MAP-based Uncertainty Quantification

Tobías Liaudat, et al.

May 24, 2023

Abstract

Abstract...

Contents

1 Variational algorithms

- Optimisation problem we are trying to tackle: Equation 1.
- Fixed point equation of the Forward-backward splitting (FB): Equation 2.
- Definition of the proximal operator: Equation 3.

$$x^* = \arg\min_{x\in\Sigma} \underbrace{[f(x) + g(x)]}_{h(x)}, \tag{1}$$

$$x = \text{Prox}_{\gamma g}\left[x - \gamma \nabla f(x)\right] , \tag{2}$$

$$\text{Prox}_{\gamma g}(y) = \arg\min_{x\in\Sigma} \left[\gamma g(x) + \frac{1}{2}\|x - y\|_2^2\right] , \tag{3}$$

1.1 Optimisation

The FB implementation shown in Algorithm 1 follows the relaxation from Combettes et al. [CW05], which is why it differs from the classic formulation from Equation 2. The difference can be seen in the application of the prox in line 6.

Algorithm 1 Forward-Backward (FB) [CW05]

Fix $\varepsilon \in]0, \min\{1, 1/\beta\}[$

for $n = 0$ to $n_{\max}$ **do**
    $\gamma_n \in [\varepsilon, 2/\beta - \varepsilon]$
    $y_n = x_n - \gamma_n \nabla f(x_n)$
    $\lambda_n \in [\varepsilon, 1]$
    $x_{n+1} = x_n + \lambda_n(\mathrm{Prox}_{\gamma_n g}(y_n) - x_n)$
    if Converged **then**
        Break

2 Convex Ridge Regulariser

The convex ridge regulariser (CRR) has been recently introduced in Goujon et al. [Gou+22]. The CRR has three main features that make it very relevant to our problem, it is *learned* (or *data-driven*), it is *convex*, and its potential (or cost) is *explicit*. This last feature means that for a given image we can compute the potential of the regulariser, which is of particular importance for our uncertainty quantification techniques.

The training of the regulariser R_θ has two main hyperparameters, which are the noise level σ used to corrupt images during the training, and the amount of denoising steps used t (from the t-step denoiser seen in Goujon et al. [Gou+22]). Once the model is trained, we can use it as a proximal denoiser and use it to solve Equation 1.

Replacing the general regulariser term g from Equation 1 with the CRR we obtain

$$x^* = \arg\min_{x \in \Sigma} \left[f(x) + \underbrace{\lambda/\mu\, R_\theta(\mu x)}_{g(x)} \right], \tag{4}$$

where the $\lambda \in \mathbb{R}^+$ hyperparameter is interpreted as the regularisation strength, and $\mu \in \mathbb{R}^+$ as a tunable scaling parameter.

For optimisation and UQ, we will need to use $g(x)$, $\nabla g(x)$ and the Lipschitz constant of g denoted L_g. These variables can be computed as follows:

- $g(x) \to \lambda/\mu\, R_\theta(\mu x)$ where $R_\theta(x)$ = `model.cost(x)`
- $\nabla g(x) \to \lambda \nabla R_\theta(\mu x)$ where $\nabla R_\theta(x)$ = `model(x)` = `model.grad(x)`
- $L_g \to \lambda\mu L_{R_\theta}$ where L_{R_θ} = `model.precise_lipschitz_bound()`

As the regulariser g is smooth, we do not need to use a proximal algorithm to solve our optimisation problem. Recall that the use of proximal algorithms is to optimise a non-smooth function, which is often the case when using Total Variation (TV) or ℓ_1 penalties with wavelet representation. Consequently, a simple gradient descent method is enough if the CRR is used as the regulariser. Nevertheless, to leverage the existing code and the proximal algorithms implemented, we can reformulate the CRR regulariser to act as a proximal operator. In this case, we would define the prox operator as

$$\mathrm{Prox}_{\gamma g}(x) = x - \gamma g(x) = x - \gamma\lambda\nabla R_\theta(\mu x), \tag{5}$$

where γ is a constant multiplying the regulariser g (possibly the optimisation step size).

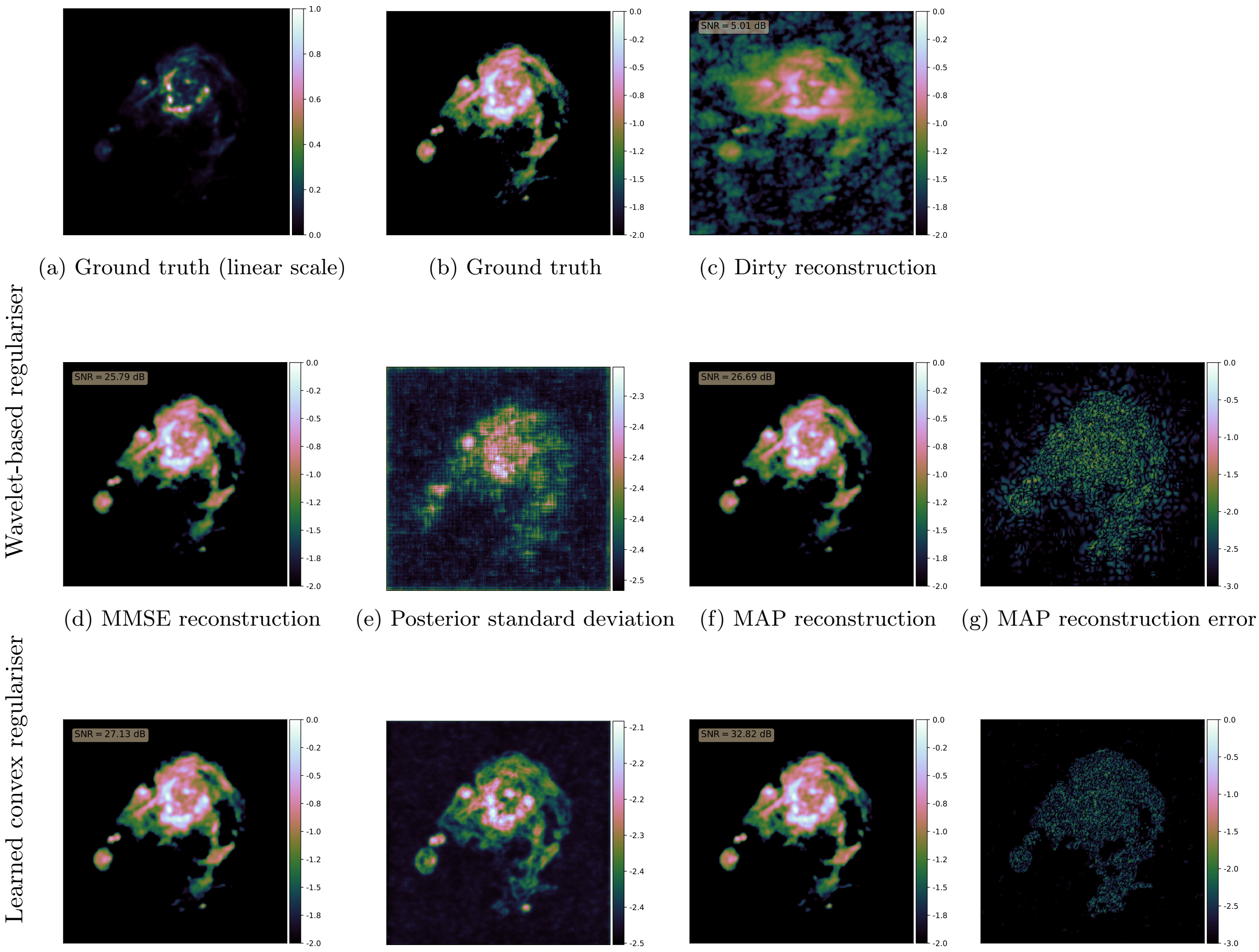
2.1 Examples

Examples of optimisation are shown in Figure 1.

3 MAP-based uncertainty quantification

Figure 4 shows an overview of the MAP-based uncertainty quantification.

Once we have obtained the MAP estimation, x_{MAP} through the optimisation procedure we can go ahead and compute an approximation of the High Posterior Density (HPD) region.



(a) Ground truth (linear scale) (b) Ground truth (c) Dirty reconstruction

(d) MMSE reconstruction (e) Posterior standard deviation (f) MAP reconstruction (g) MAP reconstruction error

(d) MMSE reconstruction (e) Posterior standard deviation (f) MAP reconstruction (g) MAP reconstruction error

Figure 1: Image reconstructions for M31.

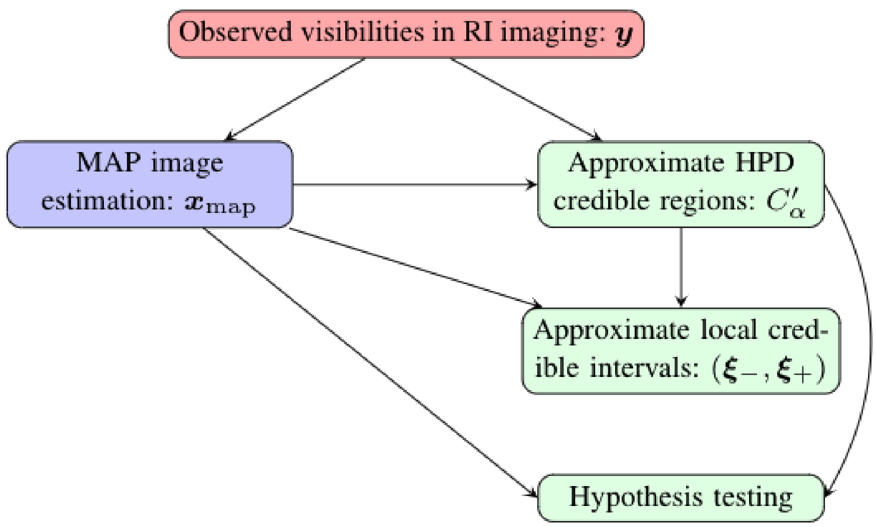
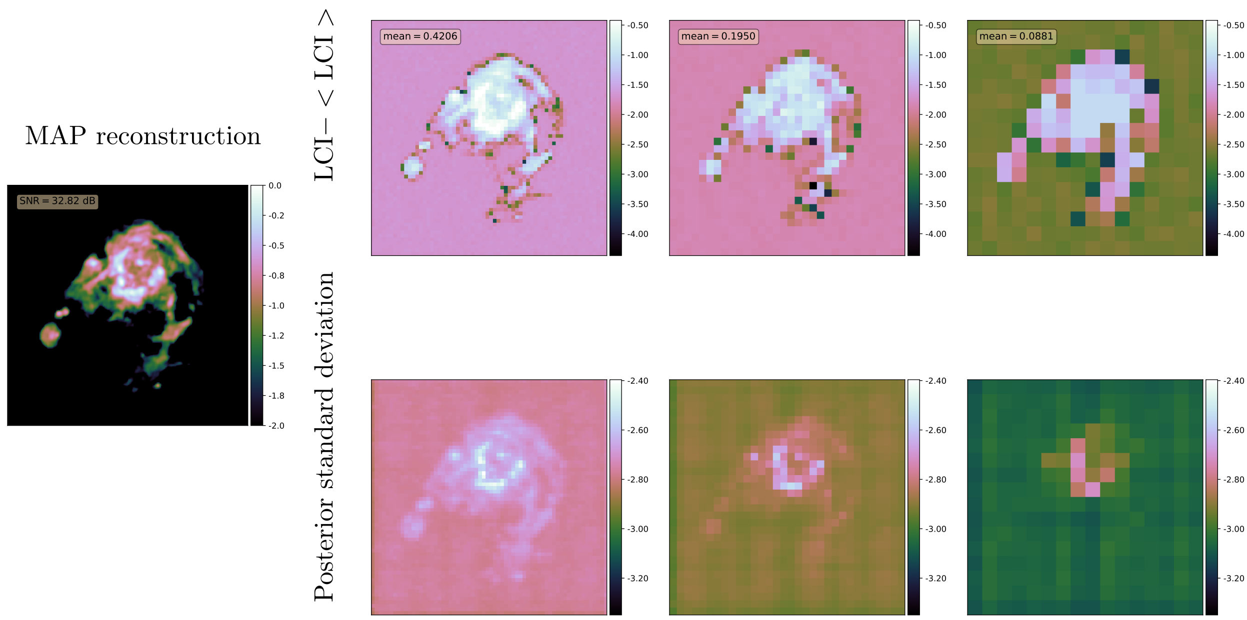


Figure 2: Overview of MAP-based UQ from Cai et al. [CPM18].



(a) Pixel size 4×4 (b) Pixel size 8×8 (c) Pixel size 16×16

Figure 3: Pixel-based uncertainty quantification for M31 with the learned convex regulariser.

The HPD region is defined as

$$C'_\alpha = \{x : f(x) + g(x) \leq \gamma'_\alpha\} \,, \tag{6}$$

where $\gamma'_\alpha \in \mathbb{R}$ is a threshold value computed as

$$\gamma'_\alpha = f(x_{\mathrm{MAP}}) + g(x_{\mathrm{MAP}}) + \tau_\alpha \sqrt{N} + N \,, \tag{7}$$

where N is the number of dimensions of the image, e.g., if $x \in \mathbb{R}^{p \times p}$ then $N = p^2$, and $\tau_\alpha \in \mathbb{R}$ is a constant computed as

$$\tau_\alpha = \sqrt{16 \log(3/\alpha)} \,, \tag{8}$$

where α defines the credible level as $100(1-\alpha)\%$. We usually use 0.01 or 0.05 as values of α, which gives us credible levels of 99% and 95%, respectively.

3.0.1 Example

3.1 Pixel-based UQ maps

Some useful references where the pixel-based UQ maps have already been defined and used are [CPM18; Pri+19].

We want to generate

We use some type of root-finding algorithm, used to find the zero of a 1D function. At the moment we are using the bisection method. There are other methods with better properties, but we have not implemented them yet. An example is the Interpolate Truncate and Project (ITP) method.

The pseudocode of the bisection method is presented in algorithm 2.

3.2 Hypothesis tests

The hypothesis test procedure is presented in Figure 4, and presented in more detail in Cai et al. [CPM18] and Price et al. [Pri+21].

Algorithm 2 Bisection method

Endpoint values: a and b
Function to bisect: f

for $n = 0$ to $n_{\max}$ **do**
    $c \leftarrow (a + b)/2$ — Compute midpoint
    if Tolerance reached **then**
        Break
    if $\operatorname{sign}(f(c)) == \operatorname{sign}(f(a))$ **then**
        $a \leftarrow c$
    else
        $a \leftarrow b$
Output c

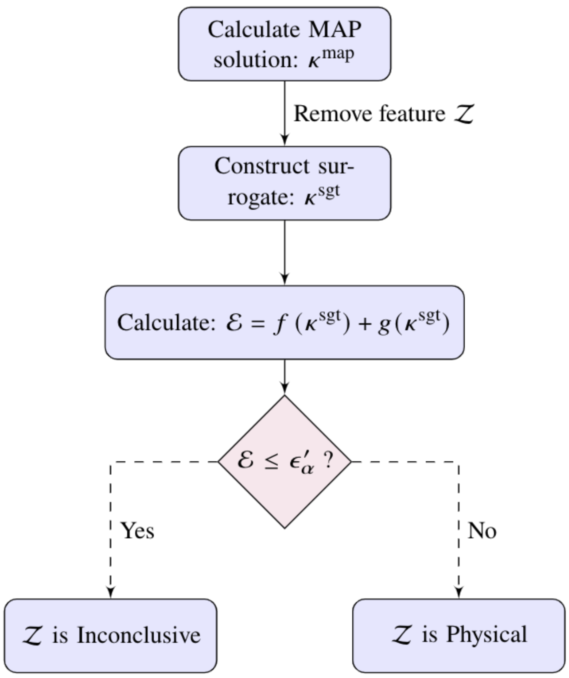


Figure 4: Hypothesis test flow from Price et al. [Pri+19].

References

[CPM18] X. Cai, M. Pereyra, and J. D. McEwen. "Uncertainty quantification for radio interferometric imaging: II. MAP estimation". In: *Monthly Notices of the Royal Astronomical Society* 480.3 (Aug. 2018), pp. 4170–4182. DOI: 10.1093/mnras/sty2015. eprint: https://academic.oup.com/mnras/article-pdf/480/3/4170/25643815/sty2015.pdf.

[CW05] P. L. Combettes and V. R. Wajs. "Signal Recovery by Proximal Forward-Backward Splitting". In: *Multiscale Modeling & Simulation* 4.4 (2005), pp. 1168–1200. DOI: 10.1137/050626090. eprint: https://doi.org/10.1137/050626090.

[Gou+22] A. Goujon, S. Neumayer, P. Bohra, S. Ducotterd, and M. Unser. "A Neural-Network-Based Convex Regularizer for Image Reconstruction". In: *arXiv e-prints*, arXiv:2211.12461 (Nov. 2022), arXiv:2211.12461. DOI: 10.48550/arXiv.2211.12461. arXiv: 2211.12461 `[eess.IV]`.

[Pri+19] M. A. Price, X. Cai, J. D. McEwen, M. Pereyra, T. D. Kitching, and L. D. E. S. Collaboration. "Sparse Bayesian mass mapping with uncertainties: local credible intervals". In: *Monthly Notices of the Royal Astronomical Society* 492.1 (Dec. 2019), pp. 394–404. DOI: 10.1093/mnras/stz3453. eprint: https://academic.oup.com/mnras/article-pdf/492/1/394/31694512/stz3453.pdf.

[Pri+21] M. A. Price, J. D. McEwen, X. Cai, T. D. Kitching, C. G. R. Wallis, and (. the LSST Dark Energy Science Collaboration). "Sparse Bayesian mass mapping with uncertainties: hypothesis testing of structure". In: *Monthly Notices of the Royal Astronomical Society* 506.3 (July 2021), pp. 3678–3690. DOI: 10.1093/mnras/stab1983. eprint: https://academic.oup.com/mnras/article-pdf/506/3/3678/42496246/stab1983.pdf.