The Journal of Open Source Software

TelescopeML – I. An End-to-End Python Package for Interpreting Telescope Datasets through Training Machine Learning Models, Generating Statistical Reports, and Visualizing Results

Ehsan (Sam) Gharib-Nezhad[1,2], **Natasha E. Batalha**[1], **Hamed Valizadegan**[3,4], **Miguel J. S. Martinho**[3,4], **and Gopal Nookula**[5]

1 Space Science and Astrobiology Division, NASA Ames Research Center, Moffett Field, CA, 94035 USA **2** Bay Area Environmental Research Institute, NASA Research Park, Moffett Field, CA 94035, USA **3** Universities Space Research Association (USRA), Mountain View, CA 94043, USA **4** Intelligent Systems Division, NASA Ames Research Center, Moffett Field, CA 94035, USA **5** Department of Computer Science, University of California, Riverside, Riverside, CA 92507 USA

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Paul La Plante

Reviewers:

- @oparisot

Submitted: 29 November 2023
Published: unpublished

In partnership with

AMERICAN ASTRONOMICAL SOCIETY

This article and software are linked with research article DOI LINK OF ApJ PAPER, published in the The Astrophysical Journal.

Statement of Need

We are on in a new era of space exploration, thanks to advancements in ground- and space-based telescopes, such as the James Webb Space Telescope (e.g., Gardner et al., 2023) and CRIRES. These remarkable instruments collect high-resolution, high-signal-to-noise spectra from extrasolar planets (e.g., Alderson et al., 2023), and brown dwarfs (e.g., Miles et al., 2023) atmospheres. Without an accurate interpretation of this data, the main objectives of space missions will not be fully accomplished. Different analytical and statistical methods, such as the chi-squared-test, Bayesian statistics as well as radiative-transfer atmospheric modeling packages have been developed (e.g., Batalha et al., 2019; MacDonald, 2023) to interpret the spectra. They utilize either forward- and/or retrieval-radiative transfer modeling to analyze the spectra and extract physical information, such as atmospheric temperature, metallicity, carbon-to-oxygen ratio, and surface gravity (Iyer et al., 2023; Line et al., 2014; Marley & Robinson, 2015). These atmospheric models rely on generating the physics and chemistry of these atmospheres for a wide range of thermal structures and compositions. In addition to Bayesian-based techniques, machine learning and deep learning methods have been developed in recent years for various astronomical problems, including confirming the classification of light curves for exoplanet validation (e.g., Valizadegan et al., 2021), recognizing molecular features (Zingales & Waldmann, 2018) as well as interpreting brown dwarfs spectra using Random Forest technique (e.g., Lueber et al., 2023). Here, we present one of the first applications of deep learning and convolutional neural networks on the interpretation brown dwarf atmospheric datasets. The configuration of a CNN and the key concepts can be found in (Goodfellow et al., 2016; Kiranyaz et al., 2021).

With the continuous observation of these objects and the increasing amount of data, there is a critical need for a systematic pipeline to quickly explore the datasets and extract important physical from them. In the future we can expand our pipeline to exoplanet atmospheres, and use it to provide insights about the diversity of exoplanets and brown dwarfs' atmospheric compositions. Ultimately, `TelescopeML` will help facilitate the long-term analysis of this data in research. `TelescopeML` is an ML Python package with Sphinx-ed user-friendly documentation that provides both trained ML models and ML tools for interpreting observational data captured by telescopes.

Functionality and Key Features

`TelescopeML` is a Python package comprising a series of modules, each equipped with specialized machine learning and statistical capabilities for conducting Convolutional Neural Networks (CNN) or Machine Learning (ML) training on datasets captured from the atmospheres of extrasolar planets and brown dwarfs. The tasks executed by the `TelescopeML` modules are outlined below and visualized in following Figure:

- **DataMaster module**: Performs various tasks to process the datasets, including:
  - Preparing inputs and outputs
  - Splitting the dataset into training, validation, and test sets
  - Scaling/normalizing the data
  - Visualizing the data
  - Conducting feature engineering
- **DeepTrainer module**: Utilizes different methods/packages such as TensorFlow to:
  - Build Convolutional Neural Networks (CNNs) model using the training examples
  - Utilize tuned hyperparameters
  - Fit/train the ML models
  - Visualize the loss and training history, as well as the trained model's performance
- **Predictor module**: Implements the following tasks to predict atmospheric parameters:
  - Processes and predicts the observational datasets
  - Deploys the trained ML/CNNs model to predict atmospheric parameters
  - Visualizes the processed observational dataset and the uncertainty in the predicted results
- **StatVisAnalyzer module**: Provides a set of functions to perform the following tasks:
  - Explores and processes the synthetic datasets
  - Performs the chi-square test to evaluate the similarity between two datasets
  - Calculates confidence intervals and standard errors
  - Functions to visualize the datasets, including scatter plots, histograms, boxplots

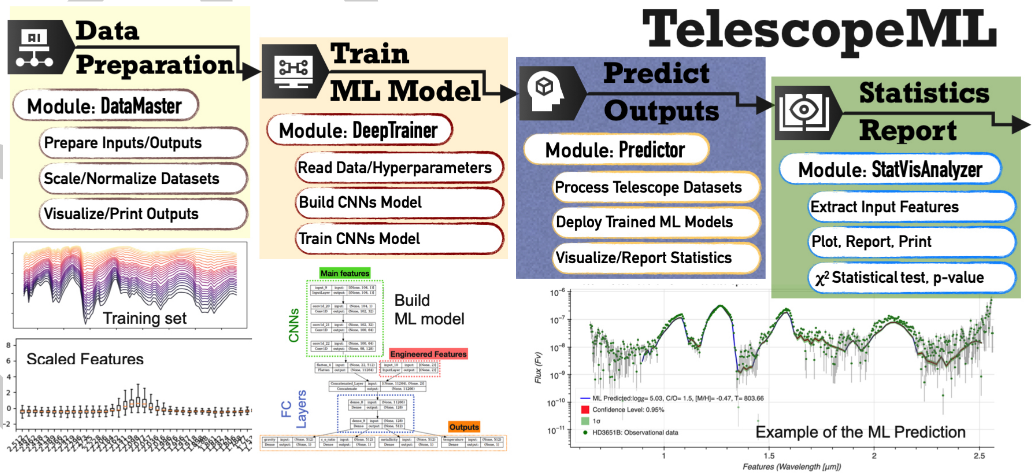

Figure 1: TelescopeML main modules to manipulate the training example, build the ML model, train and tune it, and ultimately extract the target features from the observational data.

Documentation

`TelescopeML` is available and being maintained as a GitHub repository at github.com/EhsanGharibNezhad/TelescopeML. Online documentation is hosted with *Sphinx* using *ReadtheDocs* tools and includes several instructions and tutorials as follows:

- **Main page**: ehsangharibnezhad.github.io/TelescopeML/
- **Installation**: ehsangharibnezhad.github.io/TelescopeML/installation.html
- **Tutorials and examples**: ehsangharibnezhad.github.io/TelescopeML/tutorials.html
- **The code**: ehsangharibnezhad.github.io/TelescopeML/code.html
- **ML Concepts**: ehsangharibnezhad.github.io/TelescopeML/knowledgebase.html

Users and Future Developments

Astrophysicists with no prior machine learning knowledge can deploy the `TelescopeML` package and download the pre-trained ML or CNN models to interpret their observational data. In this scenario, pre-trained ML models, as well as the PyPI package, can be installed and deployed following the online instructions. Tutorials in the Sphinx documentation include examples for testing the code and also serve as a starting point. For this purpose, a basic knowledge of Python programming is required to install the code, run the tutorials, deploy the modules, and extract astronomical features from the datasets. The necessary machine learning background and a detailed guide for package installation, along with links to further Python details, are provided to help understand the steps and outputs.

Astrophysicists with machine learning expertise and data scientists can also benefit from this package by developing and fine-tuning the modules and pre-trained models to accommodate more complex datasets from various telescopes. This effort could also involve the utilization of new ML and deep learning algorithms, adding new capabilities such as feature engineering methods, and further optimization of hyperparameters using different and more efficient statistical techniques. The ultimate outcome from these two groups would be the creation of more advanced models with higher performance and robustness, as well as the extension of the package to apply to a wider range of telescope datasets.

Utilized Underlying Packages

For processing datasets and training ML models in `TelescopeML`, the following software/packages are employed: Scikit-learn (Pedregosa et al., 2011), TensorFlow (Abadi et al., 2015), AstroPy (Astropy Collaboration et al., 2022), SpectRes (Carnall, 2017), Pandas (team, 2020), NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020), Matplotlib (Hunter, 2007), Seaborn (Waskom, 2021), Bokeh (Bokeh Development Team, 2018). Additionally, for generating training astronomical datasets, Picaso (Batalha et al., 2019) is implemented.

Acknowledgements

EGN and GN would like to thank OSTEM internships and funding through the NASA with contract number 80NSSC22DA010. EGN acknowledges ChatGPT 3.5 for proofreading some of the functions and providing helpful suggestions.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. https://www.tensorflow.org/

Alderson, L., Wakeford, H. R., Alam, M. K., Batalha, N. E., Lothringer, J. D., Adams Redai, J., Barat, S., Brande, J., Damiano, M., Daylan, T., Espinoza, N., Flagg, L., Goyal, J. M., Grant, D., Hu, R., Inglis, J., Lee, E. K. H., Mikal-Evans, T., Ramos-Rosado, L., … Zhang,

X. (2023). Early Release Science of the exoplanet WASP-39b with JWST NIRSpec G395H. *Nature*, *614*(7949), 664–669. https://doi.org/10.1038/s41586-022-05591-3

Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L., Shupe, D. L., Patil, A. A., Corrales, L., Brasseur, C. E., N"othe, M., Donath, A., Tollerud, E., Morris, B. M., Ginsburg, A., Vaher, E., Weaver, B. A., Tocknell, J., Jamieson, W., … Astropy Project Contributors. (2022). The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package. *935*(2), 167. https://doi.org/10.3847/1538-4357/ac7c74

Batalha, N. E., Marley, M. S., Lewis, N. K., & Fortney, J. J. (2019). Exoplanet reflected-light spectroscopy with PICASO. *The Astrophysical Journal*, *878*(1), 70.

Bokeh Development Team. (2018). *Bokeh: Python library for interactive visualization*. https://bokeh.pydata.org/en/latest/

Carnall, A. C. (2017). SpectRes: A Fast Spectral Resampling Tool in Python. *arXiv e-Prints*, arXiv:1705.05165. https://doi.org/10.48550/arXiv.1705.05165

Gardner, J. P., Mather, J. C., Abbott, R., Abell, J. S., Abernathy, M., Abney, F. E., Abraham, J. G., Abraham, R., Abul-Huda, Y. M., Acton, S., & al., et. (2023). The James Webb Space Telescope Mission. *135*(1048), 068001. https://doi.org/10.1088/1538-3873/acd1b5

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Iyer, A. R., Line, M. R., Muirhead, P. S., Fortney, J. J., & Gharib-Nezhad, E. (2023). The SPHINX M-dwarf Spectral Grid. I. Benchmarking New Model Atmospheres to Derive Fundamental M-dwarf Properties. *944*(1), 41. https://doi.org/10.3847/1538-4357/acabc2

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., & Inman, D. J. (2021). 1D convolutional neural networks and applications: A survey. *Mechanical Systems and Signal Processing*, *151*, 107398. https://doi.org/https://doi.org/10.1016/j.ymssp.2020.107398

Line, M. R., Knutson, H., Wolf, A. S., & Yung, Y. L. (2014). A systematic retrieval analysis of secondary eclipse spectra. II. A uniform analysis of nine planets and their c to o ratios. *The Astrophysical Journal*, *783*(2), 70.

Lueber, A., Kitzmann, D., Fisher, C. E., Bowler, B. P., Burgasser, A. J., Marley, M., & Heng, K. (2023). Intercomparison of Brown Dwarf Model Grids and Atmospheric Retrieval Using Machine Learning. *954*(1), 22. https://doi.org/10.3847/1538-4357/ace530

MacDonald, R. J. (2023). POSEIDON: A multidimensional atmospheric retrieval code for exoplanet spectra. *Journal of Open Source Software*, *8*(81), 4873. https://doi.org/10.21105/joss.04873

Marley, M. S., & Robinson, T. D. (2015). On the Cool Side: Modeling the Atmospheres of Brown Dwarfs and Giant Planets. *53*, 279–323. https://doi.org/10.1146/annurev-astro-082214-122522

Miles, B. E., Biller, B. A., Patapis, P., Worthen, K., Rickman, E., Hoch, K. K. W., Skemer, A., Perrin, M. D., Whiteford, N., Chen, C. H., Sargent, B., Mukherjee, S., Morley, C. V., Moran, S. E., Bonnefoy, M., Petrus, S., Carter, A. L., Choquet, E., Hinkley, S., … Zhang, Z. (2023). The JWST Early-release Science Program for Direct Observations of Exoplanetary

Systems II: A 1 to 20 μm Spectrum of the Planetary-mass Companion VHS 1256-1257 b. *ApJL*, *946*(1), L6. https://doi.org/10.3847/2041-8213/acb04a

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

team, T. pandas development. (2020). *Pandas-dev/pandas: pandas* (latest). Zenodo. https://doi.org/10.5281/zenodo.3509134

Valizadegan, H., Martinho, M., Wilkens, L. S., Jenkins, J. M., Smith, J., Caldwell, D. A., Twicken, J. D., Gerum, P. C., Walia, N., Hausknecht, K., Lubin, N. Y., Bryson, S. T., & Oza, N. C. (2021). ExoMiner: A Highly Accurate and Explainable Deep Learning Classifier that Validates 301 New Exoplanets. *arXiv e-Prints*, arXiv:2111.10009. https://arxiv.org/abs/2111.10009

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., … SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021

Zingales, T., & Waldmann, I. P. (2018). ExoGAN: Retrieving Exoplanetary Atmospheres Using Deep Convolutional Generative Adversarial Networks. *156*(6), 268. https://doi.org/10.3847/1538-3881/aae77c