Improved supervised learning methods for EoR parameters reconstruction

Aristide Doussot, Evan Eames, Benoit Semelin

Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

MNRAS **490,** 371–384 (2019)
Advance Access publication 2019 September 12

doi:10.1093/mnras/stz2429

Improved supervised learning methods for EoR parameters reconstruction

Aristide Doussot ,★ Evan Eames and Benoit Semelin
Sorbonne Université, Observatoire de Paris, PSL research university, CNRS, LERMA, F-75014 Paris, France

Accepted 2019 August 28. Received 2019 August 28; in original form 2019 April 5

ABSTRACT

Within the next few years, the Square Kilometre Array (SKA) or one of its pathfinders will hopefully detect the 21-cm signal fluctuations from the Epoch of Reionization (EoR). Then, the goal will be to accurately constrain the underlying astrophysical parameters. Currently, this is mainly done with Bayesian inference. Recently, neural networks have been trained to perform inverse modelling and, ideally, predict the maximum-likelihood values of the model parameters. We build on these by improving the accuracy of the predictions using several supervised learning methods: neural networks, kernel regressions, or ridge regressions. Based on a large training set of 21-cm power spectra, we compare the performances of these methods. When using a noise-free signal generated by the model itself as input, we improve on previous neural network accuracy by one order of magnitude and, using a local ridge kernel regression, we gain another factor of a few. We then reach an accuracy level on the reconstruction of the maximum-likelihood parameter values of a few per cents compared the 1σ confidence level due to SKA thermal noise (as estimated with Bayesian inference). For an input signal affected by an SKA-like thermal noise but constrained to yield the same maximum-likelihood parameter values as the noise-free signal, our neural network exhibits an error within half of the 1σ confidence level due to the SKA thermal noise. This accuracy improves to 10 per cent of the 1σ level when using the local ridge kernel. We are thus reaching a performance level where supervised learning methods are a viable alternative to determine the maximum-likelihood parameters values.

Key words: intergalactic medium – dark ages, reionization, first stars – cosmology: theory.

1 INTRODUCTION

It has been recognized for more than 20 yr that the neutral hydrogen in the inter-galactic medium (IGM) before and during the process of reionization of the universe must have emitted radiations at 21 cm that, redshifted at meter wavelengths by cosmic expansion, should be observable nowadays with adequate radiotelescopes (Madau, Meiksin & Rees 1997; Furlanetto, Peng Oh & Briggs 2006; Pritchard & Loeb 2012; Mellema et al. 2013; Koopmans et al. 2015). The main difficulty in detecting this signal is to separate it from various types of foreground emissions (galactic synchrotron, extragalactic point sources, etc., see e.g. Di Matteo, Ciardi & Miniati 2004; Jelić et al. 2008). Single dipole instruments can measure the intensity of the signal as a function of frequency integrated on the sky (global signal). Such observations have the advantage that the signal-to-noise ratio does not depend on the collecting area, but the drawback that the limited amount of collected information gives us less leverage to separate the signal from the foregrounds and encodes less knowledge about the underlying astrophysical processes. A tentative first detection of the global signal has been reported by Bowman et al. (2018) with the EDGES instrument. It is likely that the cosmic origin of the detected feature can only be ascertained with future interferometric observations that would quantify the angular fluctuations in the detected feature (in the form of a power spectrum).

A number of instruments have been attempting to measure the power spectrum of the signal, although mostly at higher frequency (and thus lower redshift) than the EDGES detection. Only upper limits have been established so far, at various wavenumbers and redshifts (Paciga et al. 2013; Beardsley et al. 2016; Patil et al. 2017; Ali et al. 2018). With this type of observations, unambiguous separation of signal and foregrounds should be possible but the calibration of the instrument is a much more difficult task. Also, the higher information content of the measured quantity comes with the requirement of a large collecting area to improve the signal-to-noise ratio. As the methods to perform the calibration improve, we may see a first interferometric detection in the next few years. Then, next-generation instruments such as Square Kilometre Array (SKA) and HERA should be able to measure the power spectrum much more

★ E-mail: aristide.doussot@obspm.fr

accurately, detect it at lower frequencies (higher redshifts) where the foregrounds are stronger, and even, in the case of SKA, image the signal in three dimensions.

There is obviously a trade-off between the amount of information in a type of observation (global signal, power spectrum, imaging) and the collecting area and integration time required to perform it with a good enough signal-to-noise ratio. But in all cases, transforming this information into astrophysical or cosmological knowledge is not a straightforward process. Indeed, the local intensity of the signal depends, in some cases non-linearly, on the hydrogen density, ionization fraction, velocity, kinetic temperature, and on the local Ly α radiation field (see Furlanetto et al. 2006). These quantities are, in turn, correlated in a non-trivial manner through the process of structure formation. Thus, the first crucial step in interpreting the signal is to build a model that can compute the signal from such basic processes as growth of density fluctuations, formation of sources of radiations (stars and active galactic nucleus), and radiative feedback of the sources on their direct environment and on the IGM. These models can be analytical (e.g. Barkana & Loeb 2005; Pritchard & Furlanetto 2007), semi-numerical (e.g. Thomas et al. 2009; Santos et al. 2010; Mesinger, Furlanetto & Cen 2011; Fialkov, Barkana & Visbal 2014; Ghara, Choudhury & Datta 2015), or the result of radiative transfer cosmological simulations (e.g. Gnedin & Shaver 2004; Mellema et al. 2006; Valdés et al. 2006; McQuinn et al. 2007; Baek et al. 2009; and subsequent works). In all cases, the models will requires the use of astrophysical parameters to describe processes either not implemented ab initio and/or below the resolution of the computation. A simple example is the efficiency of star formation that would require a mass resolution below 1 solar mass and an extremely short time-step to be modelled self-consistently.

The second important step in extracting astrophysical knowledge from the observation is to use reliable statistical methods to put constraints to the models astrophysical parameters. A number of such methods exists. Pober et al. (2014) used the Fisher information matrix to derive confidence intervals for the parameter values; Greig & Mesinger (2015), Greig & Mesinger (2017), and Greig & Mesinger (2018) use Bayesian inference enacted by Markov chain Monte Carlo (MCMC) with the seminumerical code 21CMFAST. As even using 21CMFAST for MCMC Bayesian inference is computationally expensive, Kern et al. (2017), Schmit & Pritchard (2018), and Jennings et al. (2019) build an emulator of the code, using Gaussian Processes and Neural Networks. Another approach to parameter estimation is to train supervised learning algorithms to perform inverse modelling, taking some representation of the observed signal as an input and directly predicting the parameter values. As in the case of building an emulator, the training is specific to the chosen model, and typically requires a smaller number of modelling runs as MCMC inference does. For now, neural networks trained for inverse modelling have been implemented using the power spectrum (Shimabukuro & Semelin 2017) or the full tomographic data (Gillet et al. 2019) as input to predict best-fitting parameters values. Predicting confidence levels could be done in various ways, for example using Bayesian neural networks. It is not obvious at this stage, however, that the predicted confidence levels would have the exact same meaning as in classical Bayesian inference. When predicting best-fitting parameters using neural networks (or other supervised learning algorithms) trained to perform inverse modelling, an error exists, due to the imperfect training of the network. This training error can be exactly computed if the test input signal was produced by the model itself (then we know the corresponding *true* parameters). Note that finding parameter values that do not perfectly match a test signal *not* produced with the model is an issue not specific to supervised learning methods: maximum-likelihood parameters with a low likelihood value indicate an imperfect model. In any case, for supervised learning methods to be actually usable, we need to ensure that the training error is much smaller than the typical 1σ confidence due to the thermal noise in the target observation, as estimated by Bayesian inference. This should, of course, be true not only if a noise-free signal is fed to the network but also if a noised signal is considered. Such was not really the case in Shimabukuro & Semelin (2017), where the error is of the same order as the thermal noise, or in Gillet et al. (2019) where only a noise-free signal is considered. Thus, we need to improve the performance of supervised learning method implementations, either by improving the implemented methods, or exploring new ones.

In this work, we explore both of these possibilities. First, we improve substantially on the performances reached in Shimabukuro & Semelin (2017) using neural networks, by using a larger learning set, and optimizing several steps in the process. Then we explore another supervised learning method. Neural network have encountered great success when dealing with image classification. In this situation, the dimension of the signal space is huge, typically of the order of 10^6, the number of pixels in the image. In our case, the signal is the value of the power spectrum at various wavenumbers and redshifts. The dimension of the signal space is much lower, typically of the order of 10^2. In such comparatively low dimensions, advanced versions of the classical linear regression are known to perform well. Indeed, the linear regression, using the knowledge from a set of samples to approximate a model with a linear relation, can be classified as supervised learning. As 10^2 dimensions is still very large to apply the classical linear regression, kernel regression and ridge regressions have been developed (Hastie, Tibshirani & Friedman 2009). In this work, we combined these improved regression methods and push them as far as we can in term of performance to compare them with the neural network approach.

The layout of this article is as follows. In Section 2, we present the case to which we apply supervised learning: the input signal and thermal noise, the parameters to be predicted and the model that relates them in the case of forward modelling. In Section 3, we detail the different supervised learning methods studied in this work. In Section 4, we study the accuracy of these methods in term of the error on the reconstructed parameter values. In Section 5, we present our conclusions.

2 FRAMEWORK

2.1 The model: 21CMFAST

The learning process of any supervised learning method requires a training set consisting of a sufficient number of cases (where both inputs and outputs are known). Generating such a number of cases is currently beyond the reach of full-numerical simulations designed to predict the 21-cm signal. Consequently, we have selected the seminumerical code 21CMFAST (Mesinger et al. 2011) which is fast enough to provide the required number of cases in a reasonable amount of time. Let us briefly review some salient features of 21CMFAST numerical methods.

The main feature is that 21CMFAST does not include full radiative transfer, thus saving a lot of computation time. Instead, the ionization process is based on the 'excursion-set' approach (Furlanetto, Zaldarriaga & Hernquist 2004; Mesinger & Furlanetto 2007). The basic principle is that if the number of ionizing photon produced in a region is larger than the number of neutral hydrogen atoms in

the same region, the region is considered ionized (in practice, only the region centre cell is tagged as ionized, as regions centred on all cells will be considered). The photon production rate is assumed to be proportional to the collapse fraction (fraction of baryons in a collapsed object). At each location, the collapsed fraction smoothed on scale R, $f_{\rm coll}(\boldsymbol{x}, z, R)$, is compared to an efficiency parameter $\zeta_{\rm ion}$. The comparison is performed for decreasing R values, from a large-scale $R_{\rm mfp}$ to the cell size $R_{\rm cell}$. If $f_{\rm coll}(\boldsymbol{x}, z, R) > \zeta_{\rm ion}^{-1}$ then the centre cell of the region is flagged as ionized. Finally, at $R_{\rm cell}$, the ionizing fraction of the remaining cells that are not fully ionized is set to be $\zeta_{\rm ion} f_{\rm coll}(\boldsymbol{x}, z, R_{\rm cell})$. See Mesinger et al. (2011) for further details.

Another cost-saving strategy implemented in 21CMFAST is to ignore baryonic dynamics and use simplified dark-matter dynamics. The dark-matter density field is linearly extrapolated from the primordial field using the standard Zel'Dovich approximation (Zel'dovich 1970). Baryons are simply assumed to track the dark matter exactly. See Mesinger & Furlanetto (2007) for further details. X-ray heating and Ly α contributions to the spin temperature of hydrogen are implemented in 21CMFAST, again using cost-saving strategies. However, we deactivated these processes in our study, setting ourselves in the high-spin temperature limit.

2.2 Selected EoR observables and model parameters

In our approach to supervised learning where our goal is to put constraints on model parameters using observables, the observables are the inputs of the method and the parameters values are the outputs. Let us specify which inputs and outputs have been used in this work.

2.2.1 EoR observable

In our study, we chose to focus on the power spectrum of the intergalactic 21-cm signal, assuming that the non-Gaussianities (Shaw, Bharadwaj & Mondal 2019) of the signal are not necessary to accurately reconstruct the parameters. More precisely, we chose to consider the values of the power spectrum at 12 different wavenumbers k, logarithmically sampled from 4.42×10^{-2} to 3.20 cMpc^{-1}, for integer values of the redshift z from 5 to 15. Then, the signal that is used as an input lives in a space of dimension 120. This choice allows us to work in relatively low dimension unlike, for example, Gillet et al. (2019) who deal with the full information from the light-cone using convolutional neural networks. While neural networks have shown their ability to deal with high-dimensional signals (dimension 10^6) when analysing images for example, other supervised learning methods, such as the different flavours of linear regression presented here, are well suited to lower dimensionality.

2.2.2 Choice of EoR parameters and sampling

Concerning the EoR parameters that we want to reconstruct, we have chosen three parameters that have often been considered in other works (Greig & Mesinger 2015; Greig & Mesinger 2017; Eames, Doussot & Semelin 2019; Greig & Mesinger 2018; Schmit & Pritchard 2018):

(i) $\zeta_{\rm ion}$ accounts for the ionizing efficiency of high-z galaxies and can be expressed as

$$\zeta_{\rm ion} = 30 \left(\frac{f_{\rm esc}}{0.3}\right)\left(\frac{f_*}{0.05}\right)\left(\frac{N_\gamma}{4000}\right)\left(\frac{2}{1+n_{\rm rec}}\right) \quad (1)$$

with $f_{\rm esc}$ the ionizing photon escape fraction, f_* the fraction of galactic gas in stars, N_γ the number of ionizing photons produced per baryon in stars, and $n_{\rm rec}$ the typical number of times a hydrogen atom recombines during the EoR

(ii) $R_{\rm mfp}$ is the mean free path of ionizing photons within the ionized regions, regulated by the existence of unresolved damped Ly α systems.

(iii) $T_{\rm vir}$ is a mass threshold above which haloes are allowed to form stars and begin ionizing their surroundings.

Detailed definitions of these parameters are given in Greig & Mesinger (2015).

We based our study on a learning set of 2400 labelled cases, generated for our previous study in Eames et al. (2019), corresponding to the nodes of a logarithmic $20 \times 6 \times 20$ grid in the parameter space ($\zeta_{\rm ion}$; $R_{\rm mfp}$; $T_{\rm vir}$) with the following boundaries:

(i) $\zeta_{\rm ion} \in [20, 200]$
(ii) $R_{\rm mfp} \in$ [5cMpc, 35cMpc]
(iii) $T_{\rm vir} \in [8.0 \times 10^3$ K, 10^5 K]

Let us emphasize that this sampling method by no means ensures a maximization of the information. Methods that optimize the sampling (for a fixed number of cases and fixed explored volume in parameter space) to maximize the information are presented in Eames et al. (2019) and appear to lead to a better training of, at least, neural network methods. Further details on the set-up of the 21CMFAST runs performed for each triplet of parameter values can be found in Eames et al. (2019).

2.3 SKA noise modelling

For supervised learning methods designed to constrain model parameters to be of any use, they have to be able to handle a signal affected by the observational noise. We will concentrate on the thermal noise from the SKA, neglecting other possible sources such as imperfect foreground removal, residual calibration errors, or even sample variance. To model the expected thermal noise, we consider the SKA specifications as detailed in Dewdney (2013). Following McQuinn et al. (2006), we write the detector noise covariance matrix as

$$C(\boldsymbol{k}_i, \boldsymbol{k}_j) = \frac{1}{B t_{\boldsymbol{k}_i}} \left(\frac{\lambda^2 B T_{\rm sys}}{A_e}\right)^2 \delta_{ij}, \quad (2)$$

where B is the bandwidth, $t_{\boldsymbol{k}_i}$ is the effective observing time of the instrument in the grided visibility cell corresponding to wavenumber $\boldsymbol{k}_i$, λ is the observed wavelength, $T_{\rm sys}$ is the total system temperature, and A_e is the effective area of the station. For the system temperature, we have used $T_{\rm sys} = 100 + 300 \left(\frac{\nu}{150\,\text{MHz}}\right)^{-2.55}$K (Mellema et al. 2013). Lacking data from a definitive design of the future SKA-Low antennas, we have used an effective area for a station composed of 256 antennas of: $A_e = 256 \times \min(2.56, \frac{\lambda^2}{3})$ m^2. We have assumed a bandwidth $B = 10$ MHz, and a station diameter of 35 m determining the field of view. Finally, we have computed the $t_{\boldsymbol{k}_i}$ by integrating in visibility space the trajectories of the baselines from the SKA specifications. We considered 8 h runs for a total integration time of 1000 h, and a target field with declination -30 deg (close to the zenith for SKA-Low).

From the detector noise covariance matrix, we can compute the 1σ uncertainty on the power spectrum due to thermal noise as

$$\delta P_{\Delta T}^{21}(k) = \left[\sum_{|\boldsymbol{k}|=k} \left(\frac{1}{\frac{A_e x^2 y}{\lambda^2 B^2} C(\boldsymbol{k}, \boldsymbol{k})}\right)^2\right]^{-\frac{1}{2}} \quad (3)$$

where the sum extends over Fourier-space cells in the spherical shell with radius k (and also thickness $\Delta k = k$ in our case, which is an usual but determining choice), x is the comoving distance to the observed redshift and y the depth of the field (a distance) as determined by the bandwidth and the cosmology. The resulting level of noise is very similar to that in Koopmans et al. (2015).

See McQuinn et al. (2006) for further details on establishing the formulas. Once $\delta P_{\Delta T}^{21}(k)$ is computed for our binned wavenumbers, we simply add a realization of this noise to the signal to get a noised power spectrum.

3 SUPERVISED LEARNING METHODS

A well-established way of predicting underlying astrophysical parameters using observables is Bayesian inference associated to MCMC sampling. However, it often requires numerous instances of forward modelling to predict one observable, like in 21CMMC (Greig & Mesinger 2015; Greig & Mesinger 2017; Greig & Mesinger 2018; Park et al. 2019) where the forward modelling is performed using 21CMFAST (Mesinger et al. 2011). This inherently comes with a high computational cost, even if some attempts on designing a fast 21-cm power spectrum emulator using Gaussian processes (Kern et al. 2017; Jennings et al. 2019) or support vector machine (Jennings et al. 2019) to replace 21CMFAST have significantly accelerated the process.

With supervised learning methods trained to perform inverse modelling and predict parameter values, a typically smaller number of forward modelling instances is needed to build the learning set, decreasing the required computational time compared to 21CMMC. We chose to focus on neural networks, as it appears to be the fastest method in term of computational time, and on ridge and kernel regressions that have not been explored for this purpose before.

3.1 Common features

3.1.1 Learning set and test set

Although different, the two classes of supervised learning methods studied in this work share common features. In essence, the supervised learning material consists of a set of cases, from which the algorithm can interpolate to successfully make predictions for cases not in the set. This set of labelled cases is usually called the learning set in the neural network field. To quantify the prediction quality of a method, a second set of cases, distinct of the first one, is used. In the neural network field, this sample is often referred as the test set. Performing the evaluation on the test set avoids being impacted by the well-known issue of overfitting on the learning set.

In our study, the learning set is either made of 2400 signals for the cases without instrumental noise added (that were already described in Eames et al. 2019) or of 20 noised realizations of each signals, which means 48 000 noised signals, when instrumental noise is taken into account. The test set is composed of 512 signals generated starting from random values of the three astrophysical parameters taken within the bounds of the grid-based learning set. When noise is included, we generate 40 realizations of each signal which leads to a test set composed of 20 480 noised signals.

3.1.2 Limitations to absolute performance evaluation

Any supervised learning algorithm includes, in various forms, adjustable quantities, often called weights, that encode the computation of the outputs. The learning process therefore consists on adjusting the weights to accurately recover the known (labelled) outputs of the learning set, based on its inputs, by minimizing a given error function. The function to minimize usually depends on various adjustable hyper-parameters like a learning rate η or the weight decay rate λ. Changing the values of these hyper-parameters thus leads to different error functions, different minimization results and therefore different predictions. Optimizing the values of the hyper-parameters is of paramount importance to obtain the best possible predictions. However, it is almost impossible to make sure that a set of hyper-parameter values is a global optimum, especially with neural networks where there is an infinite number of possible architectures. The comparison between methods can only be done with parameter values that are, at best, local optima in the hyper-parameter space.

Also, our learning and test sets are generated with the same seminumerical model: 21CMFAST (Mesinger et al. 2011). Any conclusion that we reach concerning the accuracy of parameter reconstruction using different methods will only hold when applied to a real observed signal if the model is able to reproduce the observed signal. More quantitatively, in the 120-dimensional signal space, the signals produced by our three-parameter model occupy a three-dimensional manifold. The observed signal will not lie in this manifold unless the model is perfect. Even when noise is included, the performances of our parameter reconstructions methods are only evaluated close to this manifolds (at distances typically corresponding to a 1σ thermal noise). If the observed signal is at a distance equivalent to many sigmas, our conclusions cannot apply.

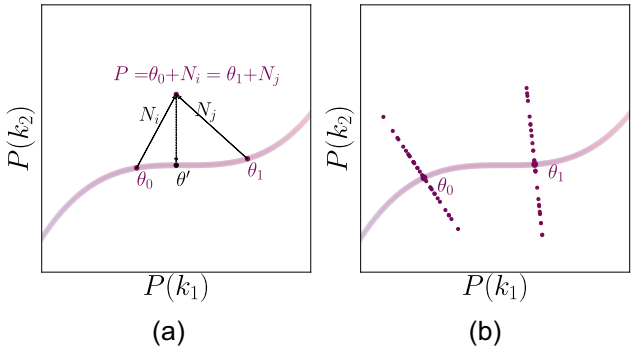

Figure 1. A toy model with a single parameter θ predicting a two-valued power spectrum. The left-hand panel shows how, when adding noise, two different parameter values can result in the same noised signal, neither of which would be the highest likelihood parameter value (θ' in this case). The right-hand panel shows how the ambiguity disappears when considering noise realizations that are perpendicular to the model-manifold.

3.2 Preparing the data

3.2.1 Labelling a noised signal: theoretical issue

A difficulty appears when using a noised signal. For a learning set sufficiently dense in the (noise-free) signal space, two neighbouring noise-free signals could be altered with two different realizations of the instrumental thermal noise in such a way as to lead to the same noised signal. This is of course why Bayesian inference predicts a distribution of possible parameter values instead of a single value. We show this problem in Fig. 1 for a toy model with a two-dimensional (2D) signal, composed of the power-spectrum values at two wavenumbers k_1 and k_2, and produced by a model with only

one parameter θ. The purple line represents the manifold of all the possible signals produced by the model, the model-manifold for short. We illustrate that the noised signal P has been obtained in two different ways, one starting from the signal corresponding to parameter value θ_0 with a noise N_i and the other starting from the signal corresponding to parameter value θ_1 with a noise N_j.

However, simple versions of neural networks cannot produce a distribution of parameter values as an output, only a single value (note however that Bayesian neural networks exist and could tackle the issue). To correctly evaluate the prediction ability of the supervised learning methods such as these single-value-output networks, it is therefore necessary to specify a single correct parameter value corresponding to a noised signal. The most logical answer is to decide that the correct parameter is the parameter corresponding to the noise-free signal on the model-manifold *closest* to the considered noised signal. If a natural distance is chosen in the signal space (e.g. L_2 norm), this minimal distance corresponds to adding the most likely realization of the thermal noise. This is in essence the maximum-likelihood value for the parameter (because the thermal noise is a Gaussian multivariate distribution). In the toy model of Fig. 1, we note this most probable parameter $\theta^{'}$. When we give to our methods the noised signal P, we thus expect them to predict the parameter $\theta^{'}$.

Finding this most probable parameter value for a given noised signal in the general case is the very purpose of methods that derive maximum-likelihood parameter values. We do not know how to do it at low cost when building our noised learning and test sets. We are however able to do so in specific cases: when the (noise-free) signal of the model-manifold closest to the noised signal under consideration belongs to our learning or test sets (for which we do know the exact parameters values). Such a noised signal belongs to a 117 dimensional hyper-plane of the signal space which is orthogonal (as defined by the chosen distance in signal space) to the model-manifold at the corresponding noise-free signal's position. Graphically, we show a finite number of these perpendicularly noised signals for the two signals corresponding to θ_0 and θ_1 in Fig. 1(b). Note that mathematically this procedure does not lift the ambiguity of determining the maximum-likelihood values in all cases.[1]

In essence, we are training our algorithms to perform an orthogonal projection on to the model-manifold. In doing so, we have to restrict our sampling to specific noised signals for the learning and test sets. We have to trust the algorithm to *interpolate* when we give it a general noised signal and perform this same orthogonal projection. This ability to interpolate (or *generalize*, using the coined word in the machine learning community) is the very purpose of machine learning.

3.3 Disentangling the inversion algorithm error from the thermal noise uncertainty

The necessity to choose a correct label (i.e. parameter value) for the signals in our learning and test sets comes with a crucial beneficial side effect. If everything worked perfectly, our inversion algorithms would predict the exact same parameters values for a given noise-free signal affected by any different realization of perpendicularized noise. In practice, this is not exactly the case as the learning process is not perfect. On the other hand, adding a general realization of the noise would displace the orthogonal projection on the model-manifold, compared to the position of the original noise-free signal, thus changing the maximum-likelihood parameter values.

Thus, using perpendicularized noise allows us to disentangle the error induced by the inversion algorithm itself on the reconstructed parameter value, from the intrinsic change to the maximum-likelihood parameter values that the general-case thermal noise can induce. Moreover, the typical range of this intrinsic change, that is typically evaluated by Bayesian confidence intervals, gives us a scale: we want, if possible, the algorithm's error to be much smaller.

3.3.1 *Generating the perpendicular noise*

The perpendicularized noise realizations that have to be generated are perpendicular to the model-manifold at the position of the signal to which they will be added. Let us first consider the case when the signals have been generated at the nodes of a grid in the parameter space, like for our learning set. In this scenario, we can use finite differences to estimate the tangent hyper-plane to the model-manifold, and thus the dual perpendicular space. For a signal $\boldsymbol{P}_{\zeta^i_{\rm ion}, R^j_{\rm mfp}, T^k_{\rm vir}}$ corresponding to the parameters $(\zeta^i_{\rm ion}; R^j_{\rm mfp}; T^k_{\rm vir})$, where i, j, and k denote the indexes on the grid, we apply an algorithm whose main steps are the following:

(i) Compute the vectors

$$\boldsymbol{V}_{\zeta_{\rm ion},i,j,k} = \boldsymbol{P}_{\zeta^{i+1}_{\rm ion}, R^j_{\rm mfp}, T^k_{\rm vir}} - \boldsymbol{P}_{\zeta^{i-1}_{\rm ion}, R^j_{\rm mfp}, T^k_{\rm vir}}, \tag{4}$$

$$\boldsymbol{V}_{R_{\rm mfp},i,j,k} = \boldsymbol{P}_{\zeta^i_{\rm ion}, R^{j+1}_{\rm mfp}, T^k_{\rm vir}} - \boldsymbol{P}_{\zeta^i_{\rm ion}, R^{j-1}_{\rm mfp}, T^k_{\rm vir}}, \text{ and} \tag{5}$$

$$\boldsymbol{V}_{T_{\rm vir},i,j,k} = \boldsymbol{P}_{\zeta^i_{\rm ion}, R^j_{\rm mfp}, T^{k+1}_{\rm vir}} - \boldsymbol{P}_{\zeta^i_{\rm ion}, R^j_{\rm mfp}, T^{k-1}_{\rm vir}} \tag{6}$$

that form a local basis in signal space generating the hyper-plane tangent to the model-manifold at signal $\boldsymbol{P}_{\zeta^i_{\rm ion}, R^j_{\rm mfp}, T^k_{\rm vir}}$.

(ii) Orthonormalize the previous basis to obtain an orthonormal basis whose elements will be referred to as $\boldsymbol{e}_{1,i,j,k}$, $\boldsymbol{e}_{2,i,j,k}$, and $\boldsymbol{e}_{3,i,j,k}$.

(iii) Generate a thermal noise $\boldsymbol{N}$ and compute

$$\boldsymbol{N}_\perp = \boldsymbol{N} - \boldsymbol{N}\cdot\boldsymbol{e}_{1,i,j,k} - \boldsymbol{N}\cdot\boldsymbol{e}_{2,i,j,k} - \boldsymbol{N}\cdot\boldsymbol{e}_{3,i,j,k}, \tag{7}$$

where dot $(\cdot)$ denotes the scalar product in signal space corresponding to our choice of distance (standard Euclidean in our case, matching the L_2 norm).

Any such $\boldsymbol{N}_\perp$ is locally perpendicular to the model-manifold. By construction, $(\zeta^i_{\rm ion}; R^j_{\rm mfp}; T^k_{\rm vir})$ are the parameters values that we will teach the inversion algorithms to predict when fed with the noised signal $\boldsymbol{P}_{\zeta^i_{\rm ion}, R^j_{\rm mfp}, T^k_{\rm vir}} + \boldsymbol{N}_\perp$.

The second case to consider is the case when to add noise to a signal that does not correspond to a node of a grid in parameter space, like in our test set. In this case, we determine the tangent hyper-plane using a weighted average of the hyper-planes for the neighbouring nodes of the grid. The details of the procedure are described in the Appendix.

It is worth noting that, with our method that uses centred finite differences, only the local bases at signals that are not at the edges of our domain can be computed; therefore, limiting the number of

[1]Two perpendicular hyper-planes from two neighbouring points of the model-manifold will intersect on the concave side of a curved manifold. Conversely, an observed noised signal (on the concave side) will belong to a single perpendicular hyper-plane only if its distance to the model-manifold is smaller than the radius of curvature of the manifold at the intersection point (speaking in terms of the 2D case).

data that can be noised in our learning set to 1296 and in our test set to 258.

3.3.2 Data pre-processing

As already mentioned in Section 3.2.1, considering the component of the noise perpendicular to the model-manifold does not completely lift the ambiguity in finding the signal on the model-manifold closest to a given observed signal, because of the manifold curvature. If the distance to the manifold is larger than the manifold inverse curvature (radius of curvature in 2D) there may be several points (or even a continuity of points) on the manifold whose perpendicular hyper-plane goes through the noised signal, only one of them being the closest. Thus, our method inherently has difficulty with observed signals far away from the manifold. If we are not careful, including a typical radio-interferometer thermal noise will automatically generate noised signals far from the manifold. Indeed, the model can easily (and does) generate the power spectrum at large wavenumbers where the thermal noise is large. In the case of a typical SKA layout, the thermal noise on the power spectrum typically increases as k^3 (see e.g. Koopmans et al. 2015). Then, including large wavenumbers without caution will break our approach. Indeed, our orthonormalized basis does not match the basis consisting of the 120 Dirac functions centred on the (k, z) values where our power spectrum is evaluated. On the Dirac functions basis, the large thermal noise components are localized on a few vectors of the basis. But when projected on the orthonormalized basis it contaminates all components. This problem would be alleviated if the noise already had components of similar amplitude on the Dirac functions basis. We arrive at the same conclusion by considering a general problem for supervised learning. If the fluctuations of a component of the signal are dominated by noise and not by variation due to changing values of the model parameter, this component will be of little help in constraining the parameters. The more relevant quantity to consider is of course the signal-to-noise ratio. This is equivalent to the traditional 'inverse variance weighting' used in radio-astronomy imaging. With this operation, the contribution of the noise will be similar for all components of the pre-processed signal. Using this pre-processing step and feeding the result to the supervised learning methods, we give ourselves a better chance that the noised signal will remain close to the model-manifold. We indeed verified that this pre-processing step generally improves the accuracy of the predictions.

3.4 Kernel regression

Let us now describe advanced versions of linear regressions that we will use as supervised learning methods.

3.4.1 Linear regression

When addressing an interpolation problem, one obvious, yet useful method that exists is the simple linear regression, which consists of performing the following minimization:

$$\min_{\alpha,\beta_j}\left[\sum_{i=1}^{N_S}\left(y_i-(\alpha+\sum_{j=1}^{N_D}\beta_j x_{i,j})\right)^2\right], \tag{8}$$

where N_S is the number of cases in the learning set, N_D is the dimension of the input, and y_i is the output. This method results in a global linear approximation of the interpolating function which is then used to predict the outputs of the data of the test set starting from their inputs. It is also equivalent to approximating the model-manifold with a single hyper-plane.

It is therefore logical to think that, for each data of the test set, a local linear approximation of the interpolated function around the considered input will give better results. This is equivalent to locally approximate the model-manifold with an hyper-plane that varies depending on the location.

3.4.2 Kernel smoothing

Introducing locality in the regression leads to the class of supervised learning methods called Kernel smoothing regression methods (Hastie et al. 2009). To achieve this, to each cases of the learning set participating in the regression we apply a weight:

$$K_\sigma(x_0,x_i)=\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-\frac{D(x_0,x_i)^2}{2\sigma^2}}, \tag{9}$$

where σ is a scale hyper-parameter describing the desired level of locality and $D(x_0,x_i)=\sqrt{\sum_{j=0}^{N_D}(x_{0,j}-x_{i,j})^2}$ is the distance in the signal space between the considered signal x_0 and the input signal x_i of the learning set. The quantity to minimize is then

$$\min_{\alpha,\beta_j}\left[\sum_{i=1}^{N_S}K_\sigma(x_0,x_i)\left(y_i-(\alpha+\sum_{j=1}^{N_D}\beta_j x_{i,j})\right)^2\right]. \tag{10}$$

The parameters α and β_j now depend on x_0.

3.4.3 Global ridge kernel regression

If the kernel smoothing method considers the local information, it does nothing to deal with an eventual degeneracy of the problem. Imagine that two of the input values that constitute the signal are perfectly correlated in the noise-free case when varying one of the model parameters. Then the prediction by the regression will be sensitive only to the average of the two β_i coefficients corresponding to these two correlated values. The two β_i could take very large values as long as the mean is correct. But then, when noise is added, which is uncorrelated at the two signal value, these two large β_i values will induce a large variance in the predicted parameter values. This situation, described in the case where noise is added, can also occur when moving from learning set to test set as the two sets can exhibit different levels of correlation between two signal values.

This problem can be alleviated by constraining the values of the coefficients, particularly relatively to one another (Hanke & Groetsch 1998; Calvetti et al. 2000; Hastie et al. 2009): this is what the ridge regression is about.

To implement this constraint, we add a penalty term in our minimization that becomes

$$\min_{\alpha,\beta_j}\left[\sum_{i=1}^{N_S}K_\sigma(x_0,x_i)\left(y_i-(\alpha+\sum_{j=1}^{N_D}\beta_j x_{i,j})\right)^2+\lambda\sum_{j=1}^{N_D}\beta_j^2\right] \tag{11}$$

where λ is an adjustable hyper-parameter. This ridge regression basically shrinks the values of the coefficients by imposing a penalty on their size. The whole coefficient shrinkage process is comparable to the weight-decay process used in neural networks and we can assimilate the hyper-parameter λ to a decay rate.

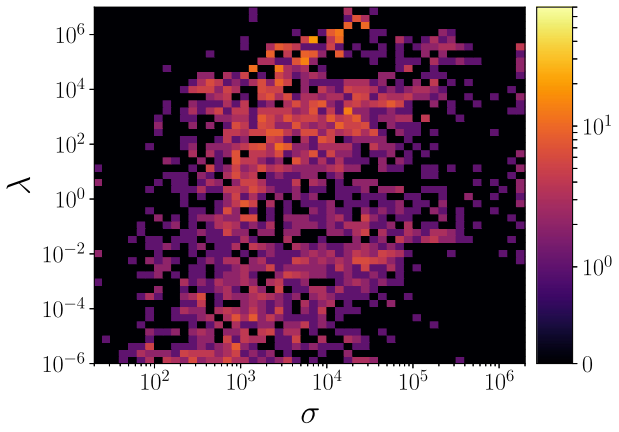

Figure 2. Histogram of the number of cases in the test set composed of noised signal, that find a particular (σ, λ) duplet of hyper-parameters as being optimal for the prediction by the local ridge kernel regression method.

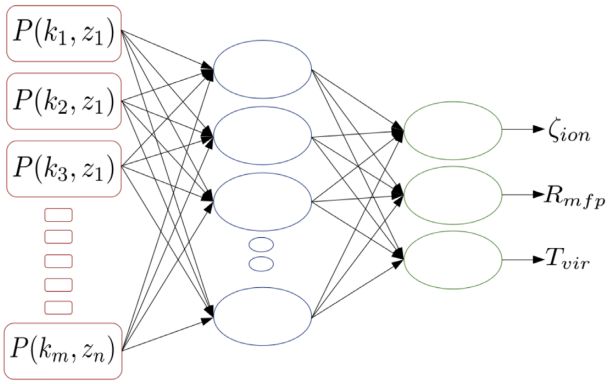

Figure 3. Schematized architecture of the neural network used in this study.

3.4.4 Local ridge kernel regression

One last step towards designing the most efficient regression is to consider the optimization of the two hyper-parameters σ and λ. It is likely that a global optimization of the hyper-parameters values on the overall domain of the signal space will result in a selection of mean values that enable most of the space to be correctly predicted but might critically fail in some area of the domain. One simple improvement is therefore to determine the best hyper-parameters values for each points of the test set, thus minimizing the quantity of equation (11) for α, β_j, σ, and λ.

We implemented the local optimization by doing a simple grid search where we allowed our hyper-parameters to vary in a vast range of value:

(i) $\sigma \in [2 \times 10^1, 2 \times 10^6]$
(ii) $\lambda \in [1 \times 10^{-6}, 1 \times 10^7]$

This wide range appeared to be necessary as the hyper-parameters indeed took vastly different values depending on the location in the signal space. To illustrate this assertion, we present in Fig. 2 an histogram of the optimized hyper-parameters values computed from all the cases in the test set composed of noised signals. The brute-force grid approach to minimization is guaranteed to find the global minimum in the explored domain. It is usable in our case because of the low number (two) of hyper-parameters. For a larger number of hyper-parameters, a minimization algorithm such as Levenberg–Marquardt should be used instead to achieve a reasonable computational cost, running the risk to converge to a local minimum.

However, whether locally or globally, optimizing the value of the hyper-parameters when predicting the outputs of the test set requires us to already know the true value of the outputs to compute an error function between the predictions and the true results. This will not be possible with an observed signal whose associated model parameters are unknown. Still, by using this knowledge in the case of the test set, we exhibit the theoretical maximum accuracy of this method. The optimization of the hyper-parameters in a real case (i.e. with an observed signal) is an open problem but one reasonable solution is to adopt for the same hyper-parameters as for the closest signal (in signal space) in the learning set. An optimization of the value of the hyper-parameters for each point of the learning set is therefore needed. Obviously, the accuracy of the predictions on the test set will be worse when using hyper-parameter values optimized for the closest signal in the learning set than when optimizing on the test set signal itself. This gap may be reduced in the future by improving the strategy to choose the optimal hyper-parameters of an observed signal.

3.5 Artificial neural network

3.5.1 Network architecture

The principle of neural networks will not be discussed in depth as it has already been extensively described in various works (for examples in this field, see Shimabukuro & Semelin 2017; Jennings et al. 2019), but we remind the reader that a neural network is basically composed of a set of calculus units, called neurons, that return an output which is the value of a function, called activation function, acting on the weighted sum of the inputs to the neuron. These units can be linked together in numerous fashion defining the architecture of the network. In our case, we use the KERAS framework[2] relying on TENSORFLOW[3] as a backend to implement a fully connected neural network with only one hidden layer of neurons, as shown in Fig. 3. Our hidden layer is composed of 80 neurons and our output layer of three neurons that each predicts the value of one of our three astrophysical parameters. With this simple architecture, a predicted parameter $y^{\rm pred}$ can be explicitly written in term of the inputs to the network as

$$y^{\rm pred} = f_2 \left(\sum_{i=1}^{80} W_i f_1 \left(\sum_{j=1}^{N_{\rm D}} w_{i,j} x_j + b_i \right) + b \right), \quad (12)$$

where $N_{\rm D}$ is the dimension of the input, x_j is the value of the jth component of the input, $w_{i,j}$ is the weight given to x_j by the ith neuron of the first layer, b_i is the bias added by the ith neuron, f_1 is the activation function of the first layer, f_2 is the activation function of the second layer, W_i is the weight given by the neuron of the second layer that predicts $y^{\rm pred}$ to the result of the ith neuron of the first layer, and b is the bias of the neuron of the second layer predicting the parameter $y^{\rm pred}$.

We chose a fairly simple architecture for our network, as it is fully connected and contains only one hidden layer. However, it has been mathematically proven (Cybenko 1989; Hornik, Stinchcombe & White 1989) that neural networks with only one hidden layer can approximate with any accuracy any function if a sufficiently large

[2] https://keras.io
[3] https://www.tensorflow.org/

Table 1. The detailed characteristics of our neural network. $N_{\rm TS}$ is the number of data in the test set, $N_{\rm P}$ is the number of predicted parameters, here 3, $y_{i,j}$ is the jth output parameter for the ith data in the test set, t is the epoch of learning, $w^t_{k,l}$ is the weight given by the lth neuron of a given layer to its kth input at the epoch of learning t.

Characteristics function	
Cost function : Mean squared logarithmic error	$C = \frac{1}{N_{\rm TS}} \sum_{i=1}^{N_{\rm TS}} C_i = \frac{1}{N_{\rm TS}} \sum_{i=1}^{N_{\rm TS}} \frac{1}{N_p} \sum_{j=1}^{N_p} \left[\log_{10} \left(\frac{y_{i,j}^{\rm pred}+1.}{y_{i,j}^{\rm true}+1} \right) \right]^2$
Optimization algorithm : RMSProp (Murugan & Durairaj 2017)	$w^{t+1}_{k,l} = w^t_{k,l} - \frac{\eta}{\sqrt{E\left[(\nabla^t_{w_{k,l}} C)^2\right]_t + 1\times10^{-7}}} \nabla^t_{w_{k,l}} C$ with $E\left[(\nabla^t_{w_{k,l}} C)^2\right]_t = 0.9E\left[(\nabla^{t-1}_{w_{k,l}} C)^2\right]_{t-1} + 0.1(\nabla^t_{w_{k,l}} C)^2$
Activation function of the hidden layer : ReLU	$f_1(x) = \begin{cases} x & \text{if} \quad x > 0 \\ 0 & \text{otherwise} \end{cases}$
Activation function of the output layer : Linear	$f_2(x) = x$
Hyper-parameters	
Learning rate η	5×10^{-4}
Batch size	128

number of neuron is used. The limit is rather in the size of the learning set describing the function to be interpolated. A network with many neurons but a small learning set will perform well on the learning set but weakly on the test set (overfitting phenomenon, or bias-variance trade-off). Evidences of the ability of this kind of network to handle the underlying denoising task have also been established (Burger, Schuler & Harmeling 2012).

3.5.2 *Network characteristics*

The learning process of a neural network is done by a gradient descent algorithm, referred as optimization algorithm. During the learning process, the optimization algorithm, using a prediction and its quality estimated by an error function, adjusts the weights given by each neurons to each inputs. The newly adjusted weights are then used to make another prediction during the next step and the weights are again re-adjusted to minimize the error function. This scheme is repeated over thousands of step, called epochs of learning, until the error function stops decreasing. For reference, we present all the characteristics of our chosen learning process in Table 1.

4 RESULTS

Throughout this section, we will mainly quantify the quality of the predictions of our method through the root-mean square (rms) relative error, computed individually for each parameters and defined as

$$\chi_y = \sqrt{\frac{1}{N_{TS}} \sum_{i=1}^{N_{TS}} \left(\frac{y_i^{\rm pred} - y_i^{\rm true}}{y_i^{\rm true}} \right)^2} \tag{13}$$

where $N_{\rm TS}$ is the number of data in the test set and y_i is the value of either $\zeta_{\rm ion}$, $R_{\rm mfp}$, or $T_{\rm vir}$ for the ith case of this set. Tables and figures in this section are computed for the signals in our test set. We first compare our different supervised learning methods for the noise-free cosmological signals in the test set described in Section 3.1.1, so we can compare our results to what has been done in Shimabukuro & Semelin (2017). We then compare our methods for predicting the astrophysical parameters from noised signals, affected by the perpendicularized noise described in Section 3.3.1. When using these very specific realizations of the noise, a noised signal should correspond to the exact same (maximum-likelihood) model parameter values as the corresponding noise-free signal. Thus, we are not exploring the intrinsic uncertainty on the predicted parameter values induced by thermal noise (i.e. the variance of the posterior distribution of the parameters values in the Bayesian approach), but rather the additional error introduced by the supervised learning algorithm in the parameter estimation. Our goal is to make sure that this additional error is small compared to the uncertainty due to thermal noise as estimated, for example, by Bayesian inference.

4.1 Noise-free cosmological signal

4.1.1 *Uncovering an unintended degeneracy in the model*

Testing our methods with a test set composed of numerous signals allows us to have the equivalent of multiple observations of the same phenomenon. This allowed us, during our investigation of the accuracy of the different methods, to reverse-engineer a feature of 21CMFAST that produces an unintended degeneracy in the $R_{\rm mfp}$ parameter. This feature, compared to other approximation, has a small impact on the accuracy of the model. However, our goal is to prove that our method is able to *invert* the model accurately, whether the model itself is realistic or not. Absolute degeneracies make perfect inversion impossible. To push beyond the accuracy limit induced by the degeneracy, we had to correct for it.

Fig. 4 depicts the predicted values of $R_{\rm mfp}$ as a function of the real ones, the black line being a perfect prediction and the yellow dot the actual predictions of our local ridge kernel regression method. It clearly shows that the predictions take mostly discrete values and are not simply exhibiting random deviations around the value used to compute the signal. Looking into the 21CMFAST source code, it appears that our method has correctly reconstructed a feature of the

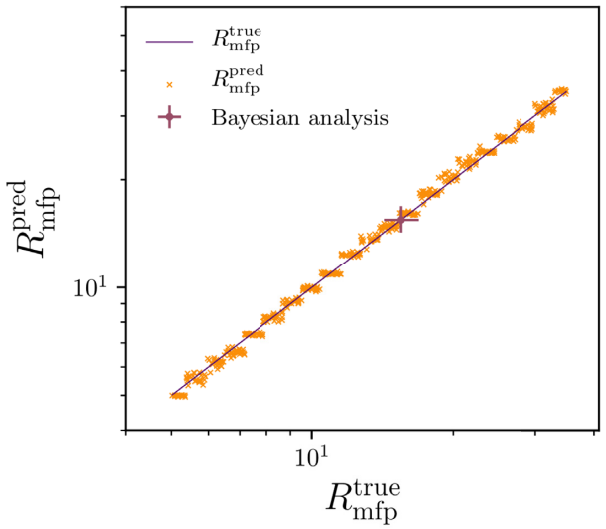

Figure 4. Predicted values of $R_{\rm mfp}$ as a function of the real ones for the theoretical perfect prediction (black line) and the actual predictions of our local ridge kernel regression method (yellow dots). For reference the 1σ uncertainty from SKA thermal noise as estimated from Bayesian inference (Greig & Mesinger 2015) is also plotted (purple cross).

code which we did not take into account when labelling signals. In this version of 21CMFAST, the radius of the regions to be tagged as ionized are investigated only in decrements of $\times$ 1.1. Consequently, only a variation of $R_{\rm mfp}$ by the same factor is guaranteed to affect the results. More precisely, in our case, the influence of $R_{\rm mfp}$ is the same for all $R_{\rm mfp}$ in $[4.8903 \times 1.1^n, 4.8903 \times 1.1^{n+1}]$ for $n \in \mathbb{N}$, explaining the observed steps in Fig. 4.

For the rest of the study, we re-labelled our signals to correct for this degeneracy by setting the 'true' value of any $R_{\rm mfp}$ in a given interval to the geometrical mean value of the interval

$$R_{\rm mfp}^{\rm true} = \sqrt{4.8903 \times 1.1^n \times 4.8903 \times 1.1^{n+1}} = 4.8903 \times 1.1^{n+\frac{1}{2}} \quad (14)$$

for $n \in \mathbb{N}$. Not correcting for this feature may have affected previous works attempting to constrain model parameters using 21CMFAST. The resulting constraints may have been less tight than they could have been. Note that the latest version of 21CMFAST does not implement $R_{\rm mfp}$ in the same way and may not be equally sensitive to this feature (although the discrete $\times$ 1.1 factor remains).

4.1.2 Performance comparison

Table 2 displays the rms relative errors of the prediction, computed individually for each parameters, on noise-free signals of the test set, for all the methods presented in Section 3 as well as results from Shimabukuro & Semelin (2017). We find that all methods of this work are better by one order of magnitude than the results presented in Shimabukuro & Semelin (2017). Beyond a more careful exploration and choice of the hyper-parameters of the learning process in the case of the neural network, the main reason for this improvement lies in the much larger learning set (justifying a network with more neurons) and the correction of the spurious degeneracy on $R_{\rm mfp}$. Our best supervised learning method for predicting a cosmological signal appears to be the local ridge kernel regression which improved Shimabukuro & Semelin (2017) results by at least a factor 50 for all three parameters, reaching a prediction rms relative error below 1 per cent. However, remember that this result shows the theoretical maximum accuracy with perfectly optimized hyper-parameters.

Fig. 5 shows the normalized distribution of the prediction relative error $\delta = \frac{y_{\rm pred}}{y_{\rm true}} - 1$ as a function of the true parameter $y_{\rm true}$ for our three astrophysical parameters $\zeta_{\rm ion}$, $R_{\rm mfp}$, and $T_{\rm vir}$ for noise-free signals of the test set, as well as the uncertainty induced by the SKA thermal noise as evaluated using Bayesian inference by Greig & Mesinger (2015) (green point) and the rms of relative error distribution computed at each bins (blue line). Note that each bin is defined for the value of one of the three parameters, the other two remaining unconstrained. Thus, each bin holds a sample of typically >25 cases. We present our three best supervised learning methods which are, from top to bottom: the neural network, the global ridge kernel regression and the local kernel ridge regression. For any of these methods, the rms relative errors of the predictions evaluated in different bins show only moderate fluctuations over the parameter range, and are inferior to a typical uncertainty induced by the SKA thermal noise. Moreover, for the local ridge kernel regression optimized on the test set the rms relative error of the prediction is only a few per cent of the uncertainty induced by the SKA thermal noise. While this needs to be confirmed in the case of the noised signal, it opens the door to using supervised learning as a method to determine the maximum-likelihood parameters associated with an observed signal.

4.2 Noised signal

When moving from a noise-free signal to a signal affected by noise, we are moving from a signal space that is effectively of dimension 3 (because we have three model parameters) to a signal space of dimension 120 (number of k bins times the number of redshifts). To improve the learning process, we need a finer sampling of the signal space. Thus, we generate 20 noised versions of the signal for each triplet of parameter values in the learning set. We do the same for the signals in the test set, generating 40 noised versions for each triplet of parameter values, to smooth our estimation of the error distributions.

4.2.1 Perpendicularized learning set and maximum-likelihood prediction

As explained in Section 3.2.1 using signals affected by a perpendicularized noise allows us to train the algorithm to predict the maximum-likelihood values of the parameters. Thus, using a learning set with signals affected by perpendicularized noise is the logical choice. Another possibility would be to trust the training of the learning methods to converge to an orthogonal projection of the signal space on to the model-manifold. This would likely happen in the case when the sampling of the signal space by the learning set is dense enough. We have no guaranty that this is the case in our situation. Thus, we experimented with learning sets affected by noise either perpendicularized or not.

We found that using a learning set with a perpendicularized noise leads to a slight improvement of around 8 per cent of the prediction accuracy for all methods. Even if the difference is not large, we decided to focus on a learning set with a perpendicularized noise, as it has a clearer theoretical interpretation. Let us now compare the performance of the different methods.

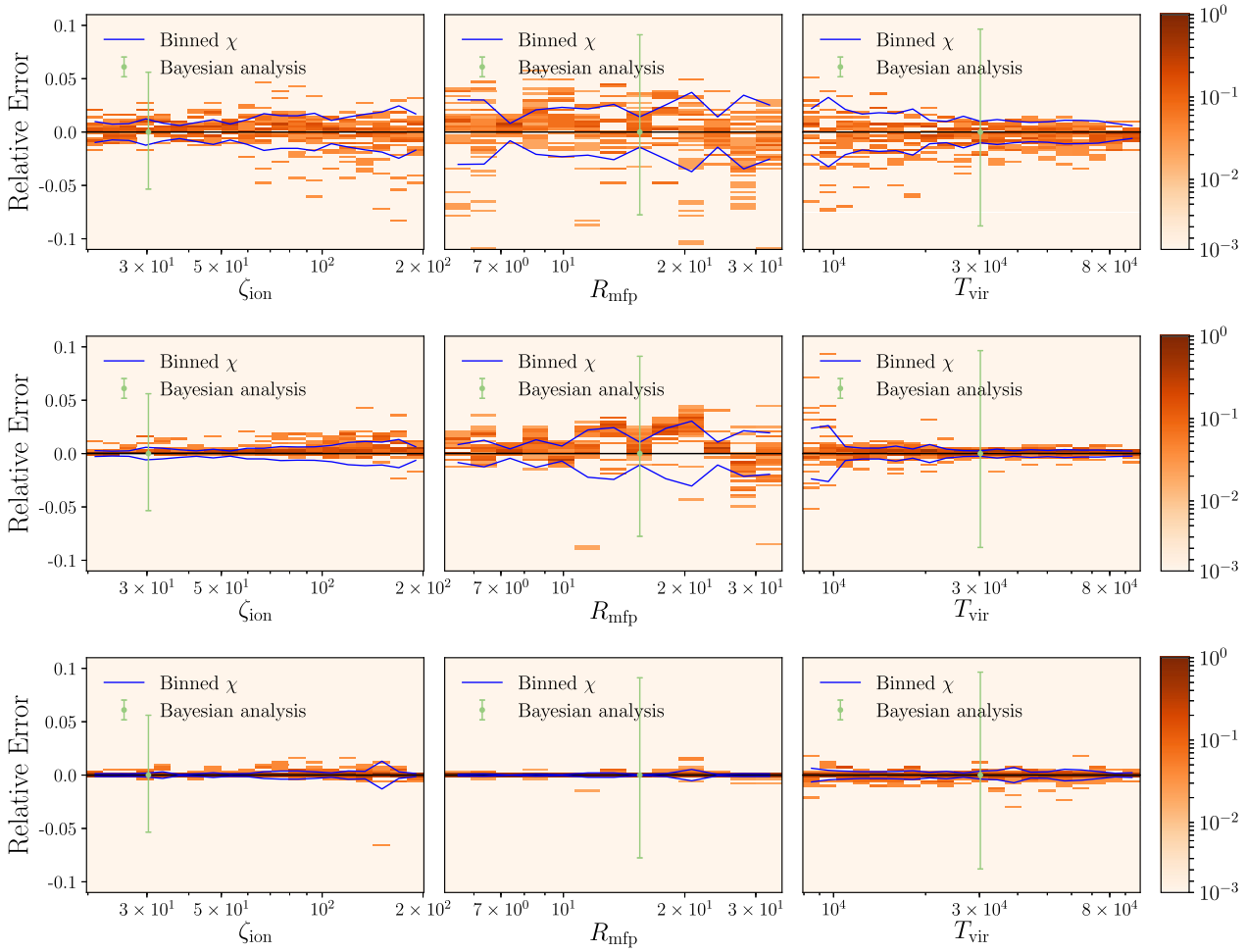

Figure 5. Normalized distribution of the prediction relative error $\delta = \frac{y_{\rm pred}}{y_{\rm true}} - 1$ as a function of the true parameter $y_{\rm true}$ for noise-free signals of the test set, for our three astrophysical parameters $\zeta_{\rm ion}$, $R_{\rm mfp}$, and $T_{\rm vir}$. The rms value of the error distribution is computed in each bin (blue line), and the 1σ uncertainty from SKA thermal noise as estimated from Bayesian inference (Greig & Mesinger 2015) is plotted for comparison (green point). All these are plotted for different supervised learning methods, from top to bottom: the neural network, the global ridge kernel regression and the local ridge kernel regression.

Table 2. Rms relative error, χ, of the prediction for noise-free signals of the test set, computed for each parameters, for all the methods presented in Section 3. The regressions have been optimized using the test set. Results from Shimabukuro & Semelin (2017) are shown for comparison.

Without noise	$\chi_{\zeta_{\rm ion}}$	$\chi_{R_{\rm mfp}}$	$\chi_{\log(T_{\rm vir})}$
(Shimabukuro & Semelin 2017)	27.1×10^{-2}	22.8×10^{-2}	2.7×10^{-2}
Linear regression	1.82×10^{-2}	8.00×10^{-2}	0.29×10^{-2}
Kernel smoothing	1.19×10^{-2}	6.13×10^{-2}	0.28×10^{-2}
Neural network	1.37×10^{-2}	2.53×10^{-2}	0.15×10^{-2}
Global ridge kernel regression	0.68×10^{-2}	1.76×10^{-2}	0.09×10^{-2}
Local ridge kernel regression	0.39×10^{-2}	0.19×10^{-2}	0.04×10^{-2}

4.2.2 Performance comparison

Table 3 presents the prediction rms relative errors computed individually for each parameters on the signals of the test set affected by perpendicularized noise as described in Section 3.3.1. We see that all our methods are reconstructing the astrophysical parameters with an accuracy of the same order of magnitude from a noised signal as from a noise-free signal, albeit worse by approximately a factor of 3. As we used a perpendicularized noise, this factor of 3 is a measure of how much larger is the error introduced by the prediction algorithm in the noised case compared to the noise-free case. It is not a measure of the impact of thermal noise on the parameter uncertainty. We show in Fig. 6 the normalized distribution of the prediction relative error $\delta = \frac{y_{\rm pred}}{y_{\rm true}} - 1$ as a function of the true parameter $y_{\rm true}$ value along with the rms value of the distribution in each bin (blue line) for our three astrophysical parameters $\zeta_{\rm ion}$, $R_{\rm mfp}$, and $T_{\rm vir}$. We show the results for our best methods which are, respectively, from top to bottom: the neural network, the global ridge kernel regression, and the local ridge kernel regression. We

Table 3. Rms relative errors χ of the predictions for signals of the test set affected by a perpendicularized noise (see Section 3.3.1), computed separately for each parameters. The rms values are given for all methods presented in Section 3 where the regressions have been optimized using the test set. Results from Shimabukuro & Semelin (2017) are presented for comparison, but note that these did not use a perpendicular noise.

With noise	$\chi_{\zeta_{\rm ion}}$	$\chi_{R_{\rm mfp}}$	$\chi_{log(T_{\rm vir})}$
Shimabukuro & Semelin (2017)	16.8×10^{-2}	17.2×10^{-2}	1.9×10^{-2}
Neural network	3.70×10^{-2}	4.04×10^{-2}	0.41×10^{-2}
Global ridge kernel regression	2.88×10^{-2}	2.84×10^{-2}	0.34×10^{-2}
Local ridge kernel regression	1.10×10^{-2}	0.60×10^{-2}	0.16×10^{-2}

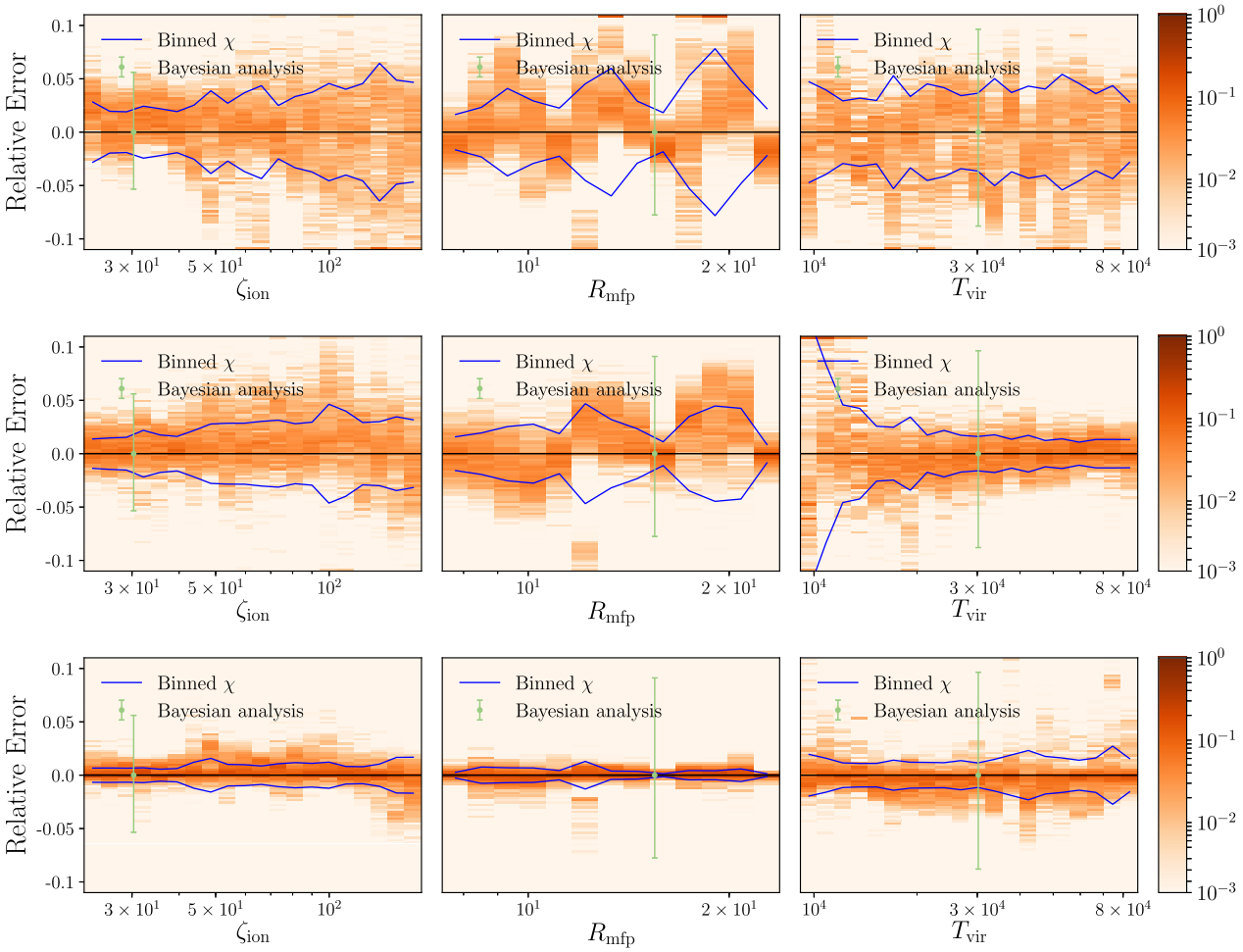

Figure 6. Normalized distribution of the prediction relative error $\delta = \frac{y_{\rm pred}}{y_{\rm true}} - 1$ as a function of the true parameter $y_{\rm true}$ for signals of the test set affected by perpendicularized noise as described in Section 3.3.1, for our three astrophysical parameters $\zeta_{\rm ion}$, $R_{\rm mfp}$, and $T_{\rm vir}$. The rms value of the distribution is computed in different bins (blue line), and the 1σ uncertainty from SKA thermal noise as estimated from Bayesian inference (Greig & Mesinger 2015) is plotted for comparison (green point). All these are plotted for different supervised learning methods, from top to bottom: the neural network, the global ridge kernel regression, and the local ridge kernel regression.

also show the 1σ uncertainty generated by SKA thermal noise as estimated with Bayesian inference (Greig & Mesinger 2015) (green bars) for comparison: we need our rms error to be, if possible, much smaller than this uncertainty. When considering the theoretical maximum accuracy of the local ridge kernel regression, which is when the hyper-parameters are optimized for the test set, we see that the prediction rms relative error is around 1 per cent. Compared to SKA thermal noise, the reconstruction error is indeed smaller by one order of magnitude. Thus, for a perfect optimization of the hyper-parameters, the local ridge kernel regression method enables a reconstruction of the highest likelihood astrophysical parameter with an error almost negligible compared to the size of the 1σ contour from Bayesian inference.

4.3 Hyper-parameters optimization strategy

Previous results for the local regression method should be considered as the theoretical maximum accuracy. As explained in

Table 4. Rms relative errors χ of the predictions for respectively: signals without noise (top) and signals affected by perpendicularized noise (bottom). The χ have been computed individually for each parameters, for the global and local ridge kernel regression presented in Section 3 and optimized using the learning set or the test set. The signals of the learning set have been noised with a perpendicularized noise. Results for our neural network are shown for comparison.

Without noise	$\chi_{\zeta_{\rm ion}}$	$\chi_{R_{\rm mfp}}$	$\chi_{log(T_{\rm vir})}$
Local ridge kernel regression using the learning set	2.69×10^{-2}	3.79×10^{-2}	0.48×10^{-2}
Global ridge kernel regression using the learning set	1.36×10^{-2}	3.78×10^{-2}	0.30×10^{-2}
Neural network	1.37×10^{-2}	2.53×10^{-2}	0.15×10^{-2}
Global ridge kernel regression using the test set	0.68×10^{-2}	1.76×10^{-2}	0.09×10^{-2}
Local ridge kernel regression using the test set	0.39×10^{-2}	0.19×10^{-2}	0.04×10^{-2}

With noise	$\chi_{\zeta_{\rm ion}}$	$\chi_{R_{\rm mfp}}$	$\chi_{log(T_{\rm vir})}$
Local ridge kernel regression using the learning set	10.3×10^{-2}	14.1×10^{-2}	4.23×10^{-2}
Global ridge kernel regression using the learning set	2.89×10^{-2}	2.83×10^{-2}	0.34×10^{-2}
Neural network	3.70×10^{-2}	4.04×10^{-2}	0.41×10^{-2}
Global ridge kernel regression using the test set	2.88×10^{-2}	2.84×10^{-2}	0.34×10^{-2}
Local ridge kernel regression using the test set	1.10×10^{-2}	0.60×10^{-2}	0.16×10^{-2}

Section 3.4.4, we have optimized our hyper-parameters by using prior knowledge of the true value of the parameters, which is obviously not possible for an observed signal. Thus, we will now consider the case when the hyper-parameters have been optimized using only the information from the learning set, a method that can directly be applied to an observed signal. In this case, the hyper-parameters values assigned to a signal of the test set are the optimized values determined for the closest signal in the learning set. Let us note that the gap between the theoretical maximum accuracy and the accuracy of this optimization on the learning set may be narrowed in the future with better hyper-parameter optimization techniques.

4.3.1 Noise-free cosmological signal

The top part of Table 4 displays the rms relative errors, computed for each parameters, for the noise-free signals from the test set described in Section 3.1.1 using neural network and global and local ridge kernel regression optimized on the learning set or the test set. For now, if we optimize the hyper-parameters with only the information from the learning set, the best method is the neural network. The local ridge kernel regression accuracy worsen by a factor of $\sim$10, implying that it is sensitive to the value of the hyper-parameters, and that the optimal value are changing fast when moving away from the grid-point used in the regression.

4.3.2 Noised signal

We also present in the bottom of Table 4 the rms relative errors χ, computed for each parameters, for a noised signal using neural network and global and local ridge kernel regression optimized on either the learning set or the test set . The results are shown for signals of the test set noised with a perpendicularized noise. When not optimizing on the test set , the global ridge kernel regression is our most accurate way to reconstruct the astrophysical parameters with a prediction accuracy of a few per cent or roughly half of SKA thermal noise. This method is barely affected by not using information from the test set , which is understandable since the optimization of the hyper-parameter is global. This simply states that the test set and learning set have comparable properties in this respect. We observe a decrease of the performance of the local ridge kernel regression when optimizing on the learning set similar to that in the case of noise-free signals.

To summarize, the error on the prediction of the parameters caused by the supervised learning methods, although much improved, is not yet quite negligible compared to the SKA uncertainty when considering the method that could be directly applied to a real signal. Algorithms to derive the optimal hyper-parameters to use on a real signal will have to be further improved.

5 CONCLUSION

In this work, we explored new supervised learning methods to constrain the underlying astrophysical parameters of the EoR. For this, we chose to base our reconstruction of the parameters on the power spectrum of the intergalactic 21-cm signal, measured at 12 wavenumbers and each integer redshift from $z = 5$ to $z = 15$. We used 21CMFAST to compute the power spectra, varying three different parameters. We chose to vary $\zeta_{\rm ion}$ which accounts for the ionizing efficiency of high-z galaxies, $R_{\rm mfp}$ which is the mean free path of ionizing photons within the ionized regions and $T_{\rm vir}$ which expressed the minimum virial temperature for haloes to be allowed to form stars.

We used a learning set of 2400 signals produced by Eames et al. (2019). They are generated on a $20 \times 6 \times 20$ grid in the parameter

space ($\zeta_{\rm ion}$; $R_{\rm mfp}$; $T_{\rm vir}$). A test set of 512 signals whose parameters are randomly picked within the bounds of the former set was also generated. To be more realistic, we also analyse the case where a SKA-type thermal noise is added to the signals. It leads us to our first main result:

(i) The signal in the test set which is used to evaluate the prediction accuracy cannot be modified with a generic noise. If this is done, the most likely parameters values associated with the noised signals are unknown: they are not, in the general case, those that were used to produce the noise-free signal. Thus, the accuracy of the prediction cannot be computed. To circumvent this issue, we have to perpendicularize the noise in the signal space relatively to the model-manifold.

We mainly implemented two supervised learning methods for our comparison. We first improved the neural network method by using a better optimization of the learning algorithm and hyper-parameters, and by using a larger learning sample, but we kept the architecture from Shimabukuro & Semelin (2017) which is a fully connected network with one hidden layer. Secondly, we studied another class of supervised learning methods which are different kinds of linear regressions and whose most advanced version is a ridge kernel regression with hyper-parameters optimized locally in the signal space. Comparing the prediction accuracy of those methods, we get the following results:

(i) For a 21-cm signal with no added noise, considering only methods which does not use information on the true value of the parameters to be predicted to optimize its learning process, the best methods is the neural network. We predict the parameters with an error of a few per cent, which is an order of magnitude better than in Shimabukuro & Semelin (2017). On the other hand, if we focus on the theoretical maximum accuracy, the best method is the local ridge kernel regression whose hyper-parameters are optimized directly using information from the test set. This information would of course not be available in the case of an observed signal. We find that, when the hyper-parameters are perfectly optimized, this method leads to a prediction rms relative error below 1 per cent, for a result 50 times better than in Shimabukuro & Semelin (2017).

(ii) When considering 21-cm signal with an added SKA thermal noise, the most accurate operational method is the ridge kernel regression globally optimized on the learning set with a prediction rms relative error of a few per cent which is approximately half the amplitude of SKA thermal noise such as predicted in Greig & Mesinger (2015). Again, from all methods the one with the theoretical maximum accuracy is the ridge kernel regression locally optimized which reconstruct the astrophysical parameters with an accuracy of the order of 1 per cent which is 10 times lower than the predicted SKA noise amplitude, meaning that, once optimized to its maximum, this methods will recover the maximum-likelihood astrophysical parameters with near negligible error due to the supervised learning method.

As explained in Section 3.1.2, our results are mitigated by the quality of our optimization of the hyper-parameters which we cannot prove to be a global optimization. Also, we optimize the performance of a neural network with only one hidden layer and do not explore the wide possibility of deep learning architecture with several hidden layer, which can very likely further improve the accuracy of the predictions.

ACKNOWLEDGEMENTS

This work was made thanks to the French ANR funded project ORAGE (ANR-14-CE33-0016). This work was performed using HPC resources from GENCI-CINES (grant no. 2018-A0050410557). The authors also want to acknowledge F. Bolgar and S. Mallat for their useful comments.

REFERENCES

Ali Z. S. et al., 2018, ApJ, 863, 201
Baek S., Di Matteo P., Semelin B., Combes F., Revaz Y., 2009, A&A, 495, 389
Barkana R., Loeb A., 2005, ApJ, 624, L65
Beardsley A. P. et al., 2016, ApJ, 833, 102
Bowman J. D., Rogers A. E. E., Monsalve R. A., Mozdzen T. J., Mahesh N., 2018, Nature, 555, 67
Burger H. C., Schuler C. J., Harmeling S., 2012, IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Providence, Rhode Island. p. 2392
Calvetti D., Morigi S., Reichel L., Sgallari F., 2000, J. Comput. Appl. Math., 123, 423
Cybenko G., 1989, Math. Control Signals Syst., 2, 303
Dewdney P. E., 2013, SKA1 System Baseline Design, http://www.skatelescope.org/wp-content/uploads/2012/07/SKA-TEL-SKO-DD-001-1_BaselineDesign1.pdf (accessed 2019 September 11)
Di Matteo T., Ciardi B., Miniati F., 2004, MNRAS, 355, 1053
Eames E., Doussot A., Semelin B., 2019, MNRAS, 489, 3655
Fialkov A., Barkana R., Visbal E., 2014, Nature, 506, 197
Furlanetto S. R., Zaldarriaga M., Hernquist L., 2004, ApJ, 613, 16
Furlanetto S. R., Peng Oh S., Briggs F. H., 2006, Phys. Rep., 433, 181
Ghara R., Choudhury T. R., Datta K. K., 2015, MNRAS, 447, 1806
Gillet N., Mesinger A., Greig B., Liu A., Ucci G., 2019, MNRAS, 484, 282
Gnedin N. Y., Shaver P. A., 2004, ApJ, 608, 611
Greig B., Mesinger A., 2015, MNRAS, 449, 4246
Greig B., Mesinger A., 2017, MNRAS, 472, 2651
Greig B., Mesinger A., 2018, MNRAS, 477, 3217
Hanke M., Groetsch C. W., 1998, J. Optim. Theory Appl., 98, 37
Hastie T., Tibshirani R., Friedman J., 2009, The Elements of Statistical Learning. Springer Series in Statistics Vol. 99. Springer New York, New York, NY
Hornik K., Stinchcombe M., White H., 1989, Neural Netw., 2, 359
Jelić V. et al., 2008, MNRAS, 389, 1319
Jennings W. D., Watkinson C. A., Abdalla F. B., McEwen J. D., 2019, MNRAS, 483, 2907
Kern N. S., Liu A., Parsons A. R., Mesinger A., Greig B., 2017, ApJ, 848, 23
Koopmans L. et al., 2015, The Cosmic Dawn and Epoch of Reionisation with SKA, Advancing Astrophysics with the Square Kilometre Array , SISSA, Trieste. PoS(AASKA14)001
Madau P., Meiksin A., Rees M. J., 1997, ApJ, 475, 429
McQuinn M., Zahn O., Zaldarriaga M., Hernquist L., Furlanetto S. R., 2006, ApJ, 653, 815
McQuinn M., Lidz A., Zahn O., Dutta S., Hernquist L., Zaldarriaga M., 2007, MNRAS, 377, 1043
Mellema G., Iliev I. T., Alvarez M. A., Shapiro P. R., 2006, NewA, 11, 374
Mellema G. et al., 2013, Exp. Astron., 36, 235
Mesinger A., Furlanetto S., 2007, ApJ, 669, 663
Mesinger A., Furlanetto S., Cen R., 2011, MNRAS, 411, 955
Murugan P., Durairaj S., 2017, p. 1
Paciga G. et al., 2013, MNRAS, 433, 639
Park J., Mesinger A., Greig B., Gillet N., 2019, MNRAS, 484, 933
Patil A. H. et al., 2017, ApJ, 838, 65
Pober J. C. et al., 2014, ApJ, 782, 66
Pritchard J. R., Furlanetto S. R., 2007, MNRAS, 376, 1680

Pritchard J. R., Loeb A., 2012, Rep. Progre. Phys., 75, 086901
Santos M. G., Ferramacho L., Silva M. B., Amblard A., Cooray A., 2010, MNRAS, 406, 2421
Schmit C. J., Pritchard J. R., 2018, MNRAS, 475, 1213
Shaw A. K., Bharadwaj S., Mondal R., 2019, MNRAS, 12, 1,
Shimabukuro H., Semelin B., 2017, MNRAS, 468, 3869
Thomas Rajat M. et al., 2009, MNRAS, 393, 32
Valdés M., Ciardi B., Ferrara A., Johnston-Hollitt M., Röttgering H., 2006, MNRAS, 369, L66
Zel'dovich Y., 1970, A&A, 5, 84

APPENDIX: PERPENDICULARIZED NOISE GENERATION AT GENERIC PARAMETER SPACE LOCATION

To generate a noise perpendicular to the model-manifold at a location that is not on our initial sampling grid, we will still use the set defined on the grid as a way to obtain the local basis generating the hyper-plane tangent to the model-manifold. For a signal $\boldsymbol{P}_{\zeta^x_{\rm ion}, R^y_{\rm mfp}, T^z_{\rm vir}}$, our algorithm is as follows:

(i) Identify the indexes i, j, and k in the grid-generated set such that $\zeta^i_{\rm ion} \leq \zeta^x_{\rm ion} \leq \zeta^{i+1}_{\rm ion}$, $R^j_{\rm mfp} \leq R^y_{\rm mfp} \leq R^{j+1}_{\rm mfp}$, and $T^k_{\rm vir} \leq T^z_{\rm vir} \leq T^{k+1}_{\rm vir}$, which determine the eight signals from the grid-generated set that form the corners of the cell containing the considered signal $\boldsymbol{P}_{\zeta^x_{\rm ion}, R^y_{\rm mfp}, T^z_{\rm vir}}$.

(ii) Using the algorithm for a grid-based set, compute the eight basis $\left(\boldsymbol{e}_{1,\alpha,\beta,\gamma}; \boldsymbol{e}_{2,\alpha,\beta,\gamma}; \boldsymbol{e}_{3,\alpha,\beta,\gamma}\right)$ with $\alpha = i$ or $i+1$, $\beta = j$ or $j+1$, and $\gamma = k$ or $k+1$, corresponding to these corner points.

(iii) Compute the distances $D_{x,y,z}(i,j,k)$ between the considered signal and each of the eight previous points, based on the same definition of the scalar product in signal space.

(iv) Compute a local basis at the considered signal $\boldsymbol{P}_{\zeta^x_{\rm ion}, R^y_{\rm mfp}, T^z_{\rm vir}}$ by making a weighted sum of the eight bases:

$$\boldsymbol{V}_{1,x,y,z} = \sum_{\alpha=i}^{i+1}\sum_{\beta=j}^{j+1}\sum_{\gamma=k}^{k+1} W(\alpha,\beta,\gamma)\boldsymbol{e}_{1,\alpha,\beta,\gamma}, \tag{A1}$$

$$\boldsymbol{V}_{2,x,y,z} = \sum_{\alpha=i}^{i+1}\sum_{\beta=j}^{j+1}\sum_{\gamma=k}^{k+1} W(\alpha,\beta,\gamma)\boldsymbol{e}_{2,\alpha,\beta,\gamma}, \tag{A2}$$

$$\boldsymbol{V}_{3,x,y,z} = \sum_{\alpha=i}^{i+1}\sum_{\beta=j}^{j+1}\sum_{\gamma=k}^{k+1} W(\alpha,\beta,\gamma)\boldsymbol{e}_{3,\alpha,\beta,\gamma}, \tag{A3}$$

where

$$W(\alpha,\beta,\gamma) = \frac{\left[D_{x,y,z}(\alpha,\beta,\gamma)\right]^{-1}}{\sum_{\alpha'=i}^{i+1}\sum_{\beta'=j}^{j+1}\sum_{\gamma'=k}^{k+1}\left[D_{x,y,z}(\alpha',\beta',\gamma')\right]^{-1}}.$$

(v) Orthonormalize the previous basis to obtain an orthonormalized basis whose elements will be referred as $\boldsymbol{e}_{1,x,y,z}$, $\boldsymbol{e}_{2,x,y,z}$, and $\boldsymbol{e}_{3,x,y,z}$.

(vi) Generate a Noise $\boldsymbol{N}$ and compute $\boldsymbol{N}_\perp$ using equation (7).

This paper has been typeset from a TEX/LATEX file prepared by the author.