Stellar Blend Image Classification Using Computationally Efficient Gaussian Processes (MuyGPs)

Rafael Bidese, Chinedu Eleh, Yunli Zhang, Roberto Molinari and Nedret Billor
Auburn University, Auburn, AL 36849, USA
{rafael, yzz0168, cae0027, robmolinari, billone}@auburn.edu

Benjamin Priest, Imene Goumiri, Amanda Muyskens and Alec Dunton
Lawrence Livermore National Laboratory, Livermore, CA 94550, USA
{priest2, goumiri1, muyskens1, dunton1}@llnl.gov

Abstract

Stellar blends are a challenge in visualizing celestial bodies and are typically disambiguated through expensive methods. To address this, we propose an automated pipeline to distinguish single stars and blended stars in low resolution images. We apply different normalizations to the data, which are passed as inputs into machine learning methods and to a computationally efficient Gaussian process model (MuyGPs). MuyGPs with N^{th} root local min-max normalization achieves 86% accuracy (i.e. 12% above the second-best). Moreover, MuyGPs outperforms the benchmarked models significantly on limited training data. Further, MuyGPs low-confidence predictions can be redirected to a specialist for human-assisted labeling.[1]

Keywords: Stellar Blends, Normalization, Gaussian Processes, Uncertainty Quantification

1 Introduction

There are several wide-aperture ground-based sky surveys that image a large area of sky, gathering large volumes of data. Some of them are the Large-aperture Synoptic Survey Telescope (LSST), Sloan Digital Sky Survey (SDSS), Dark Energy Spectroscopic Instrument (DESI), Legacy Imaging Survey and the Zwicky Transient Facility (ZTF). The data collected by those telescopes are often used to study dark matter through gravitational lensing. One inherent challenge is the distortions in universe observation caused by the matter along the line of sight between an observer and a star. When those light distortions cause two or more celestial bodies to visually overlap in an image it generates an effect called stellar blend. Stellar blends are typically disambiguated through expensive methods (i.e., spectroscopy or high-resolution images using ground-based telescopes). Therefore, to handle the high volume of data generated by those telescopes, approaches using machine learning can enable the automated classification of images into a single star or blended stars. The main difficulty is that the cutout images from those surveys are low resolution in nature and cannot be distinguished visually between blended and non-blended stars.

2 Methods

2.1 Models

One traditional approach is to manually engineer features as inputs to classification models. Sevilla-Noarbe (2015) used decision trees and Odewahn (1992) applied deep neural networks (DNN) to successfully distinguish stars from galaxies. Kim (2016) applied convolutional neural networks (CNN) to SDSS data, 5 channels and 44×44 pixels. These models have demonstrated to incorrectly classify out-of-distribution samples with confidence. Muyskens (2021) present a computationally efficient Gaussian process (GP) algorithm that has been applied successfully to other similar astronomical applications, e.g. star-galaxy disambiguation and galaxy blend detection. MuyGPs uses an approximate k-nearest neighbors with k neighbors in batches of size b to achieve a computational complexity of $O(bk^3)$ compared to a maximum likelihood expectation method that is $O(n^4)$, where n is the training set size. In the MuyGPs framework, a threshold can be defined based on the model's prediction confidence, and the ambiguous samples can be further investigated. We compare all these models (decision trees, DNN, CNN, GP) using data with different normalizations.

2.2 Normalization

Applying normalizations to the data before training a model can favor computational optimization and generate better distribution of the data. We employ several normalizations to this dataset and report results from a local min-max (Equation 1), a global min-max (Equation 2) and a combination of N^{th} root and local min-max (Equation 3). In Equations 1-3, x is the complete dataset, x_i is i^{th} sample, and $r \in \mathbb{R}_+$ and $r \in (0, 1)$. To visualize the normalized data distribution, we use Uniform Manifold Approximation and Projection (UMAP) (McInnes et al., 2018) as a dimension reduction technique and analyze the embedded space.

$$Local_{minmax}(x_i) = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)}, \tag{1}$$

$$Global_{minmax}(x_i) = \frac{x_i - min(x)}{max(x) - min(x)}, \tag{2}$$

$$N^{th}root_{Local,minmax}(x_i) = \left(\frac{x_i - min(x_i)}{max(x_i) - min(x_i)}\right)^r. \tag{3}$$

[1] Prepared by LLNL under Contract DE-AC52-07NA27344 with release number LLNL-ABS-843297.

3 Data

Single stars and stellar blends are determined by making cuts on a subset of data from the DESI Legacy Imaging Survey. Due to the nature of astronomical cataloging and innate error there is a chance of mislabeling objects (noisy label).

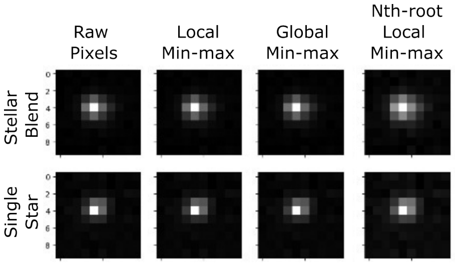

Figure 1: Sample images from the dataset for the different classes and with normalizations

The dataset consists of 27253 images extracted from ZTF, 15110 single stars and 12143 stellar blends. The stellar blends have two sub-categories: blended stars (7414) and binary stars (4729). The images are energy flux for the i-band (i.e. ~700-900 nm) of an earth-based telescope with one channel and with size 10×10 pixels. Sample images for both classes with and without normalizations are presented in Figure 1. For training and evaluation a data split of 80% for training, 10% for validation and 10% for test is used.

4 Discussion

We see the importance of normalization in Figure 2, further supported by the results presented in Table 1. Even though the samples from Figure 1 are similar, when visualizing them in an embedding as in Figure 2, we see that the classes become more easily separable with local min-max normalizations. All studied models perform better with normalization. MuyGPs performs significantly better than the other models with and without normalization (Table 1). We conclude that the proposed automated pipeline can successfully distinguish single stars from blended stars in low-resolution images. In special, when the data is limited (Figure 3) there is significantly better performance in MuyGPs. Some of it might be due to the noisy label characteristic of astronomic data and MuyGPs is able to deal better with it than the other models. Finally, in a real application, Figure 4 shows how one can set a threshold for uncertainty, determine the expected accuracy and see what proportion of the data is expected to be ambiguous and need further annotation by a different system or a human specialist.

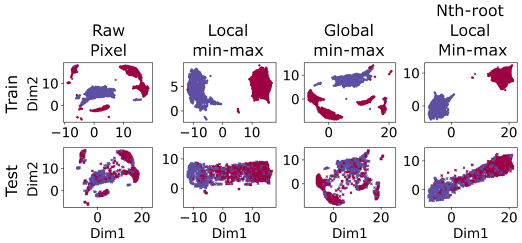

Figure 2: Normalization visualization using UMAP.

Model	Normalization	Accuracy
Decision Tree	Raw Pixel	0.64
DNN	Raw Pixel	0.65
CNN	Raw Pixel	0.62
MuyGPs	Raw Pixel	0.76
Decision Tree	N^{th} root Local Min-Max	0.69
DNN	Local Min-Max	0.70
CNN	N^{th} root Local Min-Max	0.74
MuyGPs	**N^{th} root Local Min-Max**	**0.86**

Table 1: Comparison of different model performances.

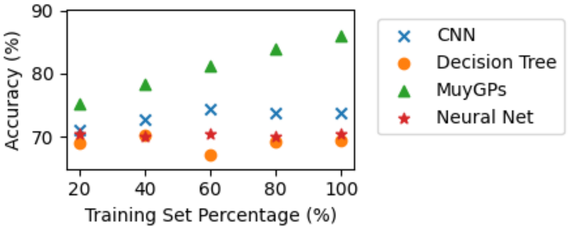

Figure 3: Accuracy for different amounts of training data.

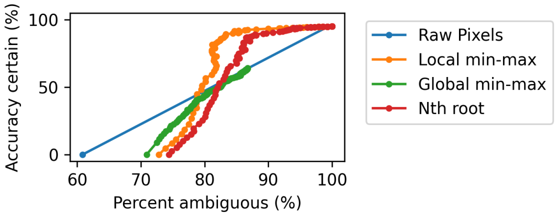

Figure 4: MuyGPs evaluation of ambiguous predictions at different confidence thresholds.

References

Edward J. Kim and Robert J. Brunner. 2016. Star–galaxy classification using deep convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 464(4):4463–4475, oct.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction.

Amanda L. Muyskens, Benjamin W. Priest, Imène Goumiri, and Michael Schneider. 2021. Muygps: Scalable gaussian process hyperparameter estimation using local cross-validation. *arXiv preprint arXiv:2104.14581*.

S. C. Odewahn, E. B. Stockwell, R. L. Pennington, R. M. Humphreys, and W. A. Zumach. 1992. Automated star/galaxy discrimination with neural networks. In H. T. MacGillivray and E. B. Thomson, editors, *Digitised Optical Sky Surveys*, pages 215–224, Dordrecht. Springer Netherlands.

Ignacio Sevilla-Noarbe and Penélope Etayo-SotosI. 2015. Effect of training characteristics on object classification: An application using boosted decision trees. *Astronomy and Computing*, 11:64–72, jun.