure 2.

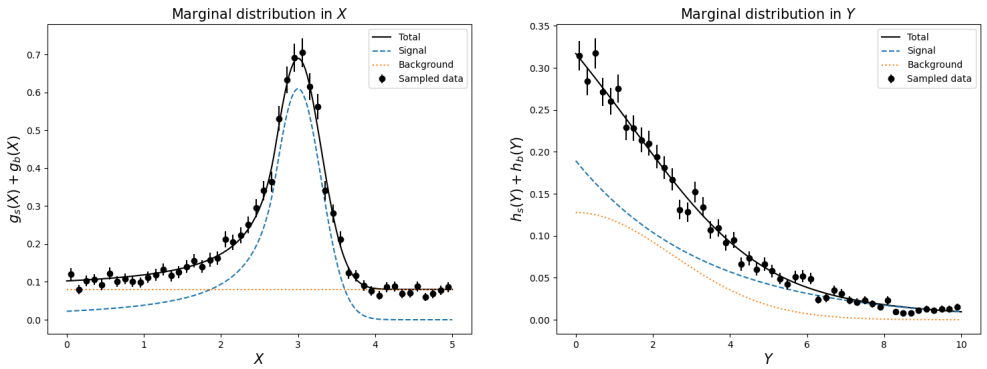
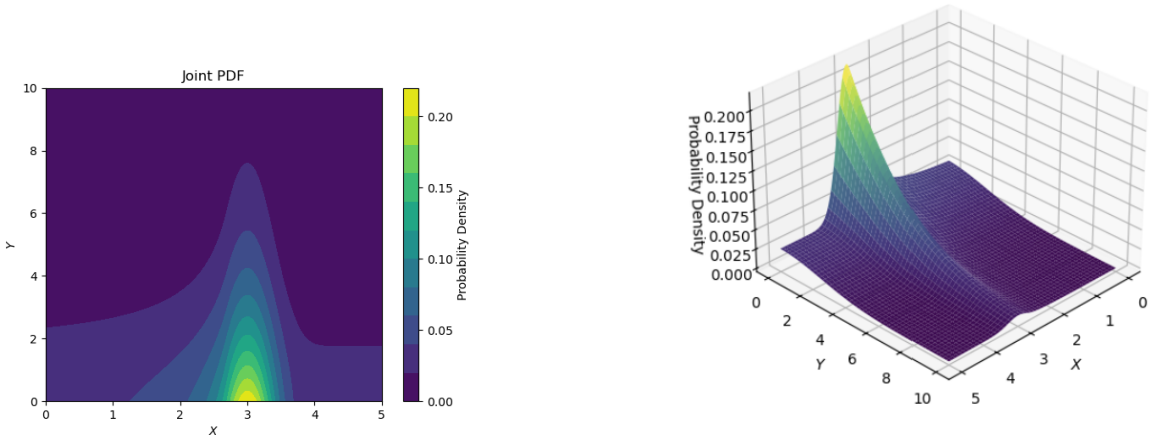


Figure 1: Marginal distribution in X and Y for both the signal, background and total p.d.f.s along with binned data from a sample containing 5000 events.



(a) 2D Joint PDF

(b) 3D Joint PDF

Figure 2: The joint probability function, $P(X, Y)$, in both 2D and 3D.

3 Part D: High-statistics sample

3.1 Generating the high-statistics sample

We used an accept-reject method with batches to generate a sample of 100,000 events. Batches improves efficiency of the code by reducing the overhead of generating and evaluating samples individually.

A 2D histogram of this sample is shown in Figure 3. The histogram contains empty bins at the low-probability regions. This is due to the inefficiency of the accept-reject method in these regions: there is a high rejection rate when the p.d.f. values are small.

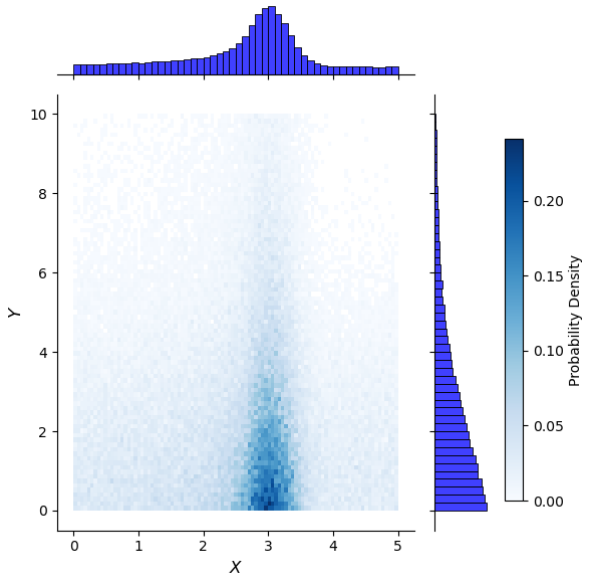


Figure 3: Histogram of high-statistics sample containing 100,000 events. The marginal histograms on the top and right of the figure show the distribution of the sample in the X and Y dimension, respectively.

3.2 EML fit

Using the `iminuit` package, we performed an EML fit for the eight parameters in the joint p.d.f. from Equation 1 and for N, the number of events in the sample. We used unbinned fits, as the estimates could be computed in a reasonable time. The EML of a sample is defined by

$$L(\nu; \overrightarrow{\theta}) = \frac{e^{-\nu}}{N!} \prod_{i=1}^{N} \nu P(X_i, Y_i; \overrightarrow{\theta}) \tag{10}$$

where L is the extended likelihood, ν is the expected number of events, N is the observed number of events, P is the joint density and $\overrightarrow{\theta}$ represents the parameters of the joint density that we are trying to fit for [1]. The cost function in an EML fit is the negative logarithm of the likelihood (NLL). The fitted parameters are the ones that minimise the NLL.

The only limits to the `iminuit` minimiser was $N > 0, \beta > 0$ and $m > 1$. We decided not to add limits that constrained the optimiser too much, and used initial guesses derived from the true parameters, perturbed by Gaussian noise to show the robustness of the optimiser. The resulting estimated parameters from these "noisy" initial guesses were nearly identical to those obtained using the true parameters as initial guesses, confirming the reliability of the fit.

The EML estimated parameters are shown in Table 1.

Name	**True Value**	**Estimated Value**	**Hesse Error**
N	100.0×10^3	100.00×10^3	0.32×10^3
μ	3.0	3.0072	0.0026
σ	0.3	0.2960	0.0024
β	1.0	0.971	0.022
m	1.4	1.48	0.07
f	0.6	0.600	0.004
λ	0.3	0.2996	0.0021
μ_b	0.0	0.02	0.08
σ_b	2.5	2.48	0.04

Table 1: Parameter estimates, Hesse errors, and true parameter values for a sample containing 100,000 events.

Many of the estimated parameters are not within the Hesse error. Reasons for this include the sample size being finite, leading to statistical fluctuations. The accept-reject method of sampling could introduce biases by undersampling low-probability regions.

There is strong correlation between parameters. For example, the correlation betweeen σ and μ is -0.547 and the correlation between β and μ is -0.871. These correlations could make it difficult for the minimisation process to independently constrain each parameter resulting in estimated values that deviate further from the true values. The calculation of uncertainties using the Hessian assumes that the likelihood surface is locally quadratic. Strong correlations between parameters can invalidate this assumption, causing the Hessian-derived variances to either under- or over-cover. The Hessian covariance matrix is shown in Figure 12 in the Appendix.

3.3 Timing the sample generation and EML fit

The time taken to generate 100,000 samples and to estimate the parameters with the EML fit using both the noisy and true parameters as starting values averaged over 100 calls is shown in Table 2. As expected, it takes much longer for the EML fit to converge to the minimum when the starting parameters are further away from the true parameters.

Function call	**Time taken (averaged over 100 calls), s**
`np.random.normal(size = 100,000)`	0.013
Accept-reject generator for 100,000 samples	0.8146
EML fit for 100,000 samples with noisy parameters as initial guesses	21.4916
EML fit for 100,000 samples with true parameters as initial guesses	9.0146

Table 2: The average time taken, over 100 runs, to generate 100,000 samples and fit the samples using both the true parameters and 'noisy' initial guesses. Additionally, the time taken to execute `np.random.normal(size=100,000)` is provided as a benchmark for computational performance.

4 Part E: EML parametric bootstrapping study

We ran a parametric bootstrap study using the EML estimated values from Table 1. These values were used instead of the true values as we often would not know the true values. From this point onwards, these values will be referred to as the 'true values'. In the study, we generated 500 bootstraps ('toys') with Poisson distributed sample sizes of 500, 1000, 2500, 5000 and 10,000. An EML fit was performed on each of these bootstraps, with the true values as initial guesses. Additional constraints were added to speed up the optimisation process but these constraints still had a large enough range not to bias the optimisation process.
For each sample size, between 5 to 25 of the toys had invalid minimums - the minimisation process failed to converge to a valid solution which is likely caused by statistical outliers. Toys with invalid minimums were discarded and we cut the final number of bootstraps per sample size to 450 such that they all had the same length for a fair analysis. Figure 4 shows histograms of the estimated λ values, the uncertainties and the pulls, which were defined to be

$$\text{pull}(\theta) = \frac{\hat{\theta} - \theta}{\sigma_\theta} \tag{11}$$

where $\hat{\theta}$ is the estimated value of the parameter and θ is the true value of the parameter.

4.1 Results

The shape of the $\hat{\lambda}$ distribution is a Gaussian centered around the true value, which is to be expected according to the central limit theorem .The estimated value of λ agrees with the true value within the uncertainty bounds for all sample sizes, except for $N = 10,000$.

This exception could arise due to a combination of reduced uncertainty at larger sample sizes and systematic biases.
The variance and pull plots are also consistent with Gaussians, although the variance plots show skew in for smaller samples. In particular, the pull plots become more similar with unit Gaussians as sample size increases, implying that the error has the correct coverage for higher sample sizes. The mean variance of λ decreases for higher sample sizes, aligning with the expectation that $\hat{V}(\lambda)$ is inversely correlated with sample size.

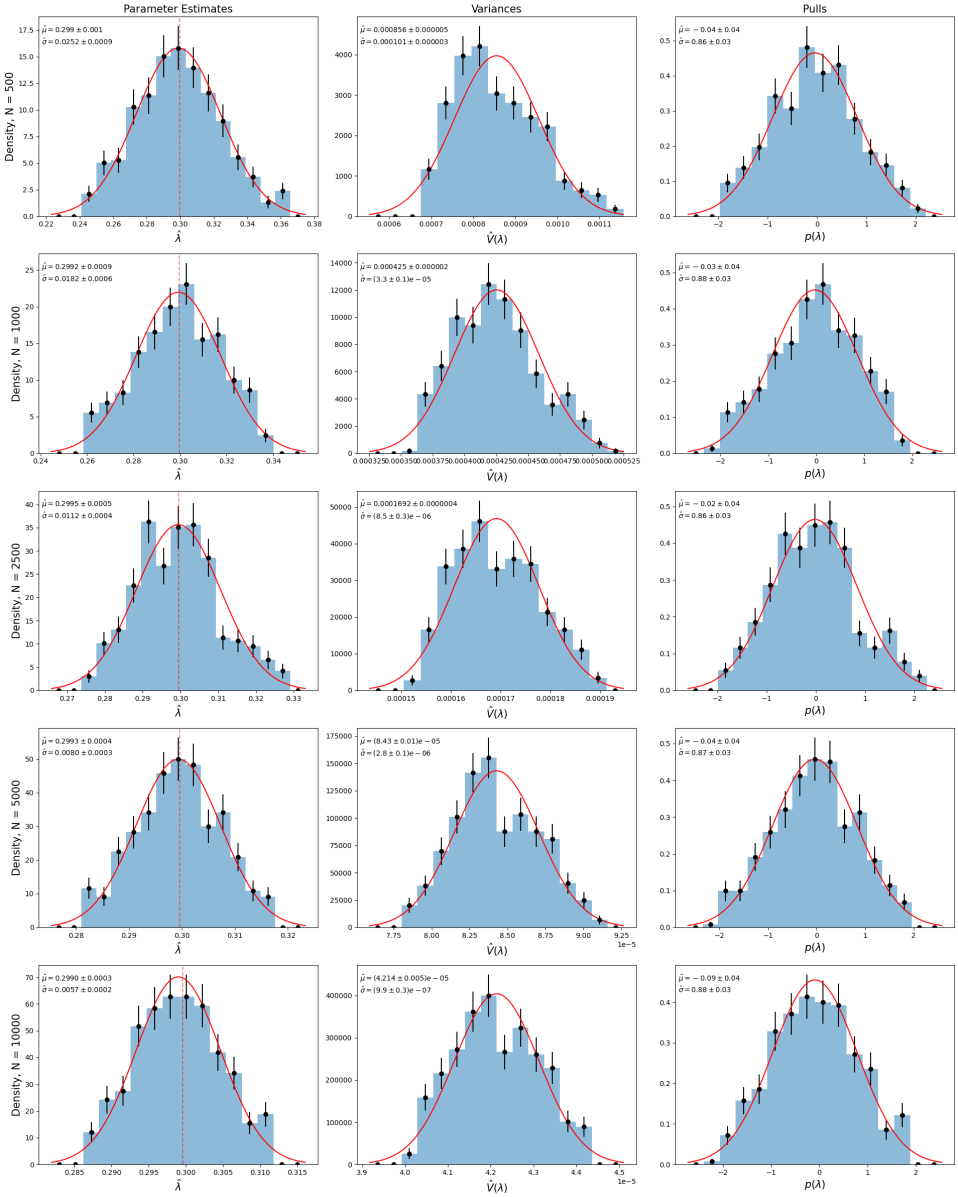


Figure 4: Comparison of $\hat{\lambda}$ parameter estimates using the EML method across different sample sizes (N). The red dashed lines in the parameter estimate plots indicate the true value of λ. The distributions are normalised and shown with fitted Gaussian curves (red lines). Text in top right show $\hat{\mu}$ and $\hat{\sigma}$ of the fitted Gaussian.

4.2 Bias

The bias of an estimated parameter is defined as

$$\text{Bias}(\hat{\theta}) = E(\hat{\theta}) - \theta \tag{12}$$

The bias for of $\hat{\lambda}$ is shown in Figure 5. Maximum likelihood estimators are biased, but are consistent: we expect the bias to decrease as the sample size increases. It is unclear why the bias increases for the samples with 5000 and $10,000$ events, as we expect the bias to converge to zero for these higher values. This could be related to the inconsistencies with the accept-reject method and the joint p.d.f.

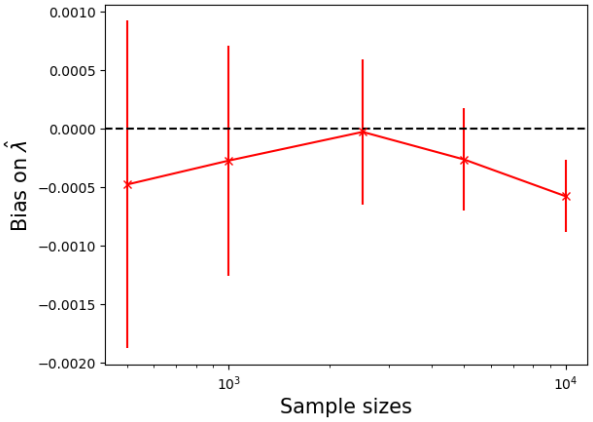


Figure 5: Bias in the estimate $\hat{\lambda}$ for different sample sizes.

4.3 Variance

We expect the estimates of the variance of λ, $\hat{V}(\lambda)$ for the different sample sizes to be inversely correlated to the sample size N. Hence, there is a linear relationship between the logarithm of the standard deviation and the sample size N:

$$\log \hat{\sigma_\lambda} = -0.5N + C \tag{13}$$

where $\hat{\sigma_\lambda}$ is the estimated uncertainty on λ and C is some arbitrary constant. Figure 6 shows the estimated standard deviations of λ for the different sample sizes N. There is a clear linear relationship between the logarithm of the standard deviation and the logarithm of the sample size with the gradient being -0.5 as expected from Equation 4.3.

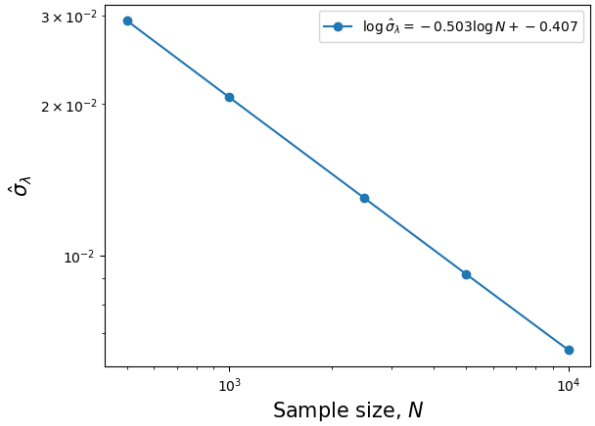


Figure 6: Estimated standard deviation of λ for the different sample sizes on a log-log plot. No error bars were plotted as they were too small to be visible.

5 Part F: sWeights

sWeights is a method used to infer the properties of a signal distribution in data consisting of both signal and background events. This method is only valid if the variables are independent for the signal and background, which is the case for our joint probability from Equation 1. sWeights allow us to extract the signal distribution in Y, $h_s(Y)$ without requiring a full parametric model for the background density $h_b(Y)$. This proves to be useful because parameterising a background density can be difficult.

The sWeights method operates by assigning weights to individual data points to extract the signal, and we can perform inference on the weighted dataset. The weights are derived using a well-parameterised discriminant variable, which in our case is X. We can then calculate the weight distribution for the signal and background densities.

The signal weight function $w_s(X)$ is determined with the properties [2]

$$f h_s(Y) = \int w_s(X) P(X, Y) dX \tag{14}$$

such that it extracts the signal in Y. There are many solutions for $w_s(X)$, but the optimal choice is the one that minimises the variance over $g(X)$:

$$w_s(X) = \frac{\alpha_s g_s(X) + \alpha_b g_b(X)}{g(X)} \tag{15}$$

where α_s and α_b are constants that can be calculated [2].

We used the package `sweights` to calculate the weights of the events in our samples. The same toys as in the EML parametric bootstrapping study were used so we could reliably compare results from both parameter estimation methods.

First, we perform an EML fit to the X distribution. However, we now fit for the signal and background yields. These yields, along with the signal and background p.d.f.s, are used to calculate the weights for each event. An example of the weight distribution for a single toy with 10,000 events is shown in Figure 7.

When applying these weight functions to our dataset, the sum of the signal weights is equal to the signal yield.

We then create a weighted histogram of the Y distribution using these weights, which

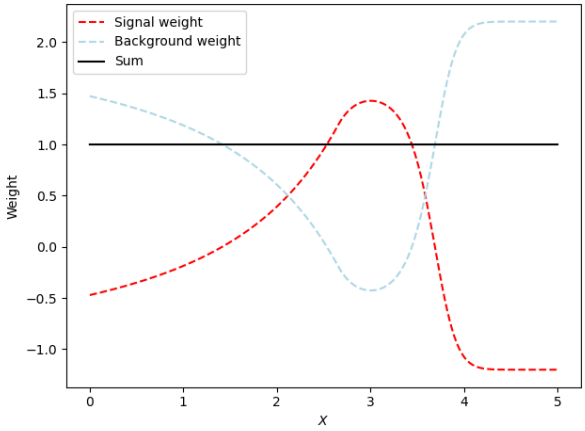


Figure 7: Signal and background weight functions determined from distribution in X using sWeights. The sum of the signal and background weights at any X is unity.

statistically isolates the signal component from the background component. Finally, we fit the signal-weighted Y histogram with $h_s(Y)$ to extract the signal parameter λ with `scipy.curvefit`.

5.1 Results

The estimated values of λ, estimated variances and pulls are shown in Figure 8. These estimates used the same toys as in the EML analysis. The estimated values of λ, the variance and the pulls all are Gaussian distributed. However, the true value of λ is within the uncertainty bounds only for sample sizes $N = 2500$ and $N = 5000$.

5.2 Bias

The bias of the sWeights estimates of λ is shown in Figure 9. Similarly to the bias on the MLE estimate, the bias decreases until the 2500 sample size then increases again, although on a smaller extent. The bias of the sWeights estimate is larger than that of the MLE estimate. While sWeights appear to introduce more bias, the estimator shows signs of consistency, as the bias generally decreases with larger sample sizes. An EML fit was used to determine parameters in X used to produce the weights - this step could introduce bias into the estimate.

5.3 Variance

The estimated standard deviation is shown in Figure 11. Similar to the EML fit, the logarithm of the standard deviation exhibits a linear relationship with the logarithm of the sample size. The gradient is approximately -0.5, within the uncertainty bounds, thus displaying the theoretical relationship in Equation 4.3.

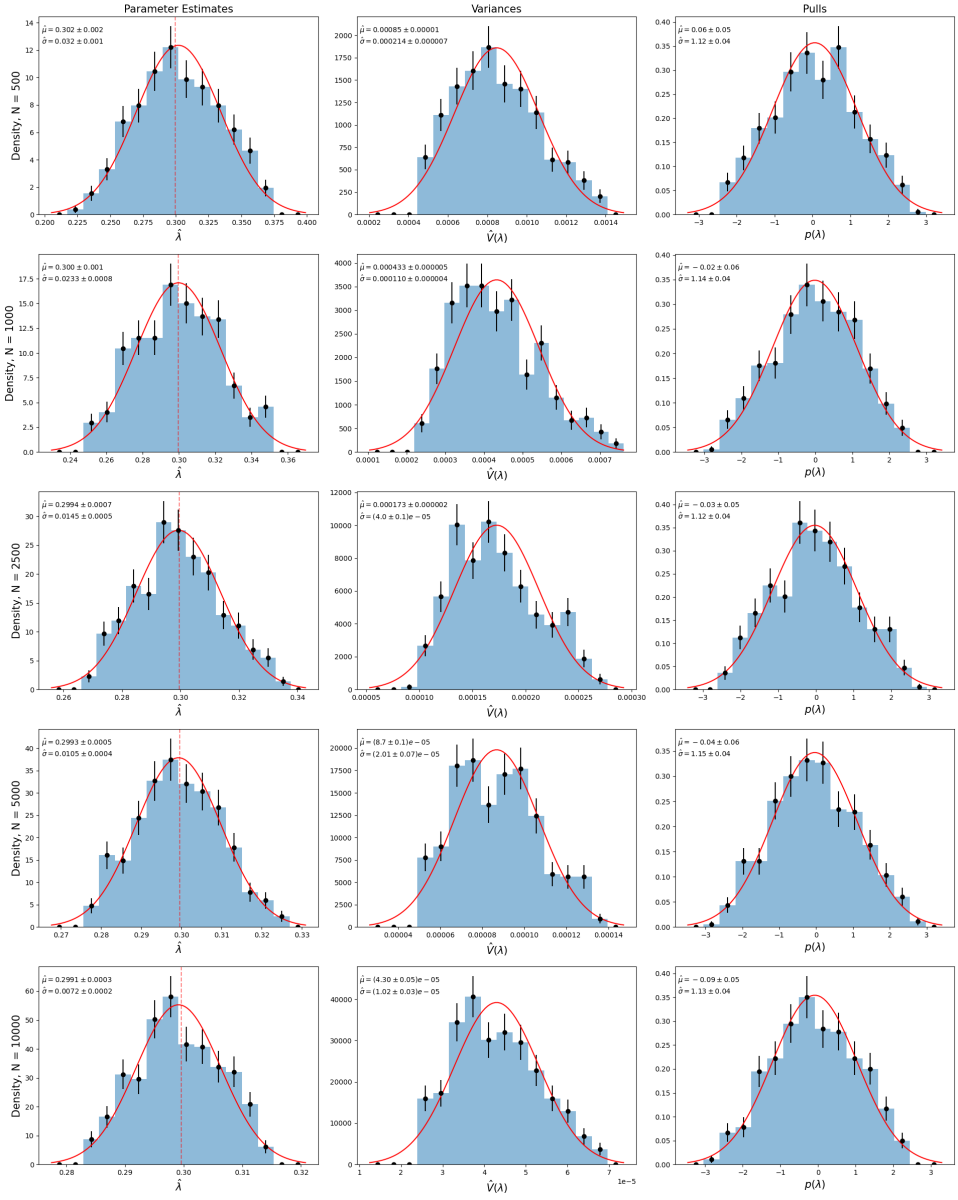


Figure 8: Comparison of $\hat{\lambda}$ parameter estimates using the sWeights method across different sample sizes (N). The red dashed lines in the parameter estimate plots indicate the true value of λ. The distributions are normalised and shown with fitted Gaussian curves (red lines). Text in top right show $\hat{\mu}$ and $\hat{\sigma}$ of the fitted Gaussian.

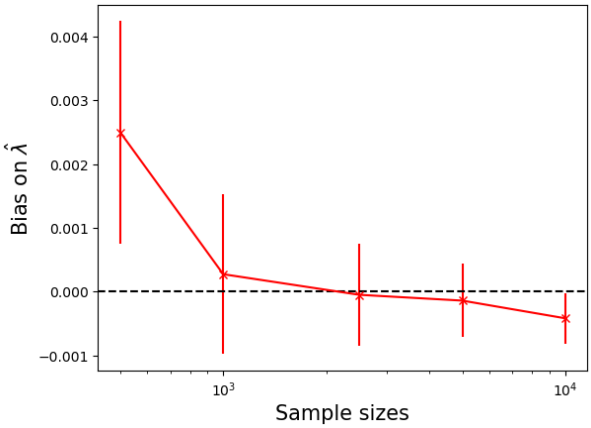


Figure 9: Bias in the sWeights estimate of $\hat{\lambda}$ for different sample sizes.

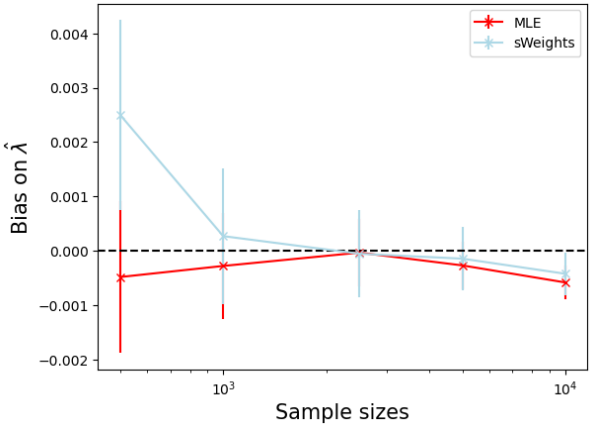


Figure 10: Bias of the estimate of λ using the EML fit and sWeights.

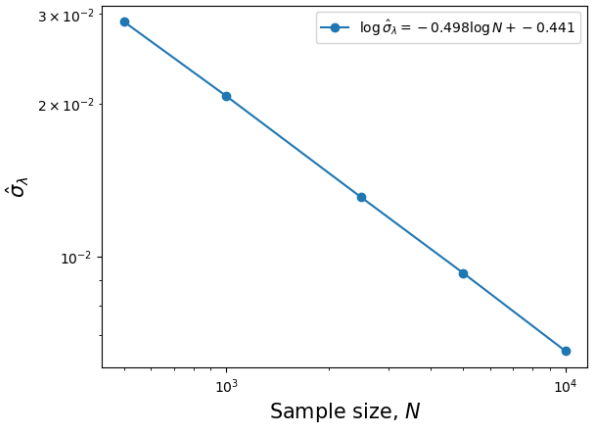


Figure 11: The estimated standard deviation of λ using sWeights for the different sample sizes on a log-log plot. No error bars were plotted as they were too small to be visible. The uncertainty on the gradient is 0.002.

6 Part G: Comparing EML and sWeights

6.1 Advantages for sWeights and disadvantages for EML

6.1.1 Curse of dimensionality for EML

The joint distribution consists of nine parameters - the optimiser has to deal with a high-dimensional parameter space. The number of possible combinations of parameters is large making it difficult to find the global maximum of the likelihood.

6.1.2 Strong correlations in parameters

The parameters in the joint distribution exhibit strong correlations that significantly influence the shape of the likelihood surface, creating flat regions where multiple parameter combinations yield similar likelihood values. This degeneracy complicates the optimisation process, making it difficult for the optimiser to converge reliably to a single solution when maximising the EML function.
Additionally, Hesse errors are valid if the likelihood surface is approximately quadratic near the maximum likelihood point. For highly correlated parameters, this assumption may break down, leading to over- or under-coverage.

6.1.3 Non-parameterised fit for Y

Although we have the parameterised models for $h_s(Y)$ and $h_b(Y)$, in real-world contexts it is unlikely that these would be available. Referring to Figure 1, the signal and background densities in X can be easily parameterised because the signal is a peaking distribution above a uniform background - the signal and background are well-seperated as they have distinct shapes. However, the signal and background distributions in Y both consist of exponential decreasing tails, as the background is the right-hand tail of a Gaussian and the signal is an exponential decaying function. Their shapes are similar and they overlap significantly, and hence can be difficult to parameterise [2]. Therefore, we could not perform an EML fit, as this method requires models for all components in all variables.

6.1.4 Computational cost

Especially for high-dimensions, the EML estimator has a higher computational cost than sWeights as it must optimise over many parameters.

6.2 Advantages for EML and disadvantages for sWeights

6.2.1 Independence of X and Y

It is a requirement for the sWeights method that the discriminant and control variables are independent in signal and background and can be seperated as in Equation 1. This is not a requirement for an EML fit.

6.2.2 Bias

From our analysis, there seems to be a greater bias on the sWeights estimate of λ for lower sample sizes. But the bias for both techniques become similar after $N = 2500$ events.

7 Conclusion

Because the joint distribution analysed in this report is independent in the signals for X and Y, and the signal in X is a peaking signal over a uniform background and is thus easily parameterised, sWeights is a better method for extracting λ from the samples. Furthermore, in real scientific contexts, it would be difficult to parameterise $h_s(Y)$. Although the bias is higher for the sWeights method, the bias decreases as sample sizes increases, suggesting that it is a consistent estimator.

8 Appendix

	N	mu	sigma	beta	m	f	lmbda	mu_b	sigma_b
N	9.94e+04	-37e-6	36e-6	0.7e-3	-0.002 **(-0.000)**	0.027e-3	18e-6	-0.002 **(-0.000)**	0.0008
mu	-37e-6	6.55e-06	-3e-6 **(-0.547)**	-29e-6 **(-0.515)**	59e-6 **(0.334)**	-0e-6 **(-0.032)**	-0e-6 **(-0.005)**	-3e-6 **(-0.013)**	2e-6 **(0.018)**
sigma	36e-6	-3e-6 **(-0.547)**	5.7e-06	24e-6 **(0.452)**	-54e-6 **(-0.330)**	3e-6 **(0.304)**	0e-6 **(0.087)**	15e-6 **(0.079)**	-9e-6 **(-0.103)**
beta	0.7e-3	-29e-6 **(-0.515)**	24e-6 **(0.452)**	0.000494	-1.3e-3 **(-0.871)**	0.010e-3 **(0.121)**	1e-6 **(0.023)**	0.1e-3 **(0.047)**	-0.1e-3 **(-0.064)**
m	-0.002 **(-0.000)**	59e-6 **(0.334)**	-54e-6 **(-0.330)**	-1.3e-3 **(-0.871)**	0.00471	-0.094e-3 **(-0.383)**	-14e-6 **(-0.100)**	-0.001 **(-0.111)**	0.0004 **(0.146)**
f	0.027e-3	-0e-6 **(-0.032)**	3e-6 **(0.304)**	0.010e-3 **(0.121)**	-0.094e-3 **(-0.383)**	1.27e-05	2e-6 **(0.264)**	0.048e-3 **(0.174)**	-0.028e-3 **(-0.219)**
lmbda	18e-6	-0e-6 **(-0.005)**	0e-6 **(0.087)**	1e-6 **(0.023)**	-14e-6 **(-0.100)**	2e-6 **(0.264)**	4.23e-06	11e-6 **(0.067)**	3e-6 **(0.046)**
mu_b	-0.002 **(-0.000)**	-3e-6 **(-0.013)**	15e-6 **(0.079)**	0.1e-3 **(0.047)**	-0.001 **(-0.111)**	0.048e-3 **(0.174)**	11e-6 **(0.067)**	0.00594	-0.0026 **(-0.924)**
sigma_b	0.0008	2e-6 **(0.018)**	-9e-6 **(-0.103)**	-0.1e-3 **(-0.064)**	0.0004 **(0.146)**	-0.028e-3 **(-0.219)**	3e-6 **(0.046)**	-0.0026 **(-0.924)**	0.00129

Figure 12: Hessian covariance matrix of the `iminuit` fit for a sample of 100,000 events.

9 Bibliography

9.1 Large Language Model (LLM) Usage

Claude 3.5 Sonnet and **ChatGPT 4.0** were used supportively in programming and report writing. The web browser version of the chatbots were used - I did not integrate generative tools within my programming software (VSCode).
Coding: All algorithms were produced by myself. LLMs were primarily used for plotting, code optimisation, big fixing and adding comments. In particular, ChatGPT was used in optimising the accept-reject generator in Part D. Whilst I wrote the base algorithm, the LLM produced the code to run the algorithm in batches. Example prompts:

- 'Help me optimise this code so it runs faster.'
- 'How do I produce 9 X 3 subplots?'

Report Writing: LLMs were used to decrease word count, optimise readability and correct grammatical errors. Example prompts:

- Please put this table into LATEX format `INSERT IMAGE OF EXCEL TABLE`.'
- 'Please correct any grammatical or spelling mistakes `INSERT TEXT EXCERPT FROM REPORT`'
- 'How do I write a pseudocode algorithm in LATEX?'

References

[1] Rojer Barlow. "Extended maximum likelihood". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 297.3 (1990), pp. 496–506. DOI: 10.1016/0168-9002(90)91334-8.

[2] Urs Langenegger et al. "Custom Orthogonal Weight functions (COWs) for Event Classification". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 603.3 (2009), pp. 339–345.