Plasma Physics and Controlled Fusion

PAPER • OPEN ACCESS

Neural network approximated Bayesian inference of edge electron density profiles at JET

To cite this article: A Pavone *et al* 2020 *Plasma Phys. Control. Fusion* **62** 045019

View the article online for updates and enhancements.

You may also like

- Fault identification using piezoelectric impedance measurement and model-based intelligent inference with pre-screening
Q Shuai, K Zhou, Shiyu Zhou et al.

- Editors' Choice—AutoEIS: Automated Bayesian Model Selection and Analysis for Electrochemical Impedance Spectroscopy
Runze Zhang, Robert Black, Debashish Sur et al.

- A Bayesian formulation for perturbed current tomography in tokamaks
N Bohlsen and M Hole

This content was downloaded from IP address 150.214.182.235 on 03/11/2023 at 10:59

OPEN ACCESS
IOP Publishing
Plasma Physics and Controlled Fusion
Plasma Phys. Control. Fusion **62** (2020) 045019 (13pp)
https://doi.org/10.1088/1361-6587/ab7732

Neural network approximated Bayesian inference of edge electron density profiles at JET

A Pavone[1], J Svensson[1], S Kwak[1], M Brix[2], R C Wolf[1] and JET Contributors[3]

[1] Max-Planck-Institut für Plasmaphysik, Teilinstitut Greifswald, D-17491 Greifswald, DE, Germany
[2] Culham Centre for Fusion Energy, Culham Science Centre, Abingdon OX14 3DB, United Kingdom

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

Received 30 August 2019, revised 6 February 2020
Accepted for publication 17 February 2020
Published 5 March 2020

Abstract

A neural network (NN) has been trained on the inference of the edge electron density profiles from measurements of the JET lithium beam emission spectroscopy (Li-BES) diagnostic. The novelty of the approach resides in the fact that the network has been trained to be a fast surrogate model of an existing Bayesian model of the diagnostic implemented within the Minerva framework. Previous work showed the very first application of this method to an x-ray imaging diagnostic at the W7-X experiment, and it was argued that the method was general enough that it may be applied to different physics systems. Here, we try to show that the claim made there is valid. What makes the approach general and versatile is the common definition of different models within the same framework. The network is tested on data measured during several different pulses and the predictions compared to the results obtained with the full model Bayesian inference. The NN analysis only requires tens of microseconds on a GPU compared to the tens of minutes long full inference. Finally, in relation to what was presented in the previous work, we demonstrate an improvement in the method of calculation of the network uncertainties, achieved by using a state-of-the-art deep learning technique based on a variational inference interpretation of the network training. The advantage of this calculation resides in the fact that it relies on fewer assumptions, and no extra computation time is required besides the conventional network evaluation time. This allows estimating the uncertainties also in real time applications.

Keywords: JET, neural network, Bayesian inference, real time, dropout, Lithium beam diagnostic, edge electron density

(Some figures may appear in colour only in the online journal)

1. Introduction

The application of neural networks (NN) to fusion experiments is not new, dating back to the mid-1990s with examples at the JET experiment of reconstruction of ion temperature profiles in real-time [1] and analysis of charge exchange spectra [2, 3]. They have been used for the inference of plasma parameters from diagnostic data as well as the prediction of disruptive events from different parameters and measured quantities [4]. More recently, they have also been used at the Wendelstein 7-X experiments for the task of reconstructing magnetic configuration properties from heat load patterns on the plasma-facing components [5, 6]; at JET for tomographic reconstruction [7]; they have been used also as surrogates for transport models as shown in [8–10]. Different machine learning algorithms as Gaussian processes

[3] See the author list of Joffrin *et al* (https://doi.org/10.1088/1741-4326/ab2276) for the JET contributors.

Original content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

 © 2020 Max-Planck-Institut für Plasmaphysik Printed in the UK

have been used in surrogate-based optimization strategy for the accelerated validation of plasma transport codes as in [11]. NN are very desirable tools especially for two reasons: they are able to identify patterns for those phenomena where a physics model describing the process is missing, e.g. plasma disruption, and they can process data at very fast time scales, e.g. in the order of tens of microseconds. The latter feature is particularly relevant today as fusion experiments produce more data than we can hope to exhaustively analyze with traditional tools.

Here we train a NN as a fast approximation, i.e. a *surrogate* model, of a Bayesian model of the JET lithium beam spectroscopy (Li-BES) diagnostic for the inference of the edge electron density profiles from experimental measurements. The principles behind the functioning of the diagnostic are given in [12], whereas details about the experimental configuration and measurements at JET can be found in [13–15]. Edge electron density profiles are useful quantities in controlling and understanding plasma phenomena as edge localized modes (ELMs), L-H transitions and turbulence transport. A model for the diagnostic, described in detail in [16], is implemented within the Minerva Bayesian modeling framework [17]. The framework provides a common way to define models and perform Bayesian inference when measurements are available. The models are strongly modular so that different modules, or *nodes* in the jargon, can be easily used to build similar models for different systems, e.g. diagnostics at different fusion machines, or test different assumptions. Currently, the framework is extensively used at the fusion experiment JET, where its application is discussed in [18] and an application to the equilibrium reconstruction using microwave diagnostics is described in [19], and W7-X, where it has been used to model a microwave radiometer calibration for the electron cyclotron emission diagnostic [20], for the inference of electron, ion, and impurity density profiles from an x-ray imaging diagnostic [21], and for the inference of ion temperature from measurements of a coeherent thomson scattering diagnostic [22].

In a previous work [23], it was shown that a NN can be trained as approximation of the Bayesian model of an x-ray imaging diagnostic at W7-X, and it was argued that the same method could be easily applied to a different system for which a Bayesian model implemented within Minerva was available. Extending such work, here we aim at validating this claim. Therefore we make use of the same method for training the network, i.e. we train the network on data generated exclusively with the Bayesian model sampling from its joint distribution, and we show that it can be successfully used to approximate the full model Bayesian inference of plasma parameters from data measured with a new physical system at a different fusion experiment, the edge electron density profiles from the Lithium beam emission spectroscopy diagnostic measurements at JET. In this way, we demonstrate that all that is required to obtain such network approximation is a Bayesian model. This is relevant because it shows that it is possible to replicate the method and achieve a fast reconstruction for any diagnostic modeled within the Minerva framework. Moreover, a major novel contribution is achieved by improving on the uncertainties calculation previously reported, which suffered from being slow and requiring limiting approximations. The calculation makes use of a novel state-of-the-art deep learning technique which can provide fast and at the same time accurate uncertainty estimates. This is of particular relevance if we think of using the network reconstructions in real time systems and control applications where we need to take decisions according to the network result and it is therefore crucial to know whether and to what extent the network output is accurate and can be trusted.

In section 2 we give an overview of the Lithium beam spectroscopy diagnostic to the extent that is relevant to this work, in section 3 we describe the Bayesian model of the diagnostic implemented within the Minerva framework, in section 4 we show how the network is trained making use of data generated with the Minerva Bayesian model in order to make predictions from experimental data, in section 5 we describe how the uncertainties of the network model can be calculated, and in section 6 we compare the network inference to the Bayesian inference carried out with the Minerva model on measurements collected at several JET pulses. We draw our conclusion in 7, where we also give an outlook on future developments.

2. The JET lithium beam spectroscopy diagnostic

The Li-BES system measures the spectral emission produced by the interaction of lithium atoms with the plasma species. The lithium atoms are injected with a beam vertically from the top of the machine, and as the beam penetrates the plasma it gradually gets excited and it is lost along the magnetic field lines when most of the atoms get ionized. A transmission grating spectrometer collects the radiation emitted along the penetration path, which is limited to the edge region of the plasma where it allows the reconstruction of the electron density. The spectrum is observed in a few nanometers wavelength range from 26 different spatial positions. A CCD camera is used to detect the photons with an integration time of typically 10 ms. A sketch of the system is shown in figure 1.

In order to understand the work presented here, details about the hardware are not as relevant as those about the model, which are given below. For the reader interested in knowing further details about the hardware set-up, detailed descriptions can be found in [13] and [14]. Here we will give an overview of the diagnostic principles and the model in order to provide the information required to understand the rest of the work. A full description of the Bayesian model and its usage to infer the electron density, including details about error treatment, modeling of the instrument function and calibration, is given in [16, 24].

The measured spectra contain different components: the Li I line radiation A, a bremsstrahlung dominated background

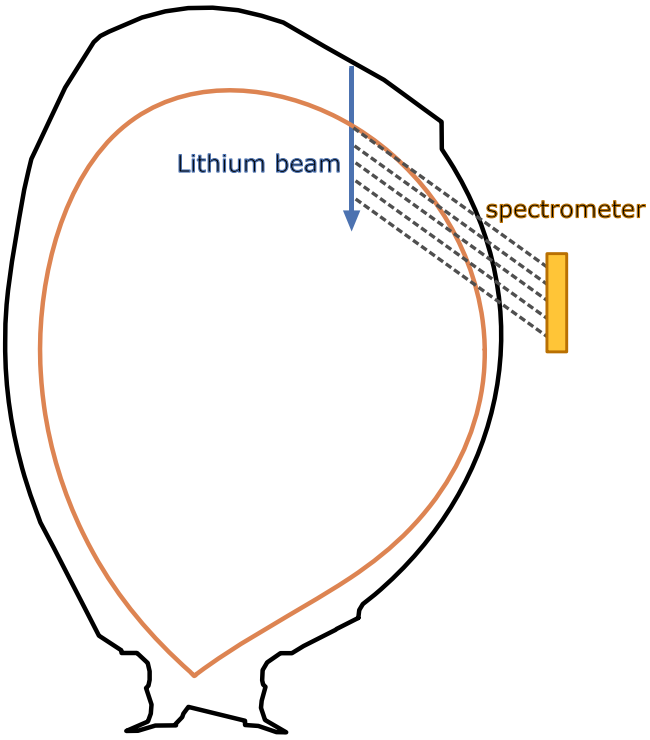

Figure 1. A schematic of the Li-BES system at JET. The lithium beam is injected vertically (blue arrow) and it penetrates the plasma volume, indicated by the orange ellipsoid, emitting light by interacting with the plasma species. The spatial positions of the measurements is indicated by the intersection of the lines of sight (dashed lines) with the lithium beam path. The light is collected by a spectrometer, in yellow in the figure.

B assumed to be constant in the wavelength range of interest, and an offset Z. The signal S can be found by taking into account an instrument function $C(\lambda)$ representing the shape of an infinitely narrow line on the detector and an interference filter function $F(\lambda)$ according to the following equation (the spectral width of the Li line is below the resolving capability of the instrument):

$$S(\lambda) = F(\lambda)[C(\lambda)A + B] + Z. \tag{2.1}$$

The quantity of interest for this study is the Li I line radiation A, which we will refer to as the measurement or observation of our system from now on. It is inferred from the measured signal S in a pre-processing stage, prior to any NN or Bayesian model evaluation, by first inferring the interference filter function F and the instrument function C from two independent and dedicated measurements without plasma, and then by inferring A, B and Z simultaneously from actual plasma experiments. We will not give further details here about how this is accomplished as it is not relevant for the rest of this work; the interested reader can find more information in [16].

The intensities of the Li I (2p-2s) line radiation come from the neutral lithium beam atoms injected into the vacuum vessel as they traverse and interact with the plasma. The atoms penetrating into the plasma undergo collisions with the plasma electrons, protons and other impurities and by mean of spontaneous emission processes they produce the line radiation that is collected by the diagnostic. The line radiation is emitted by the decay from the first excited state ($1s^2\ 2p^1$) to the ground state ($1s^2\ 2s^1$) of the beam atoms. The line intensity is then dependent on the population of the first excited stated. The change in the relative population of any excited state N_i as the beam atoms penetrate the plasma can be expressed in terms of the plasma electron density $n_e(z)$ and temperature $T_e(z)$ according to a multi-state collisional-radiative model firstly introduced in [25]:

$$\frac{dN_i(z)}{dz} = \frac{1}{v_{Li}}\sum_{j=1}^{M_{Li}}[n_e a_{ij}^e(T_e) + n_p a_{ij}^p(v_{Li}) + b_{ij}]N_j, \tag{2.2}$$

where z represents a coordinate along the penetration length of the beam. The coefficients a_{ij}^e and a_{ij}^p with $(i \neq j)$ and $a > 0$ are net population rate coefficients accounting for the contribution of plasma electrons and ions in populating the ith state from the jth state; whereas $a_{ii} < 0$ denotes a net de-population rate coefficient of the ith state accounting for excitation, de-excitation and ionization processes. The coefficients b_{ij} represent instead spontaneous emission rate coefficients or Einstein coefficients. v_{Li} is the lithium beam velocity corresponding to ≈50 keV beam energy, n_p is the density of plasma protons, and M_{Li} is the number of considered states of the neutral lithium atoms, which is 9 in this case. The dependency of the plasma profiles n_e, T_e and N_j on the z coordinate has been omitted for brevity. In order to be able to solve equation (2.2), an initial condition needs to be defined. It can be chosen to be:

$$N_i(z = 0) = \delta_{1i} \tag{2.3}$$

corresponding to the assumption that all lithium beam atoms are neutral in the ground state ($i = 1$) at $z = 0$, the position where they enter the vacuum vessel. In other words we assume $N_1(z = 0) = 1$. The population of the first excited state ($i = 2$) of the lithium atoms N_2 can then be calculated. This quantity is proportional to the observed lithium intensities $A(z)$ found from the signal measured with the CCD camera along the observation length. We therefore introduce a calibration factor α to express this relationship:

$$A(z) = \alpha N_2(z). \tag{2.4}$$

The factor is not known and it has to be inferred from the data. For the interested reader, a complete derivation and an explicit expression of α in terms of the CCD output counts can be found in [16].

Figure 2 shows an example calculation carried out with the forward model implementing the physics described so far. Given the plasma profiles in the two plots on the top, the relative population of the first excited state of the lithium atoms and then the Li I line intensity can be calculated. The plot on the bottom left representing the line intensity in arbitrary units also shows a 10% relative Gaussian noise added to the calculation (the scattered circles) in order to simulate the noise present in the measurements. As the beam

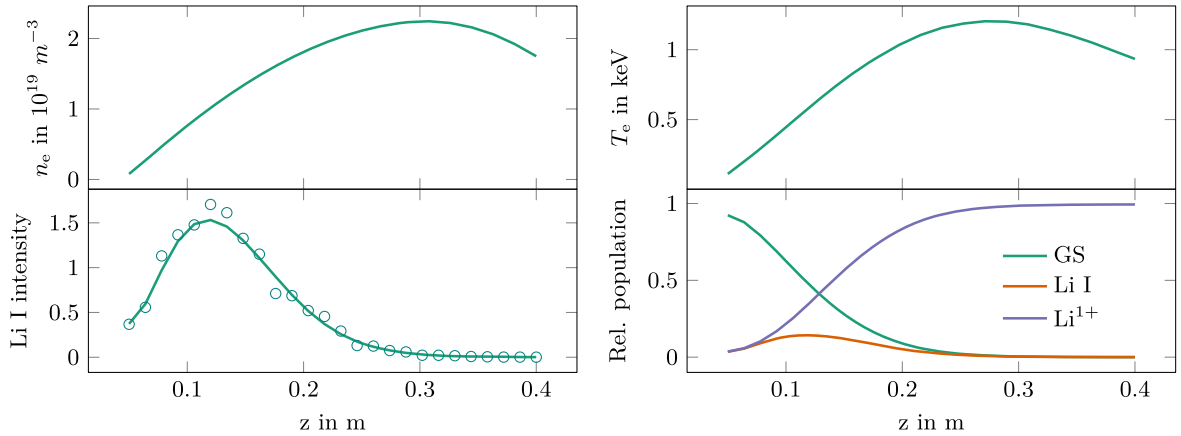

Figure 2. An example case of the Li-BES forward model calculation. In clockwise direction, from the top left plot the following quantities are shown: an electron density profile, an electron temperature profile, the Li I line intensity predicted with the forward model (solid line) together with the addition of 10% relative Gaussian noise (scattered dots), and the relative population of the ground state (GS), the first excited state Li I and the first ionized state Li^{1+} of the beam atoms. All quantities are expressed as function of the penetration distance inside the plasma, with $z = 0$ corresponding to the position where the beam enters the vacuum vessel.

atoms penetrate into the plasma they also get ionized and when this happens they follow the magnetic field lines as charged particles and do not contribute any longer to the collected emission. The ionized atom population is shown in violet in the plot on the bottom right.

The measured intensity can be used to infer the electron density profile at different edge locations along the penetration length, provided the electron temperature profile information. The latter is usually delivered at JET by the high resolution Thomson scattering diagnostic (HRTS) [26].

3. The Bayesian minerva model

The multi-state model described in section 2 is implemented within the Minerva Bayesian modeling framework. The Minerva modeling framework [17] is a framework that allows modeling complex systems and carrying out Bayesian inference with them. Models are expressed in a modular way, where the modules are called *nodes*. These modules can be easily switched and replaced so that different models can be easily built, and different assumptions can be easily tested. Nodes can represent physics quantities with associated probability distributions over the values they can assume, or they can represent deterministic calculations consumed by other nodes in the model. The models are used as forward models to predict observations from given free parameters. It makes use of graphical models [27] to represent models and the probabilistic relations between quantities in the model. An example of a Minerva graph for the lithium beam system is shown in figure 3, and it is described later in the section. Once a model has been defined within the framework, Bayesian inference can be performed with it. Thanks to the fact that model definition and Bayesian inference constitute two different and independent stages, such that the implementation details of one are abstracted away from the other, the framework offers a solution for performing scientific inference in complex systems which is general, and not strictly related to a single nuclear fusion experiment or even nuclear fusion research. As a Bayesian framework, it employs Bayesian probability theory to handle the uncertainties attributed to any modeled quantity. In Bayesian probability a prior distribution $p(T)$ is assigned to the model free parameters T and a likelihood function $p(D|T)$ is assigned to the model observations D. As measurements are available, they can be used to update the prior knowledge on the free parameters through Bayes formula:

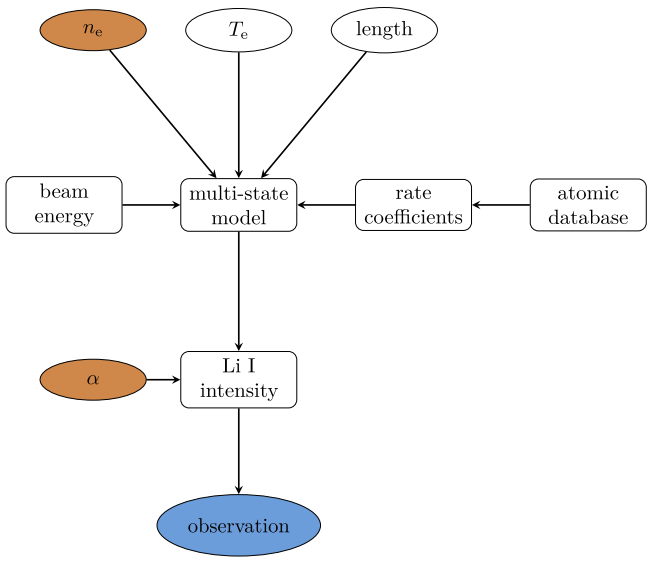

Figure 3. A simplified sketch of the Li-BES Minerva model graph. Colored nodes are probabilistic nodes, where *orange* denotes the free parameters and *blue* denotes the observed quantities. White nodes represents deterministic calculation nodes or other input parameters required by the model. The arrows represent direct or indirect dependencies in the probabilistic relations between the quantities in the probabilistic nodes.

$$p(T|D) = \frac{p(D|T)p(T)}{p(D)}. \tag{3.1}$$

The quantity $p(T|D)$ is called the posterior distribution and it reflects the new state of knowledge on the parameters T as the

observations D are taken into account. The numerator of the equation is also known as the joint distribution $p(D, T)$ of the observations and parameters. The denominator $p(D)$ is a normalization factor and it is referred to as the evidence.

3.1. Model parameters

The model free parameters are the electron density profile n_e and the absolute calibration factor α. The prior distribution for the n_e profile is modeled through a zero mean Gaussian process [28]. A Gaussian process is a stochastic process whose realizations are functions. In Bayesian inference they are used for models where the free parameters are functions, in this case 1D electron density profiles, and the observations are the values they assume in a number of domain locations. A realization of a random function drawn from the process is given by the values it assumes in a number of positions in its domain and its probability distribution is chosen to be Gaussian. Its covariance is known as *covariance function*. One common choice for it is the squared exponential, which regulates the smoothness of the function by modeling the correlation between points in the domain. For the density profiles it can be written as:

$$K(z_1, z_2) = \sigma_f^2 \exp\left(-\frac{(z_1 - z_2)^2}{2\sigma_x^2}\right) + \delta_{ij}\sigma_y^2, \tag{3.2}$$

where z_1 and z_2 are two positions along the z axis and the different σ parameters regulate the smoothness of the profile. σ_f regulates the overall variance of the profile and σ_x regulates the length scale of the changes in the profile. Small values mean that the profile can change quickly along z, whereas large values mean that it will change slowly. σ_y is used to allow for small amount of noise expected in the profile. A uniform distribution is used for the calibration factor α. The model observations are the Li I line intensities. The likelihood function is chosen to be a normal distribution centered on the forward model prediction.

3.2. Model graph

A sketch of the Minerva model graph for the Li-BES system is shown in figure 3. In the sketch, the nodes representing the free parameters n_e and α are in orange, and the node representing the observations is depicted in blue. The white nodes represent computation nodes, as the 'multi-state model' node which is used to calculate the predicted Li I line intensity, represented in the 'Li I intensity' node, or other quantities required by the model, as the energy of the lithium beam, represented by the 'beam energy' node, and the observation length, defined as the length along the beam path where the emission is observed, represented by the 'length' node. The observation length is a quantity that is known given the experimental setup and it can be different for different experiments. We make use of 20 and 26 equally spaced positions along the observation length for the profile and the observations locations, respectively. The calibration coefficient α is applied to the predicted Li I line intensities as an overall multiplicative factor. In the graph, we have also shown the dependency of the multi-state model from the rate coefficients that are taken from the Atomic Data and Analysis Structure (ADAS) database [29], a database containing data useful for modeling the radiating properties of ions and atoms in plasmas. The arrows represent direct or indirect dependencies in the probabilistic relations between the quantities in the probabilistic nodes, and should not be understood as a computational flow. All free parameters node reach, indirectly, the observation node and are not connected to each other. This expresses the fact that the joint distribution of the graph $p(D, T)$ can be factorized in terms of a conditional distribution of the observations conditioned on the free parameters $p(D|n_e, \alpha)$ and the product of two independent prior distributions over the electron density $p(n_e)$ and the calibration factor $p(\alpha)$:

$$p(D, T) = p(D|n_e, \alpha)p(n_e)p(\alpha). \tag{3.3}$$

4. NN training

Given the Bayesian model described in the previous section, we aim now at training a NN in such a way that it constitutes an approximation of the Bayesian inference that can be carried out with the full Minerva model. In order to achieve this, we use the Minerva model to generate the training data. In this section, we outline the procedure to the extent it concerns the specific case of the lithium beam system under investigation. For the interested reader, further conceptual and theoretical insights about how this method can provide a sound approximation are given in [23].

4.1. Generation of the training data

The electron temperature profile T_e and the observation length l are parameters that are known at inference time, when we perform inference with the Minerva model and the network: the former is provided by an independent measurement of the Thomson scattering diagnostic, the latter comes from the experimental setup. Both quantities constitute part of the network input, together with the measured lithium line intensities, and therefore need be generated with the Minerva model for training the network. As we aim at training the NN on the problem of inferring electron density profiles from measured Li I line intensities, the training input data are the Li I line intensities, the T_e profiles, and the length, while the training output data are the n_e profiles and the absolute calibration coefficient α. We generate the training data by sampling from the joint distribution of the model. This means that, as a first step, we draw a sample of n_e, T_e, l and α from the corresponding prior distributions and, given these values, we compute the predicted Li I line intensities and draw a sample from the likelihood function. We iterate over this process a number of times equal to the number of samples in the training set. As we need to generate data also for T_e and the observation length, we assign probability distributions also to them, so that the joint distribution of the model can be

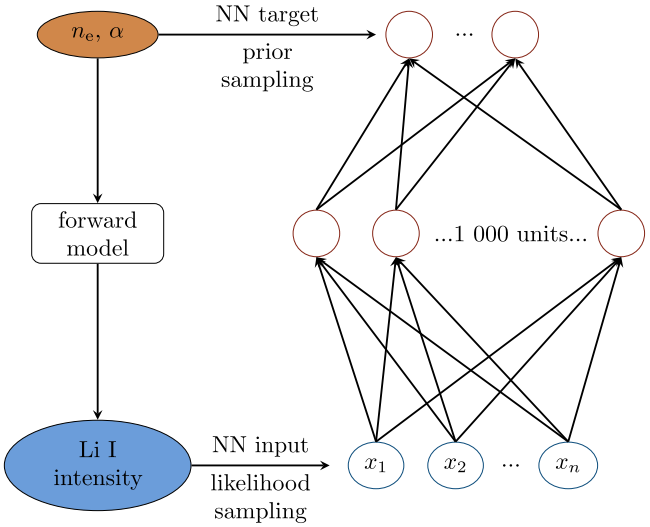

Figure 4. A sketch to illustrate the sampling procedure for the training set creation. A sketch of the Li-BES Minerva model and the neural network, having one hidden layer with 1000 units, is shown on the left and on the right, respectively. At training time, the NN takes as input the Li I line intensities generated with the Minerva model and sampled from the likelihood function together with the sampled T_e and observation length l. The sampled n_e and α used to generate the intensities are the target data of the network. The blue nodes of the neural network denote the input intensities and the two red nodes at the top denote the output points of the electron density profile.

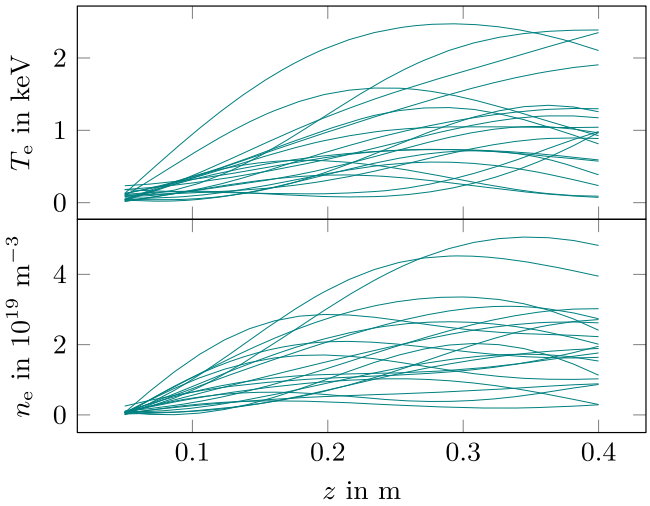

Figure 5. Samples from the Gaussian process priors for the T_e and n_e profiles, top and bottom figures, respectively. The x-axis position at 0.0 corresponds to the location where the beam atoms enter the plasma, which is at the edge of the machine. The low value constraint at such position is also visible in the shape of the sampled profiles.

written as:

$$p(D, T) = p(D, n_e, T_e, l, \alpha) \tag{4.1}$$

and

$$p(D, T) = p(D|n_e, T_e, l, \alpha)p(n_e)p(T_e)p(l)p(\alpha). \tag{4.2}$$

A sketch of the procedure is shown in figure 4.

We used for the n_e and T_e profiles a zero mean GP prior defined on a x domain of 20 linearly spaced positions between $x_0 = 0.0$ and $x_1 = 20.0$ with covariance function as in equation (3.2). The parameters of the GP are set to: $\sigma_x = 10.0$, $\sigma_y = 0.002 \times 10^{19}\,\mathrm{m}^{-3}$, $\sigma_f = 2.0 \times 10^{19}\ \mathrm{m}^{-3}$ for n_e, and $\sigma_x = 10.0$, $\sigma_y = 0.002\,\mathrm{keV}$, $\sigma_f = 1.0\,\mathrm{keV}$ for T_e. The profiles are constrained to be non-negative by rejecting the samples having negative values as they are drawn from the GP prior distributions until a positive valued sample is drawn and kept. Moreover, we constrain the profiles to assume low value at $x = 0$ corresponding to the position $z = 0$, the edge location where the beam enter the plasma. The constraint is implemented as a *virtual observation*, i.e. by implementing an observed node in the Minerva graph as a normal distribution with standard deviation 100 eV around a value of 100 eV for the T_e profile, and standard deviation $0.1 \times 10^{19}\,\mathrm{m}^{-3}$ around a value of $0.01 \times 10^{19}\,\mathrm{m}^{-3}$ for the n_e profiles. In this way, when the profiles are sampled, they are constrained by this virtual observation as if it was a real measurement, although no measurement of such kind actually occurred. Further details about how a virtual observation constraint is implemented are provided extensively in [23] and will not be treated further here, as they are not relevant to the understanding of the work that follows.

Samples from the n_e and T_e prior distributions are shown in figure 5. The distance from the location $z = 0$ at which the beam atoms enter the plasma is on the x axis. It is worth noticing that the profiles are not monotonic. For the calibration factor α we use a uniform distribution between 1.0 and 20.0. The choice for this prior was motivated by the information available from previous analysis, which showed values typically falling in this range. For the parameter l we use a uniform distribution between 0.2 and 0.4 cm. Finally, the conditional distribution of the simulated Li I intensity $P(D|T)$ is a normal distribution centered on the model prediction and with standard deviation equals to 10% relative error. In this way, we inject noise in the training input data, as we expect to have noise at evaluation time, when the input are the experimental measurements. Our training data set is made of 100 000 samples.

4.2. Network model

The NN architecture used for this problem is a multilayer perceptron (MLP) with one hidden layer with 1000 units. The activation function used in the hidden units is the so called scaled exponential linear function (SELU) [30] and the loss function used is the mean squared error:

$$L(\mathbf{w}) = \frac{1}{N}\sum_i (\mathbf{y}_i(\mathbf{w}) - \mathbf{t}_i)^2, \tag{4.3}$$

where N is the number of training samples, $\mathbf{w}$ is the vector of adaptable network weights, and $\mathbf{y}_i$ and $\mathbf{t}_i$ are the ith multi-dimensional output and target vector, respectively. The network was trained using the Adam optimizer with parameters: learning rate $= 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, see [31] for a description of the algorithm and parameters. The training data were divided in batches of 100 samples and the network weight training was terminated once 5000 passes

through the training set were reached (also called epochs). One single starting position was used for the initialization of the weights. The number of 1000 hidden units in one hidden layer was chosen by validating the network performance on a set of test data made of 1000 samples drawn from the joint distribution of the Bayesian model. The network was implemented within the TensorFlow framework [32]. The NN has been trained with dropout [33, 34]. Dropout is a technique originally introduced to prevent overfitting. Although this can be, by itself, a good reason to make use of it, there is at least another reason. Dropout training can also be used to estimate uncertainties in the network prediction; when used in this way it is referred to as *Monte Carlo (MC) dropout* [35]. In the next section we will give an overview of the theoretical framework that allows to interpret dropout training as a Bayesian inference technique. We will only touch the salient points of the derivation which are necessary to understand the current work, but for the reader interested in a deeper understanding of the theory behind it, details can be found in [35].

Before proceeding, we would like to summarize the relationship between the two key elements of this work:

- the *Minerva Bayesian model* is defined at the first step, and it is used to both carry out the full Bayesian inference of the electron density profiles from the measured experimental data, *and* to generate the training data for the NN from its joint distribution.
- the NN is first trained on data generated exclusively with the Minerva Bayesian model, afterwards it is applied to infer electron density profiles from the measured experimental data.

In this way, the full Bayesian inference and the NN inference are both based on the same Bayesian model, with the distinction that the latter approximates the former. The two inference methods will be compared in section 6.

5. NN uncertainties

Delivering uncertainties in the NN calculation is necessary in order to asses whether, and how far, the network prediction can be trusted. This is important when the network output is wanted for further calculations, and especially when a decision has to be taken according to its output, as in the case of real time control systems, e.g. feedback systems. Therefore, it is also important that the uncertainties can be calculated in a time scale comparable to the network processing speed itself. Here we give an overview of the theoretically sound and practically desirable method presented in [35].

5.1. Bayesian NNs

NN uncertainties can be calculated in a Bayesian framework known as Bayesian NNs [36]. In this context, the network training is seen as an inference problem, where the free parameters are the network weights $\mathbf{w}$ and the training target data are the observations $\mathbf{Y}$. It follows that we can write Bayes formula for the posterior of the network weights:

$$p(\mathbf{w}|\mathbf{Y}, \mathbf{X}) = \frac{p(\mathbf{Y}|\mathbf{w}, \mathbf{X})p(\mathbf{w})}{p(\mathbf{Y}|\mathbf{X})}, \tag{5.1}$$

where $\mathbf{X}$ denotes the training input data. As we have now a distribution over the network weights, we will also have a distribution over the network's predictions $\mathbf{y}^*$ for a new input vector $\mathbf{x}^*$, given by:

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y}^*|\mathbf{w}, \mathbf{x}^*)p(\mathbf{w}|\mathbf{Y}, \mathbf{X})\mathrm{d}\mathbf{w}. \tag{5.2}$$

This distribution is the one we are interested in and which prescribes the uncertainties in the network prediction.

5.2. Variational inference

For any interesting NN model, the posterior $p(\mathbf{w}|\mathbf{Y}, \mathbf{X})$ cannot be treated analytically because of the large number of weights and complex network function. We therefore make use of *variational inference* (VI) [37] in order to approximate it. In VI we choose an approximating variational distribution $q_\theta(\mathbf{w})$ parametrised by θ, which is easy to evaluate, in order to approximate the original posterior distribution. This is achieved by minimizing the Kullback–Leibler (KL) divergence with respect to θ, which can be thought as a measure of similarity between two distributions:

$$\mathrm{KL}(q_\theta(\mathbf{w})\,||\,p(\mathbf{w}|\mathbf{Y}, \mathbf{X})) = \int q_\theta(\mathbf{w})\log\frac{q_\theta(\mathbf{w})}{p(\mathbf{w}|\mathbf{Y}, \mathbf{X})}\mathrm{d}\mathbf{w}.$$

It can be shown that minimizing the KL divergence is equivalent to maximizing the so called *evidence lower bound* (ELBO) with respect to θ:

$$L_{\mathrm{VI}}(\theta) = \int_{\mathbf{w}} q_\theta(\mathbf{w})\log p(\mathbf{Y}|\mathbf{X}, \mathbf{w})d\mathbf{w} - \mathrm{KL}(q_\theta(\mathbf{w})\,||\,p(\mathbf{w})),$$

where, noticeably, the KL divergence term now is between the approximating distribution $q_\theta(\mathbf{w})$ and the prior distribution $p(\mathbf{w})$, fact that explains the name of the expression. At this point we make use of the results derived in [35], where it is shown that the conventional dropout training of a NN is equivalent to the maximization of the ELBO function.

5.3. Dropout

When a network is trained with conventional dropout, at each iteration of the training, as a new training batch sample is provided to the network, some of its units are dropped. This makes the trained network more flexible, intuitively because the units need to learn to be useful also when some of the others are missing. To be more rigorous, dropout prevents overfitting by preventing co-adaption of the units. At evaluation time all units are retained, but their output is scaled down by the probability of dropping them, since now there is a larger number of units in the network.

5.4. MC dropout

In the conventional dropout picture of training, the stochastic process is applied in the unit (or feature) space. We can switch view and see the stochastic process as applied in the

weight space, since as units are dropped, also the corresponding weights that connect them are dropped. Under this view, it is finally possible to merge the dropout training with the VI approximation of the true weight posterior. This happens by re-parametrising the weights $\mathbf{w}$ in terms of a function g such that:

$$\mathbf{w} = g(\theta, \epsilon) = \{\mathrm{diag}(\epsilon)\mathbf{M}, \mathbf{b}\}, \tag{5.3}$$

where $\theta = \mathbf{M}$, $\mathbf{b}$ and $\epsilon \sim \mathrm{Bernoulli}(p_\mathrm{b})$ where p_b is the probability of dropping the units. ϵ is then a vector of zeros and ones, $\mathbf{M}$ is a Q by D deterministic matrix of connecting weights where Q is input vector size and D output vector size, $\mathbf{b}$ is the bias vector of dimension Q, and $\mathrm{diag}(\epsilon)$ is a diagonal matrix of same size as $\mathbf{M}$ having as diagonal elements the elements of the vector ϵ. The product $\mathrm{diag}(\epsilon)\mathbf{M}$ represents a matrix multiplication whose results end up 'selecting' what connecting weight is active at a given dropout step. At this point, after some more manipulations, we can rewrite the integral in the ELBO expression as an integral over $p_\mathrm{b}(\epsilon)$ and the derivatives required for the optimization as derivatives with respect to θ. In [35] it is then shown that, optimizing a NN dropout loss function is equivalent to optimizing the function $L_\mathrm{VI}(\theta)$. In conclusion, this means that, by using a well established method for training the network, we can at the same time approximate the posterior distribution of the corresponding Bayesian network via variational inference with Bernoulli approximating variational distribution.

The only difference with the standard dropout training is that at evaluation time, instead of retaining all units, we keep dropping them as several forward passes of the network are done, so to obtain a distribution of network predictions rather than a single best estimate. This corresponds to estimating the ELBO integral with a Monte Carlo integration. The major advantage of this approach is that it scales well with large networks: forward passes of the network are typically very fast and can also be run in parallel. Therefore, calculating uncertainties in this way does not require substantial extra computation time.

We used dropout probability $p_\mathrm{b} = 0.5$ for all units in the hidden layer, and $p_\mathrm{b} = 0.0$ for the input units, i.e. all input units were retained.

We have described how variational inference and dropout can be combined in a unified view of the network training, leading to a Bayesian NN interpretation. One must be aware, though, of some caveats that have been acknowledged regarding the theoretical framework supporting this technique: see for example [38], where it is claimed that in the case of simple linear networks, this method approximates the *risk* of a process rather than the *uncertainty* of the model because the variance found in this way do not vanish at the limit of very large amount of training data; see also [39], where it is shown that the variational inference framework described in [35], specifically with regards to the choice of some approximating distributions, can lead to undefined objective function of the network, and they propose an alternative to such objective; in general, some difficulties have been recognized in the application of standard variational inference approach, as indicated in [40], where pitfalls are found in the usage of the KL divergence, and a different distance is proposed.

In the next section we will show results obtained with MC dropout estimation of the uncertainties, as we tested the network on experimental data collected at the JET tokamak.

6. Results

We evaluated the NN on data collected at several JET pulses. In order to assess the quality of the network reconstruction we can compare the reconstructed electron density profiles to those inferred with the full Bayesian model. Also, we can use the reconstructed n_e profiles as input to the forward model and simulate Li I line intensities to compare with the measured ones. This is indeed a better way to assess the quality of the network reconstruction as we can see how well the NN prediction fits the data. In the same way, the full Bayesian inference reconstruction can be compared against the measurements and the quality of the fit compared to that obtained with the NN reconstruction. We want to point out that this kind of comparison is possible because we have a model for the measurement processes, and it is the same one used for generating the network training data and the full Bayesian inference. As we previously mentioned, we are comparing two inversion methods applied to the same Bayesian model: the network inversion being a fast approximation of the full Bayesian inference.

6.1. Uncertainties

One illustrative example of such comparison is shown in figures 6 and 7 for data collected at the JET pulse 89312 at time 48.295 s, just before NBI heating started, so the plasma was in L-mode and the line integrated density was $\approx 5 \times 10^{19}\,\mathrm{m}^{-2}$. In figure 6, the NN reconstructed density profiles are compared to those inferred with the full Bayesian inference (Minerva). In figure 7, the Li I line intensities generated with the Minerva and NN reconstructed profiles are compared to the measured ones. The multiple samples represent the uncertainties. In the NN case, these are 100 samples obtained with MC dropout; in the Minerva case, these are 100 samples drawn from the full model posterior distribution which has been explored with a Markov Chain Monte Carlo sampler. From figure 6 it is evident that the uncertainties of the density profiles inferred with the network and with Minerva can be quite different. This should not surprise. It is important to realize that the uncertainties stemming from the two methods arise from two different models, the network and Minerva model, and the corresponding Bayesian inference problems. In both cases, the uncertainties are calculated in a Bayesian framework, but the models and quantities that contribute to the uncertainties in the reconstructed profiles are different, as the inference task to be solved is different. This is made evident by looking at Bayes formula and the mathematical expression of the uncertainties for the two models. In the network case, the distribution of the predicted profiles is obtained by

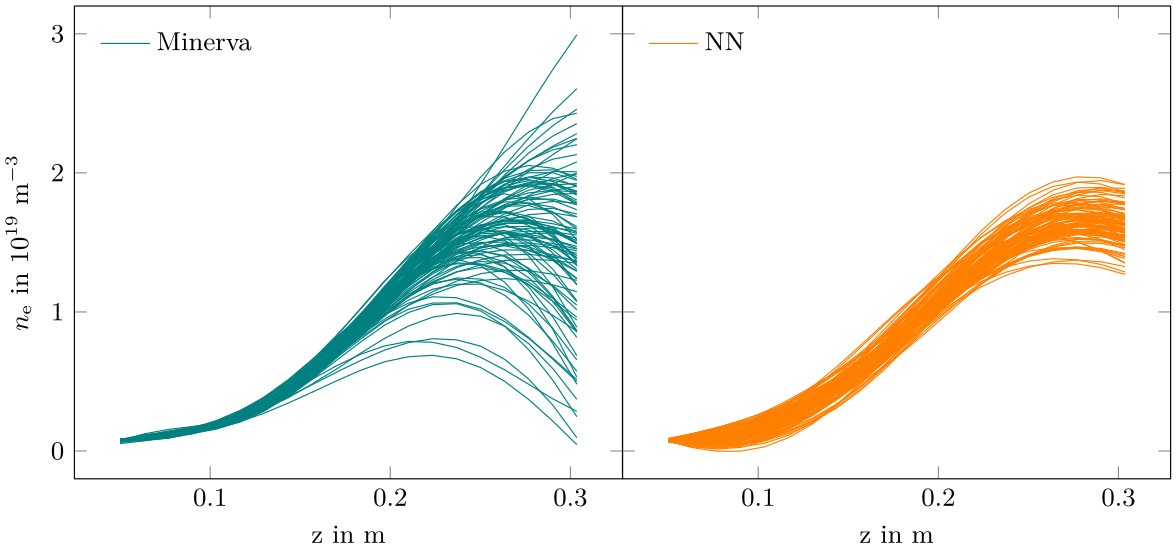

Figure 6. A comparison between the n_e profiles predicted with the NN and the full model Bayesian inference (Minerva). The samples represent the uncertainties from the MC dropout in the NN case, and the posterior distribution in the Minerva case. The data are taken from the JET pulse number 89312 at time 48.295 s.

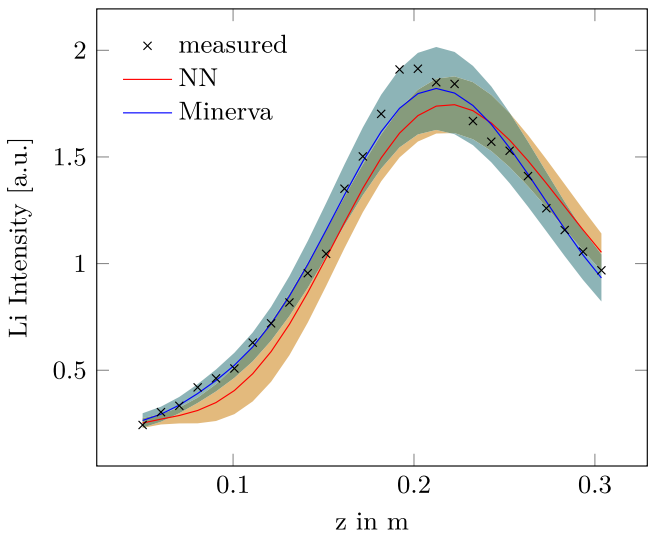

Figure 7. The Li I line intensities predicted with the NN and Minerva n_e profiles are compared to the measurements. The shadowed areas represent the uncertainties. The data are taken from the JET pulse number 89312 at time 48.295 s.

marginalization over the network weights $\mathbf{w}$ when a new input vector $\mathbf{x}^*$ is provided:

$$p(n_e|\mathbf{x}^*, \mathbf{Y}) = \int p(n_e|\mathbf{w}, \mathbf{x}^*)p(\mathbf{w}|\mathbf{Y})\mathrm{d}\mathbf{w} \tag{6.1}$$

which is the same expression of equation (5.2), in which we have omitted the dependence on the input variable $\mathbf{X}$ and substituted $\mathbf{y}^* = n_e$. When the network is evaluated on the measured line intensities, the input vector $\mathbf{x}^*$ is constituted of the electron temperature profile independently measured by a Thomson scattering diagnostic, the observation length used at that experiment, and the measured line intensities. The posterior of the network weights, instead, is given by $p(\mathbf{w}|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{w})p(\mathbf{w})$ and it is found with variational inference with dropout training as described in section 5. We do not expect dropout training to reconstruct the posterior distribution of the Minerva model $p(n_e|D)$, but to approximate the true posterior distribution of the network weights $p(\mathbf{w}|\mathbf{Y})$; then, the spread of this posterior gives rise to a spread in the predicted profiles according to equation (6.1). Whereas, in the Minerva case the distribution of the inferred profiles is given by the posterior:

$$p(n_e|D) \propto p(D|n_e)p(n_e), \tag{6.2}$$

where D represents the measured Li I line intensities. The spread of the posterior, therefore, is influenced by the model uncertainties in predicting the measured Lithium 1 line intensity $p(D|n_e)$ (e.g. measurement errors) and the prior $p(n_e)$.

To highlight the difference between the two models, it is useful to notice what is the role of the different quantities in each of them: in the Minerva model, the free parameters are the electron density profiles, and the observations are the lithium line intensities. The inference task is then to find the electron density profiles which allow to predict the measured Li line intensities, given the measurements, the physics model, and the prior. These are the boundaries of the inference problem. The final posterior distribution expresses the uncertainties in the inference of the density profiles given the model and these boundary conditions. The uncertainties that arise in this case are related to the model uncertainties in the prediction of the Li intensities—typically estimated from the measurement errors, the sensitivity of the model to different values of the electron density, and the beam attenuation. For example, because the beam is attenuated as it penetrates the plasma and gets ionized, the model is less sensitive to changes in the electron densities in the locations closer to the core of the machine, and the uncertainties are therefore larger. Quantitative details about the estimation of the error from the measurements, and quantitative considerations on the beam

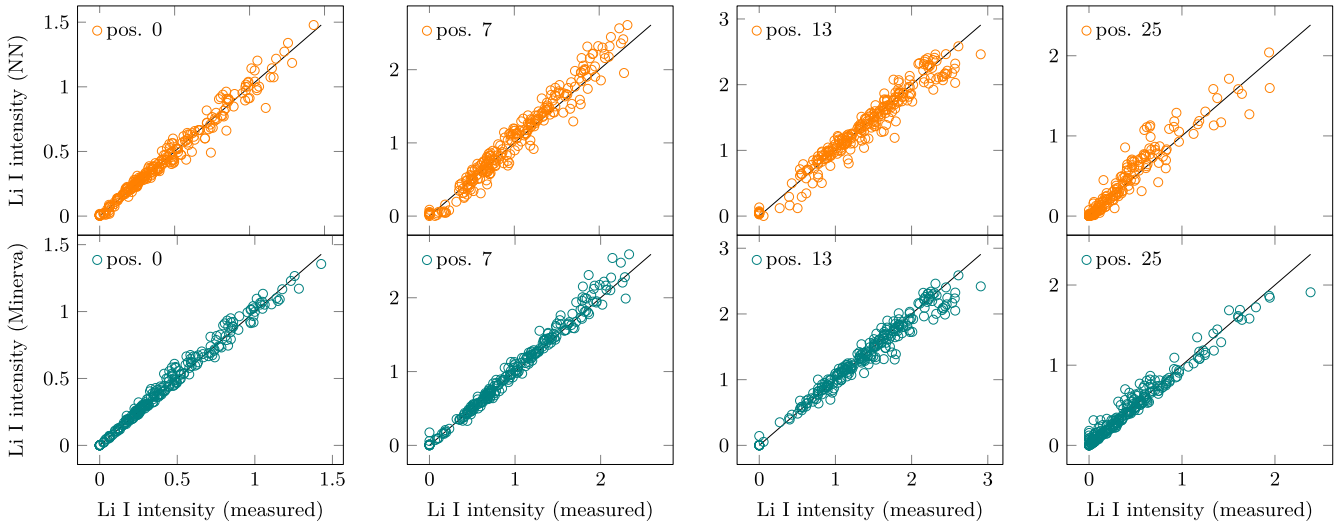

Figure 8. The Li I line intensity predicted with electron density profiles found by the network (top row) and the full model Bayesian inference (Minerva, bottom row), on the y-axis, are compared to the measurements, on the x-axis. Each column shows the comparison for a spatial position along the intensity profile. The real space coordinates of the positions can vary through the experiments, so here they are labeled by an index starting from the outermost position at index 0 to the innermost position at index 25. The solid line shows the $y = x$ line. More than 200 hundred measured data points collected across 65 pulses were used.

attenuation and sensitivity on the model are reported in previous works [16], and are not discussed here as they fall beyond the scope of this work. In the network model, the free parameters are the network weights, which lack any physics interpretation, and the observations are the set of target data in the training set, i.e. the sampled electron density profiles. The uncertainties of the network posterior depend on a combination of network structure, weight prior and approximating distribution, as it is indicated and further discussed in [35]. At training time, the network model inference task is to find the weights which allow to reconstruct the electron density profiles from the Li I line intensities, given a specific choice of network structure, weight prior, approximating distribution and training set, whose statistical properties are inherited from the Minerva model by sampling from its joint distribution. These are the boundary conditions of the inference problem for the network. The predictive distribution of equation (6.1), then, expresses the uncertainties of the model in making a prediction within these boundaries.

6.2. Li I line intensity reconstruction

The performance of the two methods is compared more extensively in figure 8, where the Li I line intensities predicted with electron density profiles found by the network (top row) and the full model Bayesian inference (Minerva, bottom row) are compared to the measurements in a scatter plot. The profiles used are the average of the MC dropout samples in the network case and the posterior distribution samples in the Minerva case. The solid line shows the $y = x$ line, where all points would lie if we had a perfect fit to the measurements. Each plot in a column shows a different spatial position along the intensity profile; since the corresponding real space coordinates may vary throughout the experiments, the positions are labeled according to an index ranging from 0 for the outermost location to 25 for the innermost one. More than 200 hundred measured data points collected across 65 pulses were considered in the analysis (see appendix for a list of the pulses). The pulses were arbitrarily chosen, without selecting for a specific set of features or plasma configurations. The pulses featured a broad range of parameters, including both L- and H-mode scenarios, low and high power and gas levels. Across all pulses, the NBI power ranged from ≈3.0 to ≈28 MW, the vacuum toroidal magnetic field from 1.6 to 3.3 T, the total ICRH power from ≈2.0 to ≈6.0 MW, the plasma current from ≈1.1 to ≈3.5 MA, and the line integrated density from $\approx 8.0 \times 10^{19}\,\mathrm{m}^{-2}$ to $\approx 2.6 \times 10^{20}\,\mathrm{m}^{-2}$. The agreement to the measurements is, in general, satisfactory for both methods. Although the network consists of a quick, approximated inversion of the full Bayesian inference, its reconstructions appear to be good enough to closely predict the data in most cases.

This is confirmed by figure 9, where we compare the mean relative error between the observations calculated with each of the two method inverted profiles and the measurements, for each position along the profile intensities:

$$E_{\mathrm{mre}} = \frac{1}{N}\sum_i \left| \frac{q_{1i} - q_{2i}}{q_{2i}} \right|, \tag{6.3}$$

where q_{1i} is the line intensity predicted by one of the methods, q_{2i} are the measured line intensities and N is the number of data points. The figure shows that the error for the network is consistently larger at every location, and it follows a trend similar to the full Bayesian inference case (Minerva). At most positions the error is below 20%, a reasonably good value, suggesting that the network inversion can provide a reliable approximated analysis.

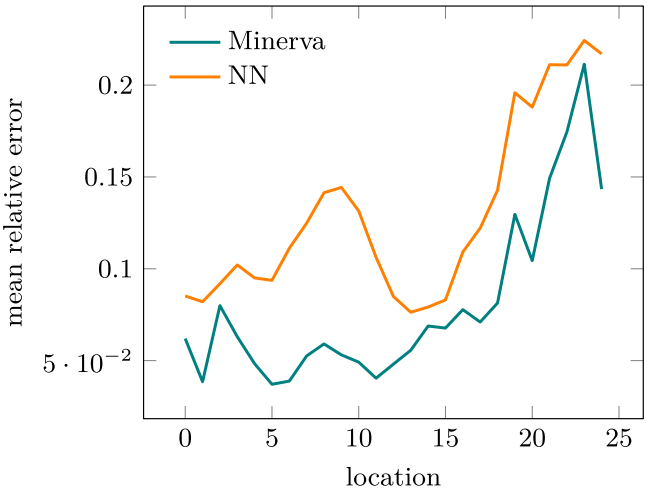

Figure 9. The mean relative error between measured Li I intensities and the intensities simulated with n_e profiles reconstructed by the network (NN) and the full Bayesian inference (Minerva) is shown at each position along the intensity profile. The calculation has been carried out for more than 200 measurements collected across 65 JET pulses.

6.3. Electron density profile inference

Finally, the electron density profiles inferred with Minerva can be compared to those found with the network as shown in figure 10. Each plot shows the n_e values at four different locations along the profile, indexed with an integer number from 0 to 19. The values are the average of the samples drawn from the posterior distribution inferred with Minerva (x-axis) and the samples found with the MC dropout network. The agreement is, in general, quite satisfactory. Indeed, an analysis of the mean relative error as defined in equation (6.3), with q_1 denoting the NN reconstructed profiles and q_2 the Minerva reconstructed profiles, shows that it is $<15\%$ at any spatial location. This can be seen in figure 11.

7. Conclusions

Extending from previous work [23], we have trained a NN as a fast, approximated Bayesian inference model for the inference of edge electron density profiles from measurements at the JET tokamak. Exploiting the NN well-known data processing speed, we can reduce the time required for the analysis from tens of minutes to tens of microseconds on a GPU, providing an approximated reconstruction. We have shown here, as it was suggested in [23], that all that is necessary in order to realize this kind of fast network approximation is the definition of a Bayesian model within the Minerva framework, since the network is trained exclusively on data generated with the model by sampling from its joint distribution. This is of particular interest because it opens the possibility to fully automate the process in order to be able to have a fast network approximation for any Bayesian model of any other diagnostic implemented within the framework.

Uncertainties can also be calculated for the network inversion. We made use of a state-of-the-art training method to approximate the network weight distribution with variational inference and calculate the uncertainties in the prediction. Compared to other existing methods, this method has the advantage of requiring essentially the same evaluation time of a standard network evaluation. It can be, therefore, particularly useful when the network is used in real time systems, which benefit of the uncertainty information when using the network prediction to make further actions or take decisions.

The network has been tested on data collected during several pulses at the JET tokamak, considering a wide range of plasma features and scenarios. A comparison of the network inferred profiles and those found with the conventional Bayesian inference shows a discrepancy in the two methods reconstructed uncertainties. This should not surprise, as they arise from two very different models with different free parameters, observed quantities, and different limitations, and therefore they are not expected to match. This discrepancy is a price that has to be paid to achieve the several orders of magnitude acceleration provided by the network. As we trained the network on a Bayesian model, we could use the same model to simulate the observations, given the network reconstructed profiles, and compare them against the measurements. We included in the comparison the full Bayesian inference reconstruction, which was carried out making use of the same model. The comparison was therefore fully consistent: the network inversion being a fast approximation of the full model one. The error in the prediction of the measurements is consistently larger when using the network predicted density profiles, as it might be expected from an approximated inversion. Still, the error is consistently below approximately 20% in all considered experimental cases, suggesting that the network inversion can be a reliable tool for fast analysis.

In future works, the NN could be used as a initial guess for the Bayesian inference carried out with the Minerva model, in this way speeding up the sampling of the posterior distribution with the MCMC by quickly providing a good starting location. The network could also be used independently, providing a fast edge profile reconstruction. For the reconstruction to be reliable, the network could be tested on a larger data set of measurements collected at previous experiments and the cases where the reconstruction fail should be investigated individually. Also, the implementation of a *novelty detection* system could be useful: this is a system which can preventively inform the user when a measurement represents an input which is unfamiliar for the network with respect to the data that had been used for training it. These cases often bring to unreliable network output and, in this way, they could be readily identified. A novelty detection method can rely on the reconstruction of the probability density of the input training data, which is then evaluated at the location of the incoming measurement input in order to asses its degree of novelty [41].

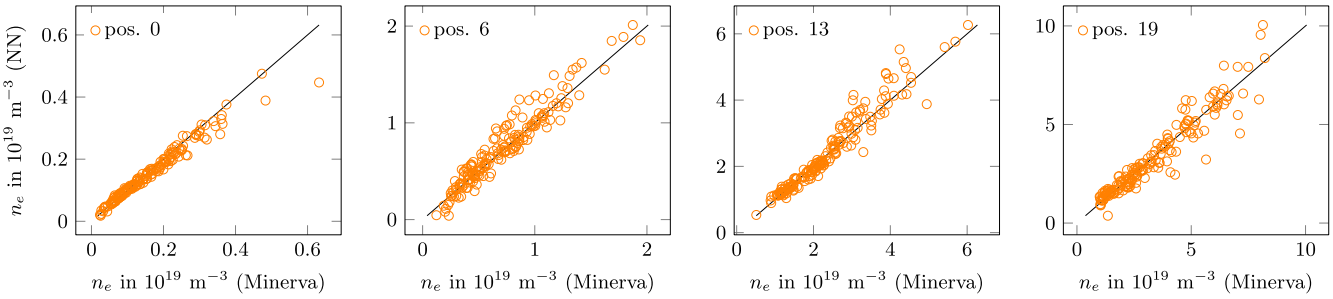

Figure 10. The electron density profile values inferred with model Bayesian inference (Minerva), on the x-axis, are compared to those inferred with the network (NN), on the y-axis. Each plot shows the comparison for a spatial position along the profile. The real space coordinates of the positions can vary through the experiments, so here they are labeled by an index starting from the outermost position at index 0 to the innermost position at index 19. The solid line shows the $y = x$ line. More than 200 hundred measured data points collected across 65 pulses were used.

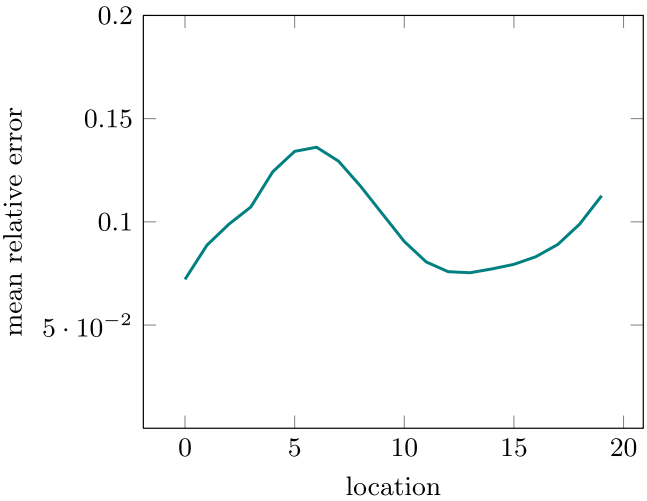

Figure 11. The mean relative error between electron density profile inferred with Minerva and with the network. The average values from the posterior samples in the Minerva case and the MC dropout sampels in the network case have been used. The calculation has been carried out for more than 200 measurements collected across 65 JET pulses.

Acknowledgments

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.

Appendix. List of JET pulses

What follows is a list of the JET pulses used in the analysis shown in figures 8 and 10, and discussed in section 6. The pulses were arbitrarily chosen, without selecting for a specific set of features or plasma configurations. The pulses featured a broad range of parameters, including both L- and H-mode scenarios, low and high power and gas levels. Across all pulses, the NBI power ranged from ≈3.0 to ≈28 MW, the vacuum toroidal magnetic field from 1.6 to 3.3 T, the total ICRH power from ≈2.0 to ≈6.0 MW, the plasma current from ≈1.1 to ≈3.5 MA, and the line integrated density from $\approx 8.0 \times 10^{19}\,\mathrm{m}^{-2}$ to $\approx 2.6 \times 10^{20}\,\mathrm{m}^{-2}$.

86685, 86687, 86902, 86906, 86911, 86913, 86918, 86983, 87080, 87091, 87094, 87143, 87184, 87260, 87261, 87283, 87411, 87412, 87487, 87518, 87562, 87790, 87792, 87825, 87864, 87865, 87873, 89094, 89095, 89110, 89174, 89193, 89231, 89237, 89248, 89312, 89341, 89342, 89343, 89344, 89345, 89346, 89347, 89349, 89351, 89353, 89387, 89390, 89391, 89392, 89393, 89395, 89425, 89426, 89427, 89448, 89449, 89450, 89451, 89705, 89707, 89708, 89727, 89728.

ORCID iDs

A Pavone https://orcid.org/0000-0003-2398-966X
R C Wolf https://orcid.org/0000-0002-2606-5289

References

[1] Svensson J *et al* 1998 Real-time ion temperature profiles in the JET nuclear fusion experiment *ICANN 98. Perspectives in Neural Computing* (https://doi.org/10.1007/978-1-4471-1599-1_30)
[2] Svensson J, von Hellermann M and König R W T 1999 Analysis of JET charge exchange spectra using neural networks *Plasma Phys. Control. Fusion* **41** 315–38
[3] Bishop C M, Roach C M and von Hellermann M G 1993 Automatic analysis of JET charge exchange spectra using neural networks *Plasma Phys. Control. Fusion* **35** 765–73
[4] Cannas B, Fanni A, Sonato P and Zedda M K 2007 A prediction tool for real-time application in the disruption protection system at JET *Nucl. Fusion* **47** 1559–69
[5] Böckenhoff D *et al* 2018 Reconstruction of magnetic configurations in w7-X using artificial neural networks *Nucl. Fusion* **58** 056009
[6] Blatzheim M *et al* 2019 Neural network regression approaches to reconstruct properties of magnetic configuration from wendelstein 7-X modeled heat load patterns *Nucl. Fusion* **59** 126029
[7] Ferreira D R, Carvalho P J, Fernandes H and (JET Contributors) 2018 Full-pulse tomographic reconstruction with deep neural networks *Fusion Sci. Technol.* **74** 47–56

[8] Meneghini O *et al* 2017 Self-consistent core-pedestal transport simulations with neural network accelerated models *Nucl. Fusion* **57** 086034
[9] Meneghini O, Luna C J, Smith S P and Lao L L 2014 Modeling of transport phenomena in tokamak plasmas with neural networks *Phys. Plasmas* **21** 060702
[10] van de Plassche K L *et al* 2019 Fast modelling of turbulent transport in fusion plasmas using neural networks *Phys. Plasmas* **27** 022310
[11] Rodriguez-Fernandez P *et al* 2018 Vitals: a surrogate-based optimization framework for the accelerated validation of plasma transport codes *Fusion Sci. Technol.* **74** 65–76
[12] Pietrzyk Z A, Breger P and Summers D D R 1993 Deconvolution of electron density from lithium beam emission profiles in high edge density plasmas *Plasma Phys. Control. Fusion* **35** 1725–44
[13] Brix M *et al* 2010 Upgrade of the lithium beam diagnostic at jet *Rev. Sci. Instrum.* **81** 10D733
[14] Brix M *et al* 2012 Recent improvements of the jet lithium beam diagnostic *Rev. Sci. Instrum.* **83** 10D533
[15] Réfy D I *et al* 2018 Sub-millisecond electron density profile measurement at the jet tokamak with the fast lithium beam emission spectroscopy system *Rev. Sci. Instrum.* **89** 043509
[16] Kwak S, Svensson J, Brix J and Ghim Y-C 2017 Bayesian electron density inference from jet lithium beam emission spectra using gaussian processes *Nucl. Fusion* **57** 036017
[17] Svensson J and Werner A 2007 Large scale bayesian data analysis for nuclear fusion experiments *IEEE Int. Symp. on Intelligent Signal Processing* pp 1–6
[18] Svensson J *et al* 2011 Modelling of jet diagnostics using bayesian graphical models *Contrib. Plasma Phys.* **51** 03
[19] Schmuck S, Svensson J, Figini L and Micheletti D 2019 Towards a bayesian equilibrium reconstruction using JET's microwave diagnostics 07 *46th European Physical Society Conference on Plasma Physics (EPS 2019) (Milan, Italy)*
[20] Hoefel U *et al* 2019 Bayesian modelling of microwave radiometer calibration on the example of the wendelstein 7-x electron cyclotron emission diagnostic *Rev. Sci. Instrumen.* **90** 043502
[21] Langenberg A *et al* 2016 Forward modeling of x-ray imaging crystal spectrometers within the Minerva Bayesian analysis framework *Fusion Sci. Technol.* **69** 560–7
[22] Abramovic I *et al* 2019 Forward modeling of collective thomson scattering for wendelstein 7-x plasmas: electrostatic approximation *Rev. Sci. Instrum.* **90** 023501
[23] Pavone A *et al* 2019 Neural network approximation of bayesian models for the inference of ion and electron temperature profiles at w7-X *Plasma Phys. Control. Fusion* **61** 075012
[24] Kwak S, Svensson J, Brix M and Ghim Y-C 2016 Bayesian modelling of the emission spectrum of the joint european torus lithium beam emission spectroscopy system *Rev. Sci. Instrum.* **87** 023501
[25] Schweinzer J *et al* 1992 Reconstruction of plasma edge density profiles from li i (2s–2p) emission profiles *Plasma Phys. Control. Fusion* **34** 1173–83
[26] Pasqualotto R *et al* 2004 High resolution thomson scattering for joint european torus (jet) *Rev. Sci. Instrum.* **75** 3891–3
[27] Pearl J 1986 Fusion, propagation, and structuring in belief networks *Artif. Intell.* **29** 241–88
[28] Rasmussen C E and Williams C K I 2006 *Gaussian Processes for Machine Learning* (Cambridge, MA: MIT Press)
[29] Summers H P 2004 *The ADAS User Manual* version 2.6 (http://www.adas.ac.uk)
[30] Klambauer G, Unterthiner T, Mayr A and Hochreiter S 2017 Self-normalizing neural networks *Proceedings of the 31st International Conference on Neural Information Processing Systems (Long Beach, CA)* pp 971–81
[31] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization arXiv:1412.6980
[32] Abadi M *et al* 2016 TensorFlow: large-scale machine learning on heterogeneous systems arXiv:1603.04467
[33] Hinton G E *et al* 2012 Improving neural networks by preventing co-adaptation of feature detectors 07 arXiv:1207.0580
[34] Srivastava N *et al* 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15** 1929–58
[35] Gal Y 2016 Uncertainty in deep learning *PhD Thesis* University of Cambridge
[36] Mackay D J C 1991 Bayesian methods for adaptive models *PhD Thesis* California Institute of Technology
[37] Jordan M I *et al* 1999 An introduction to variational methods for graphical models *Mach. Learn.* **37** 183–233
[38] Osband I 2016 Risk versus uncertainty in deep learning:bayes, bootstrap and the dangers of dropout *NIPS*
[39] Hron J *et al* 2018 *Variational Bayesian dropout: pitfalls and fixes* arXiv:1807.01969
[40] Huggins H J *et al* 2018 Practical bounds on the error of bayesian posterior approximations: a nonasymptotic approach arXiv:1809.09505
[41] Bishop C M 1994 Novelty detection and neural network validation *IEE Proc., Vis. Image Signal Process.* **141** 217–22