Exploring Spectral Analysis Methods with a PCA Emulator

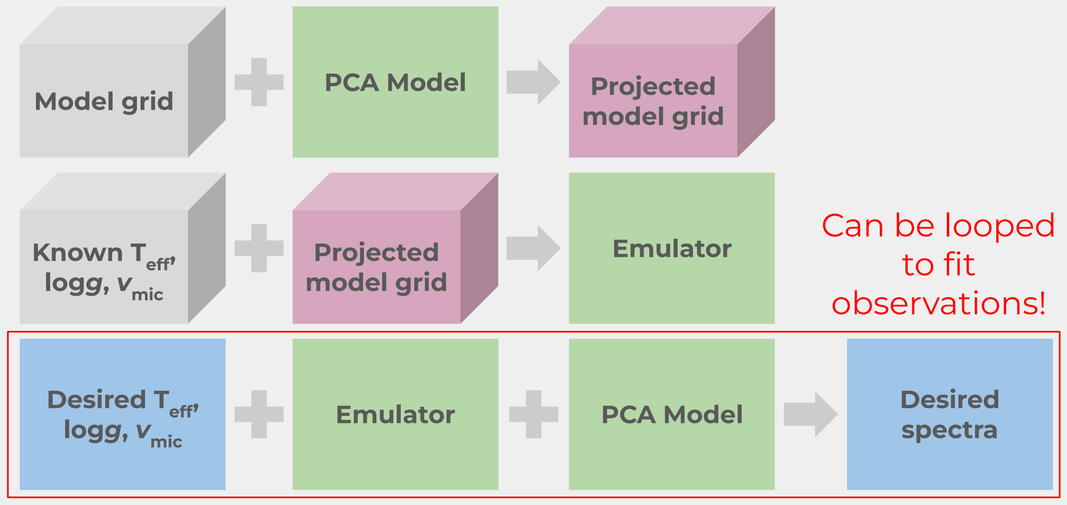
Alexandre Legault[1,2] (alegault@cab.inta-csic.es),
Francisco Najarro[1], **Miguel A. Urbaneja**[3] **& Miriam García**[1]
[1] Centro de Astrobiología, CSIC-INTA, Carretera de Ajalvir km4, 28850 Torrejón de Ardoz, Madrid, Spain.
[2] Universidad Autónoma de Madrid, Ciudad Universitaria de Cantoblanco, 28049 Madrid, Spain.
[3] Universität Innsbruck, Institut für Astro-und Teilchenphysik, Technikerstr. 25/8, A-6020 Innsbruck, Austria.

Motivation

- ➢ Modeling stellar atmospheres of massive stars and their synthetic spectra is a resource-intensive task often requiring hours of computing (e.g. CMFGEN [1], Fastwind [2], PoWR [3]). The large number of models needed to explore all the parameters playing a role make detailed quantitative spectral analysis extremely time consuming.
- ★ **Using a spectra emulator could bypass the model calculation steps, saving hours or days of computing time and analysis.**
- ➢ Such a model spectra emulator can be built using Principal Component Analysis (PCA). Routines already exist on IDL (Urbaneja, in prep.), and a description and usage of such a statistical emulator are shown in [4][5][6].
- ➢ We wish to build on this work, forward it on Python, and test the analysis of a B-type star optical spectra using the emulator.

Model Spectra Emulator

- ➢ **Principal component analysis (PCA)** is an unsupervised machine-learning method used to find patterns in data and project it in a lower dimensional space. [7]
- ➢ We can train a PCA model (which turns out to be an eigenvalue/eigenvector problem) using an initial grid of models. **Specific lines can be modeled, or the whole spectra.**
- ➢ The relation between the output of the PCA model (i.e. the grid of spectra projected in a lower dimensional space) and the associated physical parameters can be established, creating the emulator.
- ➢ This enables us to generate a new spectrum within the sampled parameter space, using the reverse process. This is effectively a **multi-dimensional statistical linear interpolator**.



*The PCA model is trained on a set of models pre-convolved for instrumental resolution. The emulated spectra will be convolved for *vsini* and v_{mac} at the future step of spectral fitting. We use the Python library Scikit-learn for all fitting and learning steps [8].

Markov Chain Monte Carlo (MCMC) Fitting with Spectra Emulator

- ➢ For testing the emulator and its fitting potential, we use the BSTAR2006 grid of precalculated stellar atmosphere models [11][12], suited for windless B stars, to fit a Galactic, main-sequence early B star spectra.
- ➢ The metallicity is fixed at solar, and the BSTAR2006 grid is used for training the emulator with 40 components.
- ➢ The emulator predictions are only physical where the spectra varies linearly with the parameters.

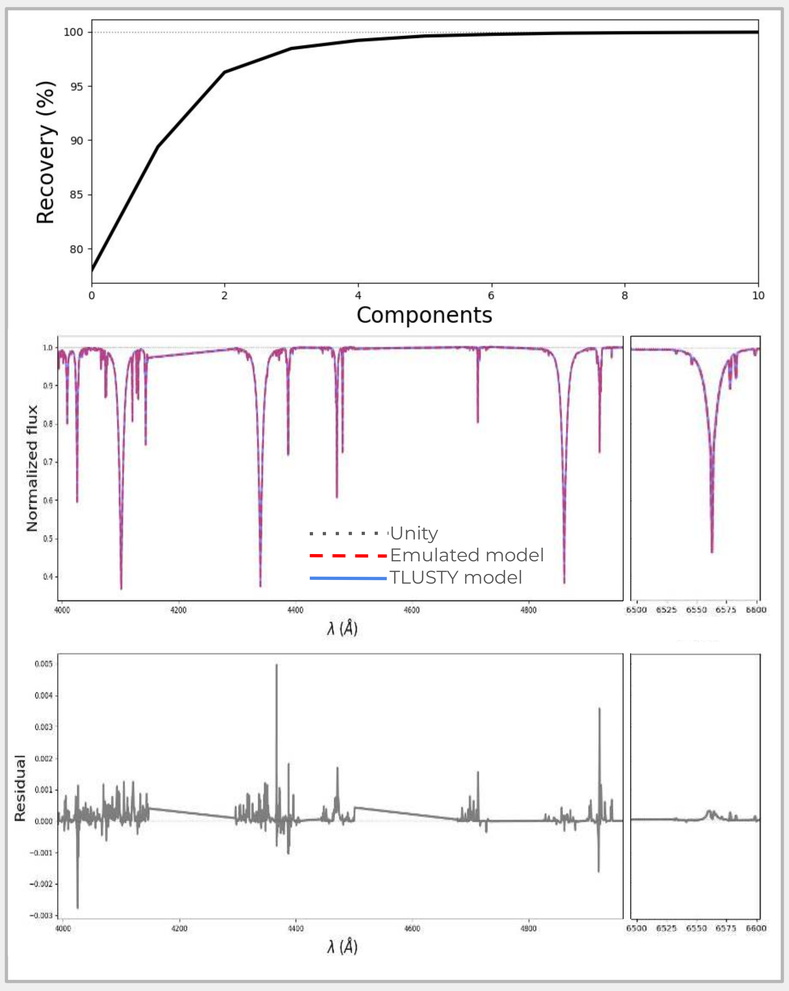


Fig. 1 - Top: Variance (or information) recovered by the emulator. At >10 preserved components, the features are nearly 100% recoverable. Middle: Superposed emulated and model spectra from BSTAR2006 (T_{eff} = 16 kK, log*g* = 3.50 dex, v_{mic} = 2 km/s, $Z/Z_{\odot}$ = 1). Bottom: Residual between the emulated and model spectra. **The larger discrepancies arise in the cores of the lines**.

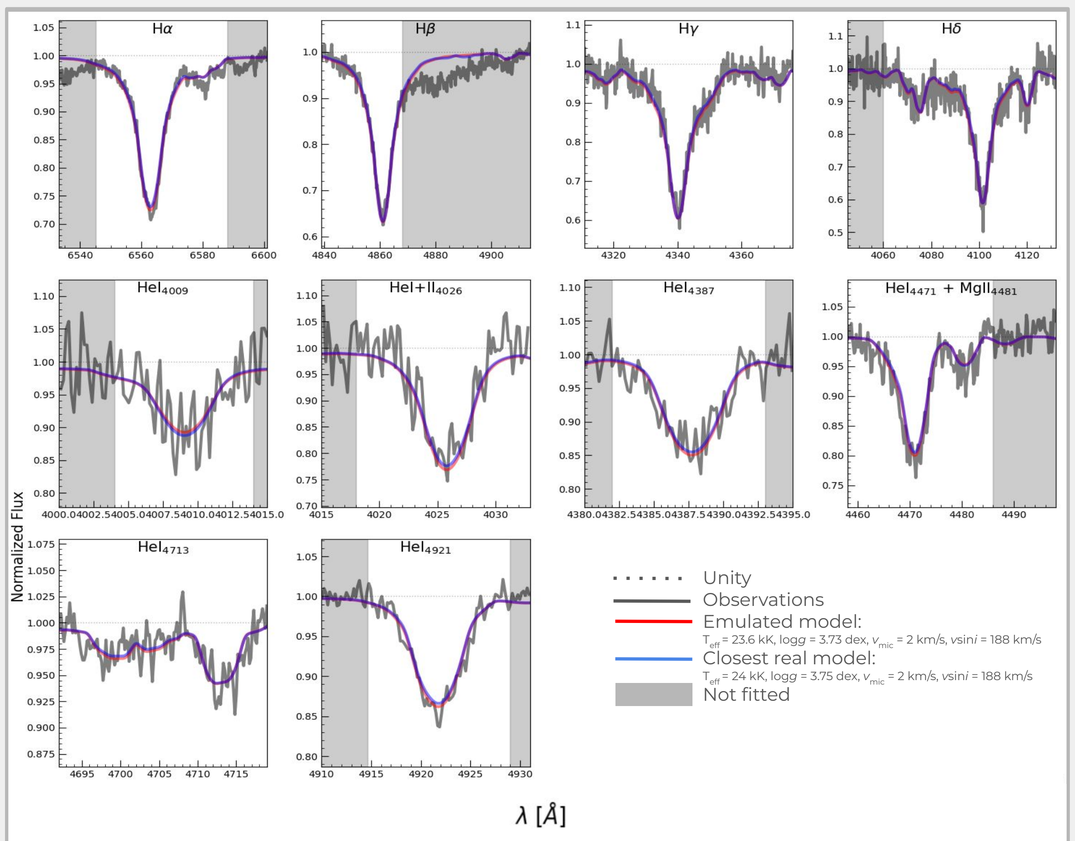


Fig. 2 - Diagnostic lines fitted from the optical spectra of vdBH245-9 (grey line), a Galactic early B-type star in the vdBH245 open cluster, part of the sample of stars analyzed in Legault et al. 2025 (in prep.). The red and blue lines show the best-fitting emulated model and the best-fitting BSTAR2006 model, respectively.

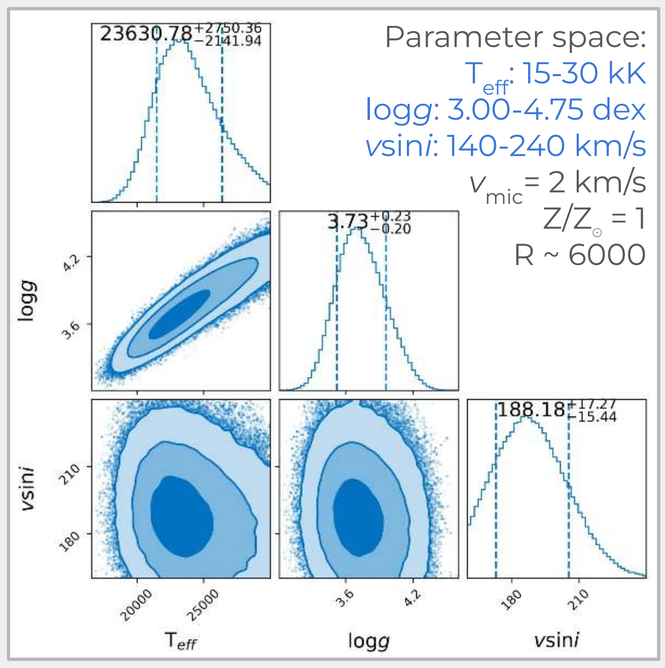


Fig. 3 - MCMC results of the emulated spectra fitting. The contour plots show the 1, 2 and 3σ intervals, with 1σ used as a formal uncertainty. We use the *bilby* package, which allows fixed parameters and incorporates *emcee*, a MCMC Python library [9][10].

Conclusions

- ➢ The emulator generates a spectra within seconds or less, and replicates properly the diagnostic features.
- ➢ MCMC is completed in reasonable time (around an hour if T_{eff}, log*g*, v_{mic}, v_{mac} and *vsini* are free parameters).
- ➢ If the emulated parameter space includes non-linearities, non-physical features appear in the emulated spectra. **More testing needed here**.
- ➢ The emulator will be used to fit OB-type multi-wavelength spectra with $N_{params} > 10$.

References

[1] Hillier, D., & Miller, D. (1998). *ApJ*, 496(1), 407-427.
[2] Puls, J., Najarro, F., Sundqvist, J., & Sen, K. (2020). *A&A*, 642, A172. (and references therein)
[3] Sander, A., Shenar, T., Hainich, R., Gímenez-García, A., Todt, H., & Hamann, W.R. (2015). *A&A*, 577, A13.
[4] Urbaneja, M., Kudritzki, R.P., Gieren, W., Pietrzyński, G., Bresolin, F., & Przybilla, N. (2017). *AJ*, 154(3), 102.
[5] Inno, L., Urbaneja, M., Matsunaga, N., Bono, G., Nonino, M., Debattista, V., Sormani, M., Bergemann, M., da Silva, R., Lemasle, B., Romaniello, M., & Rix, H.W. (2019). *MNRAS*, 482(1), 83-97.
[6] de Burgos, A., Simon-Diaz, S., Urbaneja, M., & Puls, J. (2024). *A&A, 687, A228.*
[7] F.R.S., Karl Pearson. *Philosophical Magazine Series 1* 2 (1901): 559-572.
[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, . (2011). *Journal of Machine Learning Research*, 12, 2825-2830.
[9] Ashton, G., Hubner, M., Lasky, P., Talbot, C., Ackley, K., Biscoveanu, S., Chu, Q., Divakarla, A., Easter, P., Goncharov, B., Hernandez Vivanco, F., Harms, J., Lower, M., Meadors, G., Melchor, D., Payne, E., Pitkin, M., Powell, J., Sarin, N., Smith, R., & Thrane, E. (2019). *ApJS, 241(2), 27.*
[10] Foreman-Mackey, D., Hogg, D., Lang, D., & Goodman, J. (2013). *PASP, 125, 306-312.*
[11] Lanz, T., & Hubeny, I. (2007). *ApJS, 169(1), 83-104.*
[12] Hubeny, I., & Lanz, T. (1995). *ApJ*, 439, 875.

Acknowledgements

Alexandre Legault, Francisco Najarro and Miriam García acknowledge grant PID2022-137779OB-C41. Alexandre Legault and Miriam García also acknowledge grants PREP2022-000263 and PID2022-140483NB-C22 respectively. All grants are funded by the Spanish Ministry of Science, Innovation and Universities/State Agency of Research MICIU/AEI/10.13039/501100011033.