Research Article

Marwan Gebran*, Ian Bentley, Rose Brienza, and Frédéric Paletou

Deep learning application for stellar parameter determination: III-denoising procedure

https://doi.org/10.1515/astro-2024-0010
received September 20, 2024; accepted November 13, 2024

Abstract: In this third article in a series, we investigate the need of spectra denoising for the derivation of stellar parameters. We have used two distinct datasets for this work. The first one contains spectra in the range of 4,450–5,400 Å at a resolution of 42,000, and the second in the range of 8,400–8,800 Å at a resolution of 11,500. We constructed two denoising techniques, an autoencoder, and a principal component analysis. Using random Gaussian noise added to synthetic spectra, we have trained a neural network to derive the stellar parameters $T_{\rm eff}$, $\log g$, $v_{\rm e} \sin i$, $\xi_{\rm t}$, and [M/H] of the denoised spectra. We find that, independently of the denoising technique, the accuracy values of stellar parameters do not improve once we denoise the synthetic spectra. This is true with and without applying data augmentation to the stellar parameters neural network.

Keywords: data analysis, statistical, deep learning, autoencoders, techniques: spectroscopic, noise, stars: fundamental parameters

1 Introduction

Observations in astronomy have always been associated with noise. Trying to minimize the noise is one of the needs of astronomers. Several observation techniques have been suggested to reduce the noise in spectra; however, once the observation is performed, the only way to proceed is to apply mathematical algorithms that can improve the signal-to-noise ratio (SNR) of the data. These techniques involve but are not limited to Gaussian smoothing (Chung 2020), median filtering (Kumar and Sodhi 2020), wavelet denoising (Halidou *et al*., 2023), and principal component analysis (PCA) (Bacchelli and Papi 2006, Zhang *et al*., 2010, Murali *et al*., 2012, Li 2018). More recently, and with the advancement of computational power, deep learning algorithms started to be used for that purpose. Gheller and Vazza (2022) used a convolutional denoising autoencoder to decrease the noise of synthetic images of state-of-the-art radio telescopes like LOFAR (Offringa *et al*., 2013), MeerKAT (Jonas 2009), and MWA (Tingay *et al*., 2013). The technique was applied to different kinds of corrupted input images. The autoencoder was able to effectively denoise images identifying and extracting faint objects at the limits of the instrumental sensitivity. The authors state that their autoencoder was capable of removing noise while preserving the properties of the regions of the sources with SNR as low as 1. Scourfield *et al*., (2023) used a variational autoencoder to denoise optical SDSS spectra of galaxies (York *et al*., 2000). Their main goal was to denoise the spectra while keeping the important information they can retrieve from low SNR galaxy spectra and avoiding the use of sample averaging methods (smoothing or spectral stacking). They tested the method in the context of large optical spectroscopy surveys by simulating a population of spectra with noise to mimic the ones at galaxies at a redshift of $z = 0.1$. Their results showed that the technique can recover the shape and scatter of the mass-metallicity relation in this sample.

In this work, we introduce two types of spectral denoising techniques, autoencoders (Ballard 1987, Baldi 2011) and PCA (Wold *et al*., 1987, Makiewicz and Ratajczak 1993). We test the need of the denoising technique on the derived stellar parameters: effective temperature $T_{\rm eff}$, surface gravity $\log g$, equatorial projected rotational velocity $v_{\rm e} \sin i$, microturbulence velocity $\xi_{\rm t}$, and the overall metallicity [M/H]. These stellar parameters are derived using the neural network introduced our previous work (Gebran *et al*., 2022, Gebran *et al*., 2023, Gebran 2024). The article is organized as follows: Section 2 introduces the calculation of both datasets and noisy spectra, Section 3 explains the autoencoder construction used in the

*** Corresponding author: Marwan Gebran**, Department of Chemistry and Physics, Saint Mary's College, Notre Dame, IN 46556, United States, e-mail: mgebrane@saintmarys.edu
Ian Bentley: Department of Physics, FLorida Polytechnic University, Lakeland, FL, 33805, United States
Rose Brienza: Department of Chemistry and Physics, Saint Mary's College, Notre Dame, IN 46556, United States
Frédéric Paletou: Universite de Toulouse, Observatoire Midi-Pyrenes, Irap, Cnrs, Cnes, 14 av. E. Belin, F-31400 Toulouse, France

denoising procedure, and Section 4 describes the denoising technique using PCA. Section 5 shows the results of the denoising technique using both procedures and the effect on the derived stellar parameter accuracy values. Finally, we conclude in Section 6.

2 Datasets

Two datasets were used in the context of the present study. The one analyzed in Gebran *et al.*, (2023), and the one of Gebran (2024). The characteristics of these two datasets are described in Table 1. The reason for selecting these diverse datasets is to check the procedure over different wavelength ranges and different resolving power.

The steps of calculating the datasets are detailed in Gebran *et al.*, (2022), Gebran *et al.*, (2023), and Gebran (2024). In summary, line-blanketed model atmospheres are calculated using ATLAS9 (Kurucz 1992). The models are plane parallel and in local thermodynamic equilibrium (LTE). They are in hydrostatic and radiative equilibrium. We have calculated the models using the opacity distribution function (ODF) of Castelli and Kurucz (2003). Convection was included according to Smalley's (2004) prescriptions. Convection is included in the atmospheres of stars cooler than 8,500 K using the mixing length theory. A mixing length parameter of 0.5 was used for 7,000 K ≤ $T_{\rm eff}$ ≤ 8,500 K, and 1.25 for $T_{\rm eff}$ ≤ 7,000 K.

Table 1: Range of parameters used in the calculation of the synthetic spectra for the two datasets

Parameter	Range for DB1	Range for DB2
$T_{\rm eff}$	3,600–15,000 K	
$\log g$	2.0–5.0 dex	
$v_{\rm e}\sin i$	0–300 km/s	
[M/H]	–1.5–1.5 dex	
$\xi_{\rm t}$	0–4 km/s	
Wavelength λ	4,450–5,400 Å	8,400–8,800 Å
Sampling in λ	0.05 Å	0.10 Å
Resolution ($\frac{\lambda}{\Delta\lambda}$)	42,000	11,500

We have used the radiative transfer code SYNSPEC (Hubeny and Lanz 2017) to calculate the synthetic spectra. As mentioned previously, two datasets were calculated with each one containing around 200,000 spectra. In both datasets, metal abundances were scaled with respect to the Grevesse and Sauval (1998) solar value from −1.5 dex up to +1.5 dex. The effective temperature, surface gravity, projected equatorial velocity, and microturbulence velocity were also modified according to the values displayed in Table 1. The first dataset consists of spectra having a resolution of 42,000 and a wavelength range between 4,450 and 5,400 Å. As explained in Gebran *et al.*, (2022, 2023), this wavelength range is sensitive to all stellar parameters in the spectral range of AFGK stars. The second dataset has spectra computed between 8,400 and 8,800Å at a resolution of 11,500. This region includes the Gaia radial velocity spectrometer (RVS, Cropper *et al.*, 2018). The RVS spectra contain lines sensitive to the stellar parameters and to the chemical abundance of many metals (Mg, Si, Ca, Ti, Cr, Fe, Ni, and Zr, among others) at different ionization stages. The linelist used in this work is the one used in Gebran *et al.*, (2022) and Gebran *et al.*, (2023). It contains updated values for the atomic parameters such as the wavelength of the transitions, the oscillator strengths, the damping constants, and others.

In summary, we ended up with two datasets of around 200,000 synthetic spectra each, with $T_{\rm eff}$, $\log g$, $v_{\rm e}\sin i$, [M/H], and $\xi_{\rm t}$ randomly chosen from Table 1. Figure 1 shows a

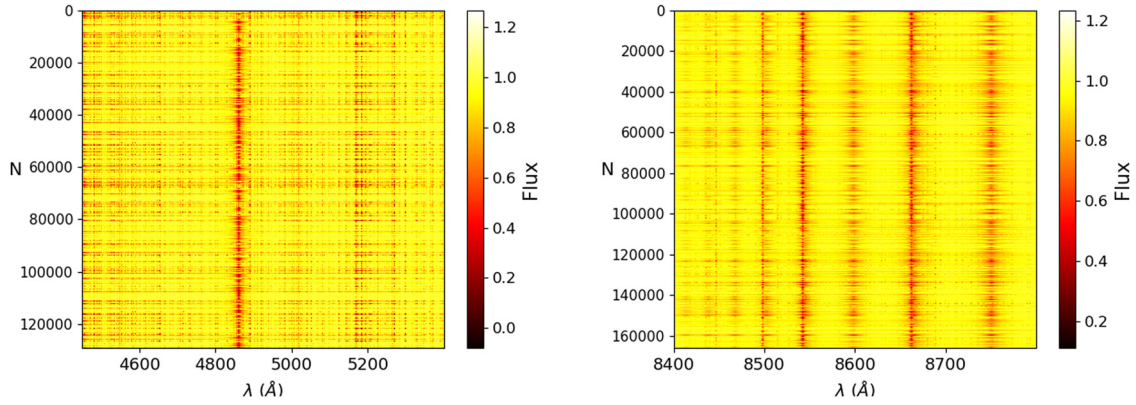

Figure 1: Color map representing the fluxes for a sample spectra of the two training datasets. The left color map represents dataset 1 and the right one represents dataset 2. The y-label represents the number of spectra, N.

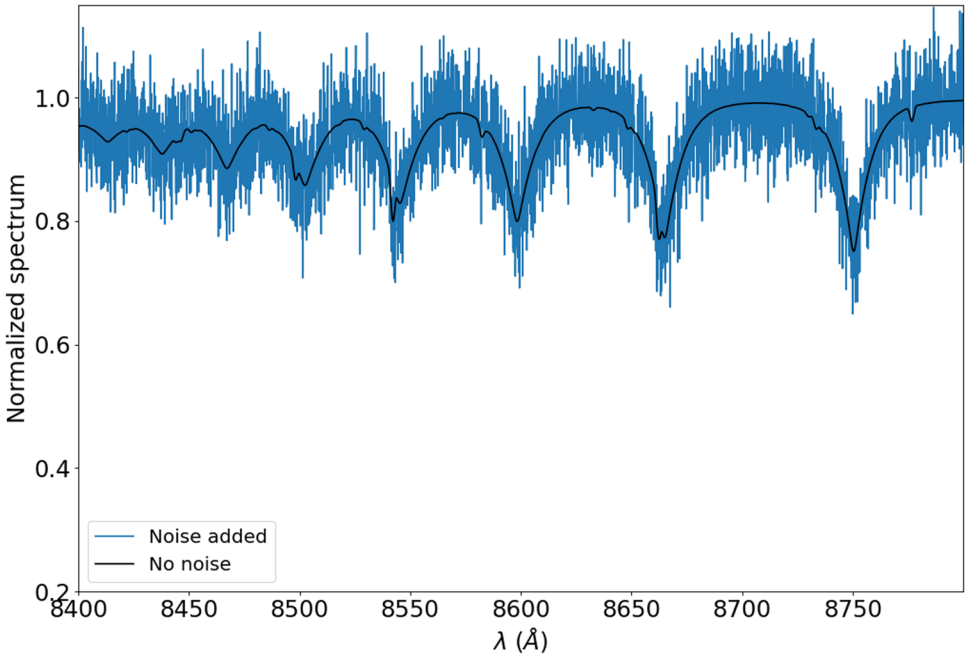

Figure 2: Example of a spectrum of dataset 2 calculated using a random selection of stellar parameters from Table 1. The black spectrum represents the synthetic spectrum calculated without noise, and the blue one corresponds to the same parameters but with an SNR of 19. The stellar parameters of the spectrum are 14,550 K, 3.05 dex, 44 km s^{-1}, –1.15 dex, and 3 km s^{-1} for $T_{\rm eff}$, $\log g$, $v_{\rm e} \sin i$, [M/H], and $\xi_{\rm t}$, respectively.

color map of a sub-sample of the datasets. The Balmer line is detected in the left color map for dataset 1, and the absorption lines of the calcium triplet (λ = 8,498, 8,542, 8,662 Å) are also shown in the color map of dataset 2 in the bottom part of the figure.

For each dataset, a set of spectra were calculated with random Gaussian noise between 5 and 300. This SNR is used to mimic the noisy observations that we will be denoising later on as they represent the average SNR encountered in real stellar spectra. An example of a spectrum calculated with and without noise in the parameter range of dataset 2 is shown in Figure 2.

2.1 Data augmentation

We have also tested the effect of data augmentation in this work, and for that reason, we have calculated extra dataset as suggested in Gebran *et al.*, (2022). Data augmentation is a regularization technique that increasing the diversity of the training data by applying different transformations to the existing one, helps in avoiding over-fitting and improves the predictions of stellar labels when applied with real observed data (Gebran *et al.*, 2023). We have used the same approach of Gebran *et al.*, (2022) in which five replicas of each spectrum in the dataset were performed. These replicas consist of

- Adding to each spectrum a Gaussian noise with an SNR ranging randomly between 5 and 300.
- The flux of each spectrum is multiplied with a scaling factor selected randomly between 0.95 and 1.05.
- The flux of each spectrum is multiplied with a new random scaling factor and noise was added.
- The flux of each spectrum is multiplied by a second-degree polynomial with values ranging between 0.95 and 1.05 and having its maximum randomly selected in the wavelength range of the dataset.
- The flux of each spectrum is multiplied by a second-degree polynomial and Gaussian noise added to it.

For more details about data augmentation, we refer the reader to Gebran *et al.*, (2022).

3 Auto-encoders

Autoencoders, usually used in denoising and dimensionality reduction techniques (Lecun 1987, Fogelman Soulie *et al.*, 1987, Ballard 1987, Baldi 2011, Schmidhuber 2014, Einig *et al.*, 2023, Scourfield *et al.*, 2023), are a type of neural networks that work in an unsupervised way. They consist of two distinct yet similar algorithms, an encoder and a decoder. The encoder's role is to transform the spectra

from a dimension of N_λ flux point to a smaller size of N_{latent} inside a latent space. The decoder re-transform the N_{latent} to the original spectrum of N_λ flux point. The choice of N_{latent} depends on the characteristics of the dataset. However, using the two datasets in this work, we found that the optimal size for the latent space is $N_{\text{latent}} = 10$. This is found by minimizing the difference between the output spectra and the input one during the training process. It is true that different values of N_{latent} could be used, but our choice of N_{latent} was based on the smallest value that gives a reconstruction error less than 0.5% as will be explained in the next steps.

The classical architecture of an autoencoder is shown in Figure 3 where the initial spectrum is introduced having 19,000 or 4,000 data points depending on the dataset and is then reduced to N_{latent} points through successive hidden layers. This first step defines the encoder part of the autoencoder. Then, the N_{latent} points are transformed to 19,000 or 4,000 data points while passing through different hidden layers. This second step defines the Decoder part of the autoencoder. The hidden layers are usually symmetrical in the encoder and decoder parts.

Two autoencoders were used in this work, one for each dataset. In both cases, the spectra are reduced to ten parameters in the latent space. The architecture of the used autoencoders is displayed in Table 2. We have used an Adam optimizer with a mean squared error (MSE) loss function.

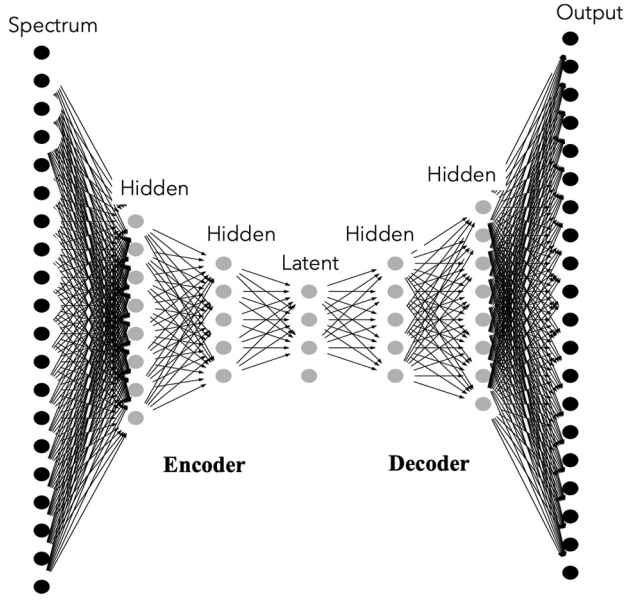

Figure 3: Sketch of an autoencoder that transforms an input spectrum of N_λ data point to a lower dimension using a series of hidden layers (encoder). The middle layer is the latent space. The decoder reconstructs the spectrum to its original dimension.

Table 2: Architecture of the autoencoder used for denoising

Layer	Characteristics
Encoder	
Input	Spectrum of N_λ data points
Hidden	1,024 Neurons
Hidden	512 Neurons
Hidden	256 Neurons
Hidden	64 Neurons
Hidden	32 Neurons
Latent space	10 Neurons
Decoder	
Hidden	32 Neurons
Hidden	64 Neurons
Hidden	256 Neurons
Hidden	512 Neurons
Hidden	1,024 Neurons
Output	Reconstructed spectrum of N_λ data points

Calculations were performed using `TensorFlow`[1] with the `Keras`[2] interface and were written in `Python`.

The training of the autoencoders was performed using the two datasets containing the synthetic spectra with no noise. The convergence is achieved when the difference between the output and the input spectra is minimized through the MSE. Convergence usually occurs after around 500 epochs. For both datasets, we achieved an R^2 score larger than 0.995, meaning that the reconstruction of the spectra is performed with an error <0.5%. Once the training is done, the denoising is performed when the trained autoencoders are applied to the noisy spectra.

4 PCA

PCA is a non-parametric mathematical transformation that extracts relevant information from a dataset (Wold *et al*., 1987, Makiewicz and Ratajczak 1993). Its goal is to compute the most meaningful basis to represent a noisy dataset. The new basis usually reveals hidden structure and filters out the noise (Shlens 2014). PCA has been used for denoising (Bacchelli and Papi 2006, Zhang *et al*., 2010, Murali *et al*., 2012, Li 2018) or spectral dimension reduction (Makiewicz and Ratajczak 1993, Paletou *et al*., 2015a, Gebran *et al*., 2016, Gebran *et al*., 2022, Gebran *et al*., 2023). The main power of PCA is that it can reduce the dimension of the data while maintaining significant patterns and trends.

1 https://www.tensorflow.org/
2 https://keras.io/

The basic idea behind the use of PCA is to derive a small number of eigenvectors and use them to recover the information in the spectra. The steps of PCA calculation are

1. The matrix containing the training dataset has N_λ flux points per spectrum; therefore, the dataset can then be represented by a matrix $\boldsymbol{M}$ of size $N_{\rm spectra} \times N_\lambda$, where $N_{\rm spectra}$ represents the number of spectra in the dataset.
2. The matrix $\boldsymbol{M}$ is then averaged along the $N_{\rm spectra}$-axis and this average is stored in a vector $\bar{M}$.
3. The variance–covariance matrix $\boldsymbol{C}$ is calculated as
$$\boldsymbol{C} = (\boldsymbol{M} - \bar{M})^{\rm T} \cdot (\boldsymbol{M} - \bar{M}), \quad (1)$$
where the superscript "T" stands for the transpose operator.
4. The eigenvectors $\mathbf{e}_k(\lambda)$ of $\boldsymbol{C}$ are then calculated. $\boldsymbol{C}$ has a dimension of $N_\lambda \times N_\lambda$. The principal components (PC) correspond to the eigenvectors sorted in decreasing magnitude.
5. Each spectrum of $\boldsymbol{M}$ is then projected on these PCs in order to find its corresponding coefficient p_{jk} defined as
$$p_{jk} = (M_j - \bar{M}) \cdot \boldsymbol{e}_k. \quad (2)$$
6. The original "denoised spectrum" can be calculated using
$$S_j = \bar{M} + \sum_{k=1}^{n_k} p_{jk} \boldsymbol{e}_k. \quad (3)$$

The PCA can reduce the size of each spectrum from N_λ to n_k. The choice of n_k depends on the many parameters, the size of the dataset, the wavelength range, and the shape of the spectra lines. We have opted for a value for n_k that reduces the mean reconstructed error to a value <0.5% according to the following equation:

$$E(k_{\rm max}) = \left\langle \left(\frac{|\bar{M} + \Sigma_{k=1}^{n_k} p_{jk} \boldsymbol{e}_k - M_j|}{M_j} \right) \right\rangle. \quad (4)$$

We have opted to a value for n_k that reduces the mean reconstructed error to a value <0.5%. This value is found to be $n_k = 50$. A detailed description of all steps of the PCA can be found in Paletou *et al.*, (2015a), Paletou *et al.*, (2015b), Gebran *et al.*, (2016, 2022, 2023), and Gebran (2024). For both datasets, we achieved an R^2 score larger than 0.996.

5 Denoising and parameter determination

The datasets that contain the synthetic spectra without any added noise are used to train the autoencoder and to find the eigenvectors of the PCA procedure. These two techniques are then used on the set of noisy spectra that are calculated in Section 2. The evaluation of the denoising procedure is tested in two ways. First, we checked the similarity of the denoised spectra with the original one with no noise added. Second, we checked the accuracy of the derived stellar parameters when we applied the procedures of Gebran *et al.*, (2022, 2023) on the denoised spectra from the autoencoder and PCA.

Autoencoders usually replace PCA because of their non-linear properties; however, both techniques showed a good reconstruction power, as shown by the R^2 score in Sections 3 and 4. A way to visualize the denoising of spectra is shown in Figure 4. The figure is divided into two parts, the upper one displays a spectrum having the parameters of dataset 1, and the bottom one has the parameters of dataset 2. In each part, the noisy spectrum is in black, the original one without noise is in dashed blue, the denoised spectrum using the autoencoder (left panel) or PCA (right panel) technique is in red, and the difference between the denoised spectrum and the original one without noise is in dashed–dotted green.

In Gebran *et al.*, (2022, 2023), we have introduced a technique to derive the stellar parameters of spectra using a neural network. We have used the same procedure to derive the accuracy of the stellar parameters once we apply the same technique to the denoised spectra. The main purpose of this step is not to evaluate if the derivation technique is accurate or not, but it is to check how similar are the derived stellar parameters of the noisy spectra to the ones derived from the original spectra with no noise added.

The networks that we used are made of several fully dense layers and are trained to derive each parameter separately. The layers are described in Table 3. The first step of the analysis is to reduce the dimension of the spectra using a PCA procedure. This PCA is not related to the one used for denoising, it is just a step for optimizing the network and making the training faster (see Gebran *et al.*, 2022 for more details).

Two different trainings are performed for each dataset. The first one is done using a dataset of only synthetic spectra with no noise added, and the second one consists of applying data augmentation with spectra having a range of SNR between 3 and 300.

Because we already know the stellar parameters of the spectra, the evaluation is performed by calculating the difference between the predicted parameter and the original one using the equation

$$\text{Accuracy} = \frac{1}{N} \sqrt{\sum_{i=1}^{N} (\text{Predicted} - \text{Original})^2}, \quad (5)$$

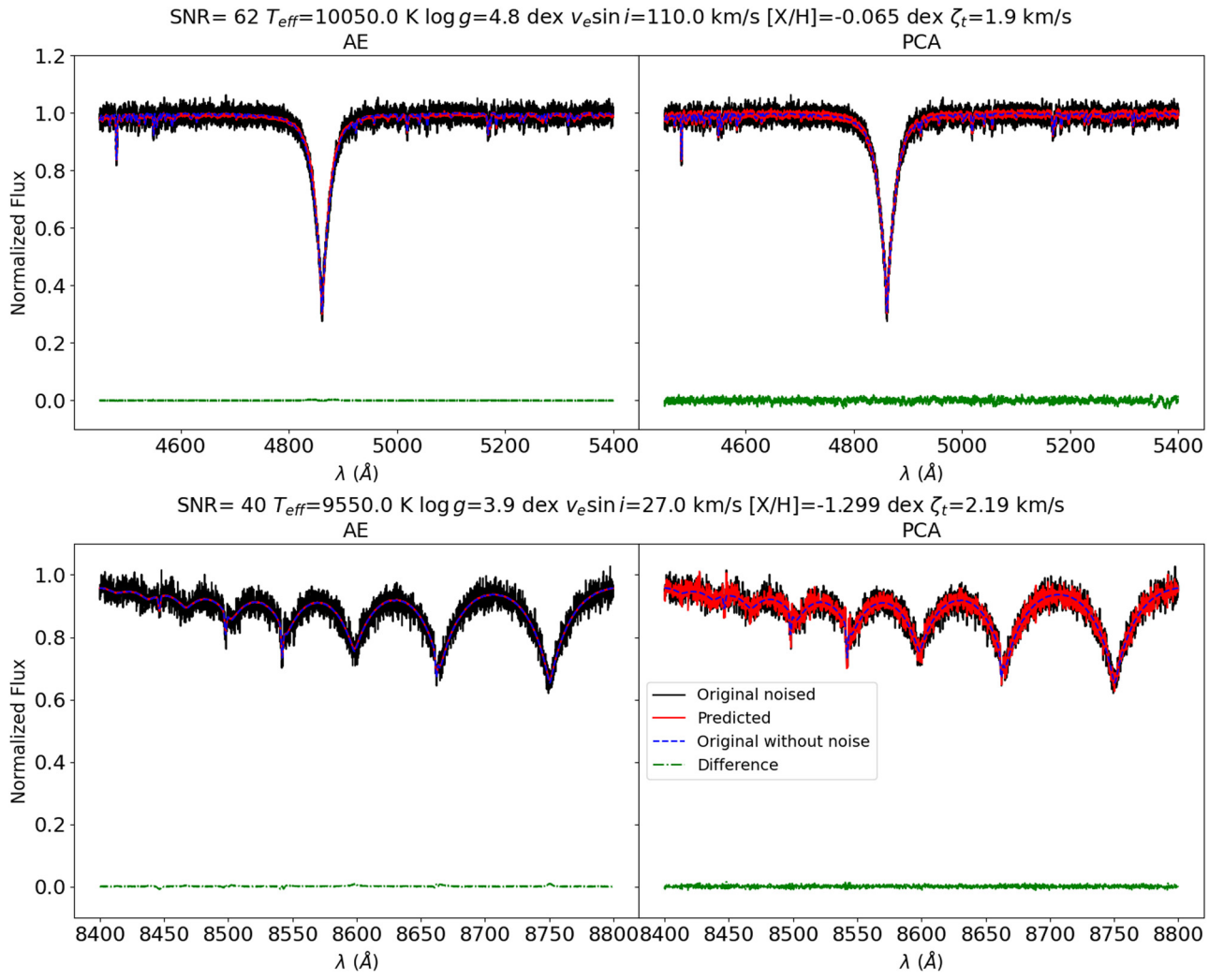

Figure 4: Denoising example of spectra having stellar parameters in both datasets. The upper plot displays a spectrum having the parameters of dataset 1, and the bottom one has the parameters of dataset 2. The noisy spectrum is in black, the original one without noise is in dashed blue, the denoised spectrum using the autoencoder (left panel) or PCA (right panel) technique is in red, and the difference between the denoised spectrum and the original one with no noise is in dashed-dotted green.

Table 3: Architecture of the neural network used for stellar parameters determination

Layer	Characteristics
Input	Spectrum of N_λ data points
PCA	Reduction to 50 data points
Hidden	4,096 Neurons
Hidden	2,048 Neurons
Hidden	1,024 Neurons
Hidden	512 Neurons
Hidden	60 Neurons
Output	1 Parameter

where N is the total number of noisy spectra used in the evaluation. This is done for $T_{\rm eff}$, log$g$, $v_{\rm e}$ sin$i$, $\xi_{\rm t}$, and [M/H]. Tables 4 and 5 display the accuracy values for the parameters for the two datasets when deriving the stellar labels of ~25,000 with no noise added (Col. 2), with random noise (Col. 3), with random noise then denoised using autoencoder of Section 3 (Col. 4) and using PCA of Section 4 (Col. 5). Each table is divided into two, one part when data augmentation is performed and one without it.

A detailed analysis of Tables 4 and 5 show that:

– Data augmentation is an important step to be applied if we need to derive the stellar parameters of noisy spectra. Without it, the model will only learn to derive the parameters of synthetic spectra without any noise added. A similar conclusion was also found in Gebran *et al.*, (2023).
– PCA denoising is capable of recovering the line profile and the details in the spectra. This is reflected by

Table 4: Accuracy values on the derived stellar parameters for the spectra calculated using the parameters of dataset 1

Parameters	No noise	With noise	Denoised (AE)	Denoised (PCA)
Augmented dataset				
$T_{\rm eff}$ (K)	52	181	240	190
log$g$ (dex)	0.017	0.089	0.160	0.100
$v_{\rm e} \sin i$ (km s^{-1})	1.80	7.58	9.74	7.60
$\xi_{\rm t}$ (km s^{-1})	0.09	0.22	0.32	0.23
[M/H] (dex)	0.021	0.071	0.103	0.071
No data augmentation				
$T_{\rm eff}$ (K)	47	219	243	222
log$g$ (dex)	0.018	0.121	0.168	0.121
$v_{\rm e} \sin i$ (km s^{-1})	1.69	9.49	10.10	9.50
$\xi_{\rm t}$ (km s^{-1})	0.11	0.40	0.40	0.48
[M/H] (dex)	0.019	0.097	0.106	0.098

The spectra are calculated with no noise added (Col. 2), with random Gaussian noise (Col. 3), with random noise and then denoised using the autoencoder network (Col. 4), and denoised using PCA (Col. 5).

Table 5: Accuracy values on the derived stellar parameters for the spectra calculated using the parameters of dataset 2

Parameters	No noise	With noise	Denoised (AE)	Denoised (PCA)
Augmented dataset				
$T_{\rm eff}$ (K)	116	300	303	345
log$g$ (dex)	0.037	0.145	0.174	0.176
$v_{\rm e} \sin i$ (km s^{-1})	6.61	13.40	16.16	13.87
$\xi_{\rm t}$ (km s^{-1})	0.16	0.63	0.79	0.63
[M/H] (dex)	0.038	0.170	0.204	0.172
No data augmentation				
$T_{\rm eff}$ (K)	124	360	576	577
log$g$(dex)	0.031	0.188	0.310	0.310
$v_{\rm e} \sin i$ (km s^{-1})	5.93	17.50	17.60	21.70
$\xi_{\rm t}$ (km s^{-1})	0.09	1.12	1.36	1.36
[M/H] (dex)	0.035	0.262	0.272	0.267

The spectra are calculated with no noise added (Col. 2), with random Gaussian noise (Col. 3), with random noise and then denoised using the autoencoder network (Col. 4), and denoised using PCA (Col. 5).

comparing the accuracy values of the derived parameters using the denoised spectra from the autoencoders and PCA (*i.e.* comparing Cols 4 and 5).

– The parameters derived using the PCA denoising technique are more accurate than the ones derived using the autoencoder denoising.
– No denoising technique is capable of improving the accuracy of the stellar parameters for the one directly derived from noisy spectra (displayed in Col. 3).
– The stellar parameter algorithm is capable of deriving the stellar labels without the need for a denoising technique.

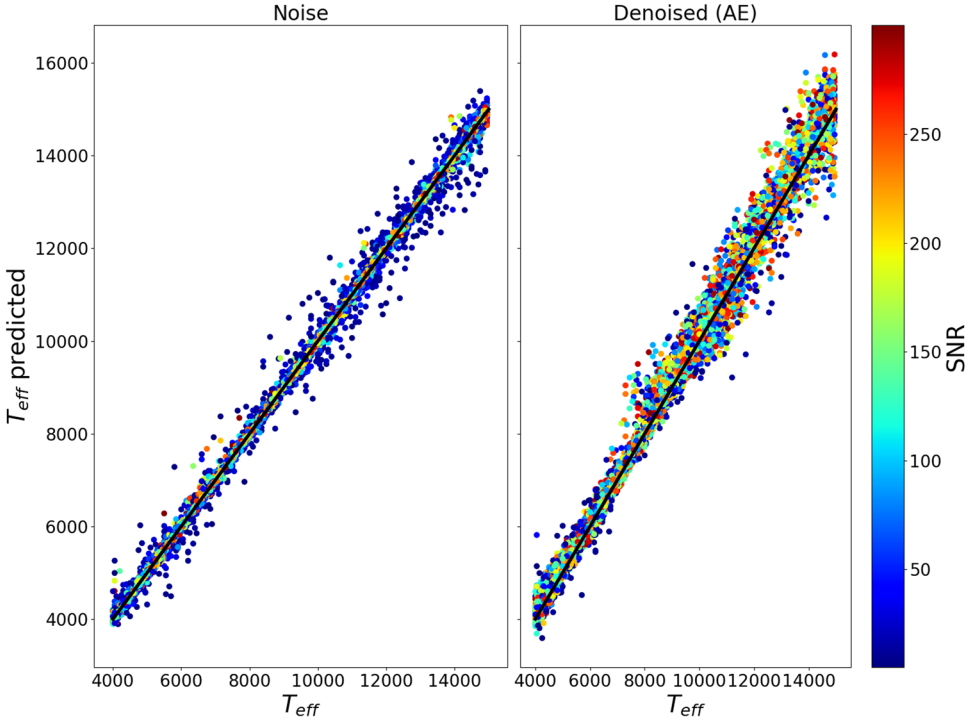

Figure 5: Predicted $T_{\rm eff}$ as a function of the true values for the data with noise (left panel) and the denoised data using autoencoder (right panel). The spectra are from the augmented dataset 2. The data are color coded to the SNR values.

These tests show mainly that data augmentation is very important when neural networks are used to derive the stellar parameters of noisy spectra, a results already found by Gebran *et al.*, (2022, 2023). As an example, Figure 5 displays the predicted $T_{\rm eff}$ with respect to the original one for the data with noise from the augmented dataset 2 (left panel) and the denoised data using autoencoder (right panel) from the same dataset. The data are color coded to the SNR values. The straight black line represents the best prediction line ($x = y$). The left panel shows that the highly dispersed results are the ones for the low SNR spectra. Once the spectra are denoised, the dispersion appears to be present for all SNR values with no specific trend or deviation. This is true for all stellar parameters. Independently of the denoising technique, there is no improvement found in the accuracy values of the derived parameters of denoised spectra when the networks were trained on noisy spectra. Applying the networks to noisy data gives more accurate results than when it is applied to denoised data.

6 Conclusion

In this work, we have applied two different denoising techniques, an autoencoder, and a PCA, on spectra with random Gaussian noise added to derive the stellar parameters using neural networks of Gebran *et al.*, (2022, 2023). The method was applied to two different spectra ranges, one in 4,450–5,400 Å and one in the Gaia RVS range from 8,400 to 8,800 Å. In this study, we do not constrain the stellar parameter derivation technique, and this was done previously in Gebran *et al.*, (2022, 2023). Interestingly, when applying the model to denoised spectra, there was no noticeable improvement in the accuracy of the derived fundamental parameters, such as $T_{\rm eff}$, $\log g$, $v_{\rm e} \sin i$, $\xi_{\rm t}$, and [M/H]. This outcome was unexpected, as denoising is typically thought to enhance the precision of predictions. However, the results indicate that data augmentation plays a more crucial role. When the model is trained on datasets that include noise, the accuracy of predictions for noisy spectra improves significantly, suggesting that the network becomes better equipped to handle real observed spectra. This highlights the importance of incorporating noisy data into training rather than relying on post-processing techniques like denoising to improve accuracy. To further validate these findings, it would be valuable to explore other denoising techniques and assess their impact on prediction accuracy. Techniques such as those presented in Alsberg *et al.*, (1997), Koziol *et al.*, (2018), and Zhao *et al.*, (2021) could be tested to see if they yield better results in reducing noise while maintaining or enhancing the precision of derived parameters. These additional experiments would help solidify the conclusion that data augmentation is more effective than denoising in improving the accuracy of noisy spectra predictions, offering deeper insights into how best to model real observational spectra.

Acknowledgement: The authors acknowledge Saint Mary's College for providing the high-power computing cluster used in this work. The authors are grateful for the reviewer's valuable comments that improved the manuscript.

Funding information: Authors state no funding involved.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and consented to its submission to the journal, reviewed all the results, and approved the final version of the manuscript. MG and RB designed the code and carried out the calculations. MG prepared the manuscript with contributions from all co-authors. IB and FP have contributed to the general assessment of the paper and the writing of parts of sections.

Conflict of interest: Authors state no conflict of interest.

References

Alsberg BK, Woodward AM, Winson MK, Rowland JJ, Kell DB. 1997. Wavelet denoising of infrared spectra. Analyst. 122:645–652.

Bacchelli S, Papi S. 2006. Image denoising using principal component analysis in the wavelet domain. J Comput Appl Math. 189(1):606–621.

Baldi P. 2011. Autoencoders, unsupervised learning, and deep architectures. In: Proceedings of ICML workshop on unsupervised and transfer learning. JMLR Workshop and Conference Proceedings. JMLR.org. p. 37–49.

Ballard DH. 1987. Modular learning in neural networks. In: Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1. AAAI'87. AAAI Press. p. 279–284.

Castelli F, Kurucz RL. 2003. New grids of ATLAS9 model atmospheres. In: Piskunov N, Weiss WW, Gray DF, editors, Modelling of Stellar Atmospheres. Proceedings of the 210th Symposium of the IAU, vol. 210, p. A20.

Chung MK. 2020. Gaussian kernel smoothing. arXiv e-prints, page arXiv:2007.09539.

Cropper M, Katz D, Sartoretti P, Prusti T, de Bruijne JHJ, Chassat F, et al. 2018. Gaia Data Release 2. Gaia Radial Velocity Spectrometer. A & A. 616:A5.

Einig L, Pety J, Roueff A, Vandame P, Chanussot J, Gerin M, et al. 2023. Deep learning denoising by dimension reduction: Application to the ORION-B line cubes. A & A. 677:A158.

Fogelman Soulie F, Gallinari P, Lecun Y, Thiria S. 1987. Automata networks and artificial intelligence. New Jersey, United States: Princeton University Press. p. 133–186.

Gebran M. 2024. Generating stellar spectra using neural networks. Astronomy. 3(1):1–13.

Gebran M, Connick K, Farhat H, Paletou F, Bentley I. 2022. Deep learning application for stellar parameters determination: I-constraining the hyperparameters. Open Astron. 31(1):38–57.

Gebran M, Farah W, Paletou F, Monier R, Watson V. 2016. A new method for the inversion of atmospheric parameters of A/Am stars. A & A. 589:A83.

Gebran M, Paletou F, Bentley I, Brienza R, Connick K. 2023. Deep learning applications for stellar parameter determination: II-application to the observed spectra of AFGK stars. Open Astron. 32(1):209.

Gheller C, Vazza F. 2022. Convolutional deep denoising autoencoders for radio astronomical images. MNRAS. 509(1):990–1009.

Grevesse N, Sauval AJ. 1998. Standard solar composition. SSR. 85:161–174.

Halidou A, Mohamadou Y, Ari AAA, Zacko EJG. 2023. Review of wavelet denoising algorithms. Multimed Tools Appl. 82(27):41539–41569.

Hubeny I, Lanz T. 2017. A brief introductory guide to TLUSTY and SYNSPEC. page arXiv:1706.01859.

Jonas JL. 2009. Meerkat the South African array with composite dishes and wide-band single pixel feeds. Proc IEEE. 97(8):1522–1530.

Koziol P, Raczkowska MK, Skibinska J, Urbaniak-Wasik S, Paluszkiewicz C, Kwiatek W, et al. 2018. Comparison of spectral and spatial denoising techniques in the context of High Definition FT-IR imaging hyperspectral data. Sci Rep. 8:14351.

Kumar A, Sodhi SS. 2020. Comparative analysis of gaussian filter, median filter and denoise autoenocoder. In: 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom). p. 45–51.

Kurucz RL. 1992. Atomic and molecular data for opacity calculations. RMXAA. 23:45.

Lecun Y. 1987. PhD thesis: Modeles connexionnistes de l'apprentissage (connectionist learning models). Universite P. et M. Curie (Paris 6).

Li B. 2018. A principal component analysis approach to noise removal for speech denoising. In: 2018 International Conference on Virtual Reality and Intelligent Systems (ICVRIS). p. 429–432.

Makiewicz A, Ratajczak W. 1993. Principal components analysis (PCA). Comput Geosci. 19(3):303–342.

Murali Y, Babu M, Subramanyam M, Prasad D. 2012. PCA based image denoising. Signal Image Process. 3:236–244.

Offringa AR, deBruyn AG, Zaroubi S, van Diepen G, Martinez-Ruby O, Labropoulos P, et al. 2013. The LOFAR radio environment. A & A. 549:A11.

Paletou F, Böhm T, Watson V, Trouilhet JF. 2015a. Inversion of stellar fundamental parameters from ESPaDOnS and Narval high-resolution spectra. A & A. 573:A67.

Paletou F, Gebran M, Houdebine ER, Watson V. 2015b. Principal component analysis-based inversion of effective temperatures for late-type stars. A & A. 580:A78.

Schmidhuber J. 2014. Deep learning in neural networks: An overview. arXiv e-prints, page arXiv:1404.7828.

Scourfield M, Saintonge A, de Mijolla, D, Viti S. 2023. De-noising of galaxy optical spectra with autoencoders. MNRAS. 526(2):3037–3050.

Shlens J. 2014. A tutorial on principal component analysis. arXiv e-prints, page arXiv:1404.1100.

Smalley B. 2004. Observations of convection in A-type stars. In: Zverko J, Ziznovsky J, Adelman SJ, Weiss WW, editors, The A-Star Puzzle. IAU Symposium 224 "The A Star Puzzle", Poprad, Slovakia, vol. 224, p. 131–138.

Tingay SJ, Goeke R, Bowman JD, Emrich D, Ord SM, Mitchell DA, et al. 2013. The Murchison widefield array: The square kilometre array precursor at low radio frequencies. PASP. 30:e007.

Wold S, Esbensen K, Geladi P. 1987. Principal component analysis. Chemom Intell Lab Syst. 2(1):37–52.

York DG, Adelman J, Anderson JE, Anderson SF, Annis J, Bahcall NA, et al., and SDSS Collaboration 2000. The sloan digital sky survey: Technical summary. AJ. 120(3):1579–1587.

Zhang L, Dong W, Zhang D, Shi G. 2010. Two-stage image denoising by principal component analysis with local pixel grouping. Pattern Recognit. 43(4):1531–1549.

Zhao X., Liu G, Sui Y, Xu M, Tong L. 2021. Denoising method for raman spectra with low signal-to-noise ratio based on feature extraction. Spectrochim Acta Part A Mol Biomol Spectrosc. 250:119374.