WPS1-PR(18) 20810

A Pavone et al.

# Neural network approximation of Bayesian models for the inference of ion and electron temperature profiles at W7-X

Preprint of Paper to be submitted for publication in
Plasma Physics and Controlled Fusion

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

# Neural network approximation of Bayesian models for the inference of ion and electron temperature profiles at W7-X

**A. Pavone[1], J. Svensson[1], A. Langenberg[1], U. Höfel[1], S. Kwak[1], N. Pablant [2], R. C. Wolf[1] and the Wendelstein 7-X Team[1]**

[1] Max-Planck-Institut für Plasmaphysik, Teilinstitut Greifswald, D-17491 Greifswald, DE
[2] Princeton Plasma Physics Laboratory, 08540 Princeton, NJ, US

E-mail: andrea.pavone@ipp.mpg.de

**Abstract.** In this paper we describe a method for training a neural network to approximate the full model Bayesian inference of plasma profiles from an X-ray imaging diagnostic measurements. The modelling is carried out within the Minerva Bayesian modelling framework where models are defined as a set of assumptions, prior beliefs on parameter values and physics knowledge. The goal is to use neural networks for fast ion and electron temperature profile inversion from measured image data. The neural network is trained solely on artificial data generated sampling from the joint distribution of the free parameter and model predictions. The training is carried out in such a way that the mapping learnt by the network constitutes an approximation of the full model Bayesian inference. The analysis is carried out on images constituted of 20x195 pixels corresponding to binned lines of sight and spectral channels respectively. Through the full model inference, it is possible to infer electron and ion temperature profiles as well as impurity density profiles. When the network is used for the inference of the temperature profiles, the analysis time can be drastically reduced, up to the scale of tens of microseconds for a single time point compared to the $\approx 4$ hours long Bayesian inference. The procedure developed for the generation of the training set does not rely on diagnostic-specific features, and therefore it is in principle applicable to any other model developed within the Minerva framework. The trained neural network has been tested on data collected during the first operational campaign at W7-X, and compared to the full model Bayesian inference results.

PACS numbers: 52.25.Xz, 52.70.La, 52.55.Hc

Submitted to: *Plasma Phys. Control. Fusion*
*Keywords*: stellarator, x-ray imaging, neural network, Bayesian inference, modelling, real time, Minerva framework

## 1. Introduction

Neural networks (NNs) are a powerful tool when it comes to speed and approximation of complex functions. Universal approximation theorems have been shown to be valid for neural networks under different assumptions, as in [1], [2] and [3]. The real time capabilities of neural networks have also been shown in different fusion experiments, e.g., ion temperautre profile inference and disruption prediction at JET as in [4], [5] and [6] and at ASDEX Upgrade [7]. Neural networks have also been used for the reconstruction of plasma parameters from diagnostic data as in the case of charge exchange spectra automatic analysis at JET for reconstruction of ion temperature, rotation velocity and impurity density [8] and [9], and in the case of electron temperature from a soft-x-ray system at NSTX [10]. In this paper, we focus on an application of neural network algorithms on X-ray Imaging Crystal Spectrometer (XICS) measurements and we will describe an approach based on a different paradigm of neural network training and reconstruction suitable when the physics model of the diagnostic is available. In particular, we will make use of the Bayesian implementation of the model within the Minerva modelling framework [11].

Neural networks applied to diagnostic data are typically trained on real measurements and the corresponding quantities of interest in situations where a model of the problem is missing. Such approach has the advantage of providing the neural network with actually measured data, but it also has the limitation of depending on a fixed and restricted amount of training samples, the feature of which depends on the performed experiments, and on a limited parameter space. An exception is described in [9], where synthetic data has been introduced in the training set by sampling from the joint distribution of the physics parameters reconstructed from those inferred from the measurements, and by synthesizing the data with a forward model. The way we try to overcome these limitations is by training the network solely on data synthesised through the Bayesian model specified within Minerva, the same that is used for the standard inference [12]. The physiscal parameters used to produce the data are sampled from the corresponding prior distributions and the synthetised observations are sampled taking into account the error model. This gives control over the features we put in the training set and, consequently, the features the neural network will be learning and will be sensitive to when evaluating on measured data. Also, an advantage of this approach concerns its generality and the possibility of performing automatic data analysis based on physics models, which comes as a consequence of the sampling procedure described in the following chapters. This becomes of greater relevance as the scale of fusion experiments grows larger and the duration of plasma shots becomes longer. During the first operation campaign (OP 1.1, see [13]) of the W7-X stellarator several diagnostics ([14], [15]) were involved in the measurements, and the number increased during the second one (OP 1.2a). Together with the number of diagnostics, a large proportion of which are currently implemented in the Minerva framework, e.g. [16] [12] [17], also the duration of the plasma shots increased. All of this makes fast and automatic data analysis very desirable. The technical implementation of our approach only makes use of features shared between all models in Minerva, and thus it is easily transferable and applicable to other diagnostics modelled in the framework. The paper will describe the modelling within the Minerva framework, the basics features of the XICS diagnostic at W7-X, the neural network architecture and training set creation scheme. We will conclude comparing the evaluation of the neural network on OP 1.1 measured data to the Bayesian inference result traditionally carried out in Minerva and discussing the results mentioning possible further improvements.

## 2. The method

In order to describe the method we used to generate training data from Minerva models, we will first describe the basic features of the XICS diagnostic installed at W7-X and the corresponding model implementation within the Minerva framework. Afterwards we will illustrate the training set creation scheme.

### *2.1. XICS diagnostic*

The X-ray Imaging Crystal Spectrometer system collects x-rays emitted in atomic processes involving ion impurities and plasma electrons, occurring in the bean shaped cross section of the W7-X stellarator. The concept behind the diagnostic is described in [18]. A sketch of the view is shown in Figure 1a, where the plasma cross section and the line of sight span is shown. The system is equipped with a spherical bent crystal and

the light is collected onto a CCD detector producing 2D images similar to the one shown in Figure 1b. The two dimensions in the image represent energy and spatial resolution respectively. The diagnostic is sensitive to the energy region of He-like Argon emission lines. The main emission lines constituting the spectrum are shown in Figure 1c and they are the w, x, y and z for the $n = 2$ to $n = 1$ transitions in addition to numerous $n >= 2$ dielectronic satellites, e.g. the k lines for $n = 2$. A study of the spectrum and the atomic processes involved can be found in [19], [20] and [21]. The detector covers the wavelength range from $\approx 3.94\,\text{Å}$ to $\approx 4.00\,\text{Å}$ along the central line of sight (LOS).

In order to interprete the measured data, a forward model of the diagnostic is implemented within the Minerva framework [12], [22]. Here we will describe the atomic processes taken into account in the model, while we will discuss Bayesian inference and Minerva modelling in the next section. A detailed description of the calculation of the emission intensity for the different lines can be found in [19]. In Table 1 we show the processes involved in the calculations relevant to this paper together with their dependence on plasma parameters.

The intensity of all lines depends on the electron temperature $T_\text{e}$ through the corresponding effective rate coefficients, a calculation of which can be found in the Appendix of [19]. The dependency on the ion temperature $T_\text{i}$ comes as well into the calculation of all of the line shapes as Voigt profiles $V(\lambda_l, \lambda)$, which is the convolution of a Gaussian and Lorentzian shape, accounting for Doppler broadening and natural broadening, respectively. The photon emission of each process also depends on the electron density $n_e$, and on the density of Argon ions in one of the ionization stages: $n_{\text{ArHe}}$ for $\text{Ar}^{16+}$, $n_{\text{ArLi}}$ for $\text{Ar}^{15+}$, and $n_{\text{ArH}}$ for $\text{Ar}^{17+}$. All of the quantities in Table 1 are defined on a 3D Cartesian coordinate space, so that they are dependent on the position $\mathbf{x}$. The emission intensity $I(\lambda)$ at a given $\lambda$ is then calculated performing an integration along the line of sight paths $L$, as in the following equation:

$$I(\lambda) = \int_L n_e^2(\mathbf{x}) \sum_l V(\lambda_l, \lambda, \mathbf{x})\, i_l(\mathbf{x}) \tag{2.1}$$

The quantity $i_l(\mathbf{x})$ is defined according to:

$$i_l(\mathbf{x}) = \sum_{j(l)} n_j(\mathbf{x}) k_{lj}(T_e(\mathbf{x}))\, \mathrm{d}\mathbf{x} \tag{2.2}$$

It denotes the overall contribution from different ionization stages $j$ to the emission line $l$. In the equations, $\lambda_l$ denotes the wavelength for the given line, $k_{lj}$ denotes the effective rate coefficient of the line $l$ in the ionization stage $j$, and $n_j$ denotes the density of ions in the ionization stage $j$.

The contributions to the overall He-like Argon spectrum arising from the different atomic processes and ionization stages are shown in Figure 2. In the spectrum depicted, the lines q, r, and a, are also visible.

### *2.2. Bayesian modelling and inference in Minerva*

When developing a Bayesian modelling and inference scheme, the first step is the definition of the model free parameters, $w$, and observed data, $d$. Probability distributions are assigned to both quantities, and they take the name of *prior* distribution, $P(w)$, and *likelihood* function, $P(d|w)$. The prior distribution represents the *a priori* knowledge that we have about the free parameters before taking the observations into account. The likelihood distribution represents instead the model uncertainties in the prediction of the data. The inference is the process of knowledge acquisition when new data are observed. According to Bayesian probability theory, it can be described as an update process of the *a priori* distribution. This process is formally expressed through Bayes formula:

$$P(w|d) = \frac{P(d|w)P(w)}{P(d)} \tag{2.3}$$

The term $P(w|d)$ is called *posterior* distribution of the free parameter $w$ given the observed data $d$, and it represents the new state of knowledge on the model free parameter as new data are collected. The quantity $P(d)$ is called *evidence* or *prior predictive* and, as first interpretation, it plays the role of normalization factor:

$$P(d) = \int P(d|w)P(w)\, \mathrm{d}w \tag{2.4}$$

The hidden relevance of this term becomes evident when we switch from Bayesian inference for parameter estimation to Bayesian model selection, see for example [23]. Since the integral in equation 2.4 is a marginalization over the free parameters of a given model, we see that $P(d)$ is the distribution of all the data that the model can describe, quantifying how likely each data point is to be generated under the assumptions of the model. Once it is evaluated on a data point $d^*$, the evidence $P(d^*)$ defines how likely our model is as an explanation of such data point. The value of this probability depends on the prior distributions of the parameters and the uncertainties attributed to our model prediction in the likelihood term. The dependency is such that broader prior distributions, or prior distributions defined over higher dimensional parameter spaces, i.e. more complex models, will be penalized, and the overall probability will always be a compromise between model complexity and good fit to the data. This is indeed an application of *Occam's razor* principle. An illuminating explanation of how Bayesian model selection naturally embodies Occam's razor is given in [24].

The numerator in Equation 2.3 corresponds to the joint distribution $P(d, w)$ of observed data and

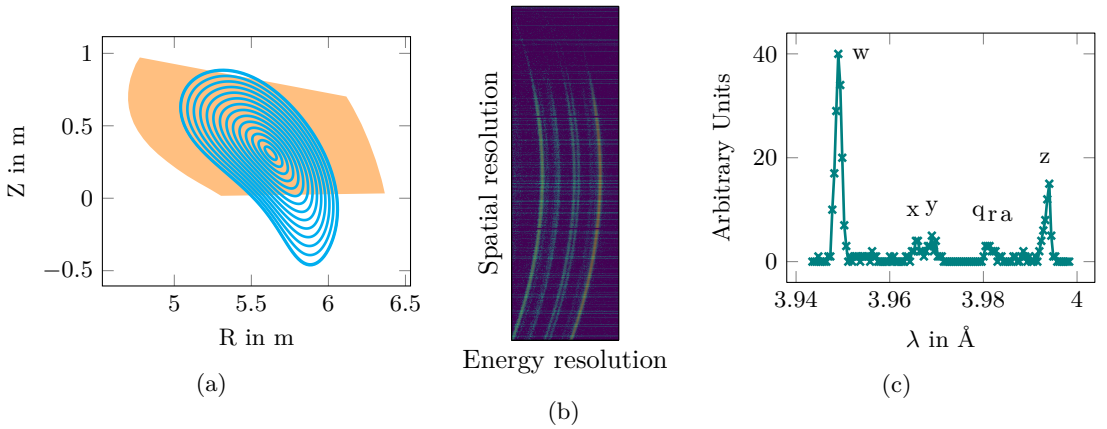


Figure 1: (a) Sketch of XICS system view on the bean shaped cross section at W7-X. (b) A measured raw image detected on the CCD detector. The typical curved feature is due to the spherical bending of the crystal. (c) He-like Argon spectrum measured along one of the central line of sights of the image in Figure (b). The main emission lines are marked with their names.

| Atomic process | Plasma parameter dependency |
|---|---|
| (1) excitation from ground state of He-like ions | $n_e, n_{ArHe}, T_e, T_i$ |
| (2) di-electronic recombination of He-like ions | $n_e, n_{ArHe}, T_e, T_i$ |
| (3) recombination of H-like ions | $n_e, n_{ArH}, T_e, T_i$ |
| (4) inner shell excitation of Li-like ions | $n_e, n_{ArLi}, T_e, T_i$ |

Table 1: Atomic process description and corresponding plasma parameter dependency.

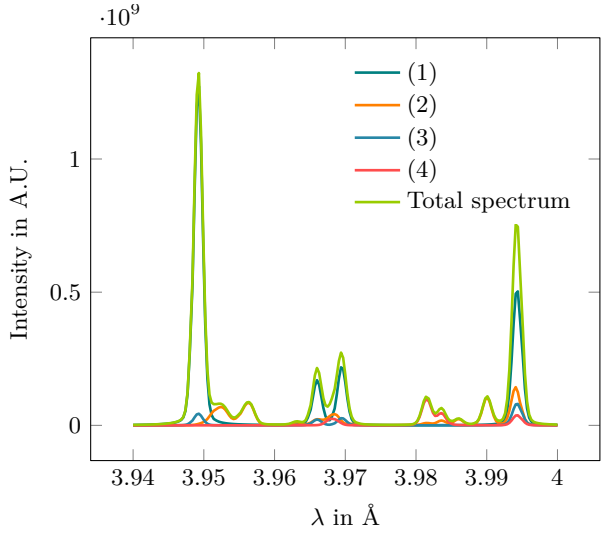


Figure 2: A He-like Argon spectrum calculated with the forward model for $T_i = 1$ keV, $T_e = 1.2$ keV, $n_e = 10^{13}$ cm$^{-3}$, $n_{ArHe} = 10^8$ cm$^{-3}$, $n_{ArLi} = n_{ArH} = n_{ArHe}/8$. The numbers in the legend refers to the atomic processes listed in Table 1

free parameter, $P(d, w) = P(d|w)P(w)$. The Minerva framework relies on graphical models [25] in order to express the conditional dependence between random variables in the model. For each model implemented in the framework, a *graph* object is created that describes the joint distribution of the free parameter and the measurements according to the forward model. A simplified version of the XICS model graph is shown in Figure 3. In the graph the coloured nodes are probabilistic nodes, where *orange* denotes the free parameters and *blue* denotes the observed quantities.

In the case of the XICS forward model, the free parameters can be the plasma parameter summarised in Table 1, right column, and the observed data are the images constituted of spectra like the one shown in Figure 2, calculated accounting for the atomic processes described in Section 2. All the distributions in the graph are chosen to be normal distributions. Note that the likelihood function is then a normal distribution centred at the model prediction, obtained with the given set of free parameter values:

$$P(d|w) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(d - y(w))}{2\sigma^2}\right) \tag{2.5}$$

where we used $y(w)$ to denote the forward model function $y$ dependent on the free parameter values $w$ and $d$ to denote a measured data point.

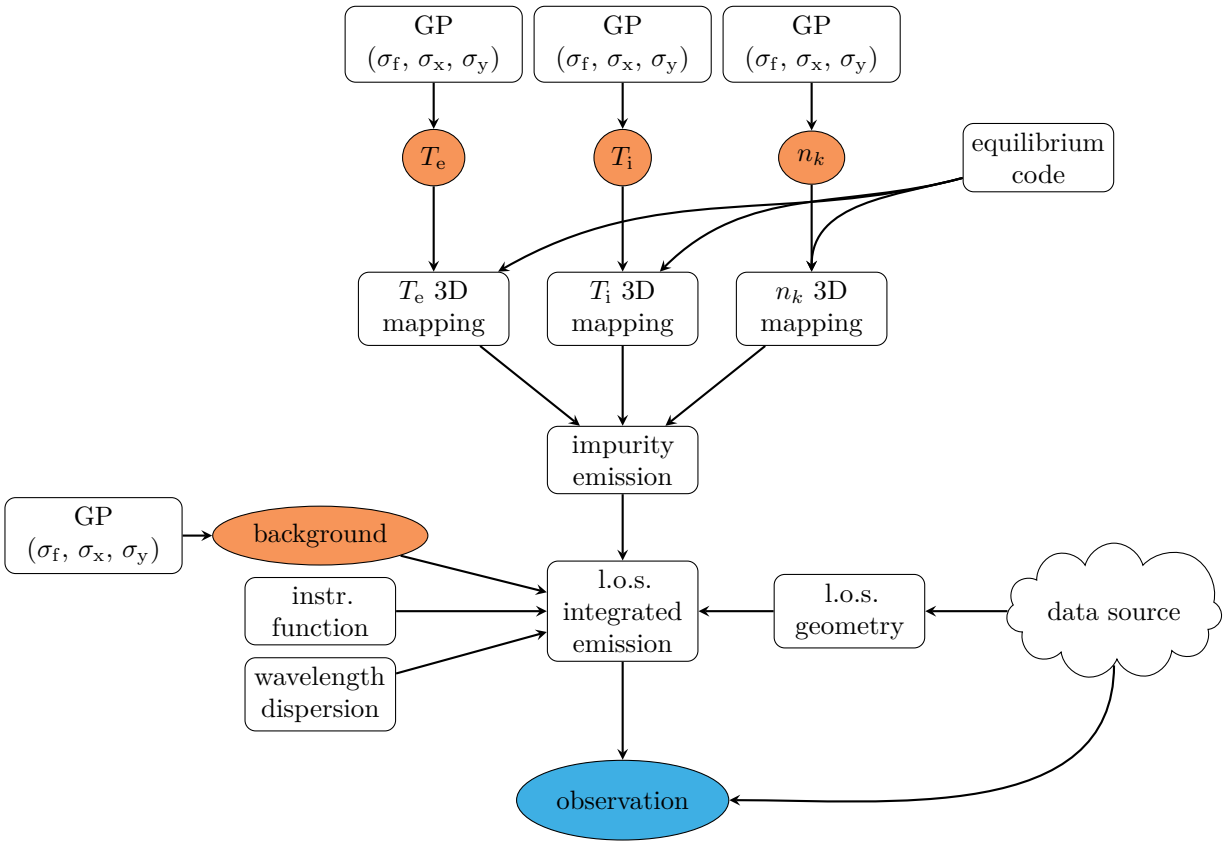


Figure 3: A simplified sketch of the XICS model graph. Coloured nodes are probabilistic nodes, where *orange* denotes the free parameters and *blue* denotes the observed quantities. White nodes represents deterministic calculation nodes. The white GP nodes represent a Gaussian process prior, and the symbols $\sigma_f$, $\sigma_x$ and $\sigma_y$ denotes the parameter in the expression of the squared exponential covariance function. The data source node is used to fetch diagnostic specific information and measurement from the W7-X Archive. The arrows represent direct or indirect dependencies in the probabilistic relations between the quantities in the probabilistic nodes.

The white squared nodes in the graph of Figure 3 represent deterministic calculation nodes, and the cloud node is used to denote a *data source*, i.e. a node that communicates with a database, here the W7-X ArchiveDB, where information about the diagnostic, e.g. geometry setup etc., are stored together with measured and analysed data. The arrows represent dependencies between nodes rather than a computational flow. For example, all arrows from the free parameters reach, directly or undirectly, the observed node, i.e. the probability distribution for the observed quantities ($d$) should be conditioned on the value of the free parameter, $P(d|w)$. The joint probability distribution represented by the whole graph can be factorized and written as: $P(w, d) = P(w)P(d|w)$.

The $T_e$, $T_i$ and $n_k$ profiles, where $n_k$ stands for Argon ion or electron density, are functions of the normalized effective radius $\rho_{\text{eff}} = \sqrt{\psi/\psi_{\text{LCFS}}}$. Thus, an equilibrium code, in this case VMEC [26], called from the corresponding node, is required to carry out the mapping to the 3-D Cartesian coordinates. The impurity emission can then be calculated locally and integrated along the line of sight. A background emission, is also added to the spectrum as a line of sight integrated quantity. In order to calculate the detector pixel response, the instrumental function and the wavelength dispersion on the chip are also required. Note that the data source node is used to provide the observed data for the inference and the l.o.s. geometry. The smoothness of the profiles is controlled by a zero mean Gaussian Process (GP) prior [27] with a squared exponential covariance function, as defined in:

$$\text{cov}_{ij} = \sigma_f^2 \exp\left(-\frac{(\rho_i - \rho_j)^2}{2\sigma_x^2}\right) + \delta_{ij}\sigma_y^2 \tag{2.6}$$

The quantity $\text{cov}_{ij}$ denotes the elements of the covariance matrix, $\rho_i$ and $\rho_j$ denotes the location of any two profile points labelled with $i$ and $j$, and the quantities $\sigma_f$, $\sigma_x$ and $\sigma_y$ denote the function variance, the length scale and the noise variance of the profiles, respectively. Note that the quantity $\text{cov}_{ij}$ corresponds to the covariance between any two points in the profile,

as function of the location $\rho_{\text{eff}}$. The length scale $\sigma_x$ describes how smooth a function is. Small length scale value means that function values can change quickly in its domain, whereas large values describe function that change slowly. The function variance $\sigma_f$ is a scaling factor and determines function value variations around the mean. Large values will allow again for bigger variation, smaller values will describe less varying function. It also determines, together with $\sigma_y$, the value of the covariance matrix elements along the diagonal, where $i = j$. The noise variance $\sigma_y$ is used to allow for noise present in the data and it specifies the amount of noise expected to be present in the data.

As data are measured, Bayesian inference can be carried out to extract information about the free parameters. The target of Bayesian inference is the posterior distribution of the parameters, $P(w|d)$. In the practical implementation within Minerva, when a single value solution is desirable, a Maximum A Posteriori (MAP) optimizer can be used to find the maximum of the posterior distribution. The full Bayesian answer to the inference problem is nevertheless provided by the full posterior distribution, the spread of which expresses the uncertainties on the inferred quantities. In order to provide this information, a Markov Chain Monte Carlo (MCMC) sampler is usually adopted to generate the posterior samples. The samples can then be stored and used in further, independent, calculations providing full non-linear uncertainty propagation. Profiles inferred using XICS data can then be used, for example, in impurity transport studies, see [28] and [22].

### *2.3. Generation of the training set*

The model depicted in Figure 3 represents a quite sophisticated inference problem: first of all, the images fed to the inversion routines go through a preprocessing stage, occurring in the data source node, where they are (1) straightened, and (2) binned along the line of sight direction in order to reduce the computational effort. At this point, the inversion problem consists of simultaneously fitting an image of 20x195 pixels along the LOS direction and the wavelength dispersion direction respectively, and doing a tomographic inversion of different plasma profiles. The full Bayesian inference takes from 1 to 4 hours for each measured image. A neural network trained on the problem of inferring plasma profiles from preprocessed observed images can process data at a time scale of tens of microseconds in good implementation conditions (e.g., on a GPU). In this paper we will focus on the inference of ion and electron temperature profiles.

In order to generate the neural network training set, we will only make use of the Minerva model. In Bayesian modelling, a model is defined by (a) its functional form, which in this case includes all the calculations based on the physics knowledge we have put in the forward model, *and* (b) two probability distributions: the prior distribution of the free parameters, $P(w)$, and the likelihood function of the observed data $P(d|w)$. Indeed, all the features of a model are reflected in its joint distribution $P(d, w) = P(d|w)P(w)$: the distribution of the variables $(d, w)$ depends on the functional form of the forward model, appearing in the likelihood term $P(d|w)$ as $y(w)$ (Equation 2.5) and which expresses the dependence of $d$ on $w$, the uncertainties on the model prediction and the prior distribution $P(w)$. As our goal is to create a (approximated) copy of the original full Bayesian model, we must provide the neural network with training data having the same properties: this is achieved by generating the training set data from samples of the full model joint distribution. Practically, such training set can be obtained by iterating over the following three steps:

i draw and store a sample from the joint prior distribution of the free parameter: $P(w) = P(T_{\text{e}}, T_{\text{i}}, n_{\text{k}}, bg) = P(T_{\text{e}})P(T_{\text{i}})P(n_{\text{k}})P(bg)$, where $bg$ denotes the background l.o.s. integrated emission profile

ii run the forward model in order to calculate a synthetic observation with the given free parameters

iii store a number of samples drawn from the likelihood function of the synthetic observations, $P(d|w)$, which is fully specified by the given set of sampled free parameter and the model uncertainties

The sampling procedure taking place at step (iii) will introduce noisy samples in the training set since the likelihood distribution expresses the uncertainties of the model prediction. This will help making the neural network stable against small perturbations in the input data when evaluated on measured images. This is equivalent to the technique known as *data augmentation*, [29] and [30]. The modifications we inject into the samples are based on the noise model that has been assumed for the problem, in this case a Gaussian noise model. The neural network is then trained on the mapping from the images to the ion and electron temperature profiles. Training on other profiles is also possible and straightforward, since it is just matter of choosing and storing another set of samples among those stored at step (i). As a consequence of such sampling procedure, the portion of the training set corresponding to the to the model prediction $d$ are samples from the evidence term $P(d)$ of Equation 2.3.

A sketch of the procedure is illustrated in Figure 4. The size of the input images is 20 pixels along the LOS dimension and 195 pixels along the wavelength dimension. The target ion and electron temperature profiles are defined with 15 points equally spaced along the effective radius. The training set is made of 500 000

samples. A test set made of 10 000 samples is used to check the generalization capabilities of the NN during training and it is generated in the same way as the training set.

Given the previously mentioned sampling procedure, an insightful interpartion can be given to the neural network mapping. A well known result [31] in the neural network field states that, under the assumption of a sum-of-squares error loss function as in Equation 2.9, large training data set and successful optimization, the neural network mapping $f$ is given by the conditional average of the target data $y_i$, conditioned on the input vector $\mathbf{x}_i$:

$$f(\mathbf{x}_i; \mathbf{w}) = \langle y_i | \mathbf{x}_i \rangle \tag{2.7}$$

In the specific contest of our study, the network's targets and input are the ion and electron temperature profiles and the synthetic XICS images. Given the fact that the training set is generated sampling from the join distribution of the XICS Bayesian model, as described in Section 2.3, the distribution of the target data $t$ given an input vector $\mathbf{x}$, $p(t|\mathbf{x})$, in the limit of large training data set, correspond to the posterior distribution of the ion or electron temperature profile of the full Bayesian model. Therefore, we can state that the neural network mapping, in ideal circumstances, is given by the mean of the full model posterior distribution. In real world circumstances, the data set size is finite, and the optimization is never perfect, so that we can say that, in this sense, the inversion provided by a neural network trained in such a way, constitute an *approximation* of the full model Bayesian inference.

### *2.4. Neural network input, output and architecture*

It is worth to summarise here what the neural network input and output are at the different stages. At **training time**:

- *input*: synthetic images, generated with the XICS forward model and the sampling procedure described in Section 2.3. These images supposedly closely resemble the actual XICS measurement (after few pre-processing operation, i.e. row binning and straightening, see next bullet point and Figure 5). They are made of 20x195 pixels/values.
- *target*: the ion or electron temperature profiles used to generate the corresponding images. These are made of 15 points, along the effective radius $\rho_{\text{eff}} = \sqrt{\psi/\psi_{\text{LCFS}}}$, where $\psi$ is the magnetic flux and $\psi_{\text{LCFS}}$ is the flux at the last closed flux surface.

At **evaluation time**:

- *input*: the pre-processed, actual XICS measurements. The measured images, originally showing the curved feature shown in Figure 1b, are straightened according to the detector and crystal geometry. Afterwards, the original 1475x195 pixels of the image are binned along the largest dimension, which corresponds to the spatial resolution, where neighbouring line of sights overlap significantly. The binned images are made of 20x195 pixels. An example of a binned image is shown on the leftmost side of Figure 5.
- *output*: the estimated ion or electron temperature profile. In case of successful training, it will match, within uncertainties, with the profile inferred through the full Bayesian model.

Essentially, two neural networks, identical for all the features except for the target profiles, have been trained and tested independently: one for the inversion of ion temperature profiles and the other for the inversion of the electron temperature profiles.

Since the network's input are 2-D images, the network architecture has been inspired by the LeNet-5 Convolutional Neural Network (CNN) [32] and is shown in Figure 5. This kind of architecture has been shown to be particularly effective when the input present a 2-D structure. It has been successfully used in many image recognition problems, achieving state-of-the-art results [33], [34]. Here we expect to recurrently find across the image the features induced in the spectrum by the ion or electron temperature profiles, which affect line width and intensities, respectively. Two convolutional layers C1 and C2, each one followed by one sub-sampling layer, are used in a hierarchical feature extraction structure. A convolutional layer apply a convolution filter or kernel to the input image, extracting information that are recurrent accross different location in the image (for more details see [32]). The kernel dimension sizes used in the convolutiona layers are respectively: (3, 16) and (2, 5), where the first and second dimensions refer to the LOS and wavelength dispersion dimension of the input images. The number of *feature maps*, i.e. sets of units whose weights are constrained to be identical [35] and [36], is set to 30 for both convolutional layers. The sub-sampling layers use *max pooling* with a resolution of 2 by 2. Two fully connected layers M1 and M2 made of 20 and 18 units respectively, constitute the final layers, which will produce the desired 15 points ion or electron temperature profile output. The activation function in the convolutional layers is the Rectified Linear Unit (ReLU):

$$f(x) = \max(0, x) \tag{2.8}$$

whereas in the fully connected layers it is the hyperbolic tangent function.

Convolutional neural networks are also especially suitable for parallelisation on GPUs, which applied to our case made the training 30 times faster than on CPUs.

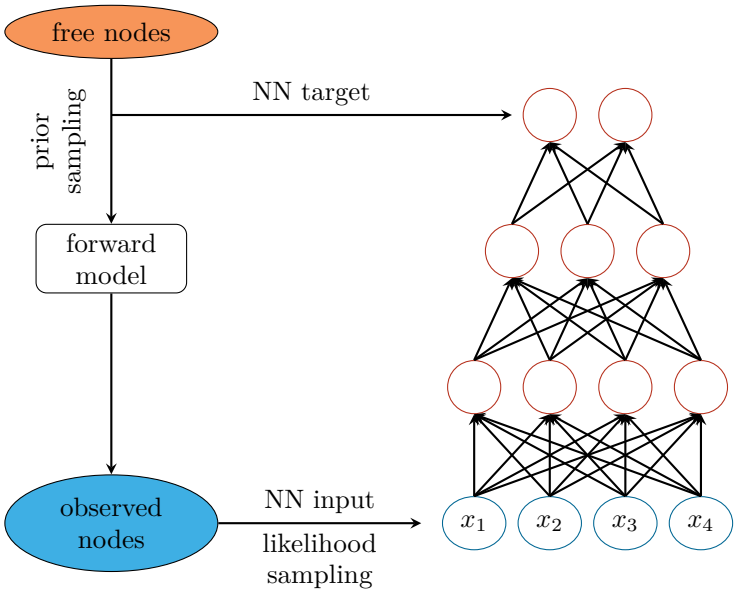


Figure 4: A sketch to illustrate the sampling procedure for the training set creation described in Section 2.3. A sketch of the Minerva model of the XICS diagnostic and the neural network are shown on the left and on the right, respectively. The NN takes as input images sampled from the likelihood function of the model, given a set of sampled free parameters. The blue nodes of the neural network denotes the input pixel of the image and the two red nodes at the top denotes the output points of the ion temperature profile.

The neural network was implemented within Theano, a Python framework for fast symbolic computation [37].

The training was stopped either according to an early stopping criteria, i.e. when the network performance on the test starts degrading, or when the decay rate of the loss function is small enough. The loss function that the NN is trained to minimize is defined as:

$$S(\mathbf{w}) = \frac{\beta}{2}\sum_{n=1}^{N}(y^n - t^n)^2 + \sum_{k,i=1}^{L,N_k}\alpha_k w_{k,i}^2 \tag{2.9}$$

where $\mathbf{w}$ denotes the network weight vector, the first term on the right-hand side is the sum-of-square error between the network output $y_n$ and the target $t_n$ and $n$ is an index that goes through the $N$ samples in the training set. The second term on the right-hand side is a regularizing term, where $w_{k,i}$ denotes the network weight of unit $i$ at layer $k$. The parameter $\beta$ and $\alpha_k$ are scale parameter which control the relative importance of the two terms. An insightful interpretation, based on a Bayesian view of the neural network training [24] [38], can be provided to such expression and can be useful in the choice of the values of $\beta$ and $\alpha_k$. The values of $\beta = 10$ and $\alpha_k = (\alpha_{C1}{=}68.00,\ \alpha_{C2}{=}58.33,\ \alpha_{M1}{=}576.67,\ \alpha_{M2}{=}5.83)$ were used. More details about Bayesian neural network training in the context of this study can be found in [39].

### *2.5. Training set comparison*

The features of the set of measurable images $D_m$ are determined by the properties that the plasma profiles have during the experiments. An absolutely free, unconstrained sampling of the 15x7 points in the plasma profiles introduced in Table 1, will produce a set of synthetic data $D$ which likely will have little in common with $D_m$. Most of the samples in such a training set will have little use to our purposes, since they would not belong to the domain of the mapping that we want the neural network to learn. In order the neural network to be able to accurately predict temperature profiles from measured images, the $D_m$ space has then to be covered densely enough by the training data set: we are not interested in generating all possible 15-dimensional output vectors, but only those which represent realistic ion or electron temperature plasma profiles. This is accomplished by refining the prior distributions in such a way that sampling from them will generate a data set of synthetic images which densely encloses $D_m$.

In principle, the full distribution of the data described by a given model is determined by the prior predictive distribution, equation 2.4. As we have seen in section 2.3, the training images constitute samples from such distribution, therefore they can be used to estimate how closely a training set resemble the actual measurements.

Several methods can be used to study the adequacy of the training set to the coverage of the $D_m$ space of the

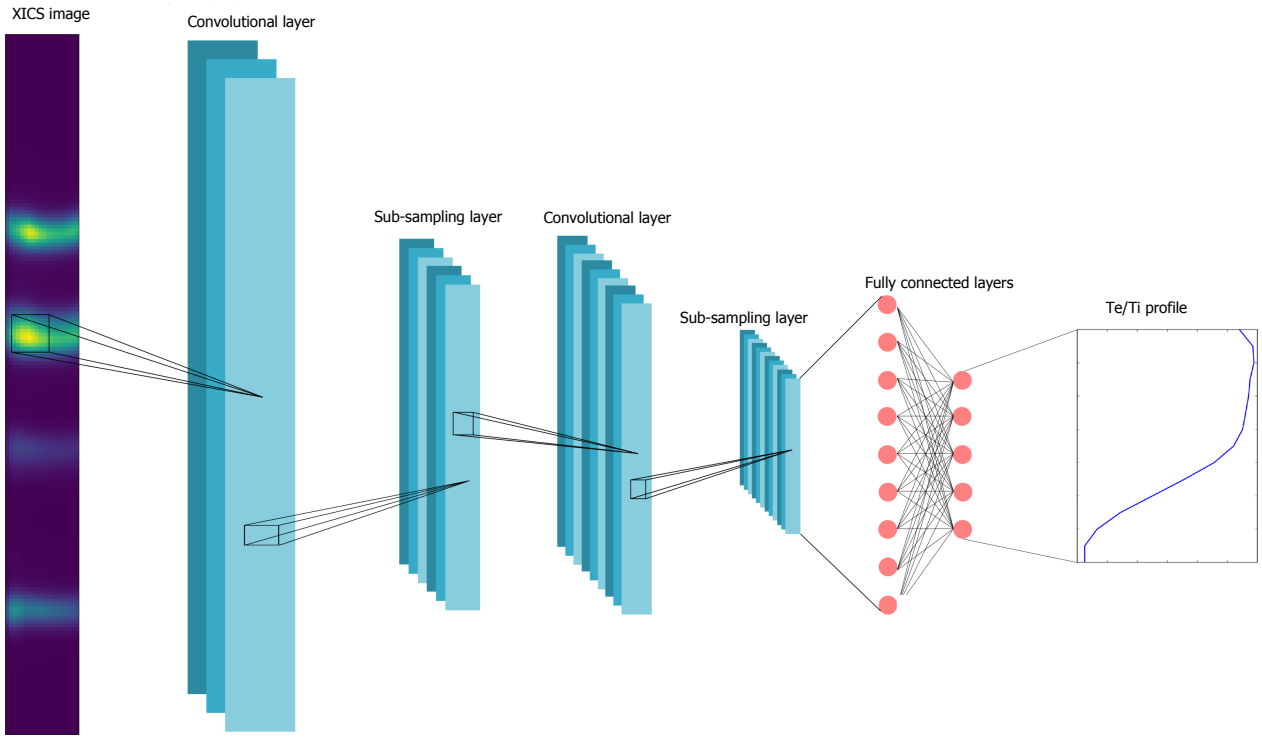


Figure 5: Architecture of the NN used. The input layer at the leftmost side is followed by a convolutional layer and a sub-sampling layer, which are followed again by a couple of convolutional and sub-sampling layers. Two fully connected feed forward layers follow up to the output layer. Each blue plane represents a feature map, where all the units share the same weights.

measured images. Here we will describe an approach which relies on the *k-nearest neighbours* algorithm (k-NN). This algorithm is used to find a number $k$ of data points in the training set that are the closest to a given observed data point, according to a metric measure. In this case the Euclidean distance has been used. We expect that the distance of a measured image from the samples in the training set to be larger in the cases where the measurement is not properly described by the training set samples compared to the distance of the test set samples, on which the neural network we know perform well, from the same training samples. Distance based methods are often used in the framework of outlier and novelty detection and similar methods are presented for example in [40] and [41]. An application in our study is shown in Figure 7, where we have compared two training sets obtained with two different models. The difference in the models is in the prior distribution of the plasma profiles: in one case, labelled as W/O, the temperature profiles were left unconstrained in the region of the Last Closed Flux Surface (LCFS), being allowed to assume any value between 0 keV and 10 keV; in the other case, labelled as W., the profiles were constrained to assume low values in the LCFS region (0.1 keV +- 0.5 keV) at $\rho_{\mathrm{eff}} = 0.99$), a feature that is typically expected in such plasma profiles. Such a constraint enters the Minerva model as a so called *virtual observation*: at the level of the Minerva graph, this corresponds to a standard observed node connected to the profiles which states that the value of the profiles at the given position $x_p$ has been "virtually measured" to have value $v_p$ with error $\epsilon_p$, as shown in Figure 6. The only difference with the other observed nodes in the graph is that it does not correspond to an actual measurement. Its role is to constrain the profile shape, and this is what observations in Bayesian models do: they constrain the solution found for the free parameters. In Figure 6, the dashed arrow and box represent the connection to the rest of the model of Figure 3. The computation node on the left of the dashed one, represents the evaluation of the $T_e$ profile at the $\rho_{eff} = 1.0$ position, which corresponds to the LCFS. This node is finally connected to the virtual observation, the blue circle, whose normal distribution is specified in term of its mean and standard deviation. If we exclude the graph represented in this figure from the rest of the model, and we then calculate the covariance matrix of the posterior distribution of the plasma profile, in this case $T_{\mathrm{e}}$, given the virtual observation, we will find a covariance matrix which can be used later on to

sample profiles which will feature the desired constraint. This has been done in order to obtain more realistic plasma profiles, and the effect of this on the training set is described in the next paragraph. Specifically, the virtual observation was implemented as a normal distribution with mean value of $0.1keV$ and standard deviation of $0.05keV$ at $\rho_{\text{eff}} = 0.99$, for the ion and electron temperature profiles.

The importance of sampling a realistic set of plasma profiles, which correspond then to a realistic set of synthetic measurements, is depicted in Figure 7. The three plots on the top show a spectrum measured along three different line of sights (blue line), and the 10 nearest neighbours (10-NN, grey lines) found among the samples in a training set where the electron temperature profiles were sampled without constraints (W/O) on the value assumed on any of the flux surfaces. The only constraint was a smoothness criteria induced by the GP prior. From left to right, the line of sight of the plots are traversing the following regions of the machine: edge, halfway to the core and core. The intensity on the y-axis is normalized to the brightest pixel in the image, in this case a w spectral line along one of the central line of sight. The three plots on the bottom, instead, show the same measured image and the 10-NN this time found among the samples in the training set with constraints (W.). The effect of such constraint on the sampled profiles is shown in Figure 8, where 100 samples from each of the two training sets are drawn. It is evident that when the constraint is applied (bottom plot), the average electron temperature through the machine is lower. This brings the training samples closer to the measured data in two ways: (1) the intensities measured along the edge line of sights (see leftmost bottom plot in Figure 7) show a smaller signal to noise ratio, (2) the ratio between different emission lines is smaller, see middle and rightmost plots in Figure 7. The distance of a measured data point from the 100000 nearest neighbours in the training set can be compared to the distance of 10 test set samples from 100000 training set samples to get an estimation of the proximity of the measured data with respect to the training samples. These values are shown in Table 2 in the two cases of training set created with and without constraints. The distance for the measured data point drops is substantially reduced when the constraint are applied to the electron temperature profile prior distribution, getting closer to the value of the test set sample. It is worth to note that, even with the constraint, the sampled profiles are not necessarily monotonically decreasing, as shown in Figure 8.

The improvement on the prediction capabilities of the neural network when applied to the measured image is remarkable and it is shown in Figure 9. The blue line in the plot denotes the mean value of 800 000 samples of ion temperature profiles drawn from the posterior distribution, inferred within Minerva, and the corresponding standard deviation. The agreement between the neural network prediction and the full Bayesian inference result is visibly better when the constraint is applied to the plasma profile prior distribution of the model.

| | Test set samples | Measured data |
|---|---|---|
| W/O constraint | 47.8 | 1186.1 |
| W constraint | 63.5 | 198.4 |

Table 2: The average distance from the measured data point to the 100000 nearest neighbours in the training set is compared to the average distance of 10 test set samples from the corresponding 100000 nearest neighbours in the training set, in the cases where the training set is created with (W) and without (W/O) constraint on the $T_{\text{e}}$ profile prior distribution.

## 3. Results

A neural network with architecture described in Section 2.3 and illustrated in Figure 5 has been evaluated on data from plasma shots of the first operational campaign at W7-X.

The prior distributions of the free parameters of the model used for the creation of the training set were all normal distribution functions with lower truncation at 0.0 keV. The $T_e$ profile prior distribution had also upper truncation at 10 keV, whereas the other profiles had none. The values of the parameters of the GP squared exponential function defined in Equation 2.6 were set to $\sigma_{\text{f}} = 2.0$, $\sigma_{\text{x}} = 0.3$, and $\sigma_{\text{y}} = 10^{-3}\,\sigma_{\text{f}}$ for the $T_i$ profile and to $\sigma_{\text{f}} = 5.0$, $\sigma_{\text{x}} = 0.3$, and $\sigma_{\text{y}} = 10^{-3}\,\sigma_{\text{f}}$ for the $T_e$. Moreover, the constraints discussed in Section 2.5 were applied to the temperature profiles but not to the density profiles. The magnetic configuration was kept fixed during the sampling procedure and the NN was tested on data from shots with such configuration. A comparison between the standard Bayesian inference carried out within the Minerva framework and the neural network inversion is shown in Figure 10, for both ion and electron temperature profiles. The different plots in the figure refer to data from different shots and time points within a shot. The profiles show a good agreement within the uncertainties. Concerning the neural network estimate, the predicted profiles have been obtained from a committee of networks: a set of 10 neural networks with same architecture, but different weight initialization, has been trained on the same training set. The error bars are calculated in a Bayesian fashion: the training procedure is seen as an inference problem

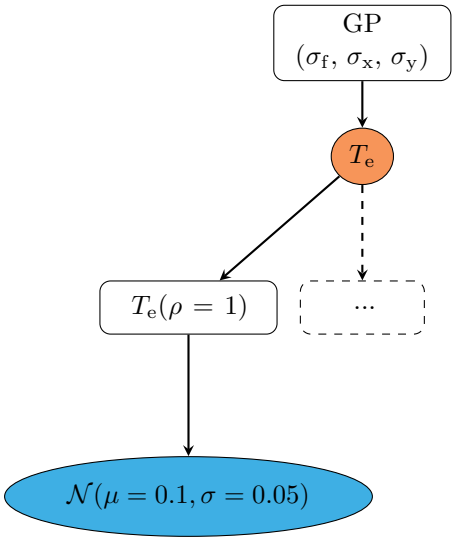


Figure 6: A sketch of the virtual observation constraint applied to the $T_e$ profile. The blue circle represents the so called *virtual observation*, which states that the $T_e$ profile has been "observed" to have value 0.1 keV with standard deviation 0.05 keV, at the LCFS ($\rho = 1$).

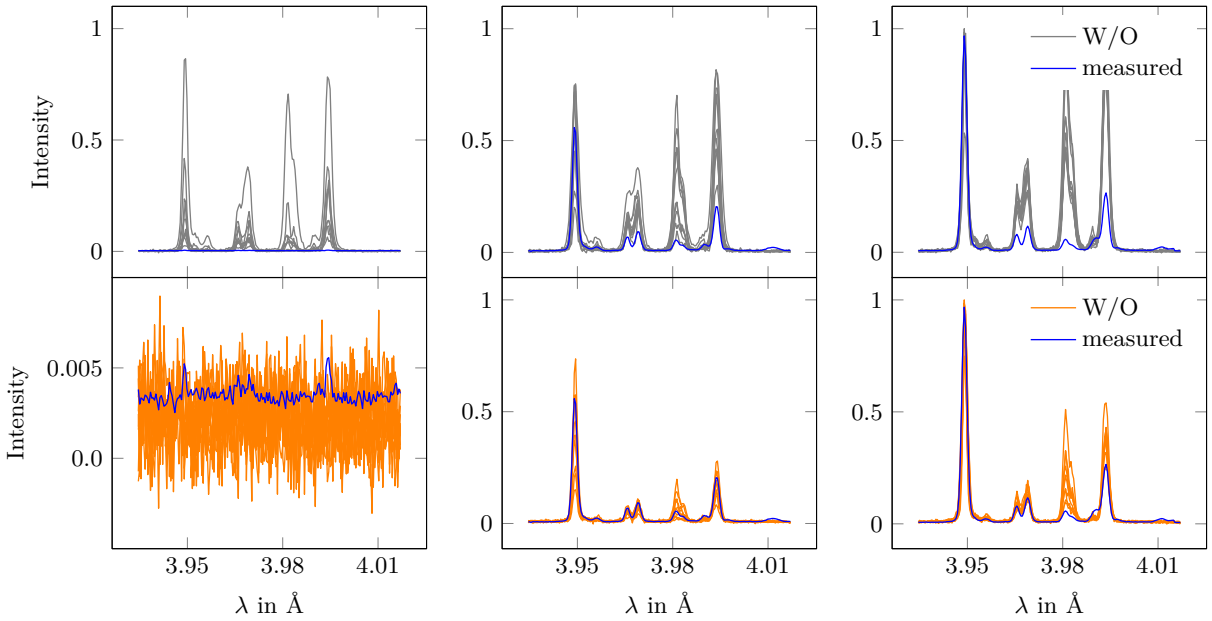


Figure 7: The k-NN algorithm is applied to find the 10 nearest neighbours to a measured image (blue line) among the training samples in two different training sets: (1) the $T_e$ profiles are let vary without (W/O) constraints on the values they assume towards the edge of the machine (three top plots), (2) the $T_e$ profiles are sampled with (W.) constraints to low values toward the edge, as described in the text (three bottom plots). The constraints are applied to the prior probability distributions. Each plot in the three columns represents the spectrum along a edge, half-way to core, and core line of sight, from left to right respectively.

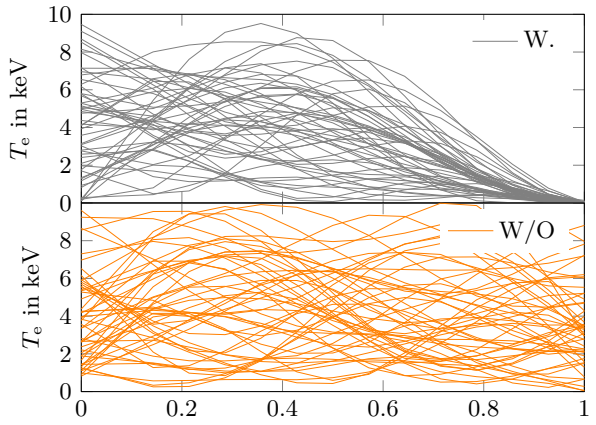


Figure 8: The $T_e$ profile samples from the two training sets. Top, the profiles are sampled without (W/O) constraints. Bottom, the profiles are sampled with constraints (W).

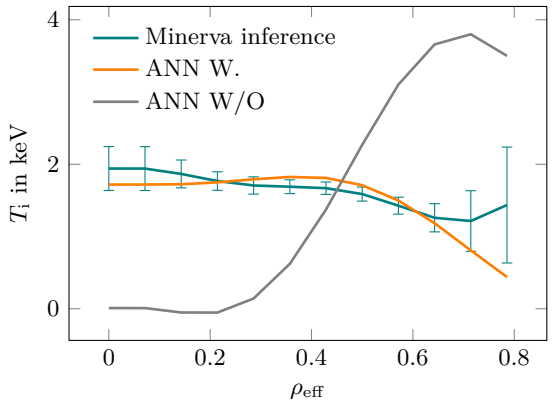


Figure 9: The standard Minerva Bayesian inference of $T_i$ profile is compared to the neural network inversion in two training cases. The orange and grey lines represent the output of the neural network trained on the training sets where the $T_e$ profiles are sampled with (W) and without (W/O) constraints, respectively.

on the network's weights, giving as result a posterior distribution of the network's weights approximated with a Gaussian distribution centred on the network's weight vector found with the training process and whose standard deviation calculation depends on the Hessian matrix of the loss function with respect to the weights. The spread in the posterior produces then a spread in the network's prediction, and this is the source of the network's error bars. This procedure has been applied to the 10 networks in the committee. The committee prediction is then obtained by sampling a random member network, sampling a set of weights from the corresponding weight's posterior and then feeding the network with a sample of the input vector drawn from the XICS noise model. This corresponds to approximating the overall weight's posterior with a multi-Gaussian approximation, where each Gaussian is obtained from the single Gaussian approximation carried out for each member of the commitee. Moreover, in this way, the error bars shown in Figure 10 includes also uncertainties in the XICS measured data. The calculation of the error bar is described in detail in [39].

The full Bayesian model prediction is obtained as the average of 800 000 samples drawn from the posterior distribution of the corresponding free parameter, obtained with a MCMC sampler, and the error bars are obtained as the standard deviation of the samples.

It is important to note that the training set has been generated sampling with the LCFS constraints described in Section 2.5: as a consequence, the profiles predicted by the networks show all small variance toward $\rho = 1$. This is the reason why the uncertainties in the ANN output in Figure 10 are systematically lower for larger values of $\rho_{\text{eff}}$.

It is worth to notice that the speed-up introduced by the neural network analysis of the data is substantial: the evaluation of one single data point takes $\sim 10\,\mu$s on a single CPU. The inference with MCMC sampling takes a few hours in similar conditions, thus the speed-up is of $10^9$ order of magnitudes.

## 4. Conclusions

We've shown a Bayesian model oriented approach to neural network training for the inference of ion and electron temperature profiles from data measured with an X-ray imaging diagnostic at W7-X. The model implemented within the Minerva framework is used to generate the training set, sampling from the prior distribution of the free parameters and from the likelihood function of the simulated data. This training procedure constitutes a different approach to conventional neural network training: starting from the full Bayesian model we are able to generate neural network approximation of full Bayesian models. Indeed, machine learning algorithms such as neural networks are applied on problems where a model describing the data is missing, thus leaving little or no room to the interpretation and control of the learning features. Instead, with the approach described here, we earn some control over the infamous black box of neural network learning: we know, for example, that the trained network will only be as good as our physics model, and we know beforehand the features of the distribution of the data in the contained training set, which is determined by the joint distribution of the Bayesian model.

Since we can manipulate the prior probability distribution functions, we can also knowingly choose the

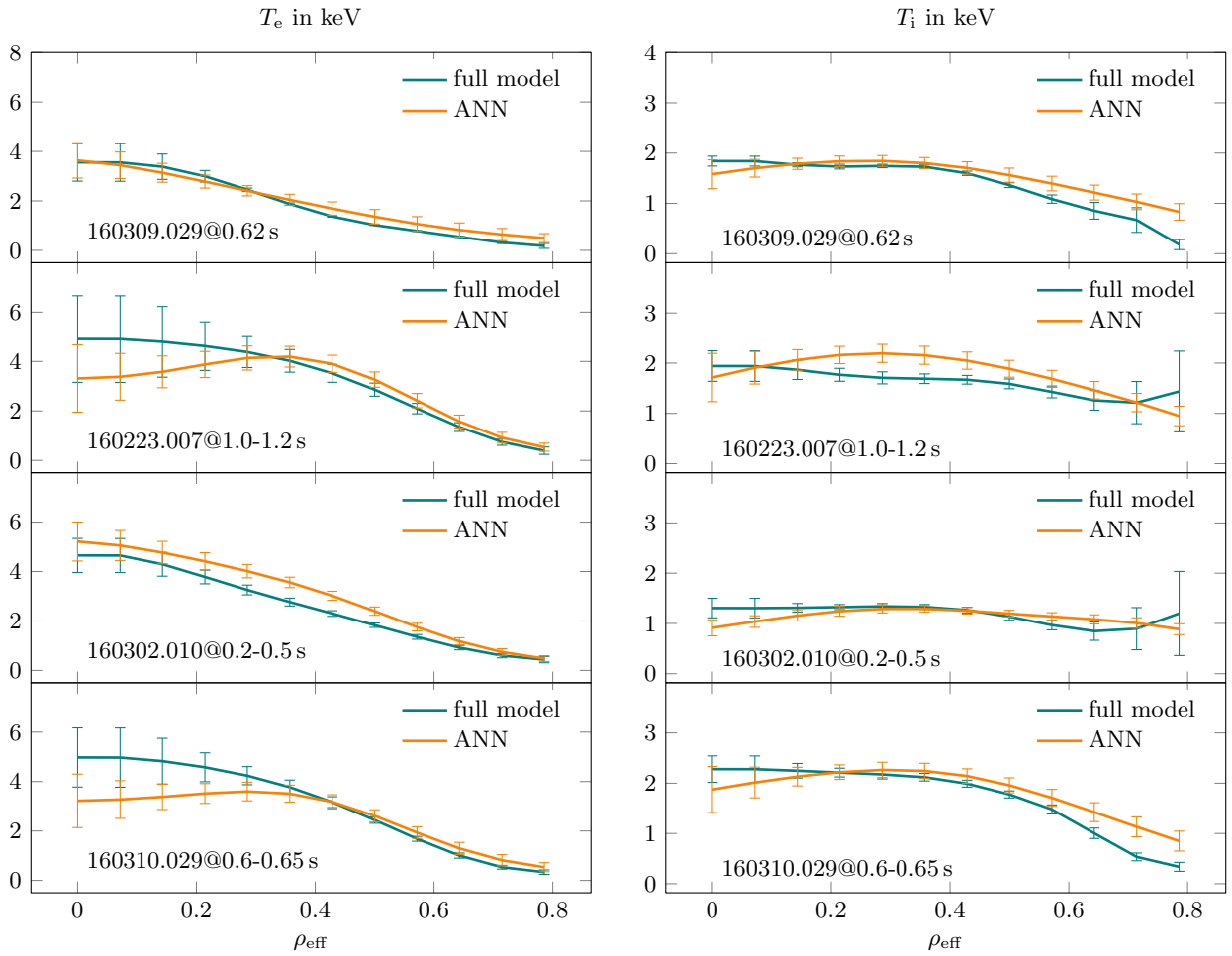


Figure 10: The results of the NN inversion compared to what obtained with standard Bayesian inference for different plasma shots. The left and right columns show $T_e$ and $T_i$ profiles respectively.

model that better describe the measurements we expect to perform. This gives us the chance to take control on the training of the network from the privileged point of view of the physics parameters that, through the forward model, describe the data we expect to measure. This is exactly what allowed us to find a better training set for more accurate neural network predictions.

The neural network has then been tested on measurements from different plasma shots from the first operational campaign at W7-X and compared with the results of the standard Bayesian inference. The first major advantage of this approach is the speed-up of the analysis, which can be carried out in tens of microseconds with NNs. The second advantage is that the sampling procedure necessary for the creation of the training set requires only abstract, not diagnostic-specific, features of the implemented model and thus is in principle automatically applicable to any other diagnostics developed within the Minerva framework.

## 5. Acknowledgement

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 and 2019-2020 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

[1] G. Cybenko. Degree of approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2:303–314, 1989.

[2] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[3] S. Sonoda and N. Murata. Neural network with unbounded activation functions is universal approximator. *arXiv e-prints*, abs/1505.03654, 2017.

[4] J. Svensson et al. Real-time ion temperature profiles in the JET nuclear fusion experiment. *ICANN 98. Perspectives in Neural Computing*, 1998.

[5] B. Cannas, A. Fanni, P. Sonato, and M. K. Zedda. A prediction tool for real-time application in the disruption protection system at JET. *Nuclear Fusion*, 47(11):1559–1569, 2007.

[6] S. Y. Wang et al. Prediction of density limit disruptions on the J-TEXT tokamak. *Plasma Physics and Controlled Fusion*, 58(5):055014, 2016.

[7] G. Pautasso and C. Tichmann. On-line prediction and mitigation of disruptions in ASDEX Upgrade. *Nuclear Fusion*, 42, 2002.

[8] C. M. Bishop, C. M. Roach, and M. G. von Hellermann. Automatic analysis of JET charge exchange spectra using neural networks. *Plasma Physics and Controlled Fusion*, 35:765–773, 1993.

[9] J. Svensson, M. von Hellermann, and R. W. T. König. Analysis of JET charge exchange spectra using neural networks. *Plasma Physics and Controlled Fusion*, 41:315–338, 1999.

[10] D. J. Clayton et al. Electron temperature profile reconstructions from multi-energy SXR measurements using neural networks. *Plasma Physics and Controlled Fusion*, 55(9), 2013.

[11] J. Svensson and A. Werner. Large Scale Bayesian Data Analysis for Nuclear Fusion Experiments. *IEEE International Symposium on Intelligent Signal Processing*, pages 1–6, 2007.

[12] A. Langenberg, J. Svensson, H. Thomsen, O. Marchuk, N. A. Pablant, R. Burhenn, and R. C. Wolf. Forward modeling of X-ray imaging crystal spectrometers within the Minerva Bayesian analysis framework. *Fusion Science and Technology*, 69(2):560–567, 2016.

[13] R. C. Wolf, C. Beidler, M. Beurskens, et al. Major results from the first plasma campaign of the Wendelstein 7-X stellarator. *Nuclear Fusion*, 2017.

[14] R. König et al. The set of diagnostics for the first operation campaign of the Wendelstein 7-X stellarator. *Journal of Instrumentation*, 10(10), 2015.

[15] M. Krychowiak et al. Overview of diagnostic performance and results for the first operation phase in Wendelstein 7-X (invited). *Review of Scientific Instruments*, 87, 2016.

[16] S.A. Bozhenkov et al. The thomson scattering diagnostic at wendelstein 7-x and its performance in the first operation phase. *Journal of Instrumentation*, 12(10):P10004, 2017.

[17] Udo Hoefel et al. Bayesian modelling of microwave radiometer calibration on the example of the wendelstein 7-x electron cyclotron emission diagnostic. 2018.

[18] M. Bitter et al. Objectives and layout of a high-resolution x-ray imaging crystal spectrometer for the large helical device. *Review of Scientific Instruments*, 81(10):1–5, 2010.

[19] O. Marchuk, R. Wolf, and H. J. Kunze. *Modeling of He-like spectra measured at the tokamaks TEXTOR and TORE SUPRA*. PhD thesis, 2004.

[20] TFR Group et al. Dielectronic satellite spectrum of heliumlike argon: A contribution to the phzsics of highlz charged ions and plasma impuritz transport. *Physical Review A*, 32(4):2374–2383, 1985.

[21] L. A. Vainshtein and U. I. Safronova. Wavelengths and transition probabilities of satellites to resonance lines of H- and He-like ions. *Atomic Data and Nuclear Data Tables*, 21(1):49–68, 1978.

[22] A. Langenberg et al. Inference of temperature and density proles via forward modeling of an x-ray imaging crystal spectrometer within the minerva bayesian analysis framework. Submitted to Review of Scientific Instruments, 2018.

[23] D. Sivia and J. Skilling. *Data Analysis: A Bayesian Tutorial*. Oxford University Press, 2006.

[24] David J C MacKay. *Bayesian Methods for Adaptive Models*. PhD thesis, California Institute of Technology, 1991.

[25] Judea Pearl. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29(3):241 – 288, 1986.

[26] S. P. Hirshman and J. C. Whitson. Steepestdescent moment method for threedimensional magnetohydrodynamic equilibria. *The Physics of Fluids*, 26(12), 1983.

[27] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[28] A. Langenberg et al. Argon Impurity Transport Studies at Wendelstein 7-X using X-ray Imaging Spectrometer Measurements. *Nuclear Fusion*, 57, 2017.

[29] Eric Kauderer-Abrams. Quantifying translation-invariance in convolutional neural networks. *arXiv e-prints*, abs/1801.01450, 2017.

[30] Sebastien C. Wong, Adam Gatt, Victor Stamatescu, and Mark D. McDonnell. Understanding data augmentation for classification: when to warp? *CoRR*, abs/1609.08764, 2016.

[31] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[32] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[33] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1725–1732, June 2014.

[34] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[35] Yann Lecun. *Generalization and network design strategies*. Elsevier, 1989.

[36] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, R. E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Handwritten digit recognition with a back-propagation network. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 396–404. Morgan-Kaufmann, 1990.

[37] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.

[38] R. M. Neal. *Bayesian Learning for Neural Networks*. PhD thesis, University of Toronto, 1995.

[39] A. Pavone et al. Bayesian uncertainty calculation in neural network inference of ion and electron temperature profiles at w7-x. *Review of Scientific Instruments*, 89(10):10K102, 2018.

[40] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proceedings of the 24rd International Conference on Very Large Data Bases*, pages 392–403, 1998.

[41] Taurus T. Dang et al. Distance-based k-nearest neighbors outlier detection method in large-scale traffic data. *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pages 507–510, 2015.