Article

Regularization, Bayesian Inference and Machine Learning methods for Inverse Problems†

Ali Mohammad-Djafari [1,2] orcid number:0000-0003-0678-7759

[1] International Science Consulting and Training (ISCT), 91440 Bures sur Yvette, France; djafari@free.fr
[2] Laboratoire des Signaux et Systèmes, CNRS, CentraleSupélec-Univ Paris Saclay, 91192 Gif-sur-Yvette, France; djafari@lss.supelec.fr
* Ali Mohammad-Djafari
† Presented as a tutorial at the 40th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, GTU, Austria, July 4, 2021.

Abstract: Classical methods for inverse problems are mainly based on regularization theory. In particular those which are based on optimization of a criterion with two parts: a data-model matching and a regularization term. Different choices for these two terms and great number of optimization algorithms have been proposed. When these two terms are distance or divergence measures, they can have a Bayesian Maximum A Posteriori (MAP) interpretation where these two terms correspond, respectively, to the likelihood and prior probability models.
The Bayesian approach gives more flexibility in choosing these terms, and in particular, the prior term via hierarchical models and hidden variables. However, the Bayesian computations can become very heavy computationally. The Machine Learning (ML) methods such as classification, clustering, segmentation and regression, based on Neural Networks (NN) and in particular Convolutional NN and Deep NN, Physics-Informed Neural Networks, etc. can become helpful to obtain approximate, but good quality and practical solutions to inverse problems.
In this tutorial, particular examples of image denoising, image restoration and Computed Tomography (CT) image reconstruction will illustrate this cooperation between ML and Inversion.

Keywords: Inverse problems; Regularization; Bayesian inference; Machine Learning; Artificial Intelligence; Gauss-Markov-Potts; Variational Bayesian Approach (VBA); Physics Informed ML

---

1. Introduction

Inverse problems arise in almost any scientific and engineering application. In fact, whenever we want to infer a quantity which is not directly measured. Noting the unknown quantity f and the measurement data g, we may have a mathematical relation between them: $g = \mathcal{H}(f)$ where f can be a 1D function (signal), 2D function (Image), 3D or more (e.g. video, hyperspectral images, etc.). $\mathcal{H}$ is a mathematical model, called forward operator and g can also be 1D, 2D, 3D or more function. In practice, we may only have discrete values of it available and for this reason the inverse problem which is inferring f from this limited data is an ill-posed problem. When discretized, we may write the relations between them as $g = \boldsymbol{H}(f) + \boldsymbol{\epsilon}$ where g contains all the data, f all the discretized representation of the unknown quantity and $\boldsymbol{H}$ a multidimensional operator connecting them. Finally, $\boldsymbol{\epsilon}$ represents all the errors of discretisation and measurement uncertainties.

Handling inverse problems, even in the discretized version linear model $g = \boldsymbol{H}f + \boldsymbol{\epsilon}$ is not easy, at least for two reasons: one is the ill-conditioning of the matrix $\boldsymbol{H}$ and its great dimensions; second is accounting for the errors.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

Classical methods for inverse problems are mainly based on regularization theory. In particular those which are based on optimization of a criterion with two parts: a data-model matching part $\Delta_1(\boldsymbol{g}, \boldsymbol{H}\boldsymbol{f})$ and a regularization term $\Delta_2(\boldsymbol{f}, \boldsymbol{f}_0)$ with a balancing term between them: $J(\boldsymbol{f}) = \Delta_1(\boldsymbol{g}, \boldsymbol{H}\boldsymbol{f}) + \lambda\Delta_2(\boldsymbol{f}, \boldsymbol{f}0)$ where Δ_1 and Δ_2 are two distances (L2, L1, etc.) or divergence measure such as Kullback-Leibler (KL) or any other divergence. $\boldsymbol{f}0$ can be equal to zero or any other prior default solution. Different choices for these two terms and great number of optimization algorithms have been proposed with success in very diversified domains and applications [1–5].

Bayesian inference based methods also had great success for handling inverse problems, in particular, when the data are noisy, uncertain, some missing and some outliers and where there is a need to account and to quantify uncertainties. In fact, when the two terms of the regularization methods are distance or divergence measures, they can have a Bayesian Maximum A Posteriori (MAP) interpretation where these two terms correspond, respectively, to the likelihood and prior probability models. Indeed, the Bayesian approach gives more flexibility in choosing these terms, and in particular, the prior term via hierarchical models and hidden variables [6–9] However, the Bayesian computations can become very heavy computationally. The Machine Learning (ML) methods such as classification, clustering, segmentation and regression, based on Neural Networks (NN) and in particular Convolutional NN and Deep NN, Physics-Informed Neural Networks, etc. can become helpful to obtain approximate, but good quality and practical solutions to inverse problems [10–13].

However, even if in many domains of Machine Learning such as classification and clustering these methods have shown success, their use in real scientific problems are limited. The main reasons are twofold: First, the users of these tools can not explain the reasons when they are successful and when they are not. The second is that, in general, these tools can not quantify the remaining uncertainties.

Model based and Bayesian inference approach have been very successful in linear inverse problems. However, adjusting the hyper parameters is complex and the cost of the computation is high. The Convolutional Neural Networks (CNN) and Deep Learning (DL) tools can be useful for pushing farther these limits. At the other side, the Model based methods can be helpful for the selection of the structure of CNN and DL which are crucial in ML success. In this tutorial paper, first an overview and a survey of the aforementioned methods are presented and the possible interactions between them are explored [14,15].

The rest of the paper is organized as follows: First a survey of inverse problems examples, analytical inversion methods, Generalized inversion and regularization methods and finally the Bayesian inference methods, is presented. Then, a discussion on the process and final objectives of imaging systems, for example in health survey systems, going from the data acquisition to image reconstruction, its segmentation, feature extraction and finally its interpretation and usage is presented to prepare the more advanced part of this tutorial. For example, the Bayesian joint reconstruction and segmentation using Gauss-Markov-Potts prior modelling [16–19]. In the third part, first an introduction to Machine Learning (ML) tools and process and basic notions and notations on Neural Networks (NN) is given. The last part is related to the relations between all these methods via forward modeling, identification, learning and inversion. These relations are shown via a few simple examples and then we discuss about the fully learned and Physics informed partially learned ML methods for inverse problems.

After mentioning some successful case studies in which the ML tools have been successful [20–24], [25–30], [31–35], we arrive at the main conclusions of this paper and the future of the possible interactions between Model based and Machine Learning tools. We conclude by mentioning the Open problems and challenges in both classical, model based and the ML tool.

2. Inverse problems example

Inverse problems arise almost every where in science and engineering, every where we want to infer on an unknown quantity $\boldsymbol{f}$ which is not accessible (observable) directly. We have only access to another observable quantity $\boldsymbol{g}$ which is related to it via a linear or non linear relation $\boldsymbol{H}$ [36–38].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

As you could see, I am going to use a color code: red for unknown quantities and blue for observed or assumed known quantities. The Forward operator linking the two quantities is noted $\boldsymbol{H}$. In general the forward operator is well-posed, but the inverse problem is ill-posed. This means that either the classical inverse operator does not exist (existence), or we can define many generalized inverse operators, so many solutions to the problem can be defined (uniqueness), or even if we can define an inverse operator, it may be unstable (stability) [39].

Let us mention a few examples of common inverse problems here.

2.1. Image restoration

Any photographic system (camera, microscope or telescope) has limited field of view and limited resolution. If we note by $rf(x,y)$ the original image and by the ${}_bg(x,y)$ the observed image and if we assume a linear and space invariant operator between them, then the forward relation can be written as a convolution operator:

$$g(x',y') = \int f(x,y)h(x'-x,y'-y)\,\mathrm{d}x\,\mathrm{d}y \tag{1}$$

where $h(x,y)$ represents the point spread function (psf) of the imaging system.

Many examples can be given [40,41]. In Figure 1, two synthetic examples are shown.

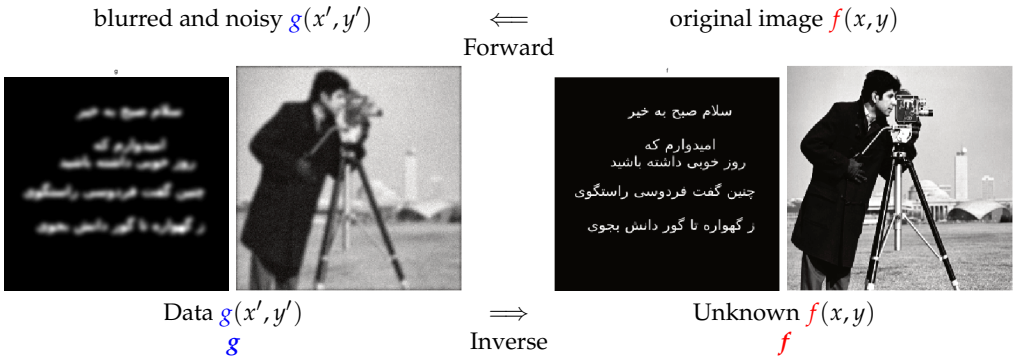

Figure 1. Forward and inverse problems in image restoration. Forward operation is a *convolution* and the inverse operation is called *deconvolution*.

2.2. X ray Computed Tomography

In X-ray Computed Tomography (CT), the relation between the data and the object can be modeled via the Radon Transform:

$$g(r,\phi) = \int f(x,y)\delta(r - x\cos\phi - y\sin\phi)\,\mathrm{d}x\,\mathrm{d}y \tag{2}$$

where $g(r,\phi)$ represents the line integrals over the lines of angles ϕ of the object function $f(x,y)$. Forward operation is called *projection* and the inversion process is called *image reconstruction*. In Figure 2, one synthetic examples is shown.

2.3. Acoustical imaging

Acoustic source localization in acoustical imaging can also be considered as an inverse problems, where the positions of acoustical sources have to estimated from the signal received by the microphone arrays. Each microphone receives the sum of the delayed sources sounds [42].

In Figure 3, one synthetic examples is shown to explain the main idea.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

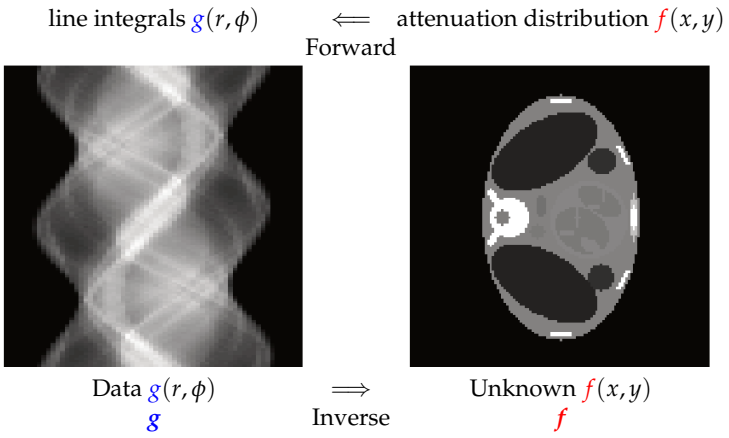

Figure 2. Forward and Inverse problems in Computed Tomography. The horizontal axis on the left is r, the vertical is ϕ and the values of $g(r,\phi)$ are presented as the gray levels. On the right the object section $f(x,y)$ is presented. Forward operation is called *projection* and the inversion process is called *image reconstruction*.

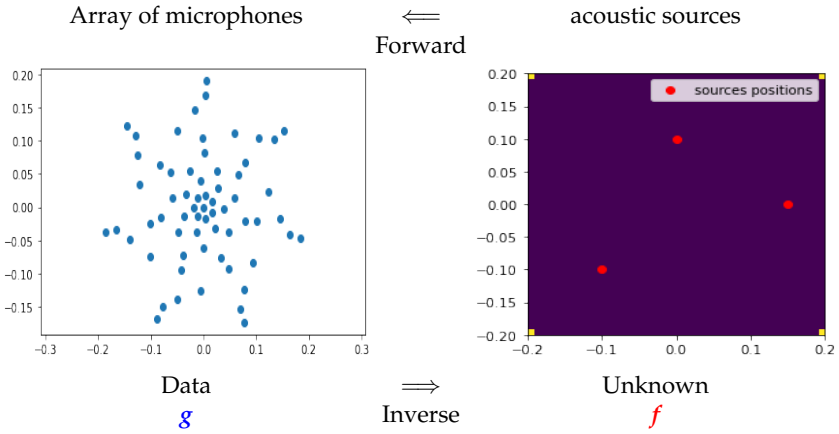

Figure 3. Forward and inverse problems in acoustical imaging. Each microphone receives the sum of the delayed sources sounds. The inverse problem is to estimate the sources distribution from the received signals by the microphones array.

2.4. Microwave imaging for Breast cancer detection

In microwave imaging, the body is illuminated by microwaves. As the elecrical properties (conductivity and permeability) of the healthy and tumor tissues are different, their corresponding induced sources are different. These differences can be measured via the electrodes outside of the breast. The inverse problem, in this case, consists in estimation these induced sources or even directly the distribution of the conductivity and permeability inside the breast. Looking to such images, the tumor area can be visualized [43,44].

2.5. Brain imaging

In Brain imaging, the electrical activity of the neurons inside the brain brain are propagated and can be measured at the surface of the sculpt via the electrodes fixed on it. These signals are called Electroencephalography (EEG). It is also possible to measure the magnetic field created by this activity. This time the signals are called (MEG). In both cases, the inversion process consists in estimating the distribution of the brain activity from the measured signals.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

Many other imaging systems to see inside the human body or inside any industrial object in Non destructive testing (NDT) applications exist. Here, a few of them have been illustrated. We can just mention a few more: Magnetic resonance imaging (MRI), Ultrasound imaging such as echography, Positron emission tomography (PET), Single emission computed tomography (SPECT), Electrical impedence tomography, Eddy current tomography [45].

3. Classification of Inverse problems methods

Inverse problems methods can be classified in the following categories:

- Analytical inversion methods
- Generalized inversion approach
- Regularisation methods
- Bayesian inference methods

In the first category, the main idea is to recognize the forward operator as one of the well known mathematical invertible operator and thus to use the appropriate inversion operator. Typical examples are Fourier Transform (FT) and Radon Transform (RT). In the second category, the notion of Generalized inversion is used. The corresponding methods are either based on Singular value decomposition (SVD) or the iterative projection based algorithms. The regularization methods are mainly based on the optimization of a criterion, often made in two parts: Data-model adequation and the regularization with a regularization parameter. Finally the Bayesian inference approach, which I consider to be the most general and complete has all the necessary tools to go beyond the regularization methods.

4. Analytical Methods

Figure 4 shows the main idea behind the analytical methods via two classical cases of image deconvolution and X ray image reconstruction. In the first case, as the forward model is a Fourier Transform (FT), the operation consists in going to the Fourier domain, doing Inverse Filtering and coming back. In the second case, the forward model is Radon transform (RT). Using the relation between FT and RT (Fourier slice theorem), the analytical inversion process becomes: i) for each angle ϕ, compute the 1D FT of $bg_\phi(r) = bg(r,\phi)$, ii) relate it to the 2D FT of $f(x,y)$ via the Fourier slice theorem and interpolate to obtain the full 2D FT of $f(x,y)$; and iii) Compute 2D IFT to obtain $f(x,y)$ [46,47].

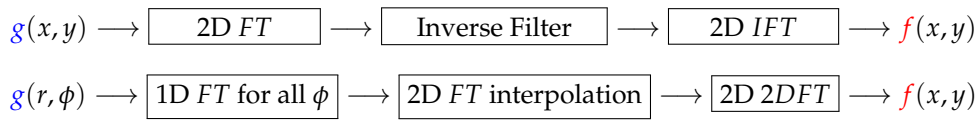

Figure 4. Transform based analytical methods. Two examples are given: Image deconvolution by inverse filtering and image reconstruction in CT by using the relation between RT and FT.

5. Generalized inversion approach

In this approach, the main idea is based on the fact that the forward operator is in general a singular one. This means there are many possible solutions to the inverse problem. In this approach there are mainly two categories of methods. The first are based on Singular values decomposition (SVD). The second is based on optimization of a criterion such as the Least Squares (LS). In both, the main idea is to define a set of possible solutions, called Generalized inverse solutions:

$$\{\boldsymbol{f}^\dagger : \boldsymbol{H}\boldsymbol{f}^\dagger = \boldsymbol{g}\} \tag{3}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

or pseudo solutions:

$$\{f^\dagger : \|Hf^\dagger - g\|^2 < \epsilon\} \tag{4}$$

Then, between those possible solutions, one tries to define a criterion, such as the minimum norm, to choose a solution. For the linear inverse problems, the corresponding solutions are given by

$$f^\dagger = [H'H]^{-1}H'g = H^\dagger g \quad \text{or} \quad f^\dagger = H'[H'H]^{-1}g = H^\dagger g \tag{5}$$

In great dimensional problems, even if we have these analytical expression, in practice, the solutions are computed by using iterative optimization algorithms, for example to optimize the LS criterion $J(f) = \|Hf - g\|^2$ by a gradient based algorithm:

$$f^{(k+1)} = f^{(k)} + \alpha H'(g - Hf^{(k)}) \tag{6}$$

with a stopping criteria or just after some fixed number of iterations. We will see in the next sections how this can lead to a Deep Learning NN structure.

6. Model Based and Regularization Approach

The model based methods are related to the notions of forward model and inverse problems approach. Figure 5 shows the main idea:

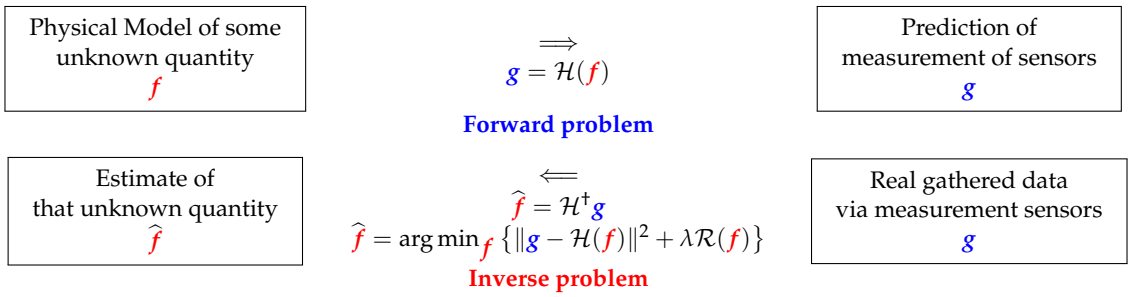

Figure 5. Model based methods: Forward and inverse problems. The solution of the inverse problem is defined either by the generalized inversion or by a regularization method.

Given the forward model $\mathcal{H}$ and the source f, the prediction of the data g can be done, either in a deterministic way: $g = \mathcal{H}(f)$ or via a probabilistic model: $p(g|f, \mathcal{H})$ as we will see in the next section. In the same way, given the forward model $\mathcal{H}$ and the data g, the estimation of the unknown source f can be done either via a deterministic method or probabilistic one. One of the deterministic method is the Generalized inversion: $f = \mathcal{H}^\dagger(g)$. A more general method is the regularization:

$$\widehat{f} = \arg\min_f \{J(f)\} \text{ with } J(f) = \|g - \mathcal{H}(f)\|^2 + \lambda\mathcal{R}(f). \tag{7}$$

As we will see later, the only probabilistic method which can be efficiently used for the inverse problems is the Bayesian approach.

6.1. Regularization Methods

Let consider the discretized linear inverse problem: $g = Hf + \epsilon$, and the regularization criterion

$$J(f) = \frac{1}{2}\|g - Hf\|_2^2 + \lambda R(f). \tag{8}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

The first main issue in such regularization method is the choice of the regularizer. The most common examples are:

$$R(\boldsymbol{f}) = \left\{ \|\boldsymbol{f}\|_2^2,\ \|\boldsymbol{f}\|_\beta^\beta,\ \|\boldsymbol{D}\boldsymbol{f}\|_2^2,\ \|\boldsymbol{D}\boldsymbol{f}\|_\beta^\beta,\ \sum_j \phi([\boldsymbol{D}\boldsymbol{f}]_j) \right\},\ 1 \leq \beta \leq 2 \tag{9}$$

The second main issue in regularization is the the choice of appropriate optimization algorithm. This, mainly depends on the type of the criterion, we have:

- $R(\boldsymbol{f})$ quadratic: Gradient based, Conjugate Gradient algorithms are appropriate.
- $R(\boldsymbol{f})$ non quadratic, but convex and differentiable: Here too the Gradient based and Conjugate Gradient (CG) methods can be used, but there are also great number of convex criterion optimization algorithms.
- $R(\boldsymbol{f})$ convex but non-differentiable: Here, the notion of sub-gradient is used.

Specific cases are:

- L2 or quadratic: $J(\boldsymbol{f}) = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|_2^2 + \lambda\|\boldsymbol{D}\boldsymbol{f}\|_2^2$.

  In this case we have an analytic solution: $\widehat{\boldsymbol{f}} = (\boldsymbol{H}'\boldsymbol{H} + \lambda\boldsymbol{D}'\boldsymbol{D})^{-1}\boldsymbol{H}'\boldsymbol{g}$. However, in practice this analytic solution is not usable in high dimensional problems. In general, as the gradient $\nabla J(\boldsymbol{f}) = -\boldsymbol{H}'(\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}) + 2\lambda\boldsymbol{D}'\boldsymbol{D}\boldsymbol{f}$ can be evaluated analytically, gradient based algorithms are used.
- L1 (TV): convex but not differentiable at zero: $J(\boldsymbol{f}) = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|_2^2 + \lambda\|\boldsymbol{D}\boldsymbol{f}\|_1$.

  The algorithms in this case use the notions of Fenchel conjugate, Dual problem, sub gradient and proximal operator [11,48–50]
- Variable splitting and Augmented Lagrangian

$$(\boldsymbol{f}, \widehat{\boldsymbol{z}}) = \arg\min_{\boldsymbol{f},\boldsymbol{z}} \left\{ \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|_2^2 + \lambda\|\boldsymbol{z}\|_1 + q\|\boldsymbol{z}\|_2^2 \right\} \text{ s.t. } \boldsymbol{z} = \boldsymbol{D}\boldsymbol{f} \tag{10}$$

  A great number of optimization algorithms have been proposed: ADMM, ISTA, FISTA, etc. [1,5, 51].

  Main limitations of deterministic regularization methods are:

- Limited choice of the regularization term. Mainly, we have: a) Smoothness (Tikhonov), b) Sparsity, Piecewise continuous (Total Variation).
- Determination of the regularization parameter. Even if there are some classical methods such as L-Curve and Cross validation, there are still controversial discussions about this.
- Quantification of the uncertainties: This is the main limitation of the deterministic methods, in particular in medical and biological applications where this point is important.

The best possible solution to push further all these limits is the Bayesian approach which has: (a) Many possibilities to choose prior models, (b) possibility of the estimation of the hyper-parameters, and most important (c) accounting for the uncertainties.

7. Bayesian Inference Methods

7.1. Basic idea

The simple case of the Bayes rule is:

$$p(\boldsymbol{f}|\boldsymbol{g}, \mathcal{H}) = \frac{p(\boldsymbol{g}|\boldsymbol{f}, \mathcal{H})\ p(\boldsymbol{f}|\mathcal{H})}{p(\boldsymbol{g}|\mathcal{H})} \text{ where } p(\boldsymbol{g}|\mathcal{H}) = \int p(\boldsymbol{g}|\boldsymbol{f}, \mathcal{H})\ p(\boldsymbol{f}|\mathcal{H})\, \mathrm{d}\boldsymbol{f} \tag{11}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

where $\mathcal{H}$ is a model, $p(\boldsymbol{g}|\boldsymbol{f},\mathcal{H})$ is the likelihood of $\boldsymbol{f}$ in the data through the model, $p(\boldsymbol{f}|\mathcal{H})$ is the prior knowledge about the unknown quantity $\boldsymbol{f}$ and $p(\boldsymbol{f}|\boldsymbol{g},\mathcal{H})$ called the posterior is the result of the combination of the likelihood and prior. The denominator $p(\boldsymbol{g}|\mathcal{H})$, called the evidence, is the overall likelihood of the model in the data $\boldsymbol{g}$.

When there are some hyper parameters, for example the parameters of the likelihood and those of the prior law, which have also to be estimated, we have:

$$p(\boldsymbol{f},\boldsymbol{\theta}|\boldsymbol{g},\mathcal{H}) = \frac{p(\boldsymbol{g}|\boldsymbol{f},\boldsymbol{\theta},\mathcal{H})\ p(\boldsymbol{f}|\boldsymbol{\theta},\mathcal{H})\ p(\boldsymbol{\theta}|\mathcal{H})}{p(\boldsymbol{g}|\mathcal{H})} \text{ where } p(\boldsymbol{g}|\mathcal{H}) = \iint p(\boldsymbol{g}|\boldsymbol{f},\boldsymbol{\theta},\mathcal{H})\ p(\boldsymbol{f}|\boldsymbol{\theta},\mathcal{H})\,\mathrm{d}\boldsymbol{\theta}\,\mathrm{d}\boldsymbol{f} \tag{12}$$

This is called the joint posterior law of all the unknowns. From that joint posterior distribution, we may also obtain the marginals:

$$p(\boldsymbol{f}|\boldsymbol{g},\mathcal{H}) = \int p(\boldsymbol{f},\boldsymbol{\theta}|\boldsymbol{g},\mathcal{H})\,\mathrm{d}\boldsymbol{f} \text{ and } p(\boldsymbol{\theta}|\boldsymbol{g},\mathcal{H}) = \int p(\boldsymbol{f},\boldsymbol{\theta}|\boldsymbol{g},\mathcal{H})\,\mathrm{d}\boldsymbol{f} \tag{13}$$

7.2. Gaussian priors case

To be more specific, let consider the case of linear inverse problems $\boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{\epsilon}$,. Then, assuming Gaussian noise, we have:

$$p(\boldsymbol{g}|\boldsymbol{f}) = \mathcal{N}(\boldsymbol{g}|\boldsymbol{H}\boldsymbol{f}, v_\epsilon \boldsymbol{I}) \propto \exp\left[\frac{-1}{2v_\epsilon}\|\boldsymbol{g}-\boldsymbol{H}\boldsymbol{f}\|_2^2\right] \tag{14}$$

Assuming also a Gaussian prior:

$$p(\boldsymbol{f}) \propto \exp\left[\frac{-1}{2v_f}\|\boldsymbol{f}\|_2^2\right] \text{ or } \exp\left[\frac{-1}{2v_f}\|\boldsymbol{D}\boldsymbol{f}\|_2^2\right], \tag{15}$$

it is easy to see that the posterior is also Gaussian and the MAP and Posterior Mean (PM) estimates become the same and can be computed as the minimizer of : $J(\boldsymbol{f}) = \|\boldsymbol{g}-\boldsymbol{H}\boldsymbol{f}\|_2^2 + \lambda R(\boldsymbol{f})$:

$$p(\boldsymbol{f}|\boldsymbol{g}) \propto \exp\left[\frac{-1}{2v_\epsilon}J(\boldsymbol{f})\right] \rightarrow \widehat{\boldsymbol{f}}_{MAP} = \arg\max_{\boldsymbol{f}}\{p(\boldsymbol{f}|\boldsymbol{g})\} = \arg\min_{\boldsymbol{f}}\{J(\boldsymbol{f})\} \tag{16}$$

In summary, we have:

$$\left\{\begin{array}{l} p(\boldsymbol{g}|\boldsymbol{f}) = \mathcal{N}(\boldsymbol{g}|\boldsymbol{H}\boldsymbol{f}, v_\epsilon \boldsymbol{I}) \\ p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|0, v_f \boldsymbol{I}) \end{array}\right. \rightarrow \left\{\begin{array}{l} p(\boldsymbol{f}|\boldsymbol{g}) = \mathcal{N}(\boldsymbol{f}|\widehat{\boldsymbol{f}}, \widehat{\boldsymbol{\Sigma}}) \\ \widehat{\boldsymbol{f}} = [\boldsymbol{H}'\boldsymbol{H} + \lambda \boldsymbol{I}]^{-1}\boldsymbol{H}'\boldsymbol{g} \\ \widehat{\boldsymbol{\Sigma}} = v_\epsilon[\boldsymbol{H}'\boldsymbol{H} + \lambda \boldsymbol{I}]^{-1}, \quad \lambda = \frac{v_\epsilon}{v_f} \end{array}\right. \tag{17}$$

This case is also summarized in (Figure 6).

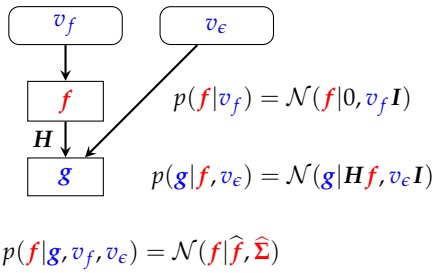

Figure 6. Bayesian inference scheme in linear systems and Gaussian priors. The posterior is also Gaussian and all the computations can be done analytically.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

7.3. Gaussian priors with unknown parameters

For the case where the hyper parameters v_ϵ and v_f are unknown (Unsupervised case), we can derive the following:

$$\left\{\begin{array}{l} p(\boldsymbol{g}|\boldsymbol{f}, v_\epsilon) = \mathcal{N}(\boldsymbol{g}|\boldsymbol{H}\boldsymbol{f}, v_\epsilon \boldsymbol{I}) \\ p(\boldsymbol{f}|v_f) = \mathcal{N}(\boldsymbol{f}|0, v_f \boldsymbol{I}) \\ p(v_\epsilon) = \mathcal{IG}(v_f|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_f) = \mathcal{IG}(v_f|\alpha_{f_0}, \beta_{f_0}) \end{array}\right. \rightarrow \left\{\begin{array}{l} p(\boldsymbol{f}|\boldsymbol{g}, v_\epsilon, v_f) = \mathcal{N}(\boldsymbol{f}|\widehat{\boldsymbol{f}}, \widehat{\boldsymbol{\Sigma}}) \\ \widehat{\boldsymbol{f}} = [\boldsymbol{H}'\boldsymbol{H} + \widehat{\lambda}\boldsymbol{I}]^{-1}\boldsymbol{H}'\boldsymbol{g} \\ \widehat{\boldsymbol{\Sigma}} = \hat{v_\epsilon}[\boldsymbol{H}'\boldsymbol{H} + \widehat{\lambda}\boldsymbol{I}]^{-1}, \ \widehat{\lambda} = \frac{\hat{v_\epsilon}}{\hat{v_f}} \\ p(v_\epsilon|\boldsymbol{g}, \boldsymbol{f}) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon) \\ p(v_f|\boldsymbol{g}, \boldsymbol{f}) = \mathcal{IG}(v_f|\widetilde{\alpha}_f, \widetilde{\beta}_f) \\ \widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon, \widetilde{\alpha}_f, \widetilde{\beta}_f \end{array}\right. \tag{18}$$

where all the details and in particular the expressions for $\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon, \widetilde{\alpha}_f, \widetilde{\beta}_f$ can be found in [19].

This case is also summarized in Figure 7.

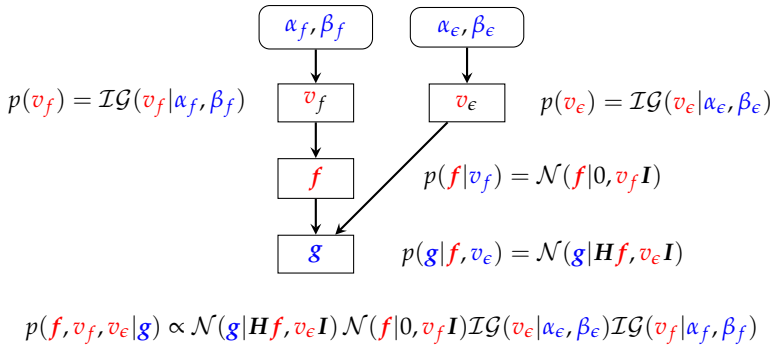

Figure 7. Bayesian inference scheme in linear systems and Gaussian priors. The posterior is also Gaussian and all the computations can be done analytically.

The joint posterior can be written as:

$$p(\boldsymbol{f}, v_\epsilon, v_\xi|\boldsymbol{g}) \propto \exp\left[-J(\boldsymbol{f}, v_\epsilon, v_\xi)\right] \tag{19}$$

From this expression, we have different expansion possibilities:

- JMAP: Alternate optimization with respect to $\boldsymbol{f}, v_\epsilon, v_f$:

$$J(\boldsymbol{f}, v_\epsilon, v_f) = \frac{1}{2v_\epsilon}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|_2^2 + \frac{1}{2v_f}\|\boldsymbol{f}\|_2^2 + (\alpha_{\epsilon_0} + 1)\ln v_\epsilon + \frac{\beta_{\epsilon_0}}{v_\epsilon} + (\alpha_{f_0} + 1)\ln v_f + \frac{\beta_{f_0}}{v_f} \tag{20}$$

- Gibbs sampling MCMC:

$$\boldsymbol{f} \sim p(\boldsymbol{f}, v_\epsilon, v_f|\boldsymbol{g}) \rightarrow v_\epsilon \sim p(v_\epsilon|\boldsymbol{g}, \boldsymbol{f}) \rightarrow v_f \sim p(v_f|\boldsymbol{g}, \boldsymbol{f}) \tag{21}$$

- Variational Bayesian Approximation: Approximate $p(\boldsymbol{f}, v_\epsilon, v_f|\boldsymbol{g})$ by a separable one $q(\boldsymbol{f}, v_\epsilon, v_f) = q_1(\boldsymbol{f})q_2(v_\epsilon)q_3(v_f)$ minimizing KL$(q|p)$ [19,52–55].

8. Imaging inside the Body: From Data acquisition to Decision

To introduce the link between the different model based methods and the Machine Learning tools, let consider the case of medical imaging, from the acquisition to the decision steps:

- Data acquisition :

$$\text{Object } \boldsymbol{f} \rightarrow \boxed{\text{CT scan, MRI, TEP, US,Microwave imaging}} \rightarrow \text{Data } \boldsymbol{g}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

- Image Reconstruction by analytical methods:

$$\text{Data } g \rightarrow \boxed{\text{Reconstruction}} \rightarrow \text{Image } \widehat{f}$$

- Post Processing (Segmentation, Contour detection, selection of Region of interest):

$$\text{Image } \widehat{f} \rightarrow \boxed{\text{Segmentation}} \rightarrow \widehat{z}$$

- Understanding and Decision:

$$\begin{array}{l}\text{Image } \widehat{f} \\ \text{Segmentation } \widehat{z}\end{array} \rightarrow \boxed{\begin{array}{l}\text{Interpretation} \\ \text{Decision}\end{array}} \rightarrow \begin{array}{l}\text{Tumor or} \\ \text{Not Tumor}\end{array}$$

8.1. Bayesian Joint reconstruction and segmentation

The questions now are: Can we join any of these steps? Can we go directly from the image to the decision? For the first one, the Bayesian approach can provide a solution:

$$\text{Data } g \rightarrow \boxed{\begin{array}{l}\text{Reconstruction} \\ \text{Segmentation}\end{array}} \begin{array}{l}\rightarrow \text{Reconstruction } \widehat{f} \\ \rightarrow \text{Segmentation } \widehat{z}\end{array}$$

The main tool here is to introduce a hidden variable which can represent the segmentation. A solution is to introduce a classification hidden variable z with $z_j = \{1, 2, \cdots, K\}$ which can be used to show the segmented image. See Figure 8

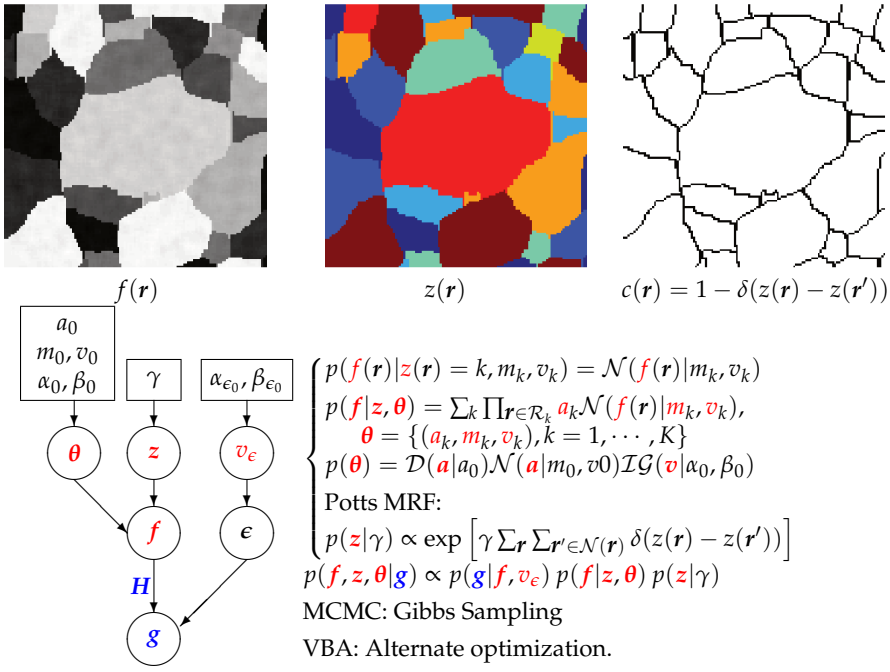

Figure 8. Gauss-Markov-Potts prior model for Bayesian image reconstruction and segmentation.

Figures 8 and 9 summarize this scheme:
A few comments for these relations:

- $p(\boldsymbol{g}|\boldsymbol{f}, \boldsymbol{z})$ does not depend on $\boldsymbol{z}$, so it can be written as $p(\boldsymbol{g}|\boldsymbol{f})$.
- We may choose a Markovian Potts model for $p(\boldsymbol{z})$ to obtain more compact homogeneous regions [18,19].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

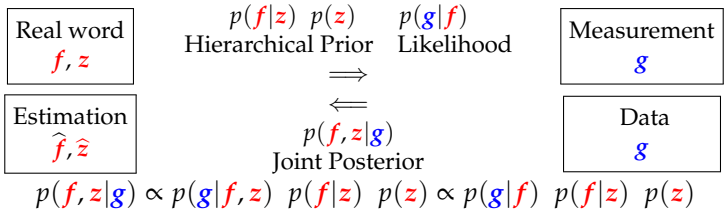

Figure 9. Bayesian approach with hierarchical prior model for joint reconstruction and segmentation.

- If we choose for $p(f|z)$ a Gaussian law, then $p(f, z|g)$ becomes a Gauss-Markov-Potts model [19].
- We can use the joint posterior $p(f, z|g)$ to infer on (f, z): We may just do JMAP: $(\widehat{f}, \widehat{z}) = \arg\max \{p(f, z|g)\}$ or trying to access to the expected posterior values by using the Variational Bayesian Approximation (VBA) techniques [19,56], [57,58], [17,55,59].

This scheme can be extended to consider the estimation of the hyper parameters too. Figure 10 shows this.

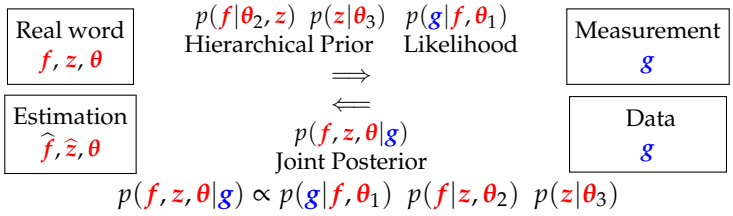

Figure 10. Advanced Bayesian approach for joint reconstruction, segmentation and parameter estimation.

Again, here, we can use the joint posterior $p(f, z, \theta|g)$ to infer on all the unknowns [17].

8.2. Advantages of the Bayesian Framework

Between the main advantages of the Bayesian framework for inverse problems, we can mention the following:

- Large flexibility of prior models prior
  - Smoothness (Gaussian, Gauss-Markov)
  - Direct Sparsity (Double Exp, Heavy-tailed distributions)
  - Sparsity in the Transform domain (Double Exp, Heavy-tailed distributions on the WT coefficients)
  - Piecewise continuous (DE or Student-t on the gradient)
  - Objects composed of only a few materials (Gauss-Markov-Potts), ...
- Possibility of estimating hyper-parameters via JMAP or VBA
- Natural ways to take account for uncertainties and quantify the remaining uncertainties.

8.3. Imaging inside the Body: From data to decision: Classical or Machine Learning

From previous sections, we see that we have many solutions to go from data to an image by inversion (image reconstruction), then extraction of interesting features (segmentation) and finally the interpretation and decision. The question that we may ask now is : *Can we do all together in a more easily way?* Machine Learning and Artificial Intelligence tools may propose such a solution. See Figure 11

To be able to use ML to go from data to decision, there is a crucial need of a great and rich data base obtained by experts to let the machine to **Learn** from that great data base. In the next section, we go a little more in details to see the advantages, limitations and drawbacks.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

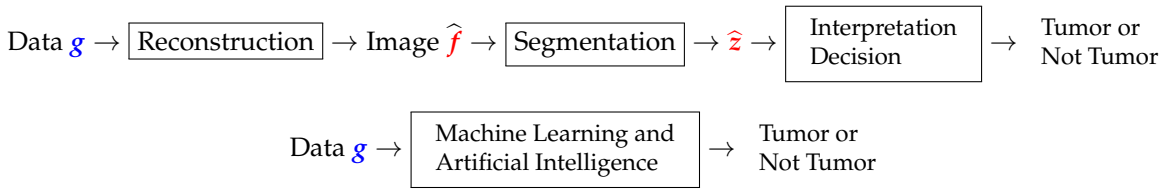

Figure 11. Two approaches going from the data to decision: Top: from data, first reconstruct an image via inversion, then post-process to obtain segmentation and do pattern recognition to extract the contours of region of interest and finally make a decision. Bottom: Try to use Machine Learning methods to go directly from data to decision.

9. Machine Learning Basic Idea

The main idea in Machine Learning is first **to learn** from a great number of data-decisions: $(g_i, d_i), i = 1, \cdots N$:

$$\text{Learning Data } (g_i, d_i)_{i=1}^{N} \rightarrow \boxed{\begin{array}{c}\text{Learning step}\\ \text{The weights } W \text{ of the NN are obtained}\end{array}} \rightarrow W$$

and then, when a new case (Test g_j) appears, it uses the learned weights W to give a decision d_j

$$\text{Test case Data } g_j \rightarrow \boxed{\begin{array}{c}\text{Testing step}\\ \text{The learned weights } W \text{ are used}\end{array}} \rightarrow \text{Tumor or Not Tumor } d_j$$

Figure 12 shows the main process of ML.

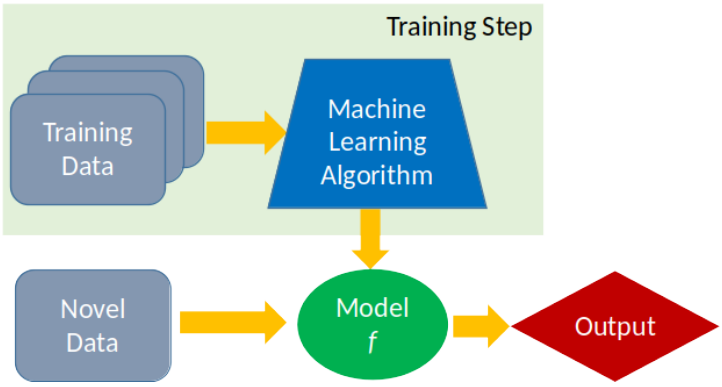

Figure 12. Basic Machine Learning process: First Learn a model, then use it. Learning step needs a rich enough data base which costs a lot. When the model is learned and tested, its use is easy, fast and its cost is low.

Nowadays, ML methods and tools have made great progress in many different area of applications. No need here to go more in details. Just mentioning a few main components of all of them. Between the basic tasks we can mention:

- Classification (supervised, semi-supervised);
- Clustering (unsupervised classification when the data have not yet labels);
- Regression (Continuos parameter estimation)

Figure 13 shows these three main tasks.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

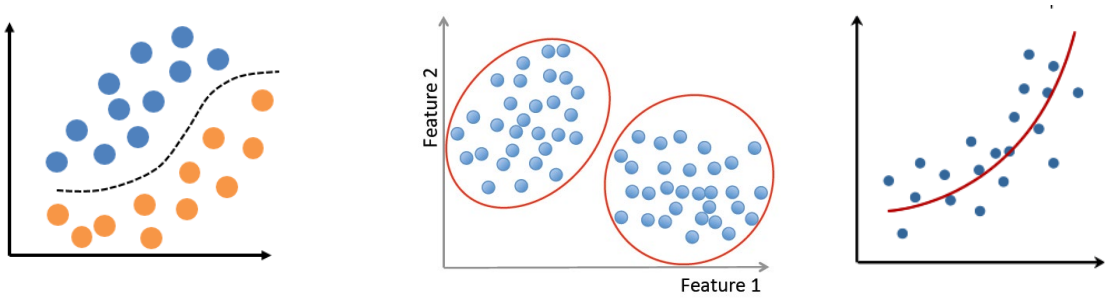

Figure 13. Basic Machine Learning Tasks: Classification, Clustering, Regression

Between the existing ML tools we may mention: Support Vector Machines (SVM), Decision-Tree learning (DT), Artificial Neural Networks (ANN), Bayesian Networks (BN), HMM and Random Forest (RF), Mixture Models (GMM, SMM, ...), KNN, Kmeans,...

Also, the combination of Imaging technology and systems, Image processing, Computer vision, Machine Learning & Artificial intelligence has been the seed for many great progress in all area of health and our environment. The frontiers between these science and technology has become less precise as it is shown in Figure 14.

Image technology	2D, 3D, Hyperspectral Acquisition, Compression, Transmission;
Image Processing	Representation, Compression, Segmentation; Enhancement, Restoration;
Computer Vision	Segmentation, Contour detection; Segments, Edges, Patterns, RoIs, Features extraction;
Machine Learning	Pattern matching and localization; Objects detection and identification;
Artificial Intelligence	2D & 3D pattern recognition, Interpretation; Classification, Clustering, Recognition, Decision making, ...

Figure 14. Frontiers between Image technology, Image processing (IP), Computer vision (CV), Machine Learning (ML) and Artificial intelligence (AI).

Between the Machine learning tools using NN, the Convolutional NN (CNN), Recurrent NN (RNN), Deep Learning (DL), Generative Artificial Networks (GAN) had greater success in different area such as Speech Recognition, Computer Vision and specifically in Segmentation, Classification and Clustering and in Multi-modality and cross-domain information fusion.

However, there are still many limitations: Lack of interpretability, reliability and uncertainty and No reasoning and explaining capabilities. To overcome, there still much to do with the Fundamentals.

10. Neural Networks, Machine Learning and Inverse problems

10.1. Neural Networks

Let starts this section by a few words on Neurons and Neural Networks. The following figures show the basic idea. The following figure shows the main idea about a Neuron in a mathematical framework. Figure 15 shows this graphically.

Figure 16 shows the components of a neuron and an example of a two layers NN.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

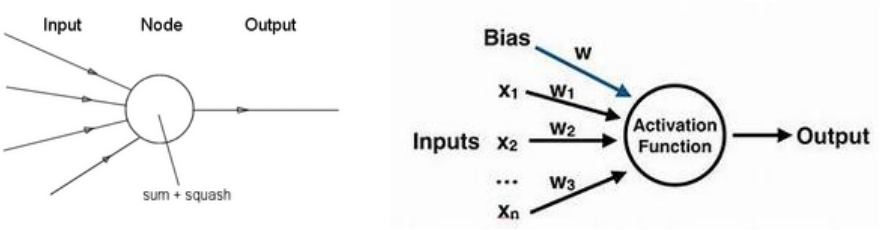

Figure 15. A neuron and its mathematical representation.

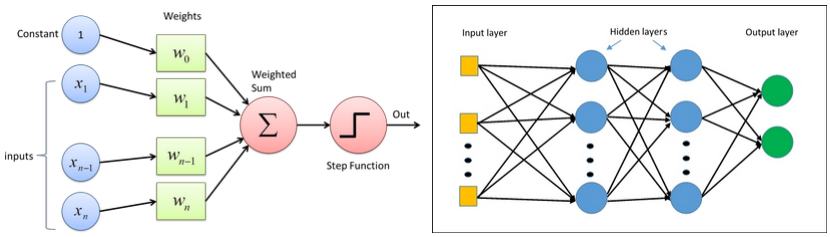

Figure 16. A neuron with its inputs and outputs and and a Neural Network with two hidden layers neurons.

10.2. NN and Learning

A Neural Network can be used for **Modeling** a universal relation between its inputs X and outputs Y. This model can be written as $Y = F_W(X)$ where W represents the parameters of the model represented by the weights of network nodes relation. They are commonly used for:

- Classification (Supervised learning)

  A set of data $\{(x_i, y_i)\}$ with labels (classes) $\{c_i\}$ are given. The objective during the training is to use them for training the network which is then used for classifying a new income (x_j, y_j)
- Clustering (Unsupervised learning)

  A set of data $\{(x_i, y_i)\}$ are given. The objective is to cluster them in different classes $\{c_i\}$.
- Regression with all data (Supervised learning)

  A set of data $\{(x_i, y_i)\}$ are given. The objective is to find a function F describing the relation between them: $F(x, y)$ or explicitly $y = F(x)$ for any x (extrapolation or interpolation).

10.3. Modeling, identification and inversion

Here, we make a connection between the classical and ML tools and show the links between Forward modeling and Inversion or Inference, Model identification and Learning or Training and Inversion and using the NN:

- Forward modeling and Inversion

$$f \rightarrow \boxed{\begin{array}{c}\text{Forward}\\ \text{modeling}\end{array}} \rightarrow \text{data } g \quad \| \quad \text{data } g \rightarrow \boxed{\begin{array}{c}\text{Inversion}\\ \text{Inference}\end{array}} \rightarrow \widehat{f}$$

- Identification of a system and Training step of NN

$$f \rightarrow \boxed{\begin{array}{c}\text{Identification}\\ W\end{array}} \rightarrow \text{data } g \quad \| \quad \{g_i, f_i\} \rightarrow \boxed{\begin{array}{c}\text{Learning}\\ \text{Training}\end{array}} \rightarrow \begin{array}{c}\text{Learned}\\ \text{Model } W\end{array}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

- Inversion (Inference) or Using the NN trained model

$$g \rightarrow \boxed{\begin{array}{c}\text{Inversion}\\ \text{Inference}\end{array}} \rightarrow \widehat{f} \quad \| \quad g \longrightarrow \boxed{\begin{array}{c}\text{Learned}\\ \text{Model } \boldsymbol{W}\end{array}} \longrightarrow \widehat{f}$$

11. ML for inverse problems

Now, to show the possibilities of the interaction between all these, nothing is better than a few examples.

11.1. First example: A linear two layers feed-forward NN

The first one is the case of linear inverse problems and quadratic regularization or the Bayesian with Gaussian priors. The solution has an analytic expression and we have the following relations:

$$g = \boldsymbol{H}f + \boldsymbol{\epsilon} \longrightarrow \widehat{f} = (\boldsymbol{H}\boldsymbol{H}^t + \lambda \boldsymbol{D}\boldsymbol{D}^t)^{-1}\boldsymbol{H}^t g = \boldsymbol{B}\boldsymbol{H}^t g$$

which can be presented schematically as

$$g \rightarrow \boxed{\boldsymbol{H}^t} \rightarrow \boxed{\boldsymbol{B}} \rightarrow \widehat{f} \text{ or equivalently } g \rightarrow \boxed{\text{Two layers linear CNN}} \rightarrow \widehat{f}$$

As we can see, this induces directly a linear feed forward NN structure. In particular, if $\boldsymbol{H}$ represents a convolution operator, then $\boldsymbol{H}^t$ and $\boldsymbol{H}^t\boldsymbol{H}$ are too and probably the operator $\boldsymbol{B}$ can also be well approximated by a convolution and the whole inversion can be modelled by a CNN [60].

11.2. Second example: Image denoising with a two layers CNN

The second example is the denoising $g = f + \boldsymbol{\epsilon}$ with L1 regularizer:

$$\widehat{f} = \boldsymbol{D}\widehat{z} \text{ and } \widehat{z} = \arg\min_{z}\{J(z)\} \text{ with } J(z) = \|g - \boldsymbol{D}z| + \lambda\|z\|_1 \tag{22}$$

where $\boldsymbol{D}$ is a filter, i.e., a convolution operator. This can also be considered as the MAP estimator with a double exponential prior. It is easy to show that the solution can be obtained by a convolution followed by a thresholding [61–63].

$$\widehat{f} = \boldsymbol{D}\widehat{z} \text{ and } \widehat{z} = S_{\frac{1}{\lambda}}(\boldsymbol{D}^t g)$$

where S_λ is a Thresholding operator.

$$g \rightarrow \boxed{\boldsymbol{D}^t} \rightarrow \boxed{\text{Thresholding}} \rightarrow \widehat{z} \rightarrow \boxed{\boldsymbol{D}} \rightarrow \widehat{f} \text{ or equivalently } g \rightarrow \boxed{\text{Two layers CNN}} \rightarrow \widehat{f}$$

11.3. Third example: A Deep learning equivalence

One of the classical iterative methods in linear inverse problems algorithm is based on just a gradient based method to optimize $J(f) = \|g - \boldsymbol{H}f\|^2$:

$$f^{(k+1)} = f^{(k)} + \alpha\boldsymbol{H}^t(g - \boldsymbol{H}f^{(k)}) = \alpha\boldsymbol{H}^t g + (\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H})f^{(k)} \tag{23}$$

where the solution of the problem is obtained recursively. Every body knows that, when the forward model operator $\boldsymbol{H}$ is singular or ill-conditioned, this iterative algorithm starts by converging, but it may diverge easily. One of the experimental method to obtain an acceptable approximate solution is just to stop the iterations after K iterations. This idea can be translated to a Deep Learning NN by using K layers. Each layer representing one iteration of the algorithm. See Figure 17 and 18

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

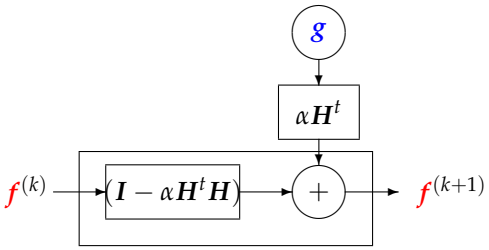

Figure 17. One bloc of iteration which can be considered as one layer of a NN

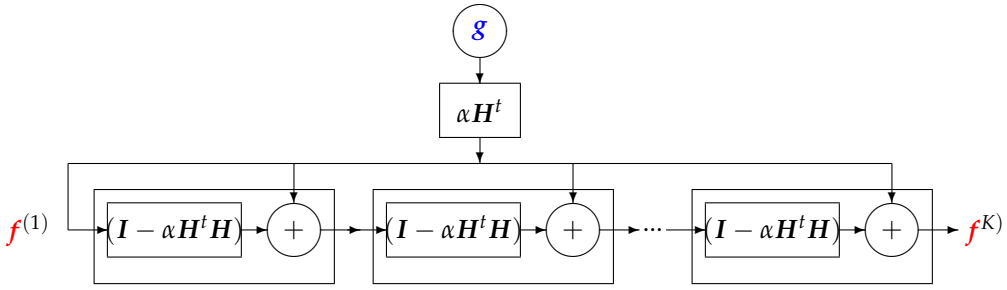

Figure 18. A K layers DL NN equivalent to K iterations of the basic optimization algorithm.

This DL structure can easily be extended to a regularized criterion: $J(\boldsymbol{f}) = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|^2 + \lambda\|\boldsymbol{D}\boldsymbol{f}\|^2$, where

$$\boldsymbol{f}^{(k+1)} = \boldsymbol{f}^{(k)} + \alpha[\boldsymbol{H}^t(\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}^{(k)}) - \lambda\boldsymbol{D}^t\boldsymbol{D}] = \alpha\boldsymbol{H}^t\boldsymbol{g} + (\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H} - \alpha\lambda\boldsymbol{D}^t\boldsymbol{D})\boldsymbol{f}^{(k)} \tag{24}$$

We just need to replace $(\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H})$ by $(\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H} - \alpha\lambda\boldsymbol{D}^t\boldsymbol{D})$.

This structure can also be extended to all the sparsity enforcing regularization terms such as ℓ_1 and Total Variation (TV) using appropriate algorithms such as ISTA, FISTA, ADMM, etc. by replacing the update expression and by adding a NL operation much like the ordinary NNs. A simple example is given in the following subsection.

11.4. Fourth example: ℓ_1 regularization and NN

Let us to consider the linear inverse problem $\boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{\epsilon}$ with ℓ_1 regularization criterion:

$$J(\boldsymbol{f}) = \|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|_2^2 + \lambda\|\boldsymbol{f}\|_1$$

and an iterative optimization algorithm, such as ISTA

$$\boldsymbol{f}^{(k+1)} = Prox_{\ell_1}\left(\boldsymbol{f}^{(k)}, \lambda\right) \stackrel{\triangle}{=} \mathcal{S}_{\lambda\alpha}\left(\alpha\boldsymbol{H}^t\boldsymbol{g} + (\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H})\boldsymbol{f}^{(k)}\right)$$

where $\mathcal{S}_\theta$ is a soft thresholding operator and $\alpha \leq |eig(\boldsymbol{H}^t\boldsymbol{H})|$ is the Lipschitz constant of the normal operator. When $\boldsymbol{H}$ is a convolution operator, then:

- $(\boldsymbol{I} - \alpha\boldsymbol{H}^t\boldsymbol{H})\boldsymbol{f}^{(k)}$ can also be approximated by a convolution and thus considered as a filtering operator;
- $\frac{1}{\alpha}\boldsymbol{H}^t\boldsymbol{g}$ can be considered as a bias term and is also a convolution operator; and
- $\mathcal{S}_{\theta=\lambda\alpha}$ is as nonlinear point wise operator. In particular when $\boldsymbol{f}$ is a positive quantity, this soft thresholding operator can be compared to ReLU activation function of NN.

In all these three examples, we directly could obtain the structure of the NN from the Forward model and known parameters. However, in this approaches there are some difficulties which consist in the determination the structure of the NN. For example, in the first example, obtaining the structure of $\boldsymbol{B}$ depends on the regularization parameter λ. The same difficulty arise for determining the shape and the threshold level of the Thresholding bloc of the network in the second example. The same need

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

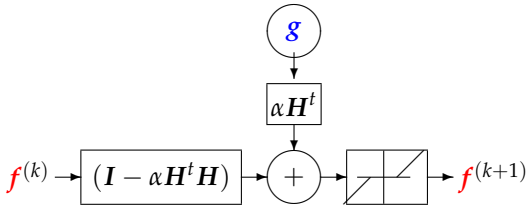

Figure 19. One block of a NN correspond to one iteration of ℓ_1 regularization

of the regularization parameter as well as many other hyper parameters are necessary to create the NN structure and weights. In practice, we can decide, for example on the number and structure of DL network, but as their corresponding weights depend on many unknown or difficult to fix parameters, ML may become of help. In the following we first consider the training part of a general ML method. Then, we will see how to include the physics based knowledge of the forward model in the structure of learning.

12. ML general approach

The ML approach can become helpfully if we could have a great number of data: inputs-outputs $\{(f,g)_k, k = 1,2,...,K\}$ examples. Thus, during the Training step, we can learn the coefficients of the NN and then use it for obtaining a new solution $\widehat{f}$ for a new data g.

The main issue is the number of data input-output examples $\{(f,g)_k, k = 1,2,...,K\}$ we can have for the training step of the network.

12.1. Fully learned method

Let consider a one layer NN where the relation between its input g_k and output f_k is given by $f_k = \boldsymbol{\phi}(\boldsymbol{W}g_k)$ where $\boldsymbol{W}$ is the weighting parameters of the NN and $\boldsymbol{\phi}$ is the point wise non linearity function of the output NN output layer. The estimation of $\boldsymbol{W}$ from the training data in the learning step is done by an optimization algorithm which optimizes a Loss function $\mathcal{L}$ defined as

$$\mathcal{L} = \sum_{k=1}^{K} \ell_k(f_k, \boldsymbol{\phi}(\boldsymbol{W}g_k)) \tag{25}$$

with

$$\ell_k(f_k, \boldsymbol{\phi}(\boldsymbol{W}g_k) = \|f_k - \boldsymbol{\phi}(\boldsymbol{W}g_k)\|^2 \tag{26}$$

a quadratic distance or any other appropriate distance or divergence or a probabilistic one

$$\ell_k(f_k, \boldsymbol{\phi}(\boldsymbol{W}g_k) = \mathrm{E}\left\{\|f_k, \boldsymbol{\phi}(\boldsymbol{W}g_k)\|^2\right\} \tag{27}$$

When, the NN is trained and we obtain the weights $\widehat{\boldsymbol{W}}$, then we can use it easily when a new case (Test g_j) appears, just by applying: $f_k = \boldsymbol{\phi}(\widehat{\boldsymbol{W}}g_k)$. These two steps of Training and Using (called also Testing) are illustrated in Figure 20

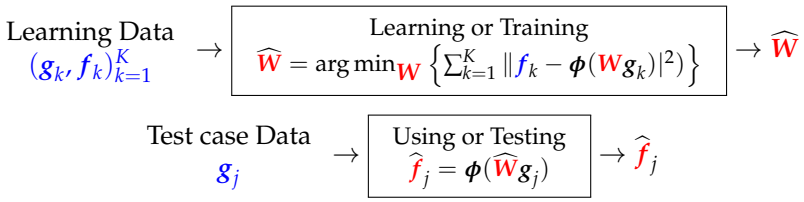

Figure 20. Training (top) and Testing (bottom) steps in a ML approach

The scheme that we presented is general and can be extended to any multi-layer NN and DL. In fact, if we had a great number of data-ground truth examples $\{(f,g)_k, k = 1,2,...,K\}$ with K much more than the number of elements $W_{m,n}$ of the weighting parameters $\boldsymbol{W}$, then, we did not even have

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

any need for forward model $\boldsymbol{H}$. This can be possible for very low dimensional problems [64–67]. But, in general, in practice we do not have enough data. So, some prior or regularizer is needed to obtain a usable solution. This can be just by adding a regularizer $R(\boldsymbol{W})$ to the loss function 25 and 26, but we can also use the physics of the forward operator $\boldsymbol{H}$.

13. Physics based ML

As mentioned above, in general, in practice, a rich enough and complete data set is not often available in particular for inverse problems. So, some prior or regularizer is needed to obtain a usable solution. Using a regularizer $R(\boldsymbol{W})$ to the loss function 25 is good, but not enough. We have to use the physics of the forward operator $\boldsymbol{H}$. This can be done in different ways.

13.1. Decomposition of the NN structure to fixed and trainable parts

The first easiest and understandable method consists in decomposing the structure of the network $\boldsymbol{W}$ in two parts: a fixed part and a learnable part. As the simplest example, we can consider the case of analytical expression of the quadratic regularization: $\widehat{\boldsymbol{f}} = (\boldsymbol{H}\boldsymbol{H}^t + \lambda \boldsymbol{D}\boldsymbol{D}^t)^{-1}\boldsymbol{H}^t\boldsymbol{g} = \boldsymbol{B}\boldsymbol{H}^t\boldsymbol{g}$ which suggests to have a two layers network with a fixed part structure $\boldsymbol{H}^t$ and a trainable one $\boldsymbol{B} = (\boldsymbol{H}\boldsymbol{H}^t + \lambda \boldsymbol{D}\boldsymbol{D}^t)^{-1}$. See Figure 21.

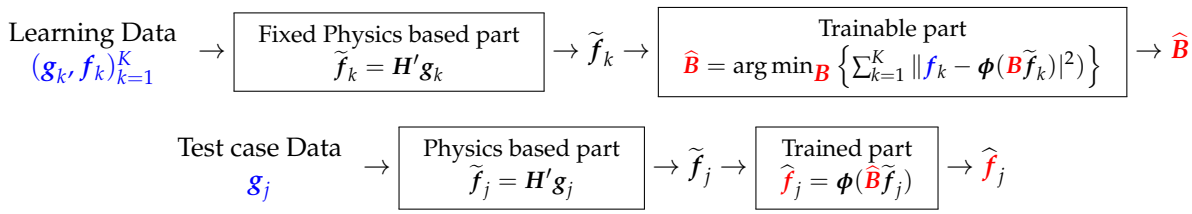

Figure 21. Training (top) and Testing (bottom) steps in the first use of physics based ML approach

It is interesting to note that in X-ray Computed Tomography (CT) the forward operator $\boldsymbol{H}$ is called *Projection*, the adjoint operator $\boldsymbol{H}'$ is called *Back-Projection (BP)* and the $\boldsymbol{B}$ operator is assimilated to a 2D filtering (convolution).

13.2. Using Singular value decomposition of forward and backward operators

Using the eigenvalues and eigenvectors of the pseudo or generalized inverse operators

$$\boldsymbol{H}^\dagger = [\boldsymbol{H}'\boldsymbol{H}]^{-1}\boldsymbol{H}' \quad \text{or} \quad \boldsymbol{H}^\dagger = \boldsymbol{H}'[\boldsymbol{H}\boldsymbol{H}']^{-1} \tag{28}$$

and Singular value decomposition (SVD) of the operators $[\boldsymbol{H}'\boldsymbol{H}]$ and $[\boldsymbol{H}\boldsymbol{H}']$ give another possible decomposition of the NN structure. Let us to note

$$\boldsymbol{H}\boldsymbol{H}' = \boldsymbol{U}\boldsymbol{\Delta}\boldsymbol{V}' \text{ or equivalently } \boldsymbol{H}'\boldsymbol{H} = \boldsymbol{V}\boldsymbol{\Delta}\boldsymbol{U}' \tag{29}$$

where $\boldsymbol{\Delta}$ is a diagonal matrix containing the singular values, $\boldsymbol{U}$ and $\boldsymbol{V}$ containing the corresponding eigenvectors. This can be used to decompose the $\boldsymbol{W}$ to four operators:

$$\boldsymbol{W} = \boldsymbol{V}'\boldsymbol{\Delta}\boldsymbol{U}\boldsymbol{H}' \quad \text{or} \quad \boldsymbol{W} = \boldsymbol{H}'\boldsymbol{V}\boldsymbol{\Delta}\boldsymbol{U}' \tag{30}$$

where three of them can be fixed and only one $\boldsymbol{\Delta}$ can be trainable. It is interesting to know that when the forward operator $\boldsymbol{H}$ has a shift-invariant (convolution) property, then the operators $\boldsymbol{U}$ and $\boldsymbol{V}'$ will correspond, respectively, to the FT and IFT operators and the diagonal elements of Λ correspond to the FT of the impulse response of the convolution forward operator. So, we will have a fixed layer corresponding to $\boldsymbol{H}'$ which can be interpreted as a matched filtering, then a fixed FT layer which is a

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2021 doi:10.20944/preprints202111.0092.v1

feed-forward linear network, a trainable filtering part corresponding the diagonal elements of $\mathbf{\Lambda}$ and a forth fixed layer corresponding to IFT. See Figure 22.

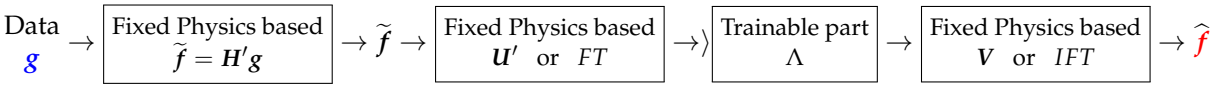

Figure 22. A four layers NN with three physics based fixed corresponding to $\boldsymbol{H}'$, $\boldsymbol{U}'$ or FT and $\boldsymbol{V}$ or IFT layers and one trainable layer corresponding to Λ.

13.2.1. DL structure based on iterative inversion algorithm

Using the iterative gradient based algorithm with fixed number of iterations for computing a GI or a regularized one as explained in previous section can be used to propose a DL structure with K layers, K being the number of iterations before stopping. Figure 23 shows this structure for a quadratic regularization which results to a linear NN and Figure 24 for the case of ℓ_1 regularization.

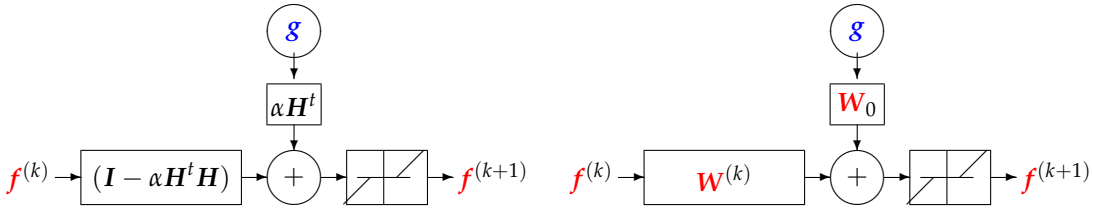

Figure 23. A K layers DL NN equivalent to K iterations of a basic gradient based optimization algorithm. A quadratic regularization results to a linear NN while a ℓ_1 regularization results to a classical NN with a nonlinear activation function. Left: supervised case. Right: unsupervised case. In both cases, all the K layers have the same structure.

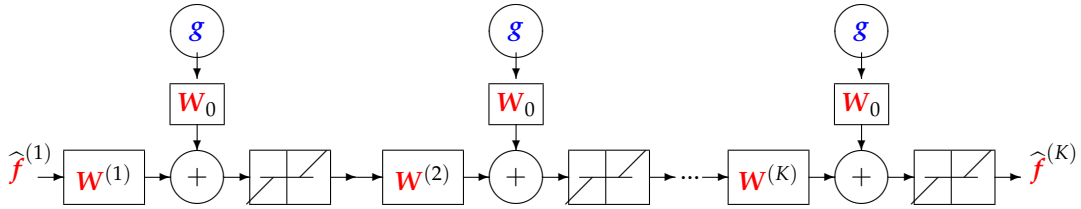

Figure 24. All the K layers of DL NN equivalent to K iterations of an iterative gradient based optimization algorithm. The simplest solution is to choose
$W_0 = \alpha H$ and $W^{(k)} = W = (I - \alpha H'H), \quad k = 1, \cdots, K.$
A more robust, but more costly is to learn all the layers $W^{(k)} = (I - \alpha^{(k)} H'H), \quad k = 1, \cdots, K.$

14. Conclusions and Challenges

Signal and image processing (SIP), imaging systems (IS), computer vision (CV), Machine learning (ML) and artificial intellegence (AI) have made great progress in the last forty years. The first category of the methods in signal and image was based on linear transformation followed by a thresholding or windowing and coming back. The second generation was model based: forward modeling and inverse problems approach. The main successful approach was based on regularization methods using a combined criterion. The third generation was model based but probabilistic and in particular using the Bayes rule, the so called Bayesian approach. Nowadays, ML, Neural Networks (NN), Convolutional NN (CNN) and Deep Learning (DL) methods have obtained great success in classification, clustering,

object detection, speech and face recognition, etc. But, they need great number of training data and lack still explanation and they may fail very easily. For inverse problems, they need still progress to do. In fact, using only data based NN without any specific structure coming from the forward mode (Physics) is just un illusion. However, the progress arrive via their interaction with the model based methods. In fact, the successful of CNN and DL methods greatly depends on the appropriate choice of the network structure. This choice can be guided by the model based methods. In this work, a few examples of such interactions are described. As we could see the main contribution of ML and NN tools can be on reducing the costs of inversion method when an appropriate model is trained. However, to obtain a good model, there is a need for sufficiently rich data and a good network structure obtained from the physics knowledge of the problem in hand. For inverse problems, when the forward models are non linear and complex, NN and DL may be of great help. However, we may still need to choose the structure of the NN via approximate forward model and approximate Bayesian inversion.

References

1. Bioucas-Dias, J.M.; Figueiredo, M.A. An iterative algorithm for linear inverse problems with compound regularizers. Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on. IEEE, 2008, pp. 685–688.
2. Ghadimi, E.; Teixeira, A.; Shames, I.; Johansson, M. Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems. *IEEE Transactions on Automatic Control* **2015**, *60*, 644–658.
3. Chambolle, A.; Dossal, C. On the convergence of the iterates of" FISTA". *Journal of Optimization Theory and Applications* **2015**, *166*, 25.
4. Ayasso, H.; Duchêne, B.; Mohammad-Djafari, A. MCMC and variational approaches for Bayesian inversion in diffraction imaging. *Regularization and Bayesian Methods for Inverse Problems in Signal and Image Processing* **2015**, *201*, 224.
5. Florea, M.I.; Vorobyov, S.A. A robust FISTA-like algorithm. Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017, pp. 4521–4525.
6. Mohammad-Djafari, A. Inverse problems in imaging science: from classical regularization methods to state of the art Bayesian methods. International Image Processing, Applications and Systems Conference, 2014, pp. 1–2. doi:10.1109/IPAS.2014.7043317.
7. Mohammad-Djafari, A. Bayesian inference with hierarchical prior models for inverse problems in imaging systems. 2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), 2013, pp. 7–18. doi:10.1109/WoSSPA.2013.6602329.
8. Mohammad-Djafari, A. Bayesian approach with prior models which enforce sparsity in signal and image processing. *EURASIP Journal on Advances in Signal Processing* **2012**, *2012*, 52.
9. Ayasso, H.; Duchêne, B.; Mohammad-Djafari, A. Bayesian inversion for optical diffraction tomography. *Journal of Modern Optics* **2010**, *57*, 765–776.
10. Ren, D.; Zhang, K.; Wang, Q.; Hu, Q.; Zuo, W. Neural Blind Deconvolution Using Deep Priors. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020); , 2020; pp. 3338–3347.
11. Bertocchi, C.; Chouzenoux, E.; Corbineau, M.C.; Pesquet, J.C.; Prato, M. Deep Unfolding of a Proximal Interior Point Method for Image Restoration. *Inverse Problems* **2019**, *36*.
12. Jospin, L.V.; Buntine, W.L.; Boussaid, F.; Laga, H.; Bennamoun, M. Hands-on Bayesian Neural Networks - a Tutorial for Deep Learning Users. *ArXiv* **2020**, *abs/2007.06823*.
13. Monga, V.; Li, Y.; Eldar, Y.C. Algorithm Unrolling: Interpretable, Efficient Deep Learning for Signal and Image Processing. *IEEE Signal Processing Magazine* **2021**, *38*, 18–44.
14. Gilton, D.; Ongie, G.; Willett, R. Model Adaptation for Inverse Problems in Imaging. *IEEE Transactions on Computational Imaging* **2021**, *7*, 661–674.
15. Repetti, A.; Pereyra, M.; Wiaux, Y. Scalable Bayesian Uncertainty Quantification in Imaging Inverse Problems via Convex Optimization. *SIAM Journal on Imaging Sciences* **2019**, *12*, 87–118.

16. Mohammad-Djafari, A. Gauss-Markov-Potts Priors for Images in Computer Tomography resulting to Joint Optimal Reconstruction and Segmentation. *International J. of Tomography and Statistics (IJTS)* **2008**, *11*, 76–92.
17. Ayasso, H.; Mohammad-Djafari, A. Joint NDT image restoration and segmentation using Gauss–Markov–Potts prior models and variational bayesian computation. *IEEE Transactions on Image Processing* **2010**, *19*, 2265–2277.
18. Féron, O.; Duchêne, B.; Mohammad-Djafari, A. Microwave imaging of inhomogeneous objects made of a finite number of dielectric and conductive materials from experimental data. *Inverse Problems* **2005**, *21*, S95.
19. Chapdelaine, C.; Mohammad-Djafari, A.; Gac, N.; Parra, E. A 3D Bayesian Computed Tomography Reconstruction Algorithm with Gauss-Markov-Potts Prior Model and its Application to Real Data. *Fundamenta Informaticae* **2017**, *155*, 373–405.
20. Chun, I.Y.; Huang, Z.; Lim, H.; Fessler, J. Momentum-Net: Fast and convergent iterative neural network for inverse problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2020**, pp. 1–1. doi:10.1109/TPAMI.2020.3012955.
21. Sandhu, A.I.; Shaukat, S.A.; Desmal, A.; Bagci, H. ANN-assisted CoSaMP Algorithm for Linear Electromagnetic Imaging of Spatially Sparse Domains. *IEEE Transactions on Antennas and Propagation* **2021**, pp. 1–1. doi:10.1109/TAP.2021.3060547.
22. Fang, Z. A High-Efficient Hybrid Physics-Informed Neural Networks Based on Convolutional Neural Network. *IEEE Transactions on Neural Networks and Learning Systems* **2021**, pp. 1–13. doi:10.1109/TNNLS.2021.3070878.
23. Husain, Z.; Madjid, N.A.; Liatsis, P. Tactile sensing using machine learning-driven Electrical Impedance Tomography. *IEEE Sensors Journal* **2021**, pp. 1–1. doi:10.1109/JSEN.2021.3054870.
24. Ongie, G.; Jalal, A.; Metzler, C.A.; Baraniuk, R.G.; Dimakis, A.G.; Willett, R. Deep Learning Techniques for Inverse Problems in Imaging. *IEEE Journal on Selected Areas in Information Theory* **2020**, *1*, 39–56. doi:10.1109/JSAIT.2020.2991563.
25. Gilton, D.; Ongie, G.; Willett, R. Neumann Networks for Linear Inverse Problems in Imaging. *IEEE Transactions on Computational Imaging* **2020**, *6*, 328–343. doi:10.1109/TCI.2019.2948732.
26. Gong, D.; Zhang, Z.; Shi, Q.; van den Hengel, A.; Shen, C.; Zhang, Y. Learning Deep Gradient Descent Optimization for Image Deconvolution. *IEEE Transactions on Neural Networks and Learning Systems* **2020**, *31*, 5468–5482. doi:10.1109/TNNLS.2020.2968289.
27. Sun, J.; Zhang, J.; Zhang, X.; Zhou, W. A Deep Learning-Based Method for Heat Source Layout Inverse Design. *IEEE Access* **2020**, *8*, 140038–140053. doi:10.1109/ACCESS.2020.3013394.
28. Peng, P.; Jalali, S.; Yuan, X. Solving Inverse Problems via Auto-Encoders. *IEEE Journal on Selected Areas in Information Theory* **2020**, *1*, 312–323. doi:10.1109/JSAIT.2020.2983643.
29. Zhang, X.; Li, B.; Jiang, J. Hessian Free Convolutional Dictionary Learning for Hyperspectral Imagery With Application to Compressive Chromo-Tomography. *IEEE Access* **2020**, *8*, 104216–104231. doi:10.1109/ACCESS.2020.2999457.
30. de Haan, K.; Rivenson, Y.; Wu, Y.; Ozcan, A. Deep-Learning-Based Image Reconstruction and Enhancement in Optical Microscopy. *Proceedings of the IEEE* **2020**, *108*, 30–50. doi:10.1109/JPROC.2019.2949575.
31. Lu, F.; Wu, J.; Huang, J.; Qiu, X. Restricted-Boltzmann-Based Extreme Learning Machine for Gas Path Fault Diagnosis of Turbofan Engine. *IEEE Transactions on Industrial Informatics* **2020**, *16*, 959–968. doi:10.1109/TII.2019.2921032.
32. Zheng, J.; Peng, L. A Deep Learning Compensated Back Projection for Image Reconstruction of Electrical Capacitance Tomography. *IEEE Sensors Journal* **2020**, *20*, 4879–4890. doi:10.1109/JSEN.2020.2965731.
33. Ren, S.; Sun, K.; Tan, C.; Dong, F. A Two-Stage Deep Learning Method for Robust Shape Reconstruction With Electrical Impedance Tomography. *IEEE Transactions on Instrumentation and Measurement* **2020**, *69*, 4887–4897. doi:10.1109/TIM.2019.2954722.
34. Yang, G.; Yu, S.; Dong, H.; Slabaugh, G.; Dragotti, P.L.; Ye, X.; Liu, F.; Arridge, S.; Keegan, J.; Guo, Y.; Firmin, D. DAGAN: Deep De-Aliasing Generative Adversarial Networks for Fast Compressed Sensing MRI Reconstruction. *IEEE Transactions on Medical Imaging* **2018**, *37*, 1310–1321. doi:10.1109/TMI.2017.2785879.
35. Stahlhut, C.; Morup, M.; Winther, O.; Hansen, L.K. Hierarchical Bayesian model for simultaneous EEG source and forward model reconstruction (SOFOMORE). 2009 IEEE International Workshop on Machine Learning for Signal Processing, 2009, pp. 1–6. doi:10.1109/MLSP.2009.5306189.

36. Idier, J. *Bayesian approach to inverse problems*; John Wiley & Sons, 2008.
37. Mohammad-Djafari, A. *Problèmes inverses en imagerie et en vision en deux volumes inséparables*; Traité Signal et Image, IC2, ISTE-WILEY, 2009.
38. Mohammad-Djafari, A. *Inverse Problems in Vision and 3D Tomography*; ISTE-WILEY, 2010.
39. Penrose, R. A Generalized Inverse for Matrices. Proceedings of the Cambridge Philosophy Society, 1955, Vol. 51, pp. 406–413.
40. Carasso, A.S. Direct Blind Deconvolution. *SIAM Journal on Applied Mathematics* **2001**, *61*, 1980–2007.
41. Chan, T.; Wong, C.K. Total variation blind deconvolution. *IEEE transactions on image processing* **1998**, *7*, 370–375.
42. Chu, N.; Mohammad-Djafari, A.; Gac, N.; Picheral, J. A variational Bayesian approximation approach via a sparsity enforcing prior in acoustic imaging. 2014 13th Workshop on Information Optics (WIO), 2014, pp. 1–4. doi:10.1109/WIO.2014.6933297.
43. Gharsalli, L.; Duchêne, B.; Mohammad-Djafari, A.; Ayasso, H. Microwave tomography for breast cancer detection within a variational Bayesian approach. 21st European Signal Processing Conference (EUSIPCO 2013), 2013, pp. 1–5.
44. Gharsalli, L.; Duchêne, B.; Mohammad-Djafari, A.; Ayasso, H. A Gauss-Markov mixture prior model for a variational Bayesian approach to microwave breast imaging. 2014 IEEE Conference on Antenna Measurements Applications (CAMA), 2014, pp. 1–4. doi:10.1109/CAMA.2014.7003385.
45. Premel, D.; Mohammad-Djafari, A. Eddy current tomography in cylindrical geometry. *IEEE Transactions on Magnetics* **1995**, *31*, 2000–2003. doi:10.1109/20.376435.
46. Kak, A.C.; Slaney, M. *Principles of computerized tomographic imaging*; SIAM, 2001.
47. Jackson, J.I.; Meyer, C.H.; Nishimura, D.G.; Macovski, A. Selection of a convolution function for Fourier inversion using gridding [computerized tomography application]. *Medical Imaging, IEEE Transactions on* **1991**, *10*, 473–478.
48. Osher, S.; Burger, M.; Goldfarb, D.; Xu, J.; Yin, W. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation* **2005**, *4*, 460–489.
49. Wang, Y.; Yang, J.; Yin, W.; Zhang, Y. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences* **2008**, *1*, 248–272.
50. Goldstein, T.; Osher, S. The split Bregman method for L1-regularized problems. *SIAM journal on imaging sciences* **2009**, *2*, 323–343.
51. Sun, Y.; Wu, Z.; Xu, X.; Wohlberg, B.; Kamilov, U.S. Scalable Plug-and-Play ADMM With Convergence Guarantees. *IEEE Transactions on Computational Imaging* **2021**, *7*, 849–863.
52. Mohammad-Djafari, A. Joint estimation of parameters and hyperparameters in a Bayesian approach of solving inverse problems. Proceedings of 3rd IEEE International Conference on Image Processing, 1996, Vol. 1, pp. 473–476 vol.2. doi:10.1109/ICIP.1996.560890.
53. Mohammad-Djafari, A.; Ayasso, H. Variational Bayes and mean field approximations for Markov field unsupervised estimation. 2009 IEEE International Workshop on Machine Learning for Signal Processing. IEEE, 2009, pp. 1–6.
54. Blei, D.; Kucukelbir, A.; McAuliffe, J. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association* **2016**, *112*.
55. Wang, L.; Mohammad-Djafari, A.; Gac, N. X-ray Computed Tomography using a sparsity enforcing prior model based on Haar transformation in a Bayesian framework. *Fundamenta Informaticae* **2017**, *155*, 449–480.
56. Wang, L.; Gac, N.; Mohammad-Djafari, A. Bayesian 3D X-ray computed tomography image reconstruction with a scaled Gaussian mixture prior model. AIP Conf. Proc, 2015, Vol. 1641, pp. 556–563.
57. Chapdelaine, C.; Mohammad-Djafari, A.; Gac, N.; Parra, E. A Joint Segmentation and Reconstruction Algorithm for 3D Bayesian Computed Tomography Using Gauss-Markov-Potts Prior Model. The 42nd IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.
58. Chapdelaine, C.; Gac, N.; Mohammad-Djafari, A.; Parra, E. New GPU implementation of Separable Footprint Projector and Backprojector : first results. The 5th International Conference on Image Formation in X-Ray Computed Tomography, 2018.
59. Chapdelaine, C. Variational Bayesian Approach and Gauss-Markov-Potts prior model. *arXiv:1808.09552* **2018**.

60. Ciresan, D.C.; Meier, U.; Masci, J.; Gambardella, L.M.; Schmidhuber, J. Flexible, High Performance Convolutional Neural Networks for Image Classification. Intl. Joint Conference on Artificial Intelligence IJCAI, 2011, pp. 1237–1242.
61. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NIPS 2012), 2012, p. 4.
62. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. Technical Report arXiv:1311.2901 [cs.CV], NYU, 2013.
63. Nan, Y.; Quan, Y.; Ji, H. Variational-EM-Based Deep Learning for Noise-Blind Image Deblurring. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020); , 2020.
64. Saul, L.K.; Roweis, S.T. Think globally, fit locally: unsupervised learning of low dimensional manifolds. *The Journal of Machine Learning Research* **2003**, *4*, 119–155.
65. Arel, I.; Rose, D.C.; Karnowski, T.P. Deep machine learning – a new frontier in artificial intelligence research. *Computational Intelligence Magazine, IEEE* **2010**, *5*, 13–18.
66. Cho, K.; Raiko, T.; Ilin, A. Enhanced gradient for training restricted Boltzmann machines. *Neural Computation* **2013**, *25*, 805–831.
67. Cho, K. Foundations and Advances in Deep Learning. PhD thesis, Aalto University School of Science, 2014.