Check for updates

RESEARCH ARTICLE

REVISED Machine learning based stellar classification with highly sparse photometry data

[version 2; peer review: 2 approved, 1 approved with reservations]

Seán Enis Cody[1], Sebastian Scher[1], Iain McDonald[2], Albert Zijlstra[2], Emma Alexander[2], Nick Cox[3]

[1]Know-Center GmbH, Graz, 8010, Austria
[2]The University of Manchester, Manchester, England, M13 9PL, UK
[3]ACRI-ST, Grasse, 06130, France

v2 **First published:** 16 Feb 2024, **4**:29
https://doi.org/10.12688/openreseurope.17023.1
Latest published: 28 Aug 2024, **4**:29
https://doi.org/10.12688/openreseurope.17023.2

Abstract

Background

Identifying stars belonging to different classes is vital in order to build up statistical samples of different phases and pathways of stellar evolution. In the era of surveys covering billions of stars, an automated method of identifying these classes becomes necessary.

Methods

Many classes of stars are identified based on their emitted spectra. In this paper, we use a combination of the multi-class multi-label Machine Learning (ML) method XGBoost and the PySSED spectral-energy-distribution fitting algorithm to classify stars into nine different classes, based on their photometric data. The classifier is trained on subsets of the SIMBAD database. Particular challenges are the very high sparsity (large fraction of missing values) of the underlying data as well as the high class imbalance. We discuss the different variables available, such as photometric measurements on the one hand, and indirect predictors such as Galactic position on the other hand.

Results

We show the difference in performance when excluding certain variables, and discuss in which contexts which of the variables should be used. Finally, we show that increasing the number of samples of a

Open Peer Review

Approval Status ? ✓ ✓

	1	2	3
version 2 (revision) 28 Aug 2024		✓ view	✓ view
version 1 16 Feb 2024	? view	✓ view	✗ view

1. **Giacomo Cordoni**, Australian National University, Canberra, Australia
2. **Nestor Sanchez**, Universidad Internacional de Valencia, Valencia, Spain
3. **John YH Soo**, Universiti Sains Malaysia, Pulau Pinang, Malaysia

Any reports and responses or comments on the article can be found at the end of the article.

particular type of star significantly increases the performance of the model for that particular type, while having little to no impact on other types. The accuracy of the main classifier is □0.7 with a macro F1 score of 0.61.

Conclusions

While the current accuracy of the classifier is not high enough to be reliably used in stellar classification, this work is an initial proof of feasibility for using ML to classify stars based on photometry.

Plain language summary

Astronomy is at the forefront of the 'Big Data' regime, with telescopes collecting increasingly large volumes of data. The tools astronomers use to analyse and draw conclusions from these data need to be able to keep up, with machine learning providing many of the solutions. Being able to classify different astronomical objects by type helps to disentangle the astrophysics making them unique, offering new insights into how the Universe works. Here, we present how machine learning can be used to classify different kinds of stars, in order to augment large databases of the sky. This will allow astronomers to more easily extract the data they need to perform their scientific analyses.

Keywords

stellar classification, photometry, astrophysics, machine learning, sparsity, XGBoost, class imbalance, sampling bias,



This article is included in the Horizon Europe gateway.



This article is included in the Astronomy, Astrophysics & Cosmology gateway.



This article is included in the Horizon 2020 gateway.

Corresponding authors: Seán Enis Cody (scody@know-center.at), Sebastian Scher (sscher@know-center.at)

Author roles: Cody SE: Formal Analysis, Investigation, Methodology, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Scher S**: Investigation, Methodology, Supervision, Writing – Original Draft Preparation, Writing – Review & Editing; **McDonald I**: Data Curation, Investigation, Methodology, Software, Writing – Original Draft Preparation, Writing – Review & Editing; **Zijlstra A**: Methodology, Supervision, Writing – Review & Editing; **Alexander E**: Investigation, Methodology, Writing – Review & Editing; **Cox N**: Funding Acquisition, Project Administration, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This work is carried out as part of the EXPLORE project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004214 (EXPLORE) and from the European Union's Horizon Europe research and innovation programme under grant agreement No. 101091534 (KNOWSKITE-X).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2024 Cody SE *et al*. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Cody SE, Scher S, McDonald I *et al.* **Machine learning based stellar classification with highly sparse photometry data [version 2; peer review: 2 approved, 1 approved with reservations]** Open Research Europe 2024, **4**:29 https://doi.org/10.12688/openreseurope.17023.2

First published: 16 Feb 2024, **4**:29 https://doi.org/10.12688/openreseurope.17023.1

REVISED **Amendments from Version 1**

Fully revised textual elements, many clarifications added to the paper to make the objectives and environment clearer. Most important of which is a brief section discussing the goals explicitly in the introduction.

Furthermore, a discussion section for results obtained through a SHAP explainability study of the final classifier, findings in regards to bands used and missing data patterns are discussed here.

Added per star HR Diagrams to the main body.

Added HR-Diagrams of the classification results on a per class basis (FN, FP, TP) to appendix. Added density plots of star locations to appendix.

Any further responses from the reviewers can be found at the end of the article

Introduction

The billions of stars surveyed across our Galaxy and beyond by modern telescopes present a broad continuum of different properties. To make sense of this parametric diversity, astronomers typically classify stars as belonging to one or more stellar types (e.g., 1). These types often represent a phase in the evolutionary pathway of a particular kind of star, and are defined according to one or more observed properties. Such stellar types can be broad, such as the entire gamut of binary stars, or narrow, such as stars exhibiting a particular type of variability in the emitted light.

The star's spectrum represents one of the primary tools we can use to identify to which classes it belongs. However, the spectra of most stars remain unobserved due to the technical challenges of observing enough light from faint objects. Instead, we rely on photometry, i.e., a measure of the star's brightness in images taken through different ultraviolet, visible and infrared filters. The combination of these different brightnesses at different wavelengths of light define an approximate spectrum of the star, which is known as a spectral energy distribution (SED). Since the majority of catalogued stars are too faint to be observed with more detailed methods, the SED represents the majority of the information known about most of the stars that have been catalogued so far[2].

If we know the distance to a star and can estimate how much of its light has been absorbed or scattered by interstellar dust, we can correct the SED and estimate the amount of light emitted by the star at the observed wavelengths (e.g., 3). By fitting the corrected SED with atmospheric models of stars, we can estimate the temperature of the stellar surface and, by integrating under the corrected SED, we can determine the luminosity of the star (the rate at which it emits light across all wavelengths). These represent two of the most fundamental properties of a star, which are often used in classification[4,5]. From these data, we can also estimate the stellar radius and, by assuming the star's mass, its surface gravity. However, with only the SED, we are ignorant of many other aspects used to classify stars. We have no information about the star's mass or composition, its movement through space, its detailed spectral features, or any information on the variability of its light, all of which are used in classification (e.g., 6).

In this respect, the problem of stellar classification relates to similar problems in other domains, such as materials characterisation and Earth imaging, where hyper-spectral remote-sensing observations are used to classify objects in different environments[7–9].

Modern astronomical catalogues are large: the *Gaia* satellite's Data Release 3[10] contains approximately 10^9 stars with usable distances, to which other catalogues of similar size can be cross-matched to form SEDs. Fitting of these SEDs, as described above, has been automated: in this paper we use the PySSED software package[11] to collect and fit the SEDs of stars observed by *Gaia*. However, an automated stellar classifier that can cope with this wealth of information has yet to be constructed.

In this paper, we use a combination of known stellar positions, distances, motions in the sky, photometry and computed temperature, luminosity, radius and surface gravity as input to a supervised ML multi-label multi-class classification algorithm based on a sample of stars with existing classifications from the SIMBAD[12] astronomical database. We specifically focus on the top-level stellar classes in SIMBAD: binary, emission-line, evolved, low-mass, massive, main-sequence, chemically peculiar and variable stars, and young stellar objects. However, we do not attempt herein to optimise the list of classes using lower-level classes in the SIMBAD hierarchy.

The principal feasibility of using ML methods to classify stellar objects has been shown in 13, albeit on synthetically generated data. A number of works (e.g. 14–17) have used an ML approach to classify point sources into galaxies, quasars, and stars. ML has also been demonstrated in differentiating specific types of star, such as planetary nebulae[18], symbiotic stars[19], variable stars[20,21], and exoplanets in orbit around stars[22]. In contrast to these works, we focus on the entire range of stellar types, but only on stars specifically, and on the problem of how to classify them into different stellar classes. Stellar classification has been succesfully carried out for hundreds of millions of stars from spectra (e.g. 10). We focus on the challenge of classifying the much larger number of stars for which spectral observations are impractical or impossible, and for which only SEDs can be obtained.

To sum up, the main objectives of our paper are (1) to demonstrate the applicability of ML in classification of the entire range of stellar SEDs, given only sparse photometric measurements based on a random sample of the SIMBAD database; (2) to measure how inherent sampling bias and associated class imbalance in astronomical literature affects the classifier's performance; and (3) assess how limitations caused by minority classes in the training database can be improved by the acquisition of more data.

Challenges

There are several key challenges in this work that differentiate it from more typical supervised-learning ML problems. Firstly, the data used are a collection of many different sky surveys: these surveys cover different regions of the sky at differing sensitivities, and some stars bright enough for one survey may be too faint for another or, conversely, some stars may be observable by one survey but saturated in another. Consequently, a large fraction of data values are not present, i.e., the dataset is very sparse. This makes it much harder to train accurate ML models on the data.

Secondly, the stars for which classifications have been produced are severely biased. Some regions of the sky have been specifically targeted for particular kinds of star (e.g., the footprints of surveys for variable stars), thus leading to a spatial bias in the recovery of certain stellar types. Furthermore, certain types of stars are globally better classified than others, e.g., large lists of massive stars have been identified as these stars are bright and well-studied, while there are very few classified main-sequence stars, since these represent the majority of stars and are therefore often considered unworthy of classification. This strong bias makes both training and evaluating the ML models challenging.

Thirdly, stars may fall into multiple categories, though certain category combinations are prohibited: e.g., a star may be both massive and variable, but cannot be both massive and low-mass. This problem adds complexity to producing the output of a ML algorithm and defining its success.

Finally, as mentioned above, specific stellar classes are identified based on data types other than those we have collected (e.g., temporal variability or characteristics of detailed spectra). Therefore, it is *a priori* not clear how well stars can be classified based on these data alone. This fundamentally differentiates our problem from certain other supervised ML problems. For example, in many image-classification problems, such as the widely used Imagenet dataset[23], humans themselves can do the classification given the images. While this does not necessarily mean that is possible in practice to manage to train an ML algorithm to do the same task, it is at least known that such problem is theoretically solvable, given a good enough training algorithm and sufficient training samples. This is not the case for our problem setting, and we thus have no *a priori* knowledge of how well, even with unlimited training samples, a classification can actually be made for certain stellar classes.

Methods

In this section we describe the data used, the PySSED spectral energy distribution fitting algorithm, the pre-processing steps, machine-learning methods and tuning procedure.

PySSED

PySSED is a Python code for fitting stellar spectral energy distributions. Given a star, list of stars, or region of the sky, it will retrieve the star's co-ordinates from SIMBAD, a database for astronomical objects outside of the Solar System[12], and perform a search using the associated VizieR database[1] for multi-wavelength photometric brightness measurements, distances and other requested parameters. PySSED will then correct the photometry for interstellar reddening at the location and distance of the star, before fitting the corrected photometry with a stellar model atmosphere to estimate the temperature and luminosity of the star. Differences between the observed and modelled fluxes are also recorded. Note that the interstellar reddening correction is based on *Gaia* spectroscopic data, meaning *Gaia*-derived stellar parameters are no longer a completely independent comparison. An in-depth analysis of the SED fitting outputs, procedures, quantification of the impact of low photometric information, and a discussion of data biases can be found in McDonald *et al.*[11], while Wesson *et al.*[24] shows an example where PySSED is used on *James Webb Space Telescope* data.

Data

Our training set of stellar classifications comes from SIMBAD. While SIMBAD compares more objects than only stars, in this study we only look at stars, and specifically at the following nine stellar classes: (1.6) binary stars, (1.10) emission-line stars, (1.4) evolved stars, (1.8) low-mass stars, (1.1) massive stars, (1.3) main-sequence stars, (1.5) chemically peculiar stars, (1.9) variable stars and (1.2) young stellar objects (YSOs). The numbers in parentheses refer to the taxonomy used in SIMBAD. The stellar classes and the abbreviations that will be used throughout the paper are summarized in Table 1.

These stellar classes are not exclusive, thus there is overlap expected between certain classes. For example, many evolved stars are variable. (However, between other classes there can be no overlap, e.g. stars can't be both massive and low-mass). Allowed and not-allowed overlaps are detailed in Table 2.

Table 1. Star types used in the study and their abbreviations.

Star type	Abbreviation
Binary or Double Star	**
Emission Line Star	em*
Evolved Star	ev*
Low-Mass Star	lm*
Massive Star	ma*
Main Sequence Star	ms*
Chemically Peculiar Star	pe*
Variable Star	v*
Young Stellar Object	y*o height

[1] https://vizier.u-strasbg.fr/viz-bin/VizieR

Table 2. Allowed overlaps between the nine star types.

Category	Binary	Emission	Evolved	Low-mass	Massive	Main-seq.	Chem. pec.	Variable	Young
Binary	—	Y	?	Y	Y	Y	Y	Y	Y
Emission	Y	—	Y	?	Y	?	Y	Y	Y
Evolved	?	Y	—	N	Y	N	Y	Y	N
Low-mass	Y	?	N	—	N	Y	?	Y	Y
Massive	Y	Y	Y	N	—	Y	?	Y	Y
Main seq.	Y	?	N	Y	Y	—	Y	Y	N
Chem. pec.	Y	Y	Y	?	?	Y	—	Y	Y
Variable	Y	Y	Y	Y	Y	Y	Y	—	Y
Young	Y	Y	N	Y	Y	N	Y	Y	—
References	25	26	27	28	29,30	31	32	33	34

Y = overlap expected, ? = some overlap possible, N = overlap disallowed.

SIMBAD contains over 1 million stars of the classes we use. In this study, we use two subsets of those stars for analysis with PySSED. The first is a subset of 10% randomly sampled stars, resulting in 117 128 stars. This choice was largely dictated by computational time, originating from evaluating multiple queries per star at the CDS SIMBAD database during the PySSED preprocessing steps. This will be referred to as the "main dataset". In the main dataset, the most prevalent star type are variable stars, followed by evolved stars and binary stars, while the least prevalent class - only 0.24% – are massive stars (see Figure 3a). While we use random sampling of the SIMBAD database, this does not correspond to a random sample of stars in reality, as the stars in SIMBAD are not a random sample of real stars. Therefore, as a second dataset, we extended the main dataset with 80,000 additional single label main-sequence stars.

Main-sequence stars are only a very small fraction of the stars in the random dataset, but are severely underrepresented in SIMBAD: in reality they represent the majority of stars on the sky. We will refer to this dataset as the "extended dataset". Comparing the results between the two datasets allows us to show how adding more samples of a specific star type affects the performance of the classifier.

The results of fitting our "main dataset" with PySSED is shown in Figure 1. Several deficiencies in the PySSED output data are apparent. Most notable among these are the set of stars stretching to unphysically high temperatures and luminosities. This is the most significant source of error in the PySSED analysis of these stars, and is caused when hot stars are over-corrected for interstellar reddening, often due to overestimated distances or inaccuracies in the coarsely sampled interstellar-reddening maps. Removal of these stars did not significantly improve the ML classifiers. Therefore, effective temperature and luminosity output by PySSED were directly used as input features for the ML models. Other sources of error, including luminosity changes from errors in distance or deficiencies in individual photometric data, are small by comparison. The H–R diagram for each class is shown in Figure 2.

Train test split

ML algorithms tend to have a large number of parameters and can therefore easily overfit on the training data. This means that when using the trained model to make predictions on the training data itself the predictions are very good, but that model does not work well on unseen data. The scores on the training set can thus be misleading. Therefore, it is essential to withhold data from training that can then be used for an independent evaluation of the algorithm. This is usually referred to as the "test" set. Additionally, if one performs model tuning – which is essentially trying out multiple algorithms – one needs a third set that is used as evaluation set in the tuning phase, or alternatively use cross-validation on the training set. Cross-validation provides a more robust and accurate measure of performance during tuning, and we therefore opted for 5-Fold Cross Validation (CV). Deciding how to split up a dataset into training and test set is not trivial. Often, random sampling is employed. In our case standard random sampling is not feasible due to the high class imbalance, as the number of samples for the minority classes in the test set would be too low for getting robust results. Therefore, we used a modified random sampling that ensures that a minimum number of each star type is contained in the test set. To that end, at least 40 instances, accounting for about 15 percent of the minority class, were included per class in the test set used for final evaluation, before randomly sampling from the remaining data. We did this independently for the main and for the extended dataset. All scores reported in this paper are on the respective independent "test" datasets. Furthermore, in accordance with avoiding nonsensical predictions, if no class

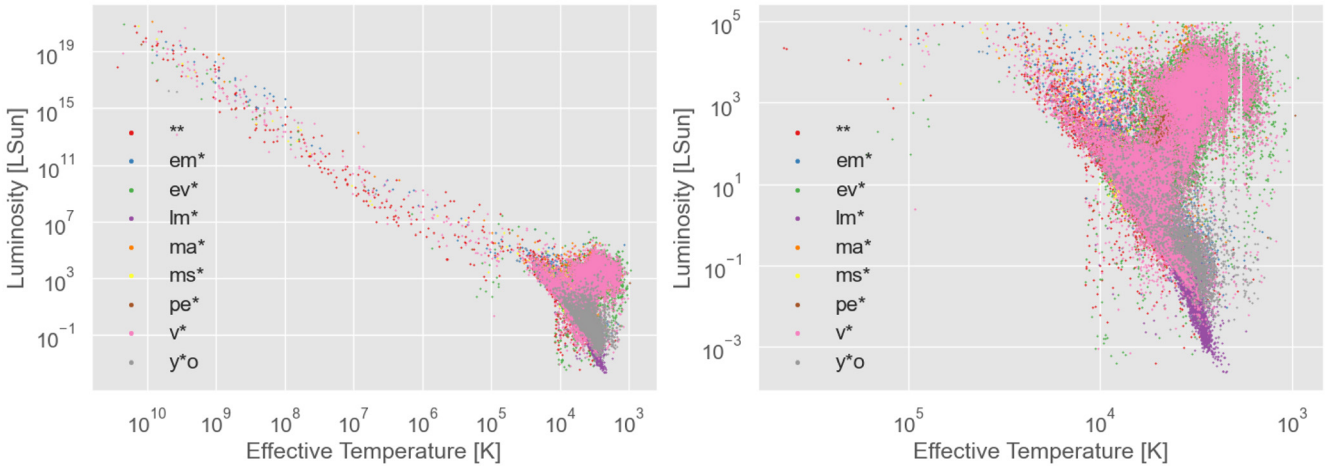


Figure 1. Hertzsprung–Russell diagram of the stars in the dataset used, with effective temperature and luminosity estimated with PySSED. The right panel is a zoom in of the left plot. The different star types are indicated by color.

is predicted in a multi-label scenario, the class of highest confidence is selected as the label prediction.

Preprocessing

The default output of PySSED contains both the original photometry and the reddening-corrected photometry, the photometry of the model fit, the temperature, luminosity, sky position and a number of other parameters ("features" in ML-jargon), resulting in 442 feature columns. Description of the default outputs and bands used are given in McDonald *et al.*[11]. Some of these can be reasonably grouped together, which we do as part of data preprocessing, to reduce the data dimensionality.

The first steps taken in preprocessing the data involved explicitly setting any missing values (which differed in representation) to NaN (not-a-number) and converting the Earth-centric location variables, RA (right ascension) and Dec (declination) to Galactic coordinates.

The largest feature engineering step conducted during preprocessing involves condensing the information contained in the photometry into a single data feature at each wavelength. To do this, we make use of the observed photometric flux in each filter (F_o), the error on that flux (ε), and the modelled flux in the same filter (F_m). We then calculate an uncertainty, σ, via

$$\sigma = \sqrt{\epsilon^2 + \left(\frac{F_o}{10}\right)^2} \quad (1)$$

This artificial addition in quadrature of 10% of the observed flux sets an uncertainty floor, which accounts for systematic issues with the observations. These can include uncertainty in the zero-point calibration of the survey, or differences between the assumed and real photometric filter transmission curves. This allows us to generate a significance (S) for each filter, representing how deviant the observation is from the model:

$$S = \frac{\log(F_o / F_m)}{\log(\sigma + 1)} \quad (2)$$

The use of logarithms accounts for the typical uncertainties on astronomical observations being given in magnitudes, and the tendency of observations to be close to the photon limit of detectors, thus have asymmetrical errors due to Poisson noise. It also prevents the most extreme deviations from dominating the feature space. By taking the ratio of observed to modelled flux, we also remove the dominant effects of stellar temperature, which is the most significant determinant of the shape of a stellar SED.

After applying the above mentioned preprocessing steps to the default PySSED output data, 96 features are retained. Still, for individual stars, a lack of observations means data are not necessarily available for all of those features. A quantitative analysis of the sparsity of the main dataset is shown in Figure 3b and Figure 3c, which depict the 74% total missing data in the main dataset. When regarding Figure 3c for example, we find that only 27 out of the 96 features contain values for at least 50 percent of samples, i.e., 69 features have missing values for more than 50 percent of samples. There are various reasons for these missing data, though two factors dominate. First, the dynamic range to bright (saturated) and faint (missing) stars varies from survey to survey, and no survey samples the entire range of stellar brightnesses; some surveys were particularly chosen because they cover a small number of very bright stars that saturate in other surveys. Second, the spatial coverage of many surveys is limited, either because the telescope used had a small field of view and performed targeted observations, rather than covering the entire sky, or because the telescope's field-of-view is limited by the hemisphere it is in.

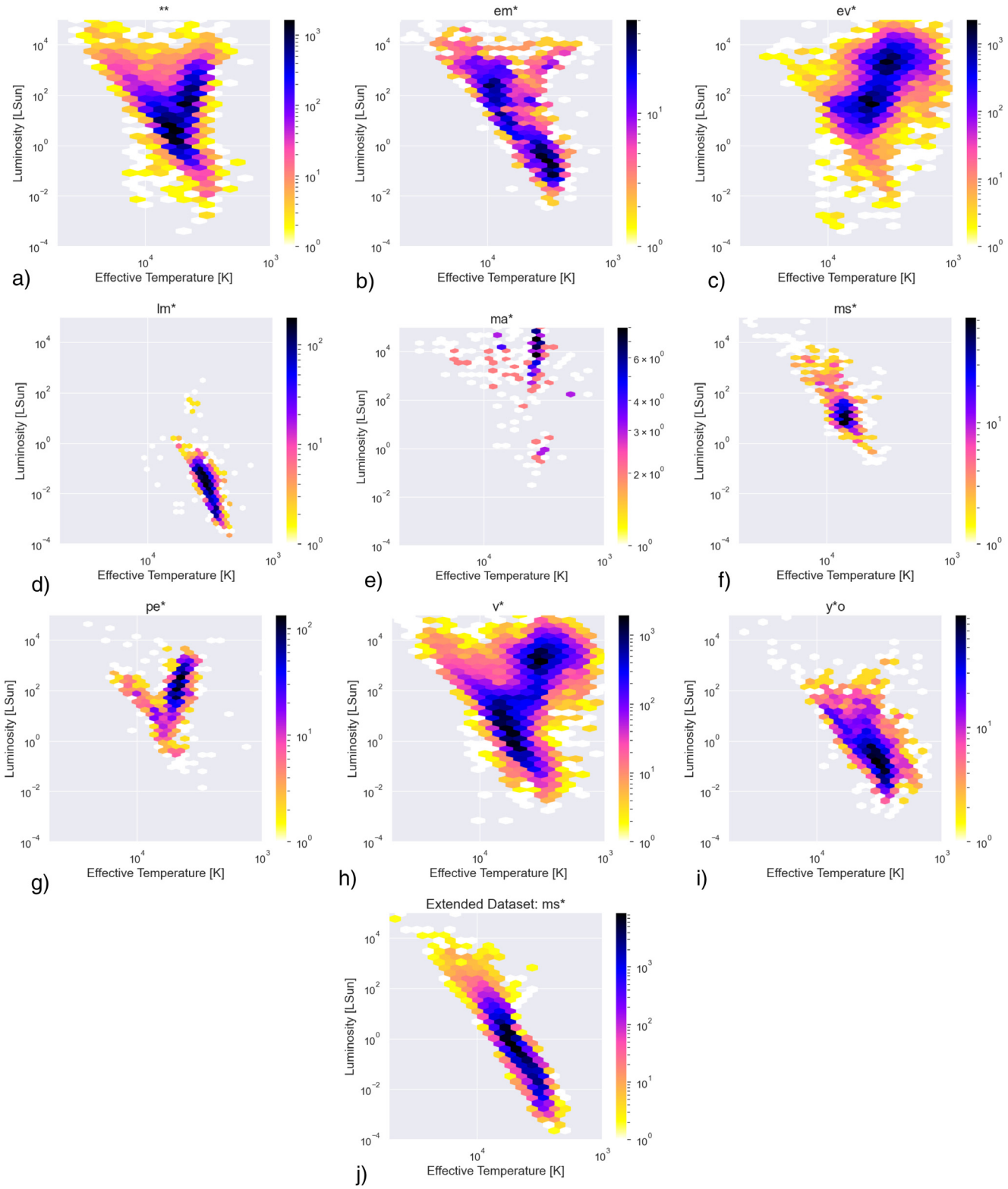


Figure 2. Density Plot HR Diagrams By Class Association.

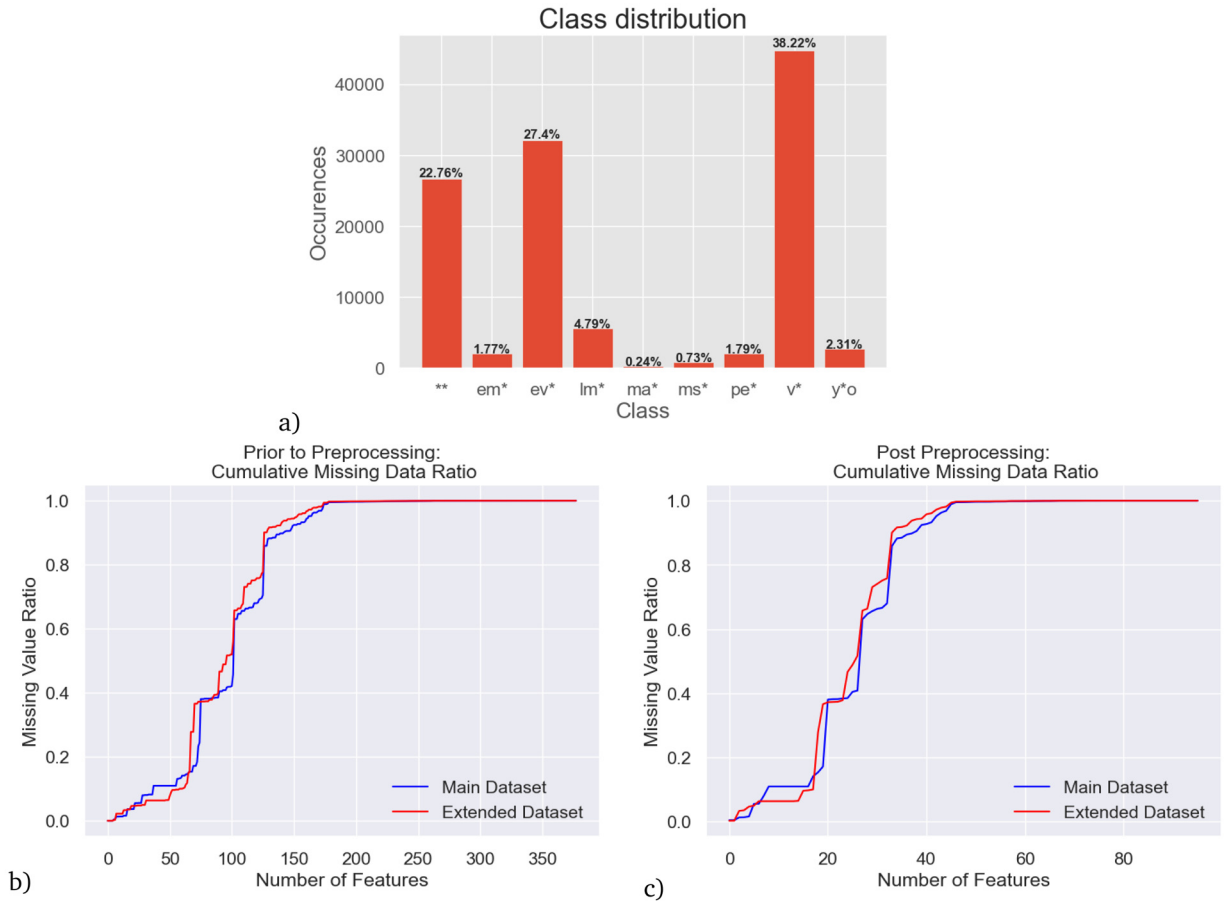


Figure 3. a) Class imbalance (distribution of star types) in the main dataset - multi-label samples are evaluated separately as single class instances for each of their class associations **b**) cumulative sparsity of the main dataset before preprocessing is applied, **c**) cumulative sparsity of the main dataset after preprocessing is done.

Feature packages

The features in the SIMBAD data can be separated into several distinct groups, with different physical and conceptual meaning.

Some of the features, such as the photometry, temperature and luminosity, have direct physical meaning for stellar classification.

In contrast, some other features, such as distance or Galactic position also do have a physical meaning, but only indirectly in the context of classifying stars. Different types of star are unequally distributed across the Galaxy, both in reality (e.g., young stars are found most frequently near the Galactic Plane) and in our dataset (e.g., the Galactic Bulge and Magellanic Clouds have been comparatively well-sampled for variable stars by microlensing surveys). Thus, position actually has predictive value for classifying stars, and a good ML training algorithm should be able to exploit this. However, the predictive skill – the performance of the classifier measured by metrics such as accuracy and f1-score – that comes from those variables is not based on the physics of the star, and it is thus debatable whether one should actually use those variables, even when they can increase the skill of the classifier. We will refer to the stellar position, proper motion across the sky and distance as "bias features" throughout this paper. In practice, the answer to whether one should or should not use those bias variables will depend on for what exact use case the classifier shall be used. If the goal is to make the best possible guess of a star type, given all information available, then the bias variables should be used. However, if the task is to infer how much we can tell about the type of the star from photometric measurements, then the bias variables should be explicitly excluded.

We grouped the variables into four categories: variables referring to position ("bias variables"), variables describing photometric measurements ("spectra variables"), effective temperature and luminosity as output by PySSED ("base variables") and a final group containing physics variables that are not direct photometric measurements (namely: the metallicity and surface gravity PySSED adopts for the star, the interstellar extinction, and any measurement of spectroscopic temperature). The feature categories are summarized in Table 3. From these four groups, we form five different feature packages: "Bias", consisting only of the bias variables; "Base",

Table 3. Feature groups.

Feature Set	Features
Base Variables	Effective Temperature (K), Luminosity (LSun)
Bias Variables	Right Ascension (degrees), Declination (degrees), Proper Motion Right Ascension (mas/year), Proper Motion Declination (mas/year) Distance (pc) Distance
Spectra Variables	Transformed Spectra [85 features (Jy)]
Physics Variables	Base Variables, Spectra Variables, Other Physics [Metallicity (dex), Surface Gravity (dex), Extinction (mag), Spectroscopic Temperature (K)]

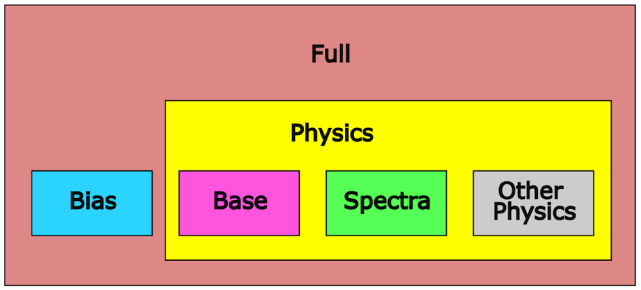


Figure 4. Overview of feature packages. There are four feature groups (Bias, Base, Spectra and Other Physics) and five feature packages built from them (Bias, Base, Spectra, Physics and Full).

consisting only of the two base variables; "Spectra", consisting only of the spectra variables; "Physics", consisting of the spectra plus the other physics variables; and "Full", consisting of all variables. This is visualized in Figure 4.

Machine-learning methods

The problem considered here is a supervised learning classification problem with continuous input features. Herein, models are tasked with either explicitly or implicitly learning some approximation of the sample data's class distribution. A common approach is to have models learn decision boundaries with the aim of separating the support sets of the class distributions, which ideally are aimed at generalizing from the sample to the population. In principle, there is a wide range of supervised learning methods for classification tasks available, e.g. logistic regression, deep neural networks, support vector machines. However, the high sparsity of the data strongly restricts which ML methods can be used. Crucially, it renders the naive approach of simply completely dropping all features with missing values, to generate a subset with no missing values impossible. As this is not possible in our case, there are two approaches left: (1) use data-imputation techniques, or (2) use ML techniques that can natively deal with missing values. Data imputation refers to techniques that fill up missing values with plausible values. There is a wide range of data imputation techniques in the literature[35]. Initial tests that we carried out with simple imputation techniques and subsequently training various ML models on them did not yield promising results. Data imputation can easily generate data that is not physically meaningful. Therefore, we settled on approach (2), using a supervised classification algorithm that can directly deal with missing data, more specifically the XGBoost algorithm[36]. XGBoost is a widely used tree-based supervised ML algorithm for multi-class classification tasks. It is especially well-suited to tabular data, where it usually outperforms deep-learning algorithms[37], and can natively deal with missing values. We thus opted to solely look at XGBoost models in detail for our problem. XGBoost – like most other ML training algorithms – has a large number of so-called hyperparameters, in addition to the parameters that are learned by fitting on the training data. These need to chosen in advance, before commencing training. Hyperparameters of XGBoost are, amongst others, the learning rate, and the maximum depth of each tree. The most common approach to choosing suitable hyperparameters is to perform systematic model tuning. For this, the model is trained with different hyperparameters on the training set, and then evaluated on the validation set, and subsequently the hyperparameter configuration that yielded the best performance on the validation set is used. Alternatively, as we have done, 5-Fold Cross Validation (CV) can be used to more accurately estimate algorithm performance, in place of a classical validation set.

The tuning procedure was performed individually for each of the five feature packages, resulting in five hyperparameter configurations. For tuning, we used Bayesian optimization as implemented in the Optuna Python library[38], utilizing the Tree-Structured Parzen Estimator (TPE)[39] for parameter search coupled with median pruning to limit the search space. The tuning grid – that is, the hyperparameter space that is explored in the tuning process – is shown in Table 4. The introduction and tuning of nine class weight parameters, to influence sample weights, was explicitly added to counteract the high class imbalance in the data. The class weights were additively combined per multi-label sample and used to directly influence the gradient computation in each boosting step, by scaling the sample's impact on the loss function. The resulting hyperparameter configurations are contained in the code repository accompanying this paper.

Evaluation metrics

Evaluating the predictions of a multi-class, multi-label classifier on highly imbalanced data is not trivial, and care has to be taken to choose the right evaluation metrics. Uncorrected accuracy (the fraction of correctly classified stars) can be highly misleading for unbalanced datasets, and can give incorrect impressions of good predictions. For multi-label predictions – where one sample can have multiple labels simultaneously – an additional complication is that it is not trivial to decide which predictions should be counted as correct and which not. E.g., if a star that is both a binary and a main-sequence star, but the classification model only predicts binary - should we count it as a correct or an incorrect prediction?

Table 4. Grid for Hyperparameter Tuning XGBoost via Baysian Optimization.

Hyperparameter	High	Low	Step Size
Class Weights 0–8	1.0	0.001	0.001
min_child_weight	20	0	2
gamma	10.0	0.0	0.2
subsample	1.0	0.2	0.1
colsample_bytree	1.0	0.3	0.1

Some standard choices we choose to visualize are the raw accuracy and F1 score. Of these demonstrated metrics, accuracy is included purely for completeness sake, as we deem it inadequate in representing the performance on a multi-label dataset with extreme class imbalance. After all, the standard definition of accuracy requires a perfect match, i.e., even if three out of four star types were predicted correctly for a sample, this would result in a classification score of 0 for that sample, masking the fact that the model prediction was almost completely correct. Depending on problem context this can be a highly significant issue. Furthermore, with extreme class imbalance, as shown in Figure 3a, a very high accuracy of about 88.38 can be achieved whilst predicting only the three majority classes correctly, hiding the fact that six out of nine stellar classes would never be predicted correctly.

$$\text{Accuracy} = \sum_{n=1}^{number_of_samples} \frac{1(\text{predicted_label} == \text{true_label})}{\text{number_of_samples}} \quad (3)$$

A common way to evaluate highly imbalanced classification tasks is to use Precision and Recall, which are defined via True Positives (TPs), False Positives (FPs) and False Negatives (FNs):

$$\text{Precision} = \frac{\text{TPs}}{\text{TPs} + \text{FPs}}, \quad (4)$$

$$\text{Recall} = \frac{\text{TPs}}{\text{TPs} + \text{FNs}} \quad (5)$$

and their harmonic mean, the F1-score

$$\text{F1} = 2 \times \frac{\text{recall} \times \text{precision}}{\text{recall} + \text{precision}}. \quad (6)$$

The precision score evaluates, in what percentage of cases, the algorithm's predicted class was truly the correct class. Recall, on the other hand, measures the percentage of stars of a given class that are not predicted as that class. In other words, a bad recall value indicates that the model tends to not detect the class label when confronted with a sample of that type. The F1 score is the harmonic mean of precision and recall.

The three scores are, however, defined for binary classification problems (problems where an object can be only one of two classes). Extending this to multi-class single-label classification – where there are multiple classes, but a star can only belong to one single class – can be done by computing the three scores separately for each class. In our case, we would thus get precision, recall and F1 for binary stars, precision, recall and F1 for main sequence stars, and so on. However, in our case, a single star can belong to multiple classes, and TP, FP and FN are thus ambiguous and need to be defined in more detail. For this, two possible approaches exist:

1. a perfect matching is performed, where every class a star belongs to must be predicted correctly. Thus, if a star belongs to one single class, then a TP means that only that class, and no other additional class was predicted.

2. a partial matching is calculated, where every class a star belongs to is seen as a separate entity. For example, if a single test star is both binary and main-sequence, but is predicted as both binary and low-mass, it is a TP for precision, recall and F1 for binary stars, a FP for precision, recall and F1 for low-mass stars, and a FN for precision, recall and F1 for main sequence stars.

   In this paper, we have chosen option (2).

As mentioned, the F1 score is defined on a by-class basis, meaning that in our case we get nine F1 scores, one for each stellar class. There are several ways of extending the F1 score in order to get a single value over all classes.

The simplest way is the **micro**-averaged F1 score. Here, all TPs, TNs and FNs over all classes are aggregated, and the F1 score computed from these aggregated values. The micro-averaged F1 score suffers from similar deficiencies in regards to class imbalance as the raw accuracy metric. Class labels are disregarded and the balance between precision and recall is evaluated for the entire test set.

Macro-averaging, on the other hand, calculates the F1 score for each class and averages the results via arithmetic mean. This is our benchmark metric of choice, as the metric is harsh on misclassification of individual classes. In other words, each class contributes equally to the final score.

Weighted averaging is based on a similar notion, however applies weights to the class-specific F1 scores, equal to their number of instances, prior to averaging. Thereby the imbalance is further exaggerated, making the metric less telling, for datasets of high class imbalance, compared to the macro approach, where all classes are equally important.

Finally, the **samples** average applies the unweighted arithmetic mean to the F1 score per sample in the test dataset. This is advantageous when dealing with a high amount of multi-label instances, however when the multi-label count is low,

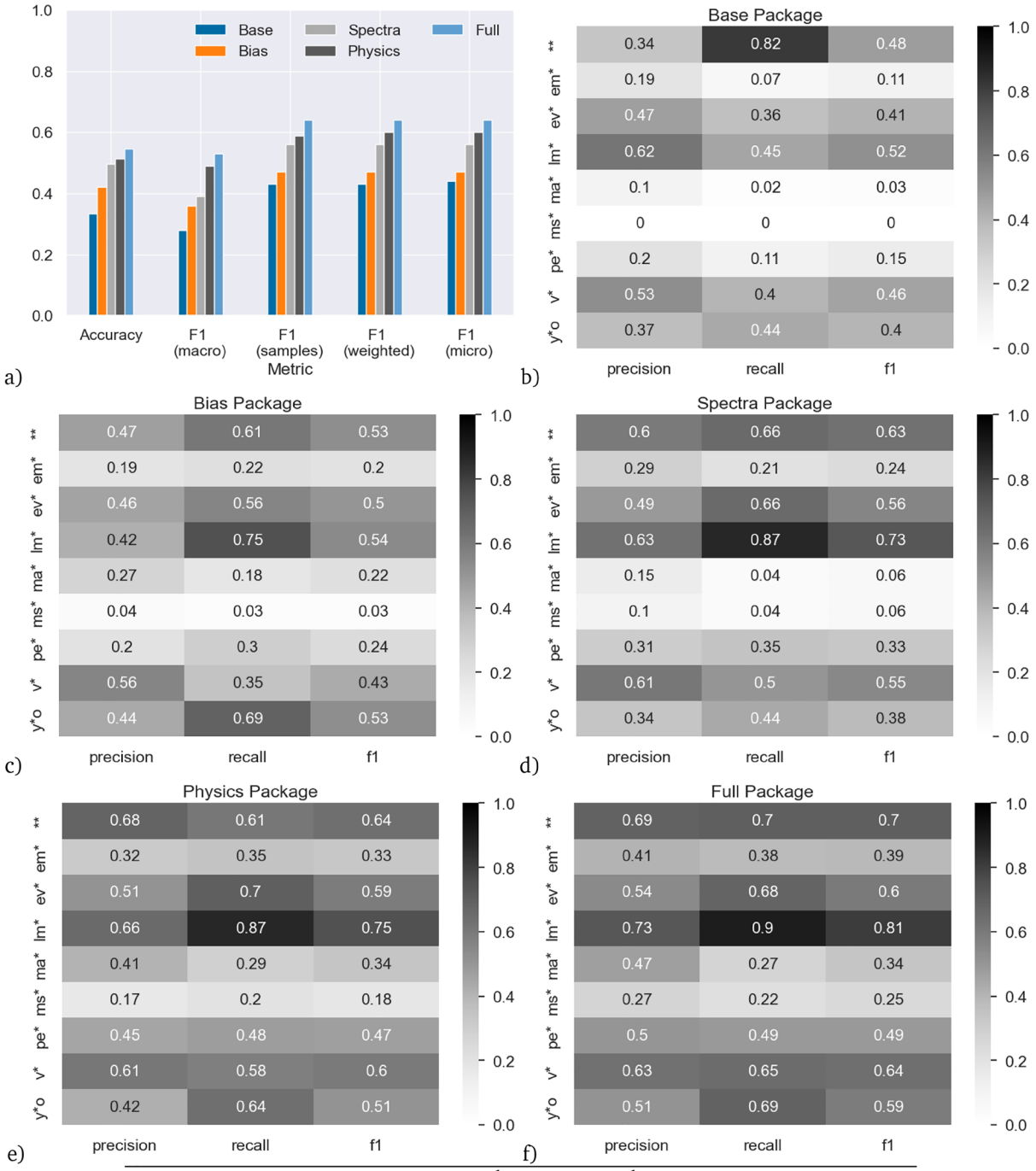
it will be close to the accuracy score we initially disregarded, suffering in part from similar drawbacks.

To get a complete picture, in this paper we use all four versions of the F1 score in parallel.

Results

"Main" dataset

We start by analyzing the skill of the classifier trained on the main dataset. The results are shown in Figure 5. Panel (a) shows the results aggregated over all stellar classes, with the metrics



Test set sample counts per class.

	**	em*	ev*	lm*	ma*	ms*	pe*	v*	y*o
Samples	5345	467	6345	1139	100	198	460	9410	564

Figure 5. Accuracy on the main (random 10%) dataset for models trained with the five different feature packages. Panel (**a**) shows the results aggregated over all star types. Panels (**b**–**f**) show the results for individual star types.

as described above in the "Metrics" section. While the absolute values of the metrics are somewhat different, there is a clear and consistent ordering in the skill of the models in terms of which feature package they use, independent of which metric is considered. The Base model – the one only using the effective temperature and luminosity from PySSED as predictors – has lowest skill, while the Full model has highest skill. The Physics model is second-best, followed by the Spectra model and the Bias model. That the Full model (the one using all available features after preprocessing) has highest skill is not surprising. The same is true for the fact that the Physics model is better than the Base model, as the Base features are a subset of the Physics features. The rest of the ordering of the models does however provide some valuable insights. That the Bias model provides higher skill than the Base model may seem intriguing, as there is no physical meaning in the Bias features; however, as discussed in the Introduction, it is debatable whether the predictive value of the Bias features should actually be used, as it represents a combination of the (real) Galactic distribution of stars and their (artifical) inhomogeneous sampling in the dataset. With the same reasoning, the skill of the Full model needs to be treated with caution, as the only difference between the Full and the Physics model is the inclusion of the Bias features. Whether this skill increase is actually valid will depend on whether one considers the predictive value of the Bias features valid, which in turn will depend on the use case the model would be used for.

More promisingly, the Spectra model is clearly better than the Base model. The Spectra model does not directly use the Base features, and yet outperforms the Base model based only on the photometry contained in the SED. This shows that the ML model is able to extract more information from the SED than what is simply encompassed in the estimation of effective temperature and luminosity alone, relying instead on an excess or deficit of flux at specific wavelengths when compared to the model. This shows the value obtained by using photometric data as direct input to the ML model.

Up to now, we only discussed prediction skill aggregated over all star types. Figure 5b-f show precision, recall and F1-score (the harmonic mean of precision and recall), split up by feature package and star type.

The skill of the classifiers greatly varies between different types of stars. For example, in terms of F1 score, the Base model works best for double stars (**), low-mass stars (lm*) and variable stars, (v*) but has near zero skill for main-sequence stars (ms*).

At the opposite end of feature inclusion, the Full model has quite high skill for low-mass stars (lm*, F1 score 0.81, recall 0.9), but the skill remains low for main-sequence stars and massive stars (ma*, $F1 \leq 0.34$).

Considering the individual classes, the poor precision and recall for the main-sequence stars is surprising, since the main sequence is defined on the Hertzsprung–Russell (H-R) diagram on which the Base model is based. This likely indicates poor training due to the lack of main-sequence stars classified as such at SIMBAD.

By comparison, low-mass stars (which are better populated at SIMBAD) invariably occupy the cooler, fainter parts of the H–R diagram, thus it is natural that the Base model should pick these up with comparatively high precision and recall. By contrast, massive stars are well-populated at SIMBAD (despite being rarer, they are intrinsically brighter, so easier to identify). They should invariably occupy the brighter and normally warmer parts of the H–R diagram, so we might expect the Base model to identify these well, yet it spectacularly fails to do so: including more features improves the precision and recall slightly, but they remain the class with the poorest skill. The reasons for this remain unclear.

Young stellar objects (y*o) also occupy a specific part of the H-R diagram, in the sparsely occupied region with cool temperatures but moderate luminosities. The Base package scores for this class are only moderately effective and the Spectra package makes little improvement. The Bias package returns a much better recall, likely due to the fact that most of these stars are found in an narrow strip in the plane of our Galaxy, and a few star-forming regions just away from the plane. This feeds into a strong recall and moderately high precision in the Full package. The precision may be impacted by the overlap between young stellar objects and variable stars.

Evolved stars (ev*) similarly occupy the cool, luminous part of the H–R diagram. Consequently, it is surprising that the Base package does not fit these with higher recall. The inclusion of the Spectra package is more effective at identifying them, possibly because they tend to show excess flux at infrared wavelengths. The low precision may again be impacted by the overlap between evolved and variable stars.

Considering the other well-fit classes, only some classes of double stars and variable stars should be identifiable on the H-R diagram, and then with low confidence. Consequently, it is more surprising that the Base combination of temperature and luminosity is able to return these well. Therefore, the high recall (82%) for double stars is particularly surprising, and it is interesting to see that this recall score decreases when more features are added.

Chemically peculiar stars (pe*) and emission-line stars (em*) have lower skill in the Base package. These kinds of stars do preferentially occupy parts of the H–R diagram, but are very heterogeneous classes that cover a wide range of physical characteristics, and normally require detailed spectral observation to classify. This breadth of properties translates into a generally poor fit across the different feature packages, as there are multiple potential characteristics in feature space that a ML analysis could declare significant, while other valid parts of feature space could be excluded.

"Extended" dataset

We now proceed to our second dataset, which extends the main dataset with an additional 80 000 main-sequence stars from random fields on the sky. We expect the addition of more main-sequence stars, the most-populous type of stars in our Galaxy but the least sampled in SIMBAD, to improve the skill of our classifier. We only chose to extend this class, due to the abundance of main sequence star data which is largely left unlabelled due to being sufficiently common.

The results are shown in Figure 6. Consistently across all feature packages and for all unified metrics, the models trained and tested on the extended data have higher skill than their counterparts on the main dataset (Figure 6a); the increase in accuracy is of order 0.1-0.2, which is an increase of 30–50% with respect to the accuracy on the main dataset.

When considering main-sequence stars as an individual class, the differences are particularly stark: the recall on the Base model increases from 0.00 in the main dataset to 0.97 in the extended dataset, indicating more main-sequence stars are being correctly identified. This appears to be mostly at the expense of correctly identifying double stars, where the recall decreases from 0.82 to 0.00, highlighting the overlap in these two types of objects on the H–R diagram. Including the SED fits in the Spectra package improves the fitting of double stars again so that both double and main-sequence stars receive a healthy F1 score in all packages including them (0.63–0.71 and 0.90–0.94, respectively). This result shows that visible traits in the SED can be exploited by the classifier, as the SIMBAD database is dominated by spectroscopic, eclipsing and various types of interacting binaries, rather than component separation in visual binaries.

Overall, the addition of the main-sequence stars does not have a significant negative impact on the F1 scores of any other class, with the exception of some classes in the Base package. In general, the recall remains similar for the other classes, indicating few stars from other classes are now being misclassified as the majority main-sequence class. The precision scores generally show an increase too, particularly for the massive-star class, indicating fewer stars in the main-sequence class are now being misclassified as well, and it is this increase in precision for massive stars, and the increase in both precision and recall for main-sequence stars, that are main drivers of the increased F1 scores for the extended dataset. This shows that this classification problem is still being limited by a lack of adequate training data for specific sub-classes of objects.

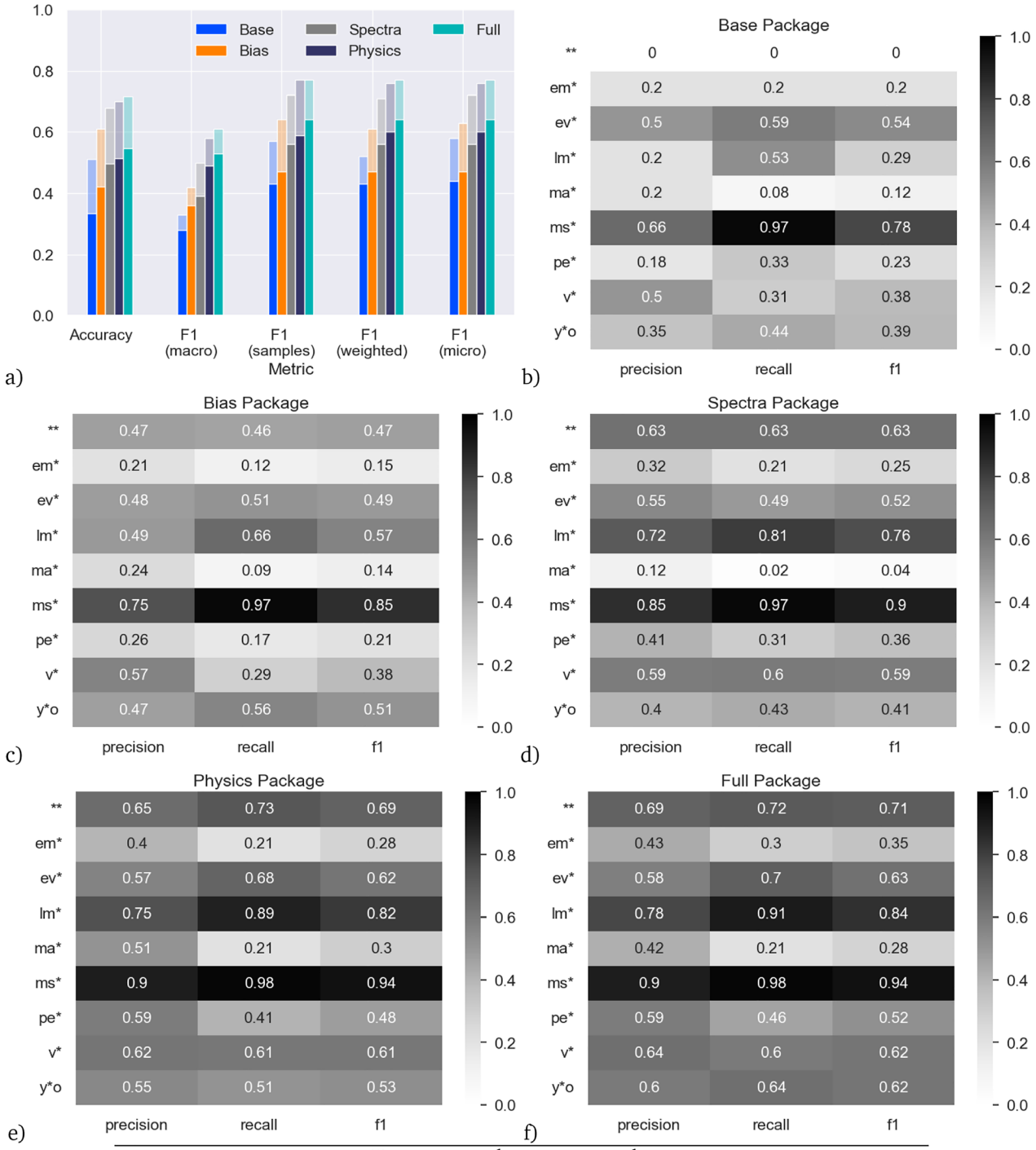
Inspired by this result, we designed a final experiment to get more insights on how decreasing or increasing the number of training samples impacts the performance of the classifier. For this, we started with the main dataset and systematically reduced the training samples in 5% steps individually for each star type. For example, we use the main dataset but with only 95%, 90%, 85%, ..., 0% of variable stars. We repeat this process for each star type in turn, resulting in 171 datasets across all types. For each dataset, we train the Full feature package model and evaluate it on the test dataset, which remains unchanged. To avoid impacting the other classes, if a star received two labels (e.g., evolved and variable star), we only removed the entry from the class undergoing reduction in that dataset.

If we were to compute the macro F1 score for each of those experiments, we would conflate two different effects: (1) the change of skill on the star type whose samples are reduced and (2) the effect on overall skill if fewer samples are available. In order to disentangle those effects, we use two different F1 scores: first, one time we compute the F1 score solely of the star type that is systematically reduced; second, we use the macro F1 score over the other eight star types. These are respectively shown in Figure 7, panels (a) and (b). Panel (a) shows the clear (and unsurprising) result that, for all star types, the skill is reduced as fewer training samples are included, finally reaching zero. However, the rate of this decay could not have been known a-priori without this analysis. This rate is informative: if the curve is very flat, then this indicates that the training algorithm has reached a plateau - in which case including more samples would likely not improve performance further. On the other hand, if performance in the experiment is gradually increasing with number of included samples, then this indicates that including more samples might likely lead to a further increase in performance.

We see that effect of removing samples very much depends on the type of star: the relatively flat curves for low-mass stars and double stars indicate that we are likely at the maximum F1 score achievable for these stellar types, while the F1 score for the other types is still increasing towards the right of the graph. Due to the computational time required, this training set only contains 10% of the stars classified in SIMBAD, however these results suggest that higher F1 scores could be obtained by analysing an increased number of objects from the SIMBAD database, and especially by including more objects from these other classes.

Gradually removing a class in this fashion seems to have little effect on the skill of the other star types (panel b). One possible exception are variable stars. Here the results suggest that reducing their number in the training set actually slightly *increases* the skill on the other star types. This is not surprising, as variable stars tend to overlap with other star types. The skill increase may suggest that if the ML algorithm learns to identify them then the algorithm may consequently also mislabel similar star types as variable stars, presuming guilt by association. An explainable AI study would be needed to confirm this suggestion.

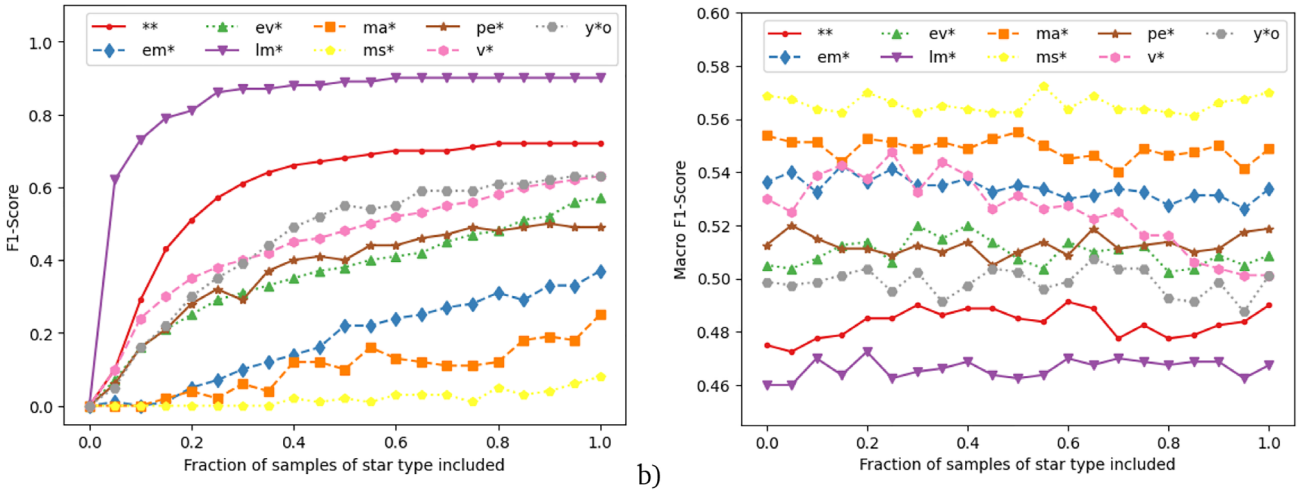
The result that decreasing one star type in the training has little impact on skill for the other star types is in congruence from the results of the experiment with the extended data,



Test set sample counts per class.

	**	em*	ev*	lm*	ma*	ms*	pe*	v*	y*o
Samples	5300	464	6484	1168	102	17028	466	9438	570

Figure 6. Accuracy on the extended dataset (main dataset plus additional main-sequence stars) and comparison to main dataset, for models trained with the five different feature packages. Panel **a**) shows the results aggregated over all star types. The solid bars are the skill with the models trained and tested on the main dataset (same as Figure 5a), the shaded bars are the skill of the models trained and tested on the extended dataset. Panels (**b**–**f**) show the results for individual star types with the extended dataset.



a) b)

Figure 7. Results from experiment systematically reducing the training samples of individual star types. a) F1 score of the star type systematically reduced, **b**) macro-averaged F1 score over all star types except the systematically reduced one. Details see text.

where the additional main-sequence stars greatly increased the skill on the main-sequence stars themselves, but not for the other star types. The overall increase in accuracy for that experiment stems from the fact that in the extended data, main-sequence stars are the majority class and thus have a large impact on overall skill.

Finally, as means of better understanding the final classifier's prediction behaviour, specifically pertaining to the treatment of missing values in XGBoost, we analysed the SHAP importance values for our test dataset.

We utilized the python library SHAP created by Lundberg and Lee[40], to extract approximations of Shapley values from coalitional game theory, which measure the contribution of each feature on the final model's prediction. These local explainability measures can in turn be aggregated across samples, giving rise to a global explainability metric, for which the results on the test set with the full feature package can be seen in Figure 8. Several findings arise from this plot, most notably in regards to the high amount of missing values in the dataset, that XGBoost seems to quite successfully learn the pattern of missingness, supporting the notion that the missing values are indeed "missing not at random". A common assumption is that more densely packed features contain more information, which at first glance may seem to be the case as 17 out of the top 20 most abundant features are contained in this list, however the most relevant feature to the model is the WISE.W4 band.

From a data-centric perspective, this is highly surprising as it is among the less densely packed features available. This implies that XGBoost, indeed learns the missing value pattern, and in turn utilizes such realizations in the final model's predictions. An example of this can be seen in the class most impacted by this band, which upon closer analysis of the local explainability, reveals that out of the top 50% most impacted main-sequence stars, 95% were missing values. Similarly, the top 5% of samples impacted most by this band, had over 92% missingness. This leads us to two significant data-centric takeaways, namely that missing value patterns are truly learned to act as discriminative features by the model's default pathway choices and that, as is to be expected, the impact of bands and features varies between star types.

From an astrophysical perspective, this is marginally less surprising. Temperature and luminosity (and, by implication, $\log(g)$, are some of the primary parameters affected by properties such as stellar mass and evolutionary stage, so are very effective discriminants of stellar class to human classifiers. Similarly, distribution on the sky in RA and Dec can be used as a secondary classifier e.g., young stellar objects will always cluster in star-forming regions and (together with massive stars) will be found almost uniquely in the plane of the Milky Way; low-mass stars will not be found at large distances, etc. The high importance of mid-infrared data, particularly from the *WISE* satellite, may be attributable to the presence or absence of excess infrared flux, as found among young stellar objects, evolved stars, and some variable and massive stars. *WISE* fluxes in excess of the fitted stellar atmosphere model can act as discriminators to positively identify these classes, hence also as evidence against the other classes. This extends to presence or absence of these data which, especially for *WISE* band 4, are limited by sensitivity absence of *WISE* observations appears to be being used by the classifier as a proxy for lack of excess infrared flux.

Conclusions

In this work, we have trained machine-learning classifiers for classifying stars. Photometric data from VizieR and higher-level

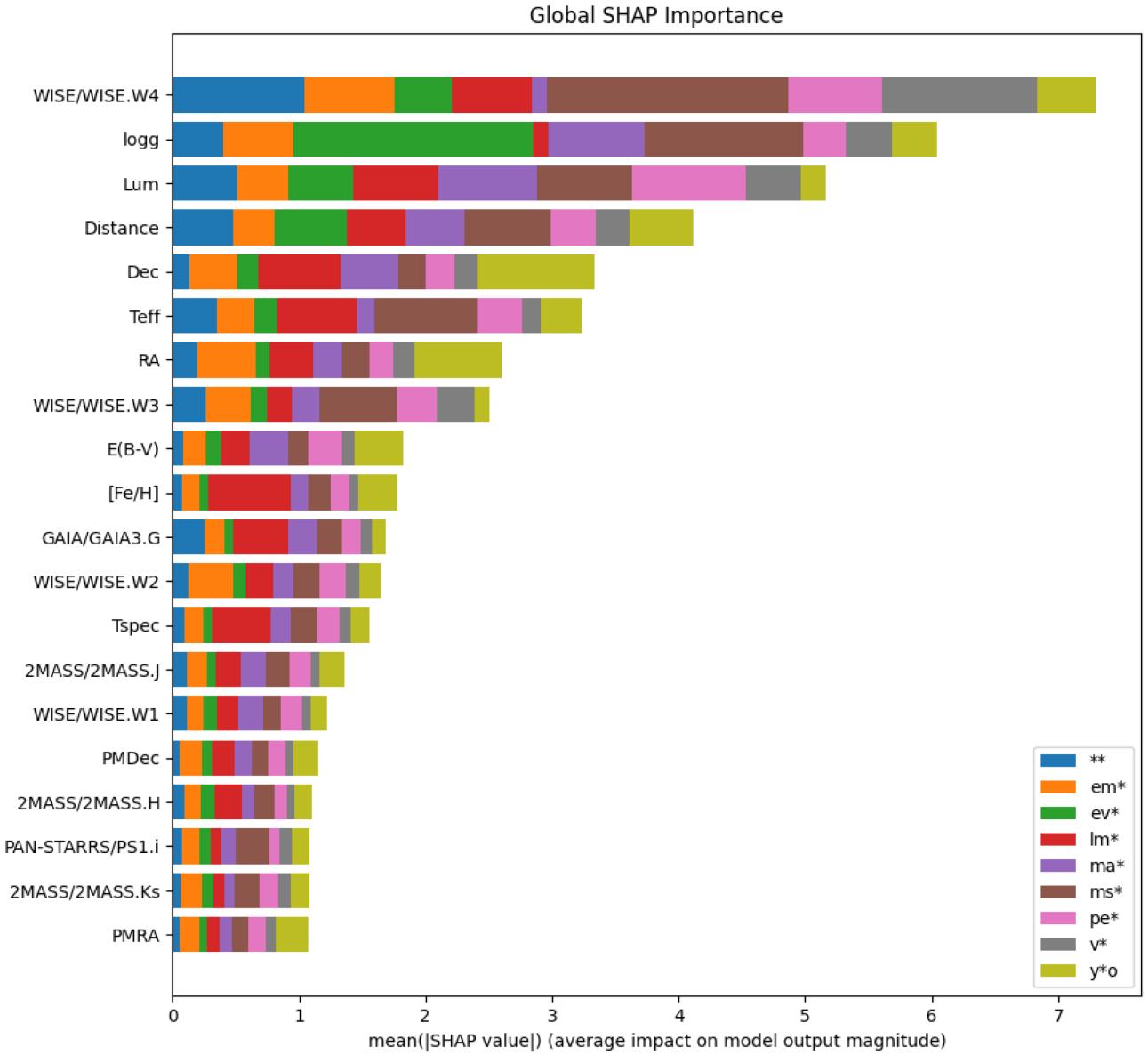


Figure 8. Top 20 feautures of the full feature set ranked by their mean absolute SHAP importance values.

data products from the PySSED spectral-energy-distribution fitting algorithm were used as inputs, for a training dataset comprising 10% of stars classified in SIMBAD as belonging to one or more of nine categories of stars. The resulting classifier has an accuracy of ~0.5 when using all available input variables. When adding additional 80 000 main-sequence stars to combat the severe class imbalance in the training data, the accuracy increased to ~0.7. In additional experiments, we disentangled the contribution of individual groups of input variables. Finally, we conducted an experiment with systematically decreasing the numbers of individual star types, and found the prevalence of a particular star type in the training set greatly influences the skill of the model on that particular star type, but has little to no effect on the skill for the other star types.

This work is a first attempt to use supervised ML techniques for stellar classification on stars from the SIMBAD database. Key challenges are the high sparsity of the data – meaning there are many missing values – and the high class imbalance, as the number of available samples varies massively between the different star types. While the skill of the best model is

not yet high enough to be usable as a reliable tool in practice, this work shows the principal feasibility of using ML for such a problem.

We suggest several avenues for further work: (1) Using more data on under-represented star types. Our results suggest that adding more data can increase the predictive skill at least for some star types. (2) Using a different taxonomy of star types. Several of the classes we adopted from SIMBAD are very heterogeneous (e.g., peculiar stars, emission line stars, variable stars and main-sequence stars). A larger number of taxonomic classes that are more cleanly separated in their physical properties may both reduce the problems of multi-label classification and provide greater utility to the astrophysical community. (3) Developing and using physics-informed methods for data imputation. In our work we solely used XGBoost as ML algorithm because it can natively deal with missing data. Initial tests with simple data-imputation methods were unconvincing. However, it would be interesting to develop data imputation methods specialized for our use case. Specifically, methods that incorporate *a priori* physical knowledge into statistical and ML methods might be promising, i.e., physics-informed methods[41]. (4) Lastly, we chose these 5 distinct feature packages with the goal of evaluating the bias in the data and the importance of the SED fitting procedure, however believe that this experiment can be further extended to the generation of different, more numerous feature packages, examining the impact of various variables, such as for example a physics feature set without the Base variables (Temperature and Luminosity). Finding a good way of filling up the large number of gaps in stellar SED data, e.g., by use of the stellar atmosphere models used in PySSED, would remove one of our main challenges, namely the high sparsity of the data. Not only is non-sparse data often easier to separate on a fundamental level, in practice it also allows using a much wider variety of ML techniques.

Appendix

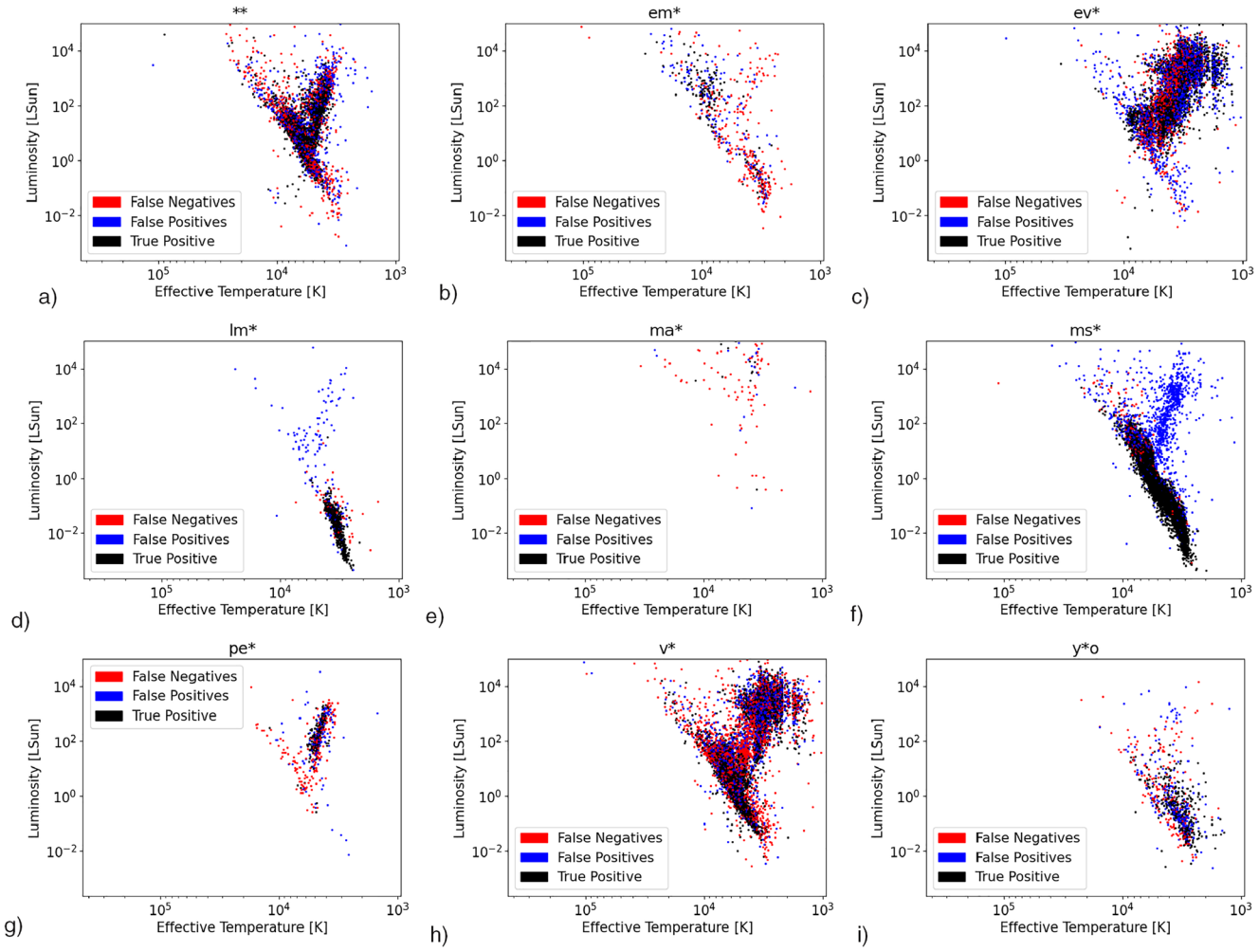


Figure 9. Extended Dataset Classification Results Visualised in HR Diagrams.

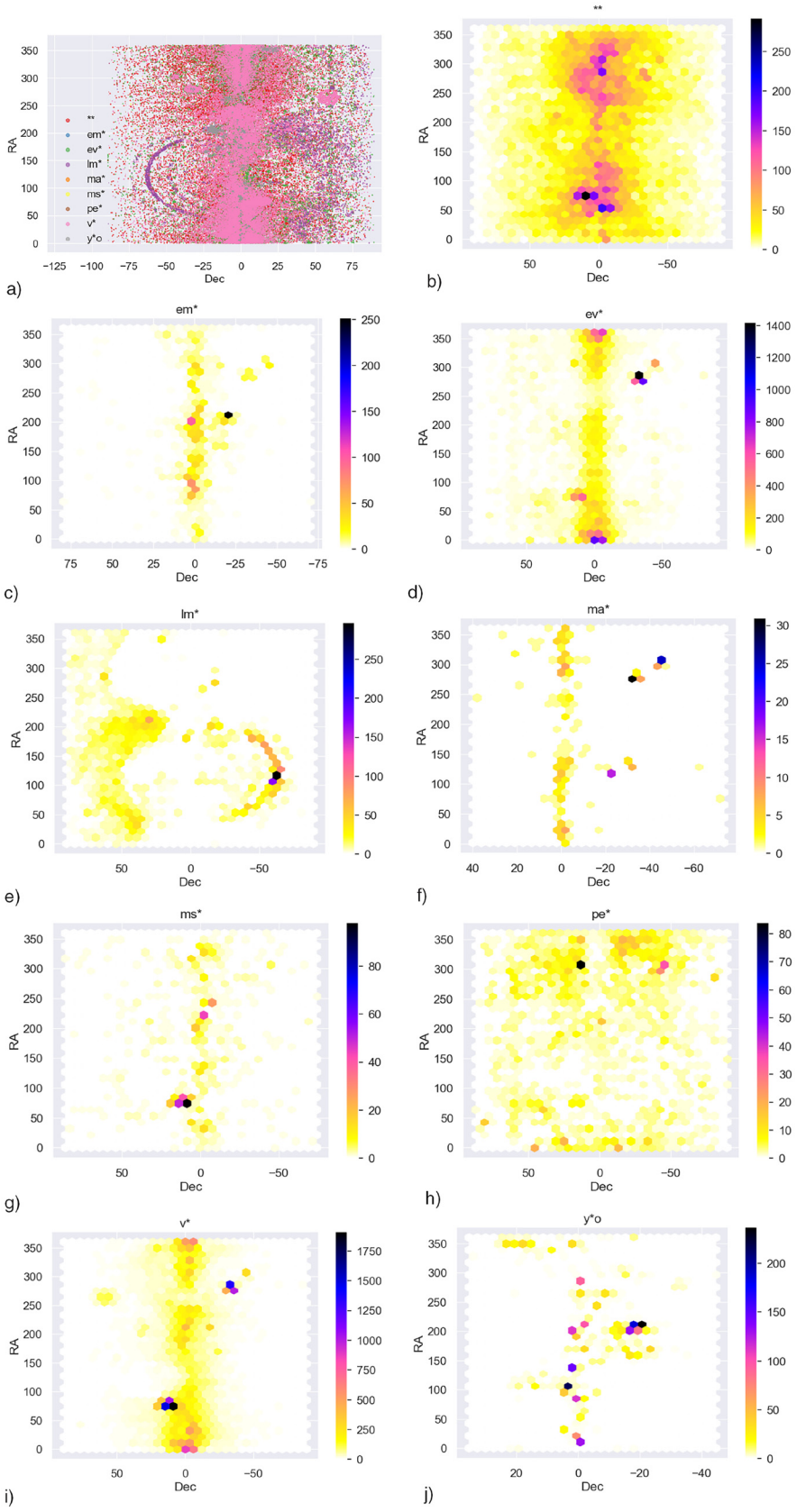


Figure 10. a) star locations of the main dataset **b**)–**j**) Density plots of individual star locations in the main dataset.

Data and software availability

Source data

Source data for this work comes from the VizieR database (https://vizier.cds.unistra.fr/viz-bin/VizieR) and is analysed using the PySSED pipeline[11]. Classification data comes from the SIMBAD database (https://simbad.u-strasbg.fr/simbad/).

Underlying data

The data used in this study is a subset of the SIMBAD dataset, extended with additional data from the PySSED Software. The complete dataset used for this study can be accessed at https://zenodo.org/records/10377253.

The data is stored as .csv files.

This project contains the following underlying data:

- 10percent_full.dat. (Main Dataset Data)
- 10percent_labels.txt. (Main Dataset Labels)
- df_plus_ms.dat. (Extended Dataset Data)
- classes_plus_ms.dat (Extended Dataset Labels)

Data are available under the terms of the Creative Commons Attribution 4.0 International data waiver.

Software availability

- Stellar Classification Code (https://github.com/explore-platform/s-phot_stellar_classifier), Archived at the time of publication: https://doi.org/10.5281/zenodo.10377253.

  Licensed under Creative Commons Attribution 4.0 International

- PySSED Code v0.3 (https://github.com/explore-platform/pyssed_explore/tree/v0.3)

Acknowledgements

We thank Roman Kern, Vedran Sabol and Manuela Rauch for their inputs and discussions.

References

1. Morgan WW, Keenan PC: **Spectral classification.** *Ann Rev Astron Astrophys.* 1973; **11**(1): 29–50.
Publisher Full Text
2. Stromgren B: **Spectral classification through photo-electric narrow-band photometry.** *Ann Rev Astron Astrophys.* 1966; **4**(1): 433–473.
Publisher Full Text
3. Fitzpatrick EL: **Correcting for the effects of interstellar extinction.** *Pub Astron Soc Pac.* 1999; **111**(755): 63.
Publisher Full Text
4. Hertzsprung E: **Über die sterne der unterabteilungen c und ac nach der spektralklassifikation von antonia c. maury.** *Astron Nachir.* 1909; **179**(24): 373.
Publisher Full Text
5. Russell HN: **Relations between the spectra and other characteristics of the stars.** *Pop Astron.* 1914; **22**: 275–294.
Reference Source
6. Woods PM, Oliveira JM, Kemper F, *et al.*: **The SAGE-spec *spitzer* legacy programme: the life-cycle of dust and gas in the large magellanic cloud–point source classification I.** *Mon Not Roy Astr Soc.* 2011; **411**(3): 1597–1627.
Publisher Full Text
7. Adep RN, shetty A, Ramesh H: **Exhype: a tool for mineral classification using hyperspectral data.** *ISPRS J Photogramm Remote Sensing.* 2017; **124**: 106–118.
Publisher Full Text
8. Peyghambari S, Zhang Y: **Hyperspectral remote sensing in lithological mapping, mineral exploration, and environmental geology: an updated review.** *J Appl Remote Sensing.* 2021; **15**(3): 031501.
Publisher Full Text
9. Blanck D, Schön A, Mamede AS, *et al.*: ***In situ* Raman spectroscopy evidence of an accessible phase potentially involved in the enhanced activity of la-deficient lanthanum orthoferrite in 3–way catalysis (twc).** *Catal Today.* 2017; **283**: 151–157. Operando V - 5th International Conference on Operando Spectroscopy, Deauville (FR) May 2015.
Publisher Full Text
10. Vallenari A, Brown AGA, Prusti T, *et al.*: ***Gaia* data release 3–summary of the content and survey properties.** *Astron Astrophys.* 2023; **674**: A1.
Publisher Full Text
11. McDonald I, Zijlstra AA, Cox NLJ, *et al.*: **PySSED: an automated method of collating and fitting stellar spectral energy distributions.** *RAS Tech Instrum.* 2024; **3**(1): 89–107.
Publisher Full Text
12. Wenger M, Ochsenbein F, Egret D, *et al.*: **The SIMBAD astronomical database. The CDS reference database for astronomical objects.** *Astron Astrophys Suppl.* 2000; **143**(1): 9–22.
Publisher Full Text
13. Gabruseva T, Zlobin S, Wang P: **Photometric light curves classification with machine learning.** *J Astron Instru.* 2020; **09**(01): 2050005.
Publisher Full Text
14. Clarke AO, Scaife AMM, Greenhalgh R, *et al.*: **Identifying galaxies, quasars, and stars with machine learning: a new catalogue of classifications for 111 million sdss sources without spectra.** *Astron Astrophys.* 2020; **639**: A84.
Publisher Full Text
15. Cunha PAC, Humphrey A: **Photometric redshift-aided classification using ensemble learning.** *Astron Astrophys.* 2022; **666**: A87.
Publisher Full Text
16. Chaini S, Bagul A, Deshpande A, *et al.*: **Photometric identification of compact galaxies, stars, and quasars using multiple neural networks.** *Mon Notices Royal Astron Soc.* 2023; **518**(2): 3123–3136.
Publisher Full Text
17. Zeraatgari FZ, Hafezianzadeh F, Zhang Y, *et al.*: **Machine learning-based photometric classification of galaxies, quasars, Emission-Line Galaxies, and stars.** *Mon Notices Royal Astron Soc.* 2024; **527**(3): 4677–4689.
Publisher Full Text
18. Awang Iskandar DNF, Zijlstra AA, McDonald I, *et al.*: **Classification of planetary nebulae through Deep Transfer Learning.** *Galaxies.* 2020; **8**(4): 88.
Publisher Full Text
19. Jia Y, Guo S, Zhu C, *et al.*: **Identifying symbiotic stars with machine learning.** *Res Astron Astrophys.* 2023; **23**(10): 105012.
Publisher Full Text
20. Naul B, Bloom JS, Pérez F, *et al.*: **A recurrent neural network for classification of unevenly sampled variable stars.** *Nature Astron.* 2018; **2**(2): 151–155.
Publisher Full Text
21. Pantoja R, Catelan M, Pichara K, *et al.*: **Semi-supervised classification and clustering analysis for variable stars.** *Mon Not R Astron Soc.* 2022; **517**(3): 3660–3681.
Publisher Full Text
22. Hayes JJC, Kerins E, Awiphan S, *et al.*: **Optimizing exoplanet atmosphere retrieval using unsupervised machine-learning classification.** *Mon Not R Astron Soc.* 2020; **494**(3): 4492–4508.
Publisher Full Text
23. Russakovsky O, Deng J, Su H, *et al.*: **ImageNet large scale visual recognition challenge.** *Int J Comput Vis.* 2015; **115**(3): 211–252.
Publisher Full Text
24. Wesson R, Matsuura M, Zijlstra AA, *et al.*: ***JWST* observations of the Ring Nebula (NGC 6720): I. Imaging of the rings, globules, and arcs.** *Mon Not R Astron Soc.* 2024; **528**(2): 3392–3416.
Publisher Full Text
25. Chen X, Liu Z, Han Z: **Binary stars in the new millennium.** *Prog Part Nucl Phys.* 2023; **134**(1–2): 104083.
Publisher Full Text

26. Kogure T, Leung KC: **The astrophysics of emission-line stars**. Springer Science & Business Media, 2010; **342**.
Reference Source

27. Herwig F: **Evolution of asymptotic giant branch stars.** *Annu Rev Astron Astrophys.* 2005; **43**: 435–479.
Publisher Full Text

28. Luhman KL: **The formation and early evolution of low-mass stars and brown dwarfs.** *Annu Rev Astron Astrophys.* 2012; **50**: 65–106.
Publisher Full Text

29. Motte F, Bontemps S, Louvet F: **High-mass star and massive cluster formation in the milky way.** *Annu Rev Astron Astrophys.* 2018; **56**: 41–82.
Publisher Full Text

30. Eldridge JJ, Stanway ER: **New insights into the evolution of massive stars and their effects on our understanding of early galaxies.** *Annu Rev Astron Astrophys.* 2022; **60**: 455–494.
Publisher Full Text

31. Schwarzschild M: **Structure and evolution of stars**. Princeton University Press, 2015; **2379**.
Publisher Full Text

32. Preston GW: **The chemically peculiar stars of the upper main sequence.** *Annu Rev Astron Astrophys.* 1974; **12**(1): 257–277.
Publisher Full Text

33. Conroy C, Strader J, van Dokkum P, *et al.*: **A complete census of luminous stellar variability on day to decade timescales.** *Astrophys J.* 2018; **864**(2): 111.
Publisher Full Text

34. Koenig XP, Leisawitz DT: **A classification scheme for young stellar objects using the *Wide-field Infrared Survey Explorer* allwise catalog: revealing low-density star formation in the outer galaxy.** *Astrophys J.* 2014; **791**(2): 131.
Publisher Full Text

35. Lin WC, Tsai CF: **Missing value imputation: a review and analysis of the literature (2006-2017).** *Artif Intell Rev.* 2020; **53**(2): 1487–1509.
Publisher Full Text

36. Chen T, Guestrin C: **Xgboost: a scalable tree boosting system.** In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016; 785–794.
Publisher Full Text

37. Shwartz-Ziv R, Armon A: **Tabular data: deep learning is not all you need.** *Inform Fusion.* 2022; **81**: 84–90.
Publisher Full Text

38. Akiba T, Sano S, Yanase T, *et al.*: **Optuna: a next-generation hyperparameter optimization framework.** In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019.
Publisher Full Text

39. Bergstra J, Bardenet R, Bengio Y, *et al.*: **Algorithms for hyper-parameter optimization.** *Adv Neural Inf Process Syst.* 2011; **24**: 2623–2631.
Reference Source

40. Lundberg SM, Lee SI: **A unified approach to interpreting model predictions.** In: I. Guyon, U. Von Luxburg, S. Ben-gio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017; **30**.
Reference Source

41. Karniadakis GE, Kevrekidis IG, Lu L, *et al.*: **Physics-informed machine learning.** *Nat Rev Phys.* 2021; **3**(6): 422–440.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ? ✓ ✓

Version 2

Reviewer Report 17 September 2024

https://doi.org/10.21956/openreseurope.19914.r43588

© **2024 Sanchez N.** This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

✓ **Nestor Sanchez**
Universidad Internacional de Valencia, Valencia, Valencian Community, Spain

The reply of the authors is Ok and they took into account all my comments and suggestions. Thanks.
NS.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Astronomy/Astrophysics, Stars, Stars Clusters, Interstellar Medium, Data Mining in Astronomy.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 29 August 2024

https://doi.org/10.21956/openreseurope.19914.r43587

© **2024 YH Soo J.** This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

✓ **John YH Soo**
Universiti Sains Malaysia, Pulau Pinang, Malaysia

I have read the detailed reply by the authors, and I commend the effort and work made to improve the manuscript, as well as the clear clarifications on many important aspects of the paper. I am convinced that the authors have done a good job making the appropriate corrections, and I therefore approve this indexing. Well done! :)

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Computational astrophysics and cosmology

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Version 1

Reviewer Report 07 May 2024

https://doi.org/10.21956/openreseurope.18396.r39765

© 2024 YH Soo J. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

John YH Soo
[1] Universiti Sains Malaysia, Pulau Pinang, Malaysia
[2] Universiti Sains Malaysia, Pulau Pinang, Malaysia

This paper has good intentions, however, the work seemed to have started on a bad footing, as the choice and construction of the data set is not very good. The classification of stars utilising XGBoost has some novelty in it, and the sample selection has been detailed. However, here are my issues with the paper:

1) The paper's objectives are not entirely clear despite obtaining some usable results, and the conclusions are not quantitative enough.

2) Several parts of the methodology are not very well explained, e.g. the exact inputs were not described, the wrong usage of the concept 'data augmentation', graphs plotted with unclear labels, issues with the preprocessing stage (training data set not enough 'signals' compared to 'background').

3) Unsatisfactory command of grammar and the incorrect usage of the word 'skill'.

4) A certain section in the results section could be removed as it is either not explained correctly, used a questionable methodology, or produced results which would not solve or help the case of the paper.

All detailed comments on the paper are shown below as part of the comments to the authors. I'm actually prone to rejection of the paper, but then again, the paper's idea has some great novelty, just that it was executed poorly, making it unsatisfying to read. However if the authors are willing to make major changes to the paper detailed below, I think it is worth a second consideration.

The following are my comments on the paper, according to the different sections:

Abstract:

- The "which variables should be used" statement should be replaced with a sentence indicating exactly what variables were used.
- I suggest that the results section could include some quantitative results.

Introduction:

- "In this paper": I suggest that the authors add further details to this introduction, listing down all if not some of the star classes which are used in this work.
- At the end of this section, the objectives of the paper are still not clear despite the general idea being given. I recommend adding a paragraph to list down in 2 to 3 points the exact objectives after the "In this paper" paragraph.

Methods:

- "solar system": capital letters.
- PySSED: please include some references to the usage of PySSED in other work.
- "augmented the main dataset": how was this done? What kind of methodology did you use to conduct the augmentation, and why only the main sequence stars? Also, are these main-sequence stars added also classified as other stars at the same time?
- Figure 1: this plot is not very informative due to the overlapping of the dots, therefore it is difficult to assess the distribution of each type of star. I suggest that the authors plot a 3 x 3 plot, of luminosity vs effective temperature, and each plot for one class only, plotted as a density plot (e.g. three circles indicating the the 68, 97, 99 percentiles based on the concentration of dots per area). With the density plots in place, you would not need to worry about the outliers on the top left as they would not be encapsulated in the plot.
- Another info I would like to know from Figure 1 is if the Augmented Dataset for main-sequence stars will look different from the main dataset. Could you visualise this in your updated Figure 1, or explain if there are no differences in the paragraph. This may heavily affect your results section.
- "Several deficiencies in the input data": the number and names of input data has not been introduced prior to this point. I suggest that such a discussion on the list of inputs should be placed before this paragraph, or a reference to a section dedicated to the discussion of the inputs.
- "Therefore, finally, ": remove 'finally'.
- Train test split: please use a grammatically correct heading.
- "442 feature columns": what are these features exactly? Please summarise this using a

table. You can shorten the grouped features, e.g. saying that columns 1-200 is the original photometry in wavelength (unit used). Also indicate the percentage of missing values here (which will help the readers understand what is going on later).

- Equation 1: please be consistent in your choice of epsilon (ϵ/ε).
- "96 features are retained": it is very unclear how this is obtained, please explain in detail how you have reduced 442 features to 96 features. If you need, use figure to visualise the condensation of features to the final number.
- "Figure 2b and Figure 2b": repeated figure here.
- Figure 2(a): could you also show the distribution of the augmented data set? Could be in the same plot, with a fainter colour in the background. This is related to the request on augmented data set earlier on. Also, this distribution is not going to train well. Imagine trying to classify massive stars with only 0.24% signal and 99.76% background, how did you expect the machine to differentiate the signals from the noises?
- Figure 2(b): it is unclear what this plot is meant to describe. The y-axis label, do you mean it to be "Percentage of data WITHOUT missing values (%)" or "Data Completeness (%)"? If yes, change this, and make it in terms of percentage (0-100%) instead of 0 to 1. Your x-axis, is it cumulative? Are you saying that, e.g. 50 features have 10% missing values, or are you saying that feature #50 has 10% missing values? For either case, the label "Number of Features" does not describe what you are plotting. Please improve the labels. You could also indicate in the plots which is before and after processing, instead of relying merely on the caption to do so.
- Table 3: add a column to show the number of input variables in that row, then a row at the bottom indicating the total number of variables.
- Change "Other Physics" to just "Physics", keep the consistency throughout the paper.
- Table 4: you can condense this table by combining Class Weights 0 to 8 into one single row. Also, please add an extra column to briefly define the hyperparameters and the class weight (or explain them in the paragraph if it takes too much text).
- Equation 3: while you are at it, you should as well define accuracy here, for completeness sake. You can show the equation when you first mentioned accuracy.

Results:

- You can remove the quotation marks for "Main" and "Augmented", as you have defined them earlier.
- "We start by analyzing the skill": replace 'skill' with 'performance' please, do this for all instances.
- Figures 4b-f: Could you add a column before precision, indicating the number of objects in each of the classes for this test set that you work on, so that we can assess the results better

and dictate if the sample sizes have affected your results.

- "The reasons for this remain unclear": it is rather obvious actually, since you only have two variables in the Base model, which are related in a certain way (for blackbodies $L \propto T^4$), giving you essentially only one independent parameter to predict 8 different classes. Therefore it is not a good 'model' to begin with. Notice also from Figure 1, how small a range in size (R) your stars are, this further makes it difficult to differentiate 8 classes with 2 seemingly dependent parameters. You really should plot the density plots for Figure 1 I suggested, and look back at this results again and see if you can explain this better.

- If you really want to find out the effect of the bias package on young stellar objects, I suggest that you plot the RA and DEC of all the classes of stars and code them by colour, and find if you see any patterns. This could be hard evidence to backup your claim here.

- The pool of double-stars observed might be biased, since you require really good resolution to distinguish them from blended objects, and since they are orbiting each other, may give rise to visible traits in the spectra. Therefore, double-stars which are far away or too close to one another may be thought to be just single stars. You should include this fact in your discussion.

- "Augmented" dataset: once again, how did you augment your main sequence, where do these extra 80k stars come from? It wasn't clear from the previous explanation. Data augmentation usually means that you generate data from existing data, e.g. rotating images, or adding Gaussian errors to produce new data from existing data. Is this what you did, or did you just add 80k additional main-sequence stars from SIMBAD on top of the sampled main-sequence stars? If this is so, you are not really 'augmenting' your data, it is merely balancing your data. This should be clarified earlier.

- Figures 5b-f: just like Figure 4, add a column of total number of stars before precision.

- "Inspired by this result": this should be a new section with a new header.

- "Details see text": please correct this grammatically incorrect sentence.

- "we only removed the entry from the class undergoing reduction in the dataset": are you saying that e.g. for an object that is both evolved and variable, since you are reducing evolved, you only set its flag for evolved class to zero (to reduce the number of evolved objects) while keeping the flag for variable to 1 (keeping the number of variable objects), thus keeping the entry but modified its properties? I hope this isn't the case, because if it is, this section's results will be invalidated and should be removed entirely, as you are not supposed to modify the entries like that. What you should have in fact, is to reduce the "noise" instead, try reducing the number of other types of stars, such that when you have a higher signal to noise in your sample, to see if your classifier work. I need more clarification on your methodology here.

- I really don't understand the purpose of Figure 6: what were you expecting to happen other than going to zero the more you cut?

- "One possible exception are variable stars": well of course it will, since it is the largest proportion among other star types. Removing variable stars completely will significantly increase the signal-to-noise ratio of the other types, therefore increasing the overall average F1 score.

- mau -> may

Overall comments:
After reading the entire paper, I suggest that instead of focusing on 8 classes, the authors should just focus on the results of the augmented sample with 4 classes only: **, ev*, ms* and v*; all other results just don't make much sense in the context of this work due to the exceedingly small signal-to-background ratio. This paper has a novel objective, but was executed poorly. I recommend major revision and rework on the paper before it is considered for indexing.

Is the work clearly and accurately presented and does it cite the current literature?
Partly

Is the study design appropriate and does the work have academic merit?
Partly

Are sufficient details of methods and analysis provided to allow replication by others?
No

If applicable, is the statistical analysis and its interpretation appropriate?
Partly

Are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions drawn adequately supported by the results?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Computational astrophysics and cosmology

I confirm that I have read this submission and believe that I have an appropriate level of expertise to state that I do not consider it to be of an acceptable scientific standard, for reasons outlined above.

Author Response 14 Aug 2024

Sean Enis Cody

This paper has good intentions, however, the work seemed to have started on a bad footing, as the choice and construction of the data set is not very good. The classification of stars utilising XGBoost has some novelty in it, and the sample selection has been detailed. However, here are my

issues with the paper:

- *The paper's objectives are not entirely clear despite obtaining some usable results, and the conclusions are not quantitative enough.*
- *Several parts of the methodology are not very well explained, e.g. the exact inputs were not described, the wrong usage of the concept 'data augmentation', graphs plotted with unclear labels, issues with the preprocessing stage (training data set not enough 'signals' compared to 'background').*
- *Unsatisfactory command of grammar and the incorrect usage of the word 'skill'.*
- *A certain section in the results section could be removed as it is either not explained correctly, used a questionable methodology, or produced results which would not solve or help the case of the paper.*

All detailed comments on the paper are shown below as part of the comments to the authors. I'm actually prone to rejection of the paper, but then again, the paper's idea has some great novelty, just that it was executed poorly, making it unsatisfying to read. However if the authors are willing to make major changes to the paper detailed below, I think it is worth a second consideration.

Answer: Dear Reviewer, thank you for both your time and effort in reading our work and providing us with your feedback. We have changed the presentation of the results (especially the plots) and parts of the discussion. We have also carefully considered your comment on the overall structure of the study. However, we think that this was mainly caused by the fact that the objectives of our research were not made sufficiently clear in the manuscript. We apologize for the confusion, and we have now clarified the objectives. Notwithstanding that other approaches might also be valuable; we are convinced that the current approach of the study is the best way to reach those objectives. You will find a detailed discussion on this in our point-by-point responses below. Finally, we have given the paper another round of careful proofreading, with special focus on grammar. *The following are my comments on the paper, according to the different sections: Abstract:*

- *The "which variables should be used" statement should be replaced with a sentence indicating exactly what variables were used.*
- *I suggest that the results section could include some quantitative results.*

We changed the formulation slightly to “in which contexts which of the variables” should be used. Listing the variables and which of them have which influence here, would exceed the scope and length of an abstract. We have, however, as you suggested, added quantitative information in the results section. Introduction:

- *"In this paper": I suggest that the authors add further details to this introduction, listing down all if not some of the star classes which are used in this work.*

We have added a couple of sentences detailing the choice of stellar classes and our reasons for selecting them in the introduction.

- *At the end of this section, the objectives of the paper are still not clear despite the general idea being given. I recommend adding a paragraph to list down in 2 to 3 points the exact objectives after the "In this paper" paragraph.*

We have added a specific list of objectives to the end of the introduction *Methods:*

- *"solar system": capital letters.*
- *PySSED: please include some references to the usage of PySSED in other work.*
  - *Added “An example of PySSED on JWST data can be found in Wesson et al. [24]”, to the manuscript, citing an existing work using PySSED (https://ui.adsabs.harvard.edu/abs/2024RASTI...3...89M/abstract).*

- *"augmented the main dataset": how was this done? What kind of methodology did you use to conduct the augmentation, and why only the main sequence stars? Also, are these main-sequence stars added also classified as other stars at the same time?*
  - We extended the dataset with single label main sequence stars. The reason for only choosing these is due to the abundance of main sequence star data and ease of labelling them. They are sufficiently common such that no one bothers to label them. This cannot be said for the other classes however.
    We added a sentence of clarification to the Extended Dataset section:
    "We only chose to extend this class, due to the abundance of main sequence star data, which is largely left unlabelled due to being sufficiently common."
- *Figure 1: this plot is not very informative due to the overlapping of the dots, therefore it is difficult to assess the distribution of each type of star. I suggest that the authors plot a 3 x 3 plot, of luminosity vs effective temperature, and each plot for one class only, plotted as a density plot (e.g. three circles indicating the the 68, 97, 99 percentiles based on the concentration of dots per area). With the density plots in place, you would not need to worry about the outliers on the top left as they would not be encapsulated in the plot.*
  - The main point we hoped to convey in Figure 1 was the artificially created outliers, however seeing as the H-R diagram is highly informative for astrophysicists, we have taken your suggestion to heart and added the density plots to the manuscript, see the new Figure 2 .
- *Another info I would like to know from Figure 1 is if the Augmented Dataset for main-sequence stars will look different from the main dataset. Could you visualise this in your updated Figure 1, or explain if there are no differences in the paragraph. This may heavily affect your results section.*
- *"Several deficiencies in the input data": the number and names of input data has not been introduced prior to this point. I suggest that such a discussion on the list of inputs should be placed before this paragraph, or a reference to a section dedicated to the discussion of the inputs.*
  - We have reformulated the sentence to "Several deficiencies in the PySSED output data are apparent." to make it clear that we were referring to the PySSED outputs. The input features for the classifier as discussed in the PySSED section and in the preprocessing section.
- *"Therefore, finally, ": remove 'finally'.*
- *Train test split: please use a grammatically correct heading.*
- *"442 feature columns": what are these features exactly? Please summarise this using a table. You can shorten the grouped features, e.g. saying that columns 1-200 is the original photometry in wavelength (unit used). Also indicate the percentage of missing values here (which will help the readers understand what is going on later).*
  - Upon careful consideration of your suggestion, we believe a further table could lead to more confusion for the reader, as table 3 already lists the variables used. For readers who like to get the gist of the paper utilizing the table 3, provides better understandability, rather than listing something that is in turn not going to be used.
    In line with providing the most complete data description possible on the default parametrization, we have now clarified that we are using the default output in the text, with reference to the original PySSED paper, which explains

the default setup in detail: “Description of the default outputs and bands used are given in \citet{mcdonaldpyssed}.”
As for the units, although the tree based model we employed is invariant to feature scaling and therefore works irrespective of which unit is used, we added these in table 3.
To address the final point raised, we added “… , which describes the 74\% total missingness in the main dataset”.

- *Equation 1: please be consistent in your choice of epsilon (ϵ/ε).*
- *"96 features are retained": it is very unclear how this is obtained, please explain in detail how you have reduced 442 features to 96 features. If you need, use figure to visualise the condensation of features to the final number.*
  - We’ve added “After applying the above mentioned preprocessing steps to the default PySSED output data, …” to clarify how the reduction has taken place. Furthermore, we added a short section to the start of the preprocessing paragraph to explain one of the goals: “… , to reduce the data dimensionality”
- *"Figure 2b and Figure 2b": repeated figure here.*
  - We have rewritten the caption and changed the titles accordingly. This should now be much clearer for the reader. Unfortunately, the two plots cannot be collapsed into one, as the x-axis is different, i.e. after preprocessing less features exist to do the missing value analysis on and would make comparison of the missingness harder.
- *Figure 2(a): could you also show the distribution of the augmented data set? Could be in the same plot, with a fainter colour in the background. This is related to the request on augmented data set earlier on. Also, this distribution is not going to train well. Imagine trying to classify massive stars with only 0.24% signal and 99.76% background, how did you expect the machine to differentiate the signals from the noises?*

We added the distribution of the extended dataset to the plots as you suggested. Thanks for pointing this out. We think there might be some confusion between the concept of signal to noise on the one hand, and the ability of a machine learning classifier to learn to separate two (or more) classes. Even if we have a very imbalanced dataset, the majority class cannot be seen as “background noise”. Instead, the majority class is indeed a class. The task of the classifier is to (directly or indirectly) learn the distribution of the features in the input feature space, dependent on the target class, including the majority class. While highly imbalanced datasets can pose a challenge, there is no principal limitation to train on such a dataset. Specifically, the *ratio* between the majority and the minority class is not the main issue at hand. The main issue is the *total* amount of samples of the minority class. What needs to be done when training with imbalanced datasets is to carefully consider the type of training algorithm, and even more so, the correct type of loss functions. Adding even more samples of the majority class would – given the correct choice of training algorithm and loss functions – normally not deteriorate the skill on the minority class, as, once again, this is not an issue of signal to noise ratio, but of learning distributions in the input feature space. This can also clearly be seen in our results from the experiment where we systematically remove samples (figure 6b), and where we found that removing samples from one class – also majority classes - has little to no impact on the classification accuracy on other classes. Additionally, we compared our standard dataset with another dataset (termed augmented

dataset in the old manuscript, and extended dataset in the new manuscript) where the prior minority class has been – via inclusion of additional samples - turned into the new majority class. This has not reduced the performance on the other classes. Indeed, the performance was clearly increased (cf. fig 4 and fig. 5). This clearly shows that signal to noise between minority and majority classes is not the issue here, and that increasing the number of majority class samples does not make correctly classifying minority stars harder. Please also note that there is no a-priori way of telling whether a dataset is too imbalanced, or whether the smallest class has – in total numbers – enough samples for successful training. Indeed, one of the main objectives of our study was to find out whether training with the given dataset is possible, and to get indications on how different classes are impacted by their specific number of included samples. Prompted by your response, we have added the following to the Machine-Learning Methods section: "... Herein, models are tasked with either explicitly or implicitly learning some approximation of the sample data's class distribution. A common approach is to have models learn decision boundaries with the aim of separating the support sets of the class distributions, which ideally are aimed at generalizing from the sample to the population. ..."

- *Figure 2(b): it is unclear what this plot is meant to describe. The y-axis label, do you mean it to be "Percentage of data WITHOUT missing values (%)" or "Data Completeness (%)"? If yes, change this, and make it in terms of percentage (0-100%) instead of 0 to 1. Your x-axis, is it cumulative? Are you saying that, e.g. 50 features have 10% missing values, or are you saying that feature #50 has 10% missing values? For either case, the label "Number of Features" does not describe what you are plotting. Please improve the labels. You could also indicate in the plots which is before and after processing, instead of relying merely on the caption to do so.*
  - Thank you for pointing this out. We have now changed he y axis label and the title to clarify that we show a fraction. With the new labels "Number of Features" should now make sense.
- *Table 3: add a column to show the number of input variables in that row, then a row at the bottom indicating the total number of variables.*
- *Change "Other Physics" to just "Physics", keep the consistency throughout the paper.*
- *Table 4: you can condense this table by combining Class Weights 0 to 8 into one single row. Also, please add an extra column to briefly define the hyperparameters and the class weight (or explain them in the paragraph if it takes too much text).*
  Thank you for the suggestion, indeed condensing the table as you have suggested makes a lot of sense, we have taken the appropriate measures and reduced the table size. The hyperparameters and weights are described in the paragraph above the table.
- *Equation 3: while you are at it, you should as well define accuracy here, for completeness sake. You can show the equation when you first mentioned accuracy.*
  - Done

Results:

- *You can remove the quotation marks for "Main" and "Augmented", as you have defined them earlier.*
- *"We start by analyzing the skill": replace 'skill' with 'performance' please, do this for all instances.*
- *Figures 4b-f: Could you add a column before precision, indicating the number of objects in*

each of the classes for this test set that you work on, so that we can assess the results better and dictate if the sample sizes have affected your results.
Done. We have added a table with the sample counts per class for the test sets.

- *"The reasons for this remain unclear": it is rather obvious actually, since you only have two variables in the Base model, which are related in a certain way (for blackbodies L ∝ T^4),* giving you essentially only one independent parameter to predict 8 different classes. Therefore it is not a good 'model' to begin with. Notice also from Figure 1, how small a range in size (R) your stars are, this further makes it difficult to differentiate 8 classes with 2 seemingly dependent parameters. You really should plot the density plots for Figure 1 I suggested, and look back at this results again and see if you can explain this better.
    - The referee's logic here is unclear to us. Indeed, luminosity, temperature and radius are related by the Stefan-Boltzmann law, but the reduction to one independent parameter only works if a tight correlation exists between two of the parameters. While the main sequence represents a partial correlation, we should still expect to see massive stars as the only objects at its end, so even a one-dimensional discriminator should be able to robustly identify them.
    - Massive stars are hot stars, therefore subject to measurement uncertainty in their Wien tails due to lack of UV photometry, which (especially with reddening corrections) cause the unphysical tail of stars noted in the first panel of Figure 1. This measurement uncertainty could be the cause of the lack of ability for the Base (H-R diagram) features to fit massive stars, but unfortunately this is difficult to assess, as it requires both judgement of which stars have been fit in error and a recomputation and retraining of the model.
- *If you really want to find out the effect of the bias package on young stellar objects, I suggest that you plot the RA and DEC of all the classes of stars and code them by colour, and find if you see any patterns. This could be hard evidence to backup your claim here.*
    - The plots have been added to the appendix, see Figure 10.
- *The pool of double-stars observed might be biased, since you require really good resolution to distinguish them from blended objects, and since they are orbiting each other, may give rise to visible traits in the spectra. Therefore, double-stars which are far away or too close to one another may be thought to be just single stars. You should include this fact in your discussion.*
    - The list of double stars at SIMBAD is dominated not by component separation in visual binaries, but by spectroscopic binaries, eclipsing binaries and various types of interacting binaries. It is precisely our hope that these have visible traits in their SED that we can exploit to train the classifier.
      We added this description to the manuscript to clarify it for the reader:
      "This result shows that visible traits in the SED can be exploited by the classifier, as the SIMBAD database is dominated by spectroscopic, eclipsing and various types of interacting binaries, rather than component separation in visual binaries."
- *"Augmented" dataset: once again, how did you augment your main sequence, where do these extra 80k stars come from? It wasn't clear from the previous explanation. Data augmentation usually means that you generate data from existing data, e.g. rotating images, or adding Gaussian errors to produce new data from existing data. Is this what*

you did, or did you just add 80k additional main-sequence stars from SIMBAD on top of the sampled main-sequence stars? If this is so, you are not really 'augmenting' your data, it is merely balancing your data. This should be clarified earlier.

- We were not referring to data augmentation in the classical ML sense and can understand how this may seem confusing to readers. The terminology was changed throughout the paper to “extended”, as it is non-random sampling from the database.

- *Figures 5b-f: just like Figure 4, add a column of total number of stars before precision.*
- *"Inspired by this result": this should be a new section with a new header.*
- *"Details see text": please correct this grammatically incorrect sentence.*
- *"we only removed the entry from the class undergoing reduction in the dataset": are you saying that e.g. for an object that is both evolved and variable, since you are reducing evolved, you only set its flag for evolved class to zero (to reduce the number of evolved objects) while keeping the flag for variable to 1 (keeping the number of variable objects), thus keeping the entry but modified its properties? I hope this isn't the case, because if it is, this section's results will be invalidated and should be removed entirely, as you are not supposed to modify the entries like that. What you should have in fact, is to reduce the "noise" instead, try reducing the number of other types of stars, such that when you have a higher signal to noise in your sample, to see if your classifier work. I need more clarification on your methodology here.*

We indeed modified the samples as you described it. Given the objective of our experiment – finding out the impact of systematically removing samples of certain start types – this is a viable approach. For the specific case of stars that have two entries, another approach would have been to drop the star from both classes. Here please note that the number of stars with double classification (e.g. evolved and variable) is very low in our dataset (2.67%), therefore, it is not essential which of the two ways is chosen. Dropping the other classes – as you suggested - would be a very different experiment, and not one that would go towards the objectives our paper. As we explained above, signal to noise is not the issue at hand in such a classification task. Please also see our discussion on signal to noise in our reply to your earlier point above. Our methodology in this experiment is further clarified in our response to your next point.

- *I really don't understand the purpose of Figure 6: what were you expecting to happen other than going to zero the more you cut?*

You are right that the performance of the star type that is dropped will of course go to zero the less samples are included. However, the rate at which this happens is not a-priori clear, but this rate is very informative. For example, the ML algorithm might have reached a plateau, in which slightly reducing the number of samples has no impact (as, in our case, happens for lm*). This indicates that including more samples will likely not improve classification performance. On the other hand, for em*, there is an immediate drop in performance. This, consequently, indicates that increasing the number of samples would likely improve classification performance. Additionally, the data shown in panel b of figure 6 shows the impact on the classification performance of the other star types then the one systematically reduced. Here, again, there is no reliable way of telling a-priori whether there will be a significant impact or not. The insights from these experiments are highly valuable for future development of this (or of similar) classifiers. We have now clarified this in the manuscript.

- *"One possible exception are variable stars": well of course it will, since it is the largest proportion among other star types. Removing variable stars completely will significantly increase the signal-to-noise ratio of the other types, therefore increasing the overall average F1 score.*

We agree that this result is not so surprising, given the overlap between variable stars and other stars. While we already discuss this in the manuscript, we added "This is not surprising, as variable stars tend to overlap with other star types." to make this clearer to the reader that this is a potentially expected result.

- *mau -> may*

Overall comments:

- *After reading the entire paper, I suggest that instead of focusing on 8 classes, the authors should just focus on the results of the augmented sample with 4 classes only: **, ev*, ms* and v*; all other results just don't make much sense in the context of this work due to the exceedingly small signal-to-background ratio. This paper has a novel objective, but was executed poorly. I recommend major revision and rework on the paper before it is considered for indexing.*

Reducing the number of star types to the four that you suggested might seem reasonable at a first glance and would likely increase the apparent classification accuracy on a classifier trained on it. However, it would be a flawed approach, and not meet the main objectives of our paper (to train a first classifier for all stellar classes). We could filter the train and the test set in a way to reduce the types of stars included. However, then the classifier could not be used on data for which the star types are not yet known, as it would per definition be impossible to do the same filtering there. Also, as we explained above, signal to noise or signal to background ratio is not the issue at hand for training a stellar classifier as in our paper. Rather the issue at hand is the class imbalance and small number of samples of the majority classes. The objective was to find out whether this is the limitation for the classifier or not, and we showed that for the minority classes, this is indeed the case. Note that there could also be other limitations (e.g. with the given features, it might be impossible to separate certain star types beyond a given accuracy, even with unlimited data). Additionally, as mentioned before, the most important thing is not so much the ratio of minority to majority, but the total number of minority samples, and here it is impossible to a-priori determine how high that number has to be to get a good classifier. Therefore, the result that more samples of the minority classes are needed is not trivial. To sum up, the main objectives are (as now after your suggestion stated explicitly in the introduction): *"(1) to provide a first look at the applicability of ML in classification of the entire stellar range, given only sparse photometric measurements based on a random sample of the SIMBAD database, (2) to answer to which extent the sampling bias and associated class imbalance that is inherent in our problem impact the classifier's performance, and (3) what impact increasing the number of samples of minority classes in training database has on the performance of the classifier."* These objectives – especially objective (1) - can only be achieved by including all the star classes. We realize that our presentation was not optimal and could have led to a misunderstanding of our objectives and on the details of our methodology as well as the motivation for choosing said methodology. We apologize for the confusion, and hope that in our revised version this is now clearer to the reader.

Competing Interests: No competing interests were disclosed.

Reviewer Report 07 May 2024

https://doi.org/10.21956/openreseurope.18396.r39772

© 2024 Sanchez N. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nestor Sanchez
[1] Universidad Internacional de Valencia, Valencia, Valencian Community, Spain
[2] Universidad Internacional de Valencia, Valencia, Valencian Community, Spain

The manuscript "Machine learning based stellar classification with highly sparse photometry data" by Cody SE, Scher S, McDonald I, Zijlstra A, Alexander E and Cox N, uses Machine Learning (ML) methods for automatically classifying stars. In general, the obtained accuracies are not high enough for the proposed method to be applied directly in subsequent works. However, the authors are aware of this fact and clearly explain the strengths and limitations of their work. Moreover, they obtain and discuss some relevant results, such as the effect of the different sets of variables (bias, base, spectra, physics) on the classification process, which could be very useful in future attempts of automated star classification. Although, as stated by the authors, their work may be considered as a "first step" in studying the feasibility of using ML techniques for star classification, the obtained results and detailed analysis are a significant contribution to the field and I recommend its indexing.

There are some minor comments that I would like the authors to consider because they could be helpful (clarifying) for any potential reader.

The issue of the distance to each source... PySSED determines the distance of each star and then corrects photometry for reddening and estimates temperatures and luminosities. However, estimating distances for large amounts of data is a complex issue, and this may be an important point of uncertainties and/or biases in the results obtained. Distances in PySSED come from different methods, not only parallaxes from Gaia, and even if distances came only from Gaia there may be significant errors or biases coming from the parallax uncertainties. It is a topic that involves PySSED, of course, and it is beyond the scope of this work, but this (potential) drawback should be clearly and explicitly mentioned because it may be one of the factors contributing to a worse performing for the "base variables" set (which includes luminosity).

In order to be clear with potential readers, the fact that the method used throughout this work is XGBoost should be mentioned in the abstract (methods), because it seems that different ML methods are going to be used. Something similar happens with the justification of using only a randomly selected subset (10%) of the complete SIMBAD sample. It is first mentioned just before the conclusions that it is due computational limitations. It would be helpful to know it in the "data" section, especially because the lack of data for a proper training in some subsets is likely one of the reasons for relatively small accuracy values.

In the abstract-conclusions, the sentence saying that achieved accuracy does not yet allow the classifier to be used by "non-experts" does not seem appropriate. It would be better to state, as in

the conclusions, that it cannot yet be reliably used for star classification, or something similar.

Finally, I noticed two typos. When introducing Table 3, it says "... are summarized in in Table 3". And shortly before the conclusions, it says "... skill increase mau suggest that if...".

Is the work clearly and accurately presented and does it cite the current literature?
Yes

Is the study design appropriate and does the work have academic merit?
Yes

Are sufficient details of methods and analysis provided to allow replication by others?
Yes

If applicable, is the statistical analysis and its interpretation appropriate?
Yes

Are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions drawn adequately supported by the results?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Astronomy/Astrophysics, Stars, Stars Clusters, Interstellar Medium, Data Mining in Astronomy.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Author Response 14 Aug 2024

Sean Enis Cody

The manuscript "Machine learning based stellar classification with highly sparse photometry data" by Cody SE, Scher S, McDonald I, Zijlstra A, Alexander E and Cox N, uses Machine Learning (ML) methods for automatically classifying stars. In general, the obtained accuracies are not high enough for the proposed method to be applied directly in subsequent works. However, the authors are aware of this fact and clearly explain the strengths and limitations of their work. Moreover, they obtain and discuss some relevant results, such as the effect of the different sets of variables (bias, base, spectra, physics) on the classification process, which could be very useful in future attempts of automated star classification. Although, as stated by the authors, their work may be considered as a "first step" in studying the feasibility of using ML techniques for star classification, the obtained results and detailed analysis are a significant contribution to the field and I recommend its indexing. There are some minor comments that I would like the authors to consider because they could be helpful (clarifying) for any potential reader. Dear Reviewer, first

of all, we would like to extend our gratitude for taking your time to review our paper and offer feedback. We are also delighted that you recommended our work for indexing and have tried to address your comments as best we could.

- *The issue of the distance to each source... PySSED determines the distance of each star and then corrects photometry for reddening and estimates temperatures and luminosities. However, estimating distances for large amounts of data is a complex issue, and this may be an important point of uncertainties and/or biases in the results obtained. Distances in PySSED come from different methods, not only parallaxes from Gaia, and even if distances came only from Gaia there may be significant errors or biases coming from the parallax uncertainties. It is a topic that involves PySSED, of course, and it is beyond the scope of this work, but this (potential) drawback should be clearly and explicitly mentioned because it may be one of the factors contributing to a worse performing for the "base variables" set (which includes luminosity).*

Answer: The PySSED technical paper is now published, which covers many of these issues. We have referred to it in the text in this context, and expanded relevant comments in the Data section.

- *In order to be clear with potential readers, the fact that the method used throughout this work is XGBoost should be mentioned in the abstract (methods), because it seems that different ML methods are going to be used. Something similar happens with the justification of using only a randomly selected subset (10%) of the complete SIMBAD sample. It is first mentioned just before the conclusions that it is due computational limitations. It would be helpful to know it in the "data" section, especially because the lack of data for a proper training in some subsets is likely one of the reasons for relatively small accuracy values.*

Answer: Thank you for pointing this out. We have added parts both to the abstract and the data section, clarifying that we are using XGBoost throughout our work and also that the data is only 10 percent due to computational cost.

- *In the abstract-conclusions, the sentence saying that achieved accuracy does not yet allow the classifier to be used by "non-experts" does not seem appropriate. It would be better to state, as in the conclusions, that it cannot yet be reliably used for star classification, or something similar.*

Answer: We agree that the formulation was not optimal, and we have changed this part of the abstract conclusions to “not high enough to be reliably used in stellar classification”.

- *Finally, I noticed two typos. When introducing Table 3, it says "... are summarized in in Table 3". And shortly before the conclusions, it says "... skill increase mau suggest that if...".*

Answer: Thank you for mentioning these, both typos have been corrected.

Competing Interests: No competing interests were disclosed.

Reviewer Report 07 May 2024

https://doi.org/10.21956/openreseurope.18396.r39764

© 2024 Cordoni G. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Giacomo Cordoni
[1] Australian National University, Canberra, Australia
[2] Australian National University, Canberra, Australia

I think the paper presents a sound and thorough discussion on how machine learning can be used to infer stellar classes. It clearly highlights the importance of having an automated tool capable of classifying large number of sources (stars in this work) in a time-efficient way. The work addresses the technical challenge of such a goal and describes how the performance of the classification can be improved in the future. The work shows promising results. I believe that the paper is clear and well written, and the scientific analysis is interesting and carried out in a rigorous way. I therefore recommend the paper for indexing.

I have a however few suggestions that I believe could improve the paper before it is indexed. I understand that addressing all these suggestions may not be feasible, but I would appreciate it if the authors could consider them and provide their thoughts.

Main comments:

1. While I think that the authors did a great job in addressing possible technical issues and sources of bias, I believe that some astrophysical aspects may have been overlooked. Specifically, there is currently a wealth of information for millions of stars in the Milky Way, and many surveys provide accurate spectro-photometric information (see e.g. SDSS-17, APOGEE, LAMOST DR9, GALAH DR3, Gaia DR3).

   These are only some of the datasets available today, and more complete and numerous surveys are planned to be released and/or carried out in the forthcoming years.

   Therefore, I believe that more information could be exploited to infer stellar types. Additionally, I wonder whether using physical variables coming from different datasets (e.g. Teff, logg from actual spectroscopic surveys, instead of from SED fitting) might improve the performance of the classification, while avoiding the SED fitting step. Indeed, SED fitting is highly sensitive on the de-reddening procedure and accurate estimates of absorption are often required to produce reliable results.

   Additionally, could the ML algorithm be trained using the same classification, but exploiting Gaia XP spectra (or stellar properties derived from XP spectra, like for example in Andrae *et al.*, 2023[1])? This could drastically increase the sample of stars to which the algorithm could be applied, while possibly increasing the accuracy of the classification (for some or all classes).

2. I am not sure the “Preprocessing” section is entirely clear. Specifically, the authors mention they retained 96 features, of which 85 belongs to the Spectra variables. Does this mean that for each star there are up-to 85 photometric measurements? If that’s the case, I think it could be informative to understand how missing values among photometric data affect the predictive skill of classes. Specifically, having less photometric information would affect the outcome both because of less information is available to the algorithm, and because the SED fitting might underperform. Therefore, could there be a minimum number of photometric bands (or a minimum number of photometric bands per wavelength region) to ensure the best performance without compromising the sample size? Additionally, could

the predictive skill of different classes be dependent on the typical number of available features of that class?

3. On a similar note, if the missing photometric measurements fall into the Gaia XP spectra wavelength region (350-1000 nm), could Data imputation be a feasible alternative? Indeed, one could substitute missing photometric values with synthethic photometry generated by means of calibrated Gaia XP spectra (see e.g. Gaia Collaboration, Montegriffo *et al.* 2021[2]). Can the authors comment upon this?

4. Finally, I believe that not all photometric magnitudes carry the same weight. Are there specific photometric features that affect the predictive skills more than others? Is there a dependence on specific class (e.g. Is the infrared more relevant for the predictive skill of some classes?)

5. I did not understand why the authors randomly selected 10% of the whole SIMBAD sample of classified stars. Does it only refer to the training, or the remaining 90% has not been used at all? Does including all the stars improve the efficiency of the ML classification or it has negligible effect? Is it a choice motivated by computational costs? If so, I believe it could be better expressed in the Data section.

6. I believe it would be informative to better show the H-R diagrams of the input sources. Indeed, it is quite difficult to visualize the different classes from the right panel of current Figure 1. Could the authors show different classes in different panels. Specifically, the H-R diagram is a highly informative tool which allows to disentangle different classes of stars. Naively, I expect that the algorithm will perform better for stars in distinct HR diagrams location, as also mentioned by the authors in their discussion. On a similar note, I believe it could be useful to show the HR diagrams of correctly/incorrectly classified stars. I would ask to see these plots in the referee report, but I leave the choice of including them in the main manuscript to the authors.

7. Would it be possible to test the predictions of the algorithm on unclassified data? Specifically, stars clusters, such as globular clusters and open clusters, could represent a valuable source of information. Indeed, while it would not be possible to test all classes, it would be quite straightforward to test the classification of low and high-mass stars, main-sequence stars, and evolved stars. There are currently multiple catalogues of cluster members based on Gaia DR3 (Cantat-Gaudin *et al.*, 2020[3]; Tarricq *et al.* 2022[4]; Hunt *et al.,* 2024[5]) that could be used to this goal. Can the authors comment upon this?

8. Would decreasing the number of classes improve the efficiency of the classification? Would removing the most under sampled classes affect the training and outcome? Specifically, as the authors mentioned that sparsity and heterogeneity if the input sample represent important challenges, I am wondering if a more homogeneous distribution among classes would help improve the efficiency of the classification. Indeed, as this is a first attempt at using ML to classify stars, I believe it would be more important to increase the efficiency of the classification of a subset of classes, rather than including all classes. If having a more homogeneous training set does not affect the outcome, that the authors can disregard this last comment.

9. Would training an algorithm to classify only one class at a time improve the accuracy of the classification? Or it would not affect the predictive skills? Do the authors have any opinion on that?

10. This comment is partially connected to comment #1. Would excluding the base variables (i.e. physics – base) strongly affect the results? If so, in what way? As the authors show that the base model generally has a lower skill, I wonder if that could be a consequence of a non-reliable SED fit. If that is the case, is it possible that removing the base model would improve the predictive skill? Additionally, as the spectroscopic temperature is already included in the "other physics" model, one would only use the information about the luminosity of the star. Could the authors comment upon this? Furthermore, would removing the SED fit step reduce the computational cost? If so, would this allow to increase the training sample size?

Minor comments:

1. At page 7, in the first lines, Figure 2b is repeated twice. Could the second instance be Figure 2c?

2. I think it could be useful to mention which photometric data is being used in the SED fitting, or at least give few examples of the SED fitting process for stars with large/medium/low available photometric information, and maybe how the retrieved parameters (Teff and log) compare with spectroscopic measurements, if available.

References

1. Andrae R, Rix H, Chandra V: Robust Data-driven Metallicities for 175 Million Stars from Gaia XP Spectra. *The Astrophysical Journal Supplement Series*. 2023; **267** (1). Publisher Full Text
2. Gaia Collaboration, Montegriffo P, Bellazzini M, De Angeli F, et al.: Gaia Data Release 3. *Astronomy & Astrophysics*. 2023; **674**. Publisher Full Text
3. Cantat-Gaudin T, Anders F, Castro-Ginard A, Jordi C, et al.: Painting a portrait of the Galactic disc with its stellar clusters. *Astronomy & Astrophysics*. 2020; **640**. Publisher Full Text
4. Tarricq Y, Soubiran C, Casamiquela L, Castro-Ginard A, et al.: Structural parameters of 389 local open clusters. *Astronomy & Astrophysics*. 2022; **659**. Publisher Full Text
5. Hunt E, Reffert S: Improving the open cluster census. III. Using cluster masses, radii, and dynamics to create a cleaned open cluster catalogue. *Astronomy & Astrophysics*. 2024. Publisher Full Text

Is the work clearly and accurately presented and does it cite the current literature?

Yes

Is the study design appropriate and does the work have academic merit?

Yes

Are sufficient details of methods and analysis provided to allow replication by others?

Yes

If applicable, is the statistical analysis and its interpretation appropriate?

Yes

Are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions drawn adequately supported by the results?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Astrophysics, Stellar astrophysics, stellar evolution, star clusters, stellar dynamics, spectro-photometric surveys.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 19 Aug 2024

Sean Enis Cody

I think the paper presents a sound and thorough discussion on how machine learning can be used to infer stellar classes. It clearly highlights the importance of having an automated tool capable of classifying large number of sources (stars in this work) in a time-efficient way. The work addresses the technical challenge of such a goal and describes how the performance of the classification can be improved in the future. The work shows promising results. I believe that the paper is clear and well written, and the scientific analysis is interesting and carried out in a rigorous way. I therefore recommend the paper for indexing. I have a however few suggestions that I believe could improve the paper before it is indexed. I understand that addressing all these suggestions may not be feasible, but I would appreciate it if the authors could consider them and provide their thoughts. *Dear reviewer, thank you for taking the time to review our work and for providing us with valuable feedback! We're happy to hear that the paper was well received. We have tried our best to address and incorporate your feedback into the revised paper. Please find detailed responses to all your comments below.* **Main comments:**

- While I think that the authors did a great job in addressing possible technical issues and sources of bias, I believe that some astrophysical aspects may have been overlooked. Specifically, there is currently a wealth of information for millions of stars in the Milky Way, and many surveys provide accurate spectro-photometric information (see e.g. SDSS-17, APOGEE, LAMOST DR9, GALAH DR3, Gaia DR3).

  These are only some of the datasets available today, and more complete and numerous surveys are planned to be released and/or carried out in the forthcoming years.

  Therefore, I believe that more information could be exploited to infer stellar types. Additionally, I wonder whether using physical variables coming from different

datasets (e.g. Teff, logg from actual spectroscopic surveys, instead of from SED fitting) might improve the performance of the classification, while avoiding the SED fitting step. Indeed, SED fitting is highly sensitive on the de-reddening procedure and accurate estimates of absorption are often required to produce reliable results.

Additionally, could the ML algorithm be trained using the same classification, but exploiting Gaia XP spectra (or stellar properties derived from XP spectra, like for example in Andrae *et al.*, 2023[1])? This could drastically increase the sample of stars to which the algorithm could be applied, while possibly increasing the accuracy of the classification (for some or all classes).

Answer: *The aim of our approach was to discern whether we could identify stellar types from photometric data alone. We agree that a large number of spectra exist, especially from Gaia, spectral and spectrophotometric means of establishing stellar types are indeed better. However, software to establish stellar types from such data already exist, and (despite the millions of spectra available) the vast majority of stars still lack spectra (e.g. only 32% of Gaia DR3 stars have spectra). A major driver behind creating this software is so that we have a classifier that can work in the absence of spectroscopy, both to provide an independent classification to spectral-based classifiers, and to have a procedure we can use among stars where spectra cannot be taken, including deeper observations of the Milky Way and external galaxies that are being taken by (e.g.) Euclid, Roman, Rubin, HST and JWST. We have highlighted this in the introduction and return to this point further down in our response. Dereddening is an issue that affects the accuracy of our fitting. In the original PySSED paper (the cited McDonald et al. 2024), the dereddening procedure was demonstrated to be robust in the majority of cases though (as Figure 1 shows) there is an asymptotically increasing sensitivity as stars become hotter. Since the dereddening procedure uses Gaia BP/RP spectrophotometric data Gaia spectrophotometry can no longer be treated as an independent result for comparison: we have highlighted this in the paper. Regardless, one item we aim to measure is the circumstellar reddening around many stars, which gives a clue to their categorisation. In essence, we consider the main limitation that limits the accuracy of our approach is that the types of star that we have tried to model (SIMBAD's top level classifications) cannot all accurately be identified from SED fitting.*

I am not sure the "Preprocessing" section is entirely clear. Specifically, the authors mention they retained 96 features, of which 85 belongs to the Spectra variables. Does this mean that for each star there are up-to 85 photometric measurements? If that's the case, I think it could be informative to understand how missing values among photometric data affect the predictive skill of classes. Specifically, having less photometric information would affect the outcome both because of less information is available to the algorithm, and because the SED fitting might underperform. Therefore, could there be a minimum number of photometric bands (or a minimum number of photometric bands per wavelength region) to ensure the best performance without compromising the sample size? Additionally, could the predictive skill of different classes be dependent on the typical number of available features of that class?

Answer: *Indeed, in our scenario, a star can have measurements within 85 different bands, which leads to a high level of sparseness in the data. As part of our early preprocessing steps we also included a feature selection based on low variance bands, under the assumption of low variance = low information. We thereby hoped to improve generalization, however the results did not improve, as the ensemble tree structure is already performing an internal feature selection*

through the tree-splitting criterion. In other words, it made sense for us to keep all features in, as XGBoost would use a feature only in making decisions if the internal "gain" is great enough (if truly all features with low variance were worse than the other more densely packed features, their impact on the classification would be negligible). The quality of the SED fitting in the majority of cases does not strongly depend on the number of photometric datapoints, but rather on their distribution in wavelength. A surprisingly good fit is normally achieved even if only a handful of the 85 possible bands have been observed (see the cited McDonald et al. 2024). In most cases, the majority of the impact from sparse data is therefore on the data available to the classifier for predicting classes. As for the predictive skill of classes being tied to sparseness of its bands, we utilized SHAP importance measures to investigate this further.

At first glance our findings seemed to align with the natural assumption, that features with more entries contained more information. For example, for the Physics Feature set, we found that 18 out of the 20 features that contributed most to the classification were among the 20 most densely populated features. Interestingly however, the WISE/WISE.W4 band which was the most impactful feature overall was not among these. In fact, upon observing the samples most impacted by this band via local explainability, we found that out of the top 50% most impacted main sequence star samples, 95% were missing values. Similarly, the top 5% of samples impacted most by this band, had over 92% missingness. This indicates that the predictive performance of different classes is not entirely dependent on the typical number of available features of that class, but rather that missing value patterns can be learned to act as discriminative features using XGBoost's default pathway choices. We have added a brief section at the end of the results section, right before the conclusions, in response to this question.

- On a similar note, if the missing photometric measurements fall into the Gaia XP spectra wavelength region (350-1000 nm), could Data imputation be a feasible alternative? Indeed, one could substitute missing photometric values with synthethic photometry generated by means of calibrated Gaia XP spectra (see e.g. Gaia Collaboration, Montegriffo *et al.* 2021[2]). Can the authors comment upon this?

Answer: *While this could certainly be done in principle, we are (as mentioned above) primarily interested in identifying the stellar classification that can be performed with photometry alone, as it has wider applications beyond the subset of objects for which spectra are available.*

- Finally, I believe that not all photometric magnitudes carry the same weight. Are there specific photometric features that affect the predictive skills more than others? Is there a dependence on specific class (e.g. Is the infrared more relevant for the predictive skill of some classes?)

Answer: *Yes, some wavelength bands are more important than others. We also investigated this using SHAP importance (see updated Fig. 8) and found that specific wavelengths were more indicative for some classes. In many cases this increase in predictive skill correlated with the abundance of sample entries at that specific wavelength. We did however also find highly sparse wavelengths, which had missing values, that were used to make reasonable predictions. This is largely to do with XGBoost's default pathways for missing values providing the capabilities to use missing values as discriminative features values. The Result for the Full feature set is shown in R1 for the top 20 most important features in the sample extended dataset, with class specific importances highlighted by varying colour. We have added the plot and a brief discussion on this to our results section of the paper, right before the conclusions.*

- I did not understand why the authors randomly selected 10% of the whole SIMBAD

sample of classified stars. Does it only refer to the training, or the remaining 90% has not been used at all? Does including all the stars improve the efficiency of the ML classification or it has negligible effect? Is it a choice motivated by computational costs? If so, I believe it could be better expressed in the Data section.

Answer: *Only 10% of SIMBAD's classified stars were used as inputs to the classifier due to the requirement that these stars were first pre-processed by PySSED. Processing a list of stars this way takes a long time due to the database setup at CDS and the multiple queries per star that need to occur as a result. Hence, the remaining 90% of stars haven't been used at all. In the discussion surrounding Figure 6, where we dropped samples per class, we noted that the problem is likely rooted in the underrepresentation of minority classes and not in the class imbalance itself. Therefore, as you have surmised, any increase in data leading to an increase in minority samples is likely to improve the classifier's performance. We realize that this was not made clear enough in the manuscript and have edited the data section accordingly. We added ", the choice of which was largely dictated by computational time, originating from evaluating multiple queries per star at the CDS SIMBAD database during the PySSED preprocessing steps" after specifying the 10 percent sample size.*

- I believe it would be informative to better show the H-R diagrams of the input sources. Indeed, it is quite difficult to visualize the different classes from the right panel of current Figure 1. Could the authors show different classes in different panels. Specifically, the H-R diagram is a highly informative tool which allows to disentangle different classes of stars. Naively, I expect that the algorithm will perform better for stars in distinct HR diagrams location, as also mentioned by the authors in their discussion. On a similar note, I believe it could be useful to show the HR diagrams of correctly/incorrectly classified stars. I would ask to see these plots in the referee report, but I leave the choice of including them in the main manuscript to the authors.

Answer: *Thanks for bringing this up, indeed it is hard to distinguish the classes in Figure 1. The main message we hoped to convey here was the artificial outliers generated as part of the SED fitting procedure, which in turn ties into our further motivation for the choice of XGBoost or rather the missing outlier removal preprocessing step (as the model is robust to these outliers). Therefore, we had chosen not to add these plots to the paper, as the message we hope to discuss would be harder to identify. We take note however that, as you say, the H-R diagram is highly informative to astrophysicists and have therefore included the requested plots as the updated figure 2. Regarding the HR diagrams of correctly/incorrectly classified stars, we expanded this to false positives, false negatives and true positives and added it to the appendix, see updated Figure 9.*

- Would it be possible to test the predictions of the algorithm on unclassified data? Specifically, stars clusters, such as globular clusters and open clusters, could represent a valuable source of information. Indeed, while it would not be possible to test all classes, it would be quite straightforward to test the classification of low and high-mass stars, main-sequence stars, and evolved stars. There are currently multiple catalogues of cluster members based on Gaia DR3 (Cantat-Gaudin *et al.*, 2020[3]; Tarricq *et al.* 2022[4]; Hunt *et al.,* 2024[5]) that could be used to this goal. Can the authors comment upon this?

Answer: *This is a very sensible suggestion. There is substantial additional science that can come from analysis of both globular and open clusters with SED fitting (e.g.*

2011ApJS..193...23M, 2009ApJ...705..746B, 2009MNRAS.394..831M). For this reason, we would prefer to perform this kind of analysis as part of a separate study focussing on SED fitting of globular clusters in general, and we have applied for funding to undertake this.

- Would decreasing the number of classes improve the efficiency of the classification? Would removing the most under sampled classes affect the training and outcome? Specifically, as the authors mentioned that sparsity and heterogeneity if the input sample represent important challenges, I am wondering if a more homogeneous distribution among classes would help improve the efficiency of the classification. Indeed, as this is a first attempt at using ML to classify stars, I believe it would be more important to increase the efficiency of the classification of a subset of classes, rather than including all classes. If having a more homogeneous training set does not affect the outcome, that the authors can disregard this last comment.

Answer: *The choice and number of classes discussed in this paper is certainly sub-optimal, due to constrains on time and funding for this work. However, we felt it important to publish those components of research we could optimise, in order that the community could take this research forward in future. The broad SIMBAD classes that we examine each contain a heterogeny of different types. For example, many classes of variable star (e.g. novae sources, T Tau stars or Mira variables) could be easily identifiable due to infrared excess caused by circumstellar dust, whereas other classes (e.g. eclipsing binary stars or spotted stars) may be very difficult to identify from SED fitting. Ideally, we would like to progress through the entire SIMBAD hierarchical tree of classes, identifying those where our classifier can obtain results with a high degree of confidence (F1 score), and creating a (potentially much larger) set of classes to choose between, each of which could better inform the classes of stars further up the hierarchical tree. However, that level of decision and subsequent optimisation is a corpus of work in its own right.*

- Would training an algorithm to classify only one class at a time improve the accuracy of the classification? Or it would not affect the predictive skills? Do the authors have any opinion on that?

Answer: *In our experiments this was not the case, we also briefly studied a one vs rest (OVR) approach, where the result was based on a set of binary classifiers, however the results were quite similar. We did this largely based on results from the literature (https://www.aanda.org/articles/aa/pdf/2022/10/aa43135-22.pdf), pertaining to advantages of OVR in stellar classification. Since the results were similar enough though, we avoided this approach in subsequent steps, as the final model obtained with an OVR approach would also have been more complex to explain. We therefore retained the multiclass model approach.*

- This comment is partially connected to comment #1. Would excluding the base variables (i.e. physics – base) strongly affect the results? If so, in what way? As the authors show that the base model generally has a lower skill, I wonder if that could be a consequence of a non-reliable SED fit. If that is the case, is it possible that removing the base model would improve the predictive skill? Additionally, as the spectroscopic temperature is already included in the “other physics” model, one would only use the information about the luminosity of the star. Could the authors comment upon this? Furthermore, would removing the SED fit step reduce the computational cost? If so, would this allow to increase the training sample size?

Answer: *There are two categories of information encoded by the SED fit. The first are the properties of the SED fit itself, which forms the Base features. The second are the discrepancies between the SED fit and the stellar atmosphere model used to create the fit. Removing the Base*

variables is possible, but essentially this removes information about where on the H-R diagram a star sits. For example, this removes much of the ability to distinguish between main-sequence and evolved stars, since the effects of surface gravity on SEDs are relatively minor. These Base variables are correspondingly some of the most critical in fitting stars (as identified by their high SHAP importance). Again, only a fraction of stars have spectroscopic temperatures, so we do not wish to rely on it. Removing the SED fit entirely is not possible due to the second category of information. In essence, we fit the difference between the stellar model and the star itself. Without the SED fitting, we remove not only the astrophysical parameter generation, but also the extinction correction, the removal of erroneous photometry, and the model normalisation. Since the computational bottleneck is simply in obtaining the basic photometry for each star from CDS, removing the SED fit come at a significant cost without any significant benefit to computational speed. **or comments:**

- At page 7, in the first lines, Figure 2b is repeated twice. Could the second instance be Figure 2c?

Answer: *The differences are minor, as the shape remains the same, however the first is the unpreprocessed data and the second post spectra transformation. The message conveyed is more or less identical, yet for completeness' sake, the non-preprocessed raw data is also shown. To make this clearer to the reader we have now adjusted the titles of the plots. Furthermore, we have added the extended dataset to these plots.*

- I think it could be useful to mention which photometric data is being used in the SED fitting, or at least give few examples of the SED fitting process for stars with large/medium/low available photometric information, and maybe how the retrieved parameters (Teff and log) compare with spectroscopic measurements, if available.

Answer: *Examples of this are given in the PySSED technical paper, which is now published. We have added text to the PySSED section to reference this paper and discuss these issues in this context.*

Competing Interests: No competing interests were disclosed.