

---

---

## Special Syllabus Part II

### Fully Bayesian Unfolding For Experimental $\gamma$ -ray Spectra

---

---

Jens Bratten Due

Department of physics, UiO

#### **Abstract**

In this project, the method of fully bayesian unfolding (FBU) has been implemented and applied to the problem of unfolding  $\gamma$ -ray spectra. The results has been compared to the current method used by the nuclear physics group at UiO, the Folding Iteration Method, implemented in the OMpy package [1]. For implementation of FBU, the PyFBU-package, based on the Bayesian modelling package PyMC3, is used [2][3]. Both methods are performed on the first excited state of a synthetic data set, a simplified version of a physical case. The ensuing results show a similarly strong performance by both methods, with a slight advantage to FBU. This, along with the flexibility of having access to the posterior distribution, leads to the conclusion that further exploration of FBU is definitely of interest. The data used in this project is explained to carry the risk of hiding limitations of the method, and a suggested next step is to investigate performance on more complex and realistic data sets.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>The Response Matrix</b>	<b>2</b>
<b>3</b>	<b>The Folding Iteration Method</b>	<b>3</b>
3.1	The Compton Subtraction Method . . . . .	3
<b>4</b>	<b>Fully Bayesian Unfolding</b>	<b>4</b>
4.1	Sampling . . . . .	5
4.2	Posterior inference . . . . .	5
4.2.1	Point estimates . . . . .	5
4.2.2	Credible intervals . . . . .	5
<b>5</b>	<b>Synthetic data</b>	<b>6</b>
<b>6</b>	<b>Prior</b>	<b>9</b>
<b>7</b>	<b>Results &amp; Discussion</b>	<b>10</b>
7.1	Posterior distributions . . . . .	10
7.2	Final spectra . . . . .	12
<b>8</b>	<b>Conclusion</b>	<b>14</b>
	<b>Appendices</b>	<b>16</b>
<b>A</b>	<b>Implementation</b>	<b>16</b>

# 1 Introduction

This project comprises the second part of a special syllabus on the topic of Bayesian statistics, currently being undertaken by yours truly. The goal of this part is to apply the toolkit of Bayesian thinking to an existing problem and assess the results and possible advantages it might bring over other methods. The problem in question is that of unfolding, i.e. taking the output from an imperfect detector and attempting to reconstruct the true signal. No detectors are perfect, and determining the exact source of a specific output is of great interest for the experimental process. More specifically, the aim is to investigate the unfolding of  $\gamma$ -ray spectra using Fully Bayesian Unfolding (FBU) [4], and compare with the Folding Iteration Method, the current method in use by the nuclear physics group at UiO [5].

## 2 The Response Matrix

The response matrix contains what we know about the circumstances of the experiment, a large part of which are the characteristics of the detector. The response matrix has elements given by:

$$R_{rt} = P(\text{measured in bin r} \mid \text{true in bin t}) \quad (1)$$

[6]. This can be read as the probability of observing an event in energy bin i, given the true event happening in bin j. In a nutshell, the response matrix describes how a signal is smeared over the other bins in the spectrum.

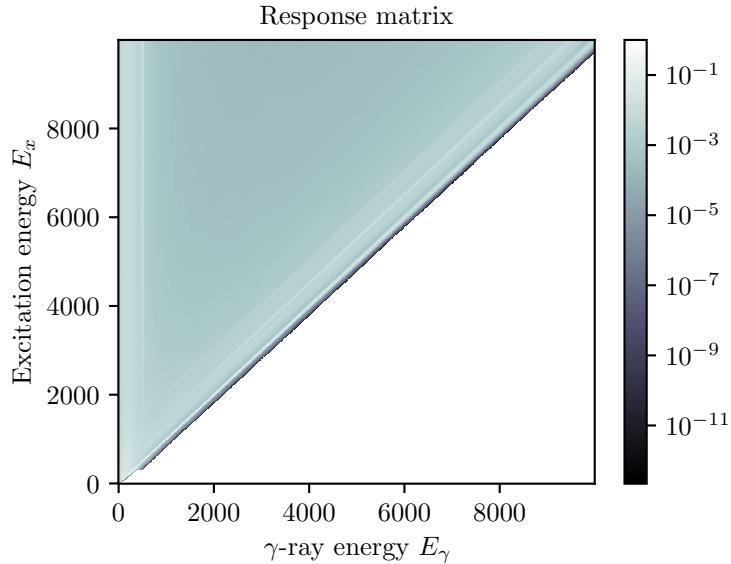


Figure 1: The response matrix collected from the OMpy library [7][8].

### 3 The Folding Iteration Method

The following section describes the methods developed by Guttormsen et al. [5].

The folded spectrum  $f$  is on the form

$$f = \mathbf{R}u, \quad (2)$$

where  $\mathbf{R}$  is the response matrix and  $u$  is the expectation values for the true spectrum. The iterative method can then be described in 4 parts.

- First we use the measured spectrum  $r$  as an initial guess,  
 $u_0 = r$
- We then fold this with the response matrix,  
 $f_0 = \mathbf{R}u_0$
- The difference between the folded and the raw spectrum is calculated and added to the initial guess, and we end up with the next trial spectrum,  
 $u_1 = u_0 + (r - f_0)$
- This is then repeated according to the following iteration scheme,  
 $u_{i+1} = u_i + (r - f_i)$

This method is performed until  $f_i \approx r$  within the experimental uncertainties [5]. It is important to note that for each new iteration, the oscillations between channels increase, as the solution approaches the inverted matrix solution  $u = \mathbf{R}^{-1}r$ , which exhibits large oscillations. [6][5]

#### 3.1 The Compton Subtraction Method

As the resulting spectrum from the Folding Iteration Method often contains some degree of fluctuations, the Compton subtraction method is performed to obtain a significantly more stable spectrum.

The first step is to define a new spectrum  $v(i)$  as the observed data excluding the Compton contribution:

$$v(i) = p_f(i)u(i) + w(i), \quad (3)$$

where  $u(i)$  is the spectrum obtained from the Folding Iteration Method, which multiplied with  $p_f$  gives the full energy contribution. The remaining contributions are contained in  $w(i) = u_s + u_d + u_a = p_s(i)u(i) + p_d(i)u(i) + \sum p_{511}(i)u(i)$ , representing single escape, double escape and annihilation (note the missing Compton contribution " $u_c$ "). To match the observed energy resolution, each contribution is then smoothed with a Gaussian function. Next, we subtract this from the raw spectrum to obtain the Compton background spectrum:

$$c(i) = r(i) - v(i). \quad (4)$$

This spectrum may exhibit significant oscillations, and is thus further smoothed. This smoothing carries a low risk of loss of important information due to the nature of the spectrum not containing

any sharp, narrow peaks. After this smoothing procedure on the individual contributions, we now "return" to the unfolded spectrum like so:

$$u(i) = \frac{r(i) - c(i) - w(i)}{p_f}. \quad (5)$$

Finally, to get closer to the true number of events, we correct for the total detector efficiency:

$$U(i) = \frac{u(i)}{\epsilon_{tot}(i)}. \quad (6)$$

This final spectrum shows higher stability compared to the result of the iteration method, while keeping similar statistical fluctuations to the raw spectrum. [5]

## 4 Fully Bayesian Unfolding

Bayes theorem succinctly describes what we are asking for in the problem of unfolding, showing the relation between the truth spectrum  $\mathbf{T}$ , and the data we have obtained  $\mathbf{D}$ .

$$P(\mathbf{T}|\mathbf{D}) \propto L(\mathbf{D}|\mathbf{T}) \cdot P(\mathbf{T}) \quad (7)$$

The expected truth spectrum  $T$  and the raw spectrum  $D$  are binned with  $N_t$  and  $N_r$  bins, respectively. Each bin in the  $T$  is assigned a prior probability distribution, describing our belief of the number of events expected to be present. We assume the data follows a Poisson distribution, meaning

$$L(\mathbf{D}|\mathbf{T}) = \prod_{r=1}^{N_r} \frac{f_r^{D_r}}{D_r!} e^{-f_r} \quad (8)$$

where

$$f_r = \sum_{t=1}^{N_t} T_t \cdot R_{rt}. \quad (9)$$

Here,  $R_{rt}$  is the element of the response matrix  $R^{N_r \times N_t}$ , corresponding to the probability that an event produced in the truth bin  $t$  is observed in the response bin  $r$ :  $P(r|t)$ . If we wish to include the background, all we have to do is add it to the sum:

$$f_r = \mathbf{B}_r + \sum_{t=1}^{N_t} T_t \cdot R_{rt}. \quad (10)$$

The next step is to employ a sampling scheme of the parameter space, usually a MCMC algorithm, to calculate  $L(\mathbf{D}|\mathbf{T}) \cdot P(\mathbf{T})$  and arrive at a posterior distribution per bin in the expected truth spectrum.

## 4.1 Sampling

There are several sampling methods possible for the problem of unfolding, a common example being Markov Chain Monte Carlo (MCMC) algorithms such as the Metropolis-Hastings algorithm. For the PyFBU-package, a variant of a Hamiltonian Monte Carlo (HMC) Markov Chain Monte Carlo algorithm is the default sampler and the one used in this project. HMC aims to be much more efficient than regular MCMC algorithms by avoiding both sensitivity to correlated parameters and random walk tendencies [9]. A drawback to this is a significant sensitivity to step size and number of steps, requiring manual tuning of these parameters. To circumvent this, Hoffman and Gelman created the No U-turn Sampler (NUTS), a variant of HMC which removes having to specify the number of steps. They also implemented an adaptive step size, meaning no manual tuning is necessary for running NUTS. Furthermore, they observed similar to better performance than other fine-tuned HMC algorithms [9]. The NUTS algorithm is implemented in the PyMC3 package [3] and used by PyFBU.

## 4.2 Posterior inference

Now that the unfolding has been performed, how do we interpret the resulting posterior distribution? While other methods may only return a point value, not necessarily accompanied by the uncertainties, FBU allows us to directly look at the final distribution per bin, and thus observe the result and its corresponding degree of belief. Of course, we are able to quantify these concepts in multiple ways. Here, we take a look at some of the methods of posterior inference.

### 4.2.1 Point estimates

We will consider three different point estimates, the posterior mean, median and mode:

- Posterior mean: The mean of the posterior distribution which minimizes the mean squared error (MSE) [10].
- Posterior median: The median of the posterior distribution which minimizes the expected absolute error [10].
- Posterior mode: The mode of the posterior distribution, also called the Maximum A Posteriori (MAP), which represents the most likely value for the parameter in question. This does not take into account any skewness of the posterior nor the existence of multiple modes of similar magnitudes.

### 4.2.2 Credible intervals

The credible interval is the Bayesian version of the frequentist confidence interval. It depends on the posterior and is defined as any interval that encompasses a certain percent of the posterior density. The difference between confidence and credible intervals is subtle, but not negligible. In the case of frequentist inference, the parameter in question, lets say  $\theta$ , is treated as an unknown, but fixed value. The limits of the confidence interval are treated as random variables. Therefore, a confidence level of 95% means that for 100 repeated experiments, 95 of the confidence intervals will contain  $\theta$ . Note that this does not mean there is a 0.95 probability of finding  $\theta$  in every confidence interval. [11][12]

For bayesian inference however, the random-trait is switched, with credible interval limits being fixed, and  $\theta$  treated as the random variable. The credible interval takes our prior belief into account, while the confidence interval relies only upon the data. A 95% credible interval covers 95% of the posterior and can then be said to contain  $\theta$  with a probability of 0.95. [11][12]

There are many types of credible intervals, the only requirement being that it covers a certain amount of area of the posterior. Some examples of ways of constructing credible intervals are:

- Using the mean as the interval center.
- Making sure the probability of being outside the interval is equal on all sides (equal tailed).
- Making the interval as narrow as possible, the Highest Posterior Density interval (HPD). This will include the most likely values, as well as the mode of the posterior if it is unimodal.

We will be using the HPD interval which, together with the point estimates mentioned above, will give a solid basis of comparison against the true energies of the  $\gamma$ -ray spectrum.

## 5 Synthetic data

We use a synthetic data set, a simplified representation of a physical case, containing three known excited states. This allows us to compare the result of unfolding directly with the true excitation energies, making it possible to evaluate the accuracy towards the actual problem. In the real world, where the true energies are unknown, the closest we could get would be by refolding our estimates on the output from FBU and observing their differences to the detector output. Furthermore, synthetic data leads to easier and more predictable testing, allowing us to not worry about background contributions and other noise, for better or worse.

Despite the convenience, using a simplified case carries the risk of masking any possible weaknesses to data with higher complexities or dimensionalities. Further testing on more advanced data sets is thus necessary to arrive at a more complete picture of the method and its limitations, but as mentioned, using the synthetic data provides a great insight into the method itself and its core functionality.

The raw data shown in Figure 2 has been folded with the response matrix and smoothed with a Gaussian to resemble detector output. The first excited state, the lowest bar in the figure, is projected on the  $E_\gamma$ -axis (Figure 3) and used as the input data to PyFBU. Projections of the second and third excited states are shown in Figure 4 and Figure 5.

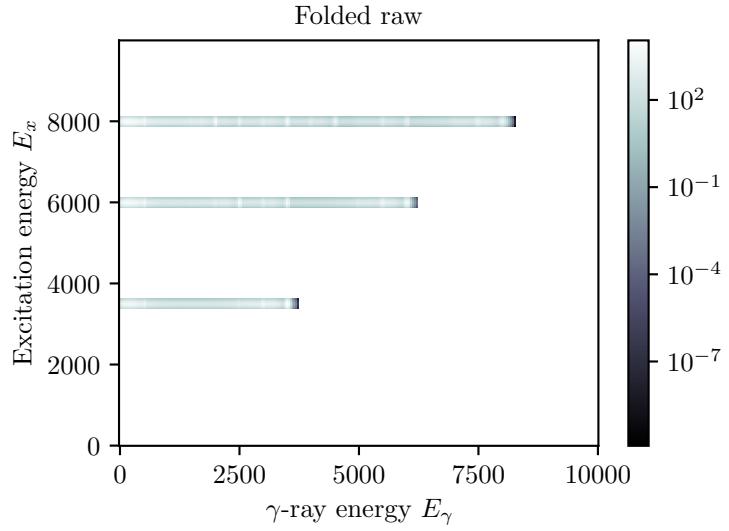


Figure 2: The raw data, representing detector output. It contains three different excited states, each of which have been folded with the response and smoothed with a Gaussian (FWHM = 100 keV) along the  $E_x$ -axis to mimic the detector resolution.

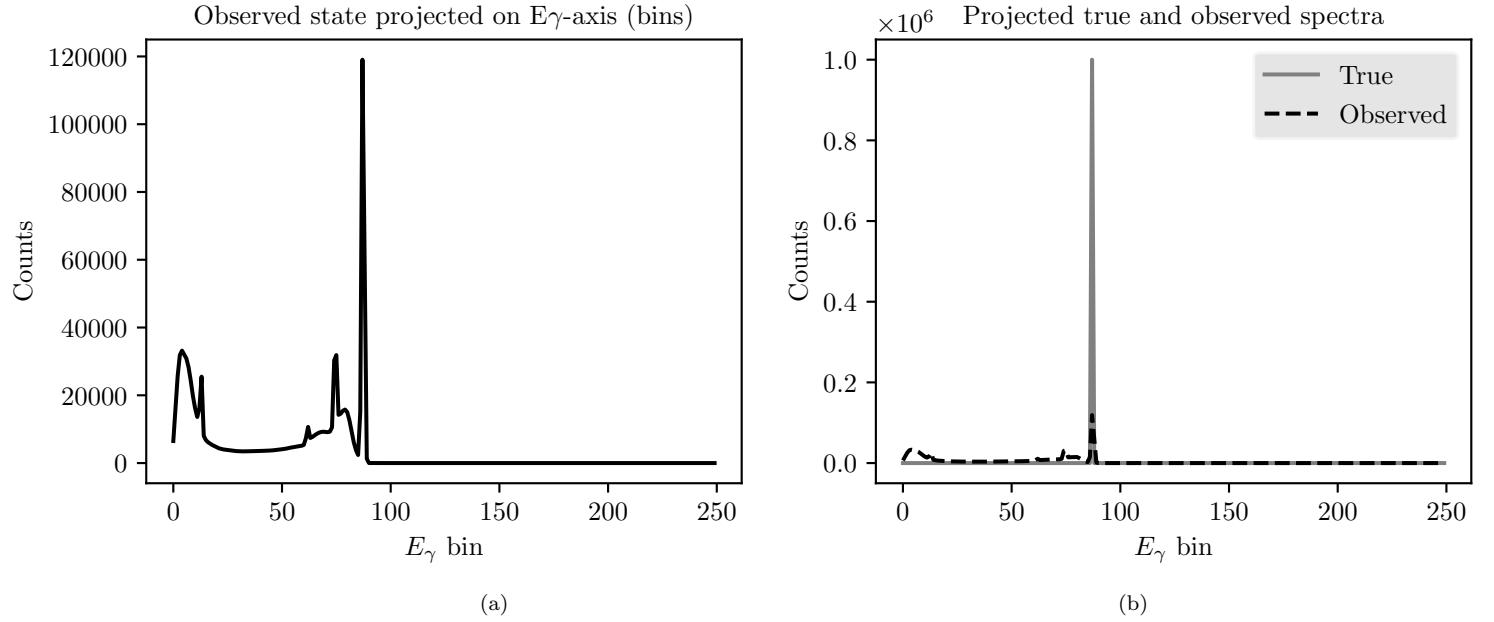
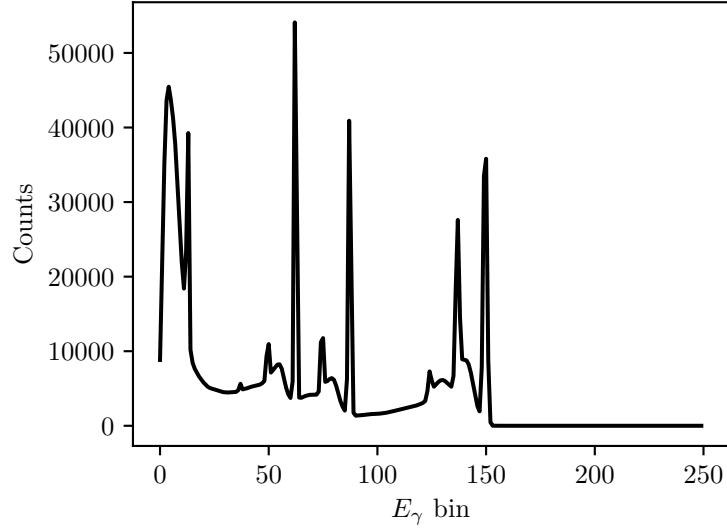
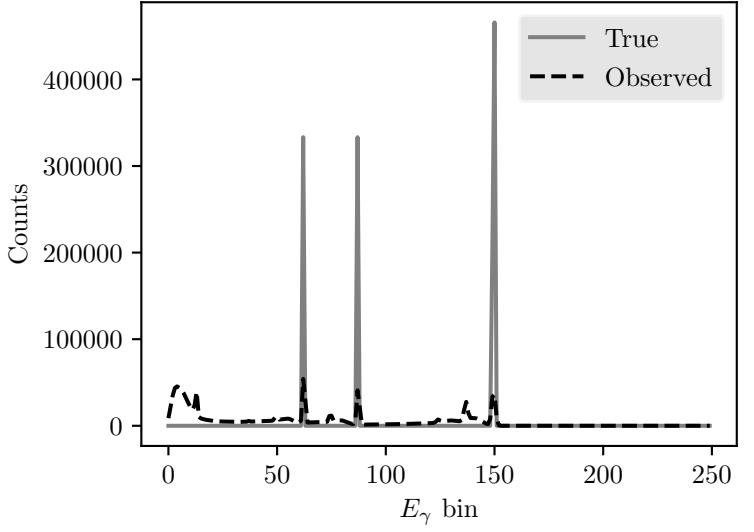


Figure 3: In (a) we see the first excited state projected on the  $E_\gamma$ -axis. In (b) we see the same state compared with the projected true spectrum. The peak of the true spectrum is located much higher than that of the observed spectrum, showing the impact of the response matrix, i.e. the smearing of the detector.

Observed state projected on  $E_\gamma$ -axis (bins)

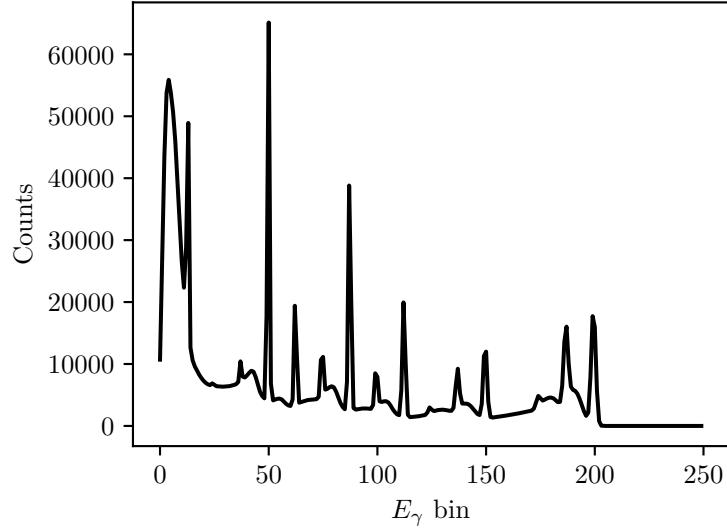
(a)

Projected true and observed spectra



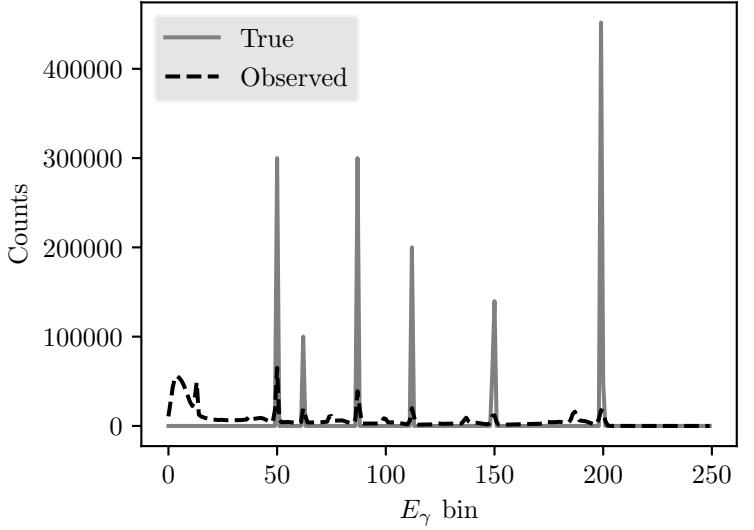
(b)

Figure 4: In (a) we see the second excited state projected on the  $E_\gamma$ -axis. In (b) we see the same state compared with the projected true spectrum. Again, we see the differences in peak height, as well as the introduction of other peaks in the observed data, which do not correspond to any true peak.

Observed state projected on  $E_\gamma$ -axis (bins)

(a)

Projected true and observed spectra



(b)

Figure 5: In (a) we see the third excited state projected on the  $E_\gamma$ -axis. In (b) we see the same state compared with the projected true spectrum. We see the same story as above, the counts from the true spectrum are significantly distributed across the bins when processed by the detector.

## 6 Prior

There are an infinite amount of choices one can make for assigning a prior, depending on what knowledge one has beforehand. In this project, an uninformative prior is chosen, based on the raw observed data. For each bin, a prior interval is constructed with lower bound at zero and upper bound at 10 times the raw data in that bin. On this interval, a flat uniform prior is chosen, meaning we believe all values in this interval are equally likely to be the true value. Stacking all these intervals gives us the prior area shown in Figure 6.

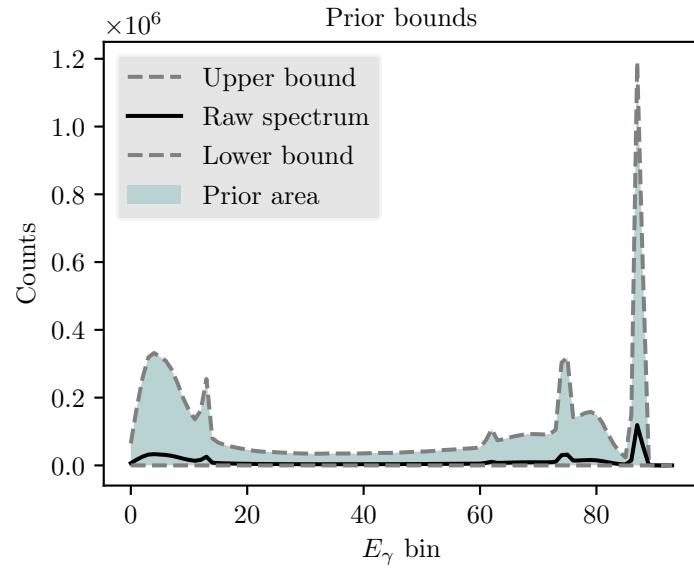


Figure 6: The prior area to be sampled from in FBU along with the raw spectrum. Each bin has an associated uniform prior with boundaries  $[0, \text{raw}[\text{bin}] \times 10]$ .

## 7 Results & Discussion

### 7.1 Posterior distributions

After performing FBU, we end up with a set of posterior samples for each bin in the spectrum. These are then made into histograms, depicting the posterior distributions, and shown below.

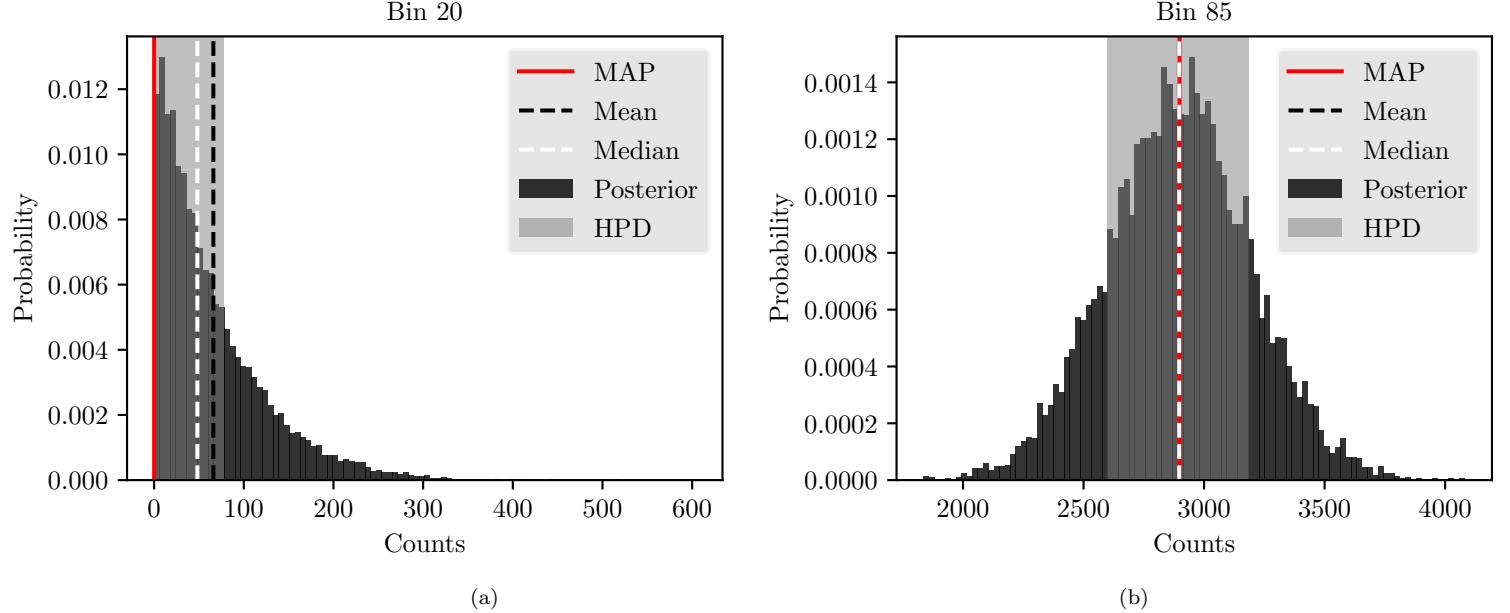


Figure 7: Posterior distributions for 2 bins including the HPD interval and point estimates (MAP, Mean, Median). The posterior in (a) lies close to and has its MAP at 0, meaning we can expect there to be close to 0 counts in this bin. The posterior in (b) is centered below 3000 counts, showing an increase in expected counts as we approach the full energy peak at bin 87. This posterior also exhibits a wider profile and larger HPD interval, and thereby a greater uncertainty.

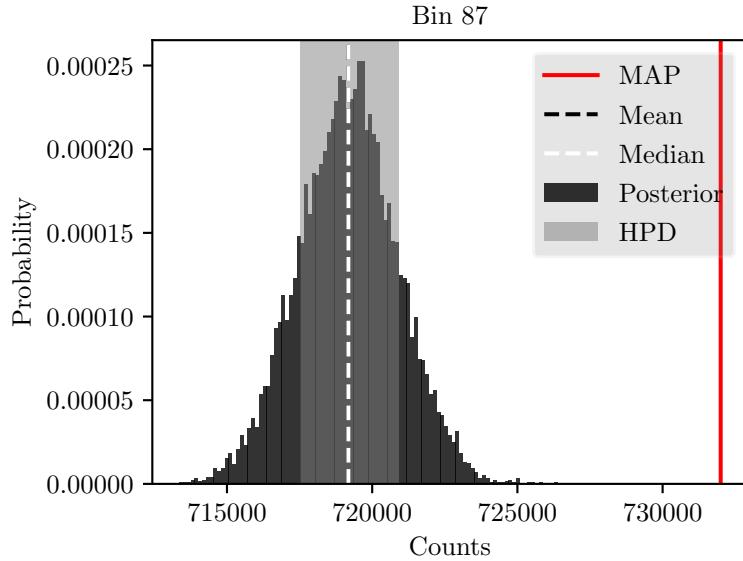


Figure 8: The posterior for bin 87, containing the full energy peak. This is centered on a much larger number of counts than the previous posteriors, showing an expected peak which matches the one in the true spectrum. Interestingly, the MAP lies outside the posterior, which defies its definition as the mode of the posterior distribution. This phenomenon is likely due to the calculation of MAP in PyMC3, which uses an algorithm to approximate the actual MAP. This algorithm is thus likely to be sensitive to the magnitude ( $\sim 7 \cdot 10^5$  here) of the count-values. When this happens, one might be better off directly calculating the maximum of the posterior, but in this case, doing so does not impact the final result significantly. In cases of high dimensionality or complexity, a direct calculation may not be computationally feasible, if even possible. Leaving the MAP as-is helps make the picture clearer regarding the limitations of the methods in these packages, and allows for this discussion to be had.

## 7.2 Final spectra

Taking the estimates above and applying to all bins, we arrive at candidates for the unfolded spectrum, which can be directly compared with the true data. A second comparison is also made for the unfolded result produced by the Folding Iteration Method in the OMpy library. Both comparisons are found in Figure 9, with accompanying error metrics in Table 1.

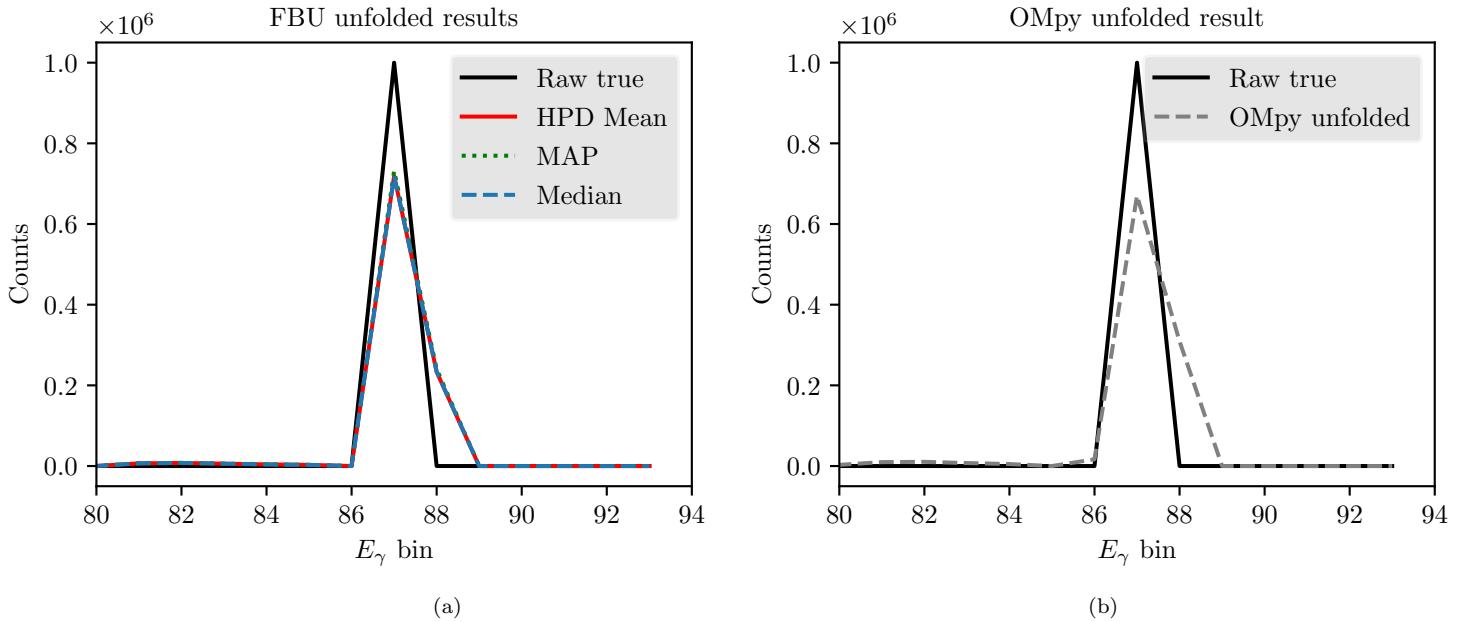


Figure 9: The results of unfolding from (a) FBU and (b) the Folding Iteration Method in the OMpy library. The plots are zoomed to better show the differences about the full energy peak. In (a) we have included the per-bin MAP, Median and Mean of the HPD interval and observe a close match between the point estimates. In (b), the peak of the unfolded spectrum is slightly lower than those of FBU.

Table 1: Evaluations for unfolded results (rounded). Both methods have a mean absolute error of a few thousands counts, FBU being slightly better than the Folding Iteration Method (OMpy). Compared to the full energy peak, these are quite small errors. In areas we expect counts to be 0 however, this can be a drastic difference. The scale differences makes it harder to interpret the mean absolute error as good or bad. The  $R^2$ -score is based on relative difference, painting a clearer picture of the accuracy of each method. Here, FBU comes closer to the ideal value of 1, showing overall better accuracy.

	MAP	Median	HPD Mean	OMpy
Mean absolute error	5702	5960	5953	7787
$R^2$ -score	0.87	0.86	0.86	0.79

Another interesting comparison can be made by refolding the results above and comparing with the folded true data, aka the observed raw spectrum. This is shown in Figure 10 and the error metrics are shown in Table 2.

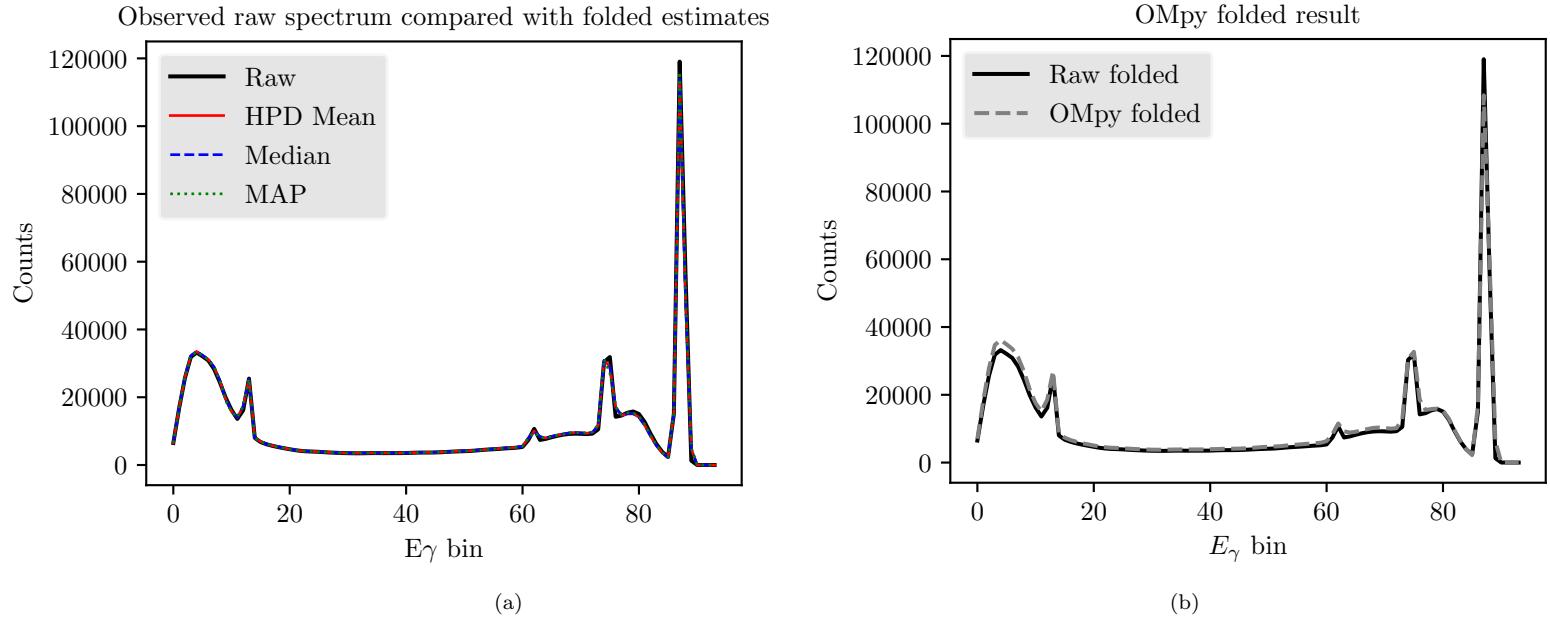


Figure 10: Comparison between the refolded results from above, this time showing all bins. A close match is observed for both methods with a slight edge to FBU.

Table 2: Evaluations for refolded results (rounded). Again, FBU scores slightly but consistently better on both metrics. Both methods see an increase in accuracy after refolding, due to the impact of the response matrix.

	MAP	Median	HPD Mean	OMPy
<b>Mean absolute error</b>	408	440	433	1003
<b>R<sup>2</sup>-score</b>	0.995	0.994	0.994	0.987

## 8 Conclusion

This project has aimed to use the method of Fully Bayesian Unfolding and compare its results with the method most commonly in use by the nuclear physics group at UiO, the Folding Iteration Method. The two methods were found to perform quite similarly on the case investigated, the first excited state in the synthetic data set. Using plots and error metrics (mean absolute error and the  $R^2$ -score), FBU is found to perform slightly better than its counterpart for both unfolded and refolded spectra. In addition, FBU supplies a posterior distribution for each bin, allowing for both direct observation of degree-of-belief as well as calculation of credible intervals and point estimates. In conclusion, FBU presents a strong case for the unfolding of  $\gamma$ -ray spectra. By not only matching, but slightly beating the performance of the standard method, as well as providing the mentioned flexibility of the posterior distribution, further exploration of FBU may prove very exciting. As mentioned before, the simplified data may not expose potential weaknesses in the method, and thus further testing on higher complexities, as well as real experimental data, is recommended.

## Bibliography

1. Midtbø, J. E., Zeiser, F., Lima, E., Ingeberg, V. W. & RK, M. *oslocyclotronlab/ompy: OmPy v1.1.0* version v1.1.0. June 2020. <https://doi.org/10.5281/zenodo.3898281>.
2. Gerbaudo, D., Helsens, C. & Rubbo, F. *PyFBU* <https://github.com/pyFBU/fbu>.
3. Salvatier, J., Wiecki, T. V. & Fonnesbeck, C. Probabilistic programming in Python using PyMC3. *PeerJ Computer Science* **2**, e55 (2016).
4. Choudalakis, G. *Fully Bayesian Unfolding* 2012. arXiv: 1201.4612 [physics.data-an].
5. Guttormsen, M., Tveter, T., Bergholt, L., Ingebretsen, F. & Rekstad, J. The unfolding of continuum  $\gamma$ -ray spectra. *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment* **374**, 371–376. ISSN: 0168-9002 (1996).
6. Cowan, G. *A survey of unfolding methods for particle physics* in *Prepared for Conference on Advanced Statistical Techniques in Particle Physics, Durham, England* (2002), 18–22.
7. Zeiser, F. & Tveten, G. M. *oslocyclotronlab/OCL\_GEANT4: Geant4 model of OSCAR* version v1.0.3. Aug. 2018. <https://doi.org/10.5281/zenodo.1339347>.
8. Zeiser, F. et al. *The energy response of the Oslo Scintillator Array OSCAR* 2020. arXiv: 2008.06240 [physics.ins-det].
9. Hoffman, M. D. & Gelman, A. *The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo* 2011. arXiv: 1111.4246 [stat.CO].
10. Jaynes, E. T. *Probability Theory: The Logic of Science* (ed Bretthorst, G. L.) (Cambridge University Press, 2003).
11. Lee, P. M. *Bayesian statistics : an introduction* eng. Chichester, West Sussex ; 2012.
12. Edwards, W., Lindman, H. & Savage, L. J. Bayesian statistical inference for psychological research. *Psychological Review* **70**. Publisher: US: American Psychological Association, 193. ISSN: 1939-1471. <http://psycnet.apa.org/fulltext/1964-00040-001.pdf> (1964).

# Appendices

## A Implementation

All code for this project is found at the following GitHub repository:  
<https://github.com/jensbd/Special-Syllabus>