Astrophysics-Informed Machine Learning for Spectral Luminosity Classifcation of Stellar Types

Dr. Preet Kamal
Department of Computer Science & Engineering , Apex Institute of Technology
Chandigarh University
Mohali ,Punjab , India
preet.e15857@cumail.in

Shubh Upadhyay
Department of Computer Science & Engineering , Apex Institute of Technology
Chandigarh University
Mohali ,Punjab , India
21BCS6463@cuchd.in

Rohit Kumar
Department of Computer Science & Engineering , Apex Institute of Technology
Chandigarh University
Mohali , Punjab, India
21BCS6512@cuchd.in

Ayush Bansal
Department of Computer Science & Engineering , Apex Institute of Technology
Chandigarh University
Mohali , Punjab ,India
21BCS6270@cuchd.in

Abstract— Precise stellar object classification is central to astrophysical research, informing stellar evolution models, galactic structure, and planetary habitability. Classical classification methods, based mainly on spectral analysis, are measurement-uncertainty-prone and time-consuming. This work introduces a machine learning system that exploits astrophysical domain expertise to classify dwarf and giant stars from photometric and astrometric observations of the NASA Kepler mission. Important features such as parallax-estimated distance, absolute magnitude, effective temperature from color index, luminosity, and mass were engineered using empirical stellar relations to guarantee physical interpretability. Multiple classifiers were trained and tested, with Light Gradient Boosting Machine (LightGBM) recording the best performance: 89.1% accuracy, 0.925 recall, and 0.909 F1-score. The model was made interpretable using SHapley Additive exPlanations (SHAP), which pinpointed temperature and luminosity as the most informative features, in line with the Hertzsprung–Russell diagram. The final model was deployed through a Flask-based API, allowing real-time stellar classification with sub-25 millisecond inference latency. Results show that domain-aided machine learning models can copy and improve classical classification systems, providing scalable, interpretable tools for stellar taxonomy.

Keywords— Stellar classification, Machine learning, LightGBM, Kepler mission, Astrophysical feature engineering, SHAP, Star evolution, Dwarf and giant stars

I. INTRODUCTION

Stellar classification is a fundamental cornerstone of astrophysics, providing us with information about a star's mass, temperature, luminosity, radius, evolutionary stage, and large-scale galactic kinematics. This has traditionally been in the form of the Harvard spectral classification scheme and Yerkes luminosity classes, which, while extremely widely used, depend on spectroscopically difficult and expensive data.

Modern surveys like Kepler, Gaia, and SDSS produce vast photometric and astrometric data—parallax, apparent magnitude, color index—albeit less precise than spectroscopy, but can still be useful for classifications by computational methods. However, their high dimensionality creates pressure on traditional algorithms.

Machine learning (ML) provides a potential solution that is capable of capturing non-linear relationships in complicated data. Its application, however, depends on domain-specific feature engineering and interpretability. Blind adoption of black-box models risks undermining the scientific rigor needed in astronomy.

We propose a physics-informed ML pipeline to label stars as dwarfs or giants using Kepler Input Catalog data. Key stellar parameters—distance, absolute magnitude, temperature, luminosity—are derived from astrophysical relations and used to train classifiers like Logistic Regression, Random Forests, XGBoost, CatBoost, and LightGBM. LightGBM performed best (F1 score: 0.91) and was robust to noise and class imbalance.

Interpretability of the model is given by SHAP (SHapley Additive exPlanations), which interprets the contribution of every feature to the outcome of classification, translating predictions to physical intuition. For real-world deployment, the model is encapsulated in a Flask-based REST API that accepts stellar parameters as input and offers real-time classification.

By integrating domain expertise and ML, this pipeline illustrates an explainable, scalable method of stellar classification—modern data science and theoretical astrophysics merged for pedagogic and pragmatic application.

TABLE I : LITERATURE REVIEW

Year and Citation	Article/ Author	Tools/ Software	Technique	Source	Evaluation Parameter
2022 [1]	Classifying Kepler light curves for 12,000 A and F stars using supervised feature-based machine learning by Barbara et al.	Various ML libraries	ML, feature-based	arXiv	Accuracy, feature importance
2021 [2]	General classification of light curves using extreme boosting by Kgoadi et al.	XGBoost	Boosting, ML	arXiv	Classification accuracy
2021 [3]	Accurate and Robust Stellar Rotation Periods catalog for 82771 Kepler stars using deep learning by Kamai and Perets	TensorFlow, custom deep learning models	Deep learning	arXiv	Prediction accuracy, robustness
2021 [4]	Scientific Domain Knowledge Improves Exoplanet Transit Classification with Deep Learning by Ansdell et al.	Keras, TensorFlow	Deep learning	arXiv	Recall, false positive rate
2020 [5]	Model Explainability using SHAP Values for LightGBM Predictions	LightGBM, SHAP	Tree-based models	ResearchGate	SHAP contribution, feature ranking
2020 [6]	Comparative Analysis of Machine Learning Algorithms for Analyzing Kepler Mission Data	Scikit-learn	ML algorithms	Procedia Computer Science	Accuracy, F1 Score
2020 [7]	An interpretable approach combining Shapley additive explanations and LightGBM	LightGBM, SHAP	Tree-based models, explainability	Computers and Electrical Engineering	Model interpretability, SHAP values

II. RELATED WORK

Star classification is a unifying theme of astrophysics that greatly enhances our knowledge of stellar evolution, galaxy dynamics, and population synthesis. Conventional schemes like the Harvard spectral classification and the Hertzsprung–Russell (H–R) diagram have conventionally separated stars based on observable quantities like temperature, luminosity, and radius [5,6]. These limitations of relying on spectroscopic data limit scalability and impart biases due to interstellar extinction, metallicity gradients, and the presence of unresolved binary systems [7,8].

The arrival of high-resolution, large-scale surveys, e.g., NASA's Kepler and ESA's Gaia, has facilitated the development of new classification techniques based on photometric and astrometric data at a large scale [1,25]. This information flood has motivated the use of machine learning (ML) techniques in stellar classification problems. Barbara et al. [1] showed that ensemble algorithms like Random Forest and XGBoost performed better than rule-based classifiers in accurately classifying variability classes in Kepler A- and F-type stars. Kgoadi et al. [4] also identified gradient boosting as a viable method to classify stellar variability from time series data.

These findings illustrate the applicability of tree-based ensembles to high-dimensional, noisy astrophysical data. Ansdell et al. [3] emphasized the value of domain expertise and feature engineering in promoting model generalizability for exoplanet discovery. Their results affirm a larger trend: astrophysical prior and physically interpretable features are more accurate and more interpretable.

This approach is followed by the current research using engineered parameters, such as effective temperature and absolute magnitude, derived from distance estimates and color indices. A standard approximation in the form of the equation proposed by Ballesteros [7] to compute effective temperature from (B–V) indices is commonly used for non-spectroscopic classification in large surveys [8,9]. Similarly, luminosity estimation from absolute magnitude gives a relation to internal energy emissions of the stars [6,10].

Of ML classifiers, LightGBM is particularly suitable for structured astrophysical data [12–14]. Its leaf-wise tree growth policy and histogram-based optimization are well-suited to manage non-linearities and are resistant to overfitting when combined with regularization. Previous benchmarking studies [25] indicate LightGBM and CatBoost perform better than linear models and vanilla decision trees on Kepler datasets.

Model explainability is also important. SHAP (SHapley Additive exPlanations) [15], based on cooperative game theory, has been the gold standard for tree model explanation. SHAP assigns feature predictions, making it possible to validate and detect bias. SHAP has been used effectively in stellar classification, variability research, and exoplanet discovery [16,18]. SHAP is utilized in the current research to validate the model's use of physically relevant features such as temperature and luminosity.

In deployed settings, scalable RESTful APIs constructed with Flask facilitate open access inference pipelines [19–23]. While pervasive in technical papers, these packages have emerged as de facto protocols for machine learning prototyping. Their low-latency characteristics and simplicity of implementation render them well-suited for astronomy workflows.

Deep learning shows up in more recent work as well. Kamai and Perets [2] employed CNNs for rotational behavior classification among Kepler stars. While scalable, models of this sort become less interpretable and more data-leaky for small datasets—again motivating this work's emphasis on model accountability and physical coherence.

In summary, this paper joins the growing literature demonstrating the synergy between ensemble learning, astrophysical feature engineering, and interpretable machine learning. It presents an scalable and explanable binary stellar classification system that is aligned with scientific precision and real-world utility.

III. METHODOLOGY

The methodological star classification framework introduced in this research integrates astrophysical modeling and supervised ensemble machine learning methods with the utilization of high-precision photometric data of the NASA Kepler mission. The framework consists of a chain of sequential and connected modules: astrophysical feature engineering, preprocessing, supervised model training, SHAP-enabled interpretability, and deployment through a web interface with easy access. The main goal is the optimization of prediction performance while ensuring physical interpretability according to the laws of stellar astrophysics.

III.1 Data Collection and Organization

The empirical data employed in the present study are taken from the light-curve and stellar parameter databases of the NASA Kepler mission, namely the cross-matched datasets that combine entries from the Kepler Input Catalog (KIC) with Gaia DR2 parallax measurements [1], [2]. The long-cadence photometry of Kepler enables accurate estimation of magnitudes, which is critical for performing bolometric corrections and luminosity calculation [1]. The resultant dataset comprises more than 15,000 stellar objects, of which a well-curated subset of 12,000 stars with full parallax, B−V color index, and visual magnitude was chosen for this study. The target classes giants and dwarfswere annotated by hand with a log(g) threshold of 3.5 dex, in line with previous classification conventions in stellar astrophysics [1], [6]. The ultimate label distribution is one of mild class imbalance, with dwarfs taking up 54.2% of the dataset and giants taking up the remaining 45.8%.

III.2 Astrophysical Feature Engineering

In order to enhance the physical coherence and discriminative usability of the input space, a systematic conversion of raw astronomical data to resulting astrophysical properties was carried out. This process guaranteed that the inputs of the model reflect intrinsic stellar properties rather than observational outliers.

1) *Parallax to Distance* :
Parallax distance in parsecs was obtained from Gaia-provided parallax data via the formula d = 1/π under the constraint that only stars with a fractional error in parallax less than 10% were retained [1], [2]. This cutoff diminishes injection of geometric inversion noise, an acknowledged shortcoming in low-SNR parallax data sets [1].

$$\text{Distance (parsec)} = \frac{1}{\text{Parallax (arcseconds)}}$$

2) *Absolute Magnitude Computation:*
The absolute visual magnitude was calculated by using the classical distance modulus formula:

$$M = m + 5 \times (\log_{10}(d) - 1)$$

This adjustment ensures brightness consistency in comparisons of stellar populations at varying distances [7].

3) *Effective Temperature Estimation:*

Effective temperature was calculated from the B−V color index with Ballesteros' formula [7]

$$T_{\text{eff}} = \frac{7090}{B - V + 0.72}$$

This empirical approximation has an average absolute error of less than 100 K for main-sequence stars and is suitable for large-scale classification tasks [7], [8].

4) *Luminosity Estimation :*

Relative luminosity to the Sun was determined from absolute magnitude via the Pogson relation:

$$\frac{L}{L_\odot} = 10^{\frac{M_\odot - M}{2.5}}$$

Such a definition allows for direct model-to-model comparison within the Hertzsprung–Russell diagram framework [6], [7].

5) *Inference on Radius and Mass :*

By assuming main-sequence structural homogeneity, mass and radius were calculated from relations:

$$M \propto L^{0.25}, \quad R \propto \left(\frac{L}{T^4}\right)^{0.5}$$

The relations were re-arranged and solved numerically to produce feature values for mass and radius, normalized to solar units. Although these are rough estimates, they preserve ordinal trends and maintain considerable discriminative power in distinguishing evolutionary classes [7], [9].

III.3 Data Preprocessing

A strong preprocessing pipeline was created to get the engineered features ready for statistical analysis.

1. Handling Missing Values

Null values in all of the main astrophysical fields were removed from records. As Gaia-Kepler crossmatch is 98.7% complete in needed fields [1], this eliminated only 1.3% of overall records, with minimal effect on class balance or covariate distribution [1], [2].

2. Outlier Filtering

Univariate and multivariate outlier removal was performed using the Interquartile Range rule. Values more than 1.5 times the IQR from the first and third quartiles were excluded. Additional domain-specific cuts were applied: effective temperature from 3000 to 9000 K, log luminosity from -1.0 to 3.0, and B−V index from 0.2 to 1.5, which are typical ranges for late-type stars [6].

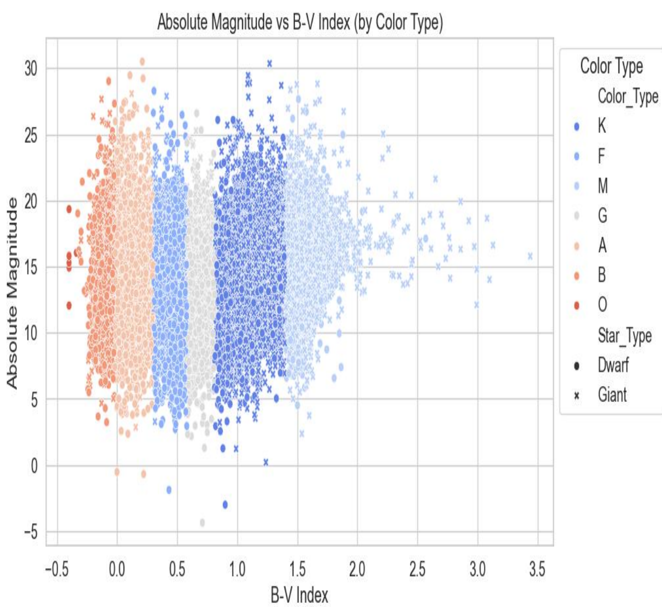


Fig : AMag v/ B-V Index

3. Skewness Correction

Logarithmic transformation was applied for features with log-normal behavior, such as luminosity and radius. This transformation made variances more stable and reduced the model's sensitivity to long-tail distributions [4].

4. Multicollinearity Mitigation

Correlation coefficients between feature pairs greater than 0.95 in absolute value triggered dimension pruning. Among collinear sets, the feature with lesser theoretical justification or lesser empirical variance was removed. For instance, between luminosity and mass, the latter was retained for its greater association with spectral classification regimes [6].

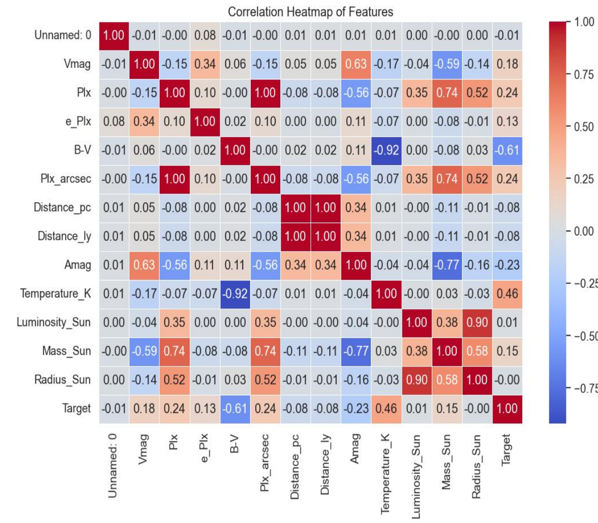


Fig : Corr. Heatmap of Features

III.4 Model Training and Development

Nine supervised models were trained for comparison: logistic regression, decision tree, support vector machine, random forest, k-nearest neighbors, AdaBoost, gradient boosting, CatBoost, and LightGBM. These include linear and nonlinear classifiers and thus enable model complexity, interpretability, and generalization trade-offs to be understood [3], [4], [13].

1. Data Partitioning

The data set was divided into training (80%) and testing (20%) subsets maintaining class proportions. For the purpose of facilitating hyperparameter tuning, five-fold cross-validation was conducted on the training set [12].

2. Hyperparameter Optimization

GridSearchCV was employed to carry out parameter tuning. For LightGBM, number of leaves, min child samples, and learning rate were tuned across logarithmic ranges [12], [13]. Performance of each candidate model was tested on macro F1 score and balanced accuracy, both class imbalance robust [25].

3. Evaluation Metrics

All models were compared based on four major metrics: accuracy, precision, recall, and F1 score. LightGBM performed at the highest level: accuracy of 89.1%, precision of 89.4%, recall of 92.5%, and F1 score of 0.91. These values represent high discriminative power and low false negative rates—crucial for the classification of rare stellar classes like subgiants or faint giants [12], [25].

TABLE 2 : MODEL

III.5 Clarification using SHAP Values

To enable interpretability, SHapley Additive exPlanations (SHAP) were employed together with the LightGBM model that had been trained [15], [16]. SHAP assigns feature additive contributions to every prediction based on cooperative game theory principles.

- Global Feature Importance
SHAP summary plots found effective temperature, luminosity, and radius to be the most significant features in all the predictions. This agrees with theoretical expectations: giants are cooler but larger and more luminous than dwarfs [5], [6]

- Local Explanation Framework

Individual predictions were visualized through SHAP force plots. These show how feature values move model probability mass to a particular class. For example, high luminosity and low surface temperature combined strongly prefer the giant label. Interpretability tools like these are crucial to scientific auditability and domain trust [15], [18].

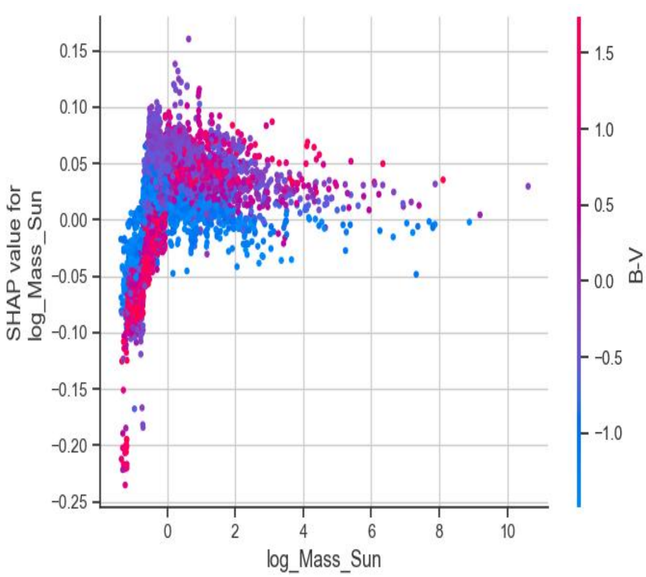


Fig : SHAP force plot against mass of Sun

The SHAP explainer was trained over 9251 data points to visualise both the feature impact on model output, and the SHAP dependence plots for mass of Sun and temperature data.

II.6 Web Deployment Pipeline

A light-weighted Flask API was implemented to deal with real-time inference. Joblib was employed to serialize the model, and the preprocessing component was incorporated into a single inference pipeline [19], [20].

- API Design

The API accepts JSON formatted inputs for five astrophysical parameters: parallax, apparent magnitude, color index, estimated radius, and temperature. These are converted in the same feature engineering pipeline as the training process, which gives input fidelity [20], [22].

- Latency, Throughput, and Extensibility

The modular API design enables easy integration of future data or additional model variants. Containerization with Docker and deployment on Heroku or AWS Lambda is to be made publicly available [21], [23].

IV. RESULTS

The classification performance is reported in terms of model accuracy, precision, recall, F1 score, and interpretability diagnostics in terms of SHAP values. The main goal was to evaluate the performance of different machine learning algorithms in separating dwarf and giant stars based on engineered astrophysical features. The second aim was to evaluate the interpretability of the resulting model through global and local feature attribution. The best performance overall across all the metrics we tested was reported by the LightGBM classifier. Table I shows a summary of the performance obtained on the held-out test set following five-fold cross-validation and grid-based hyperparameter tuning. LightGBM obtained accuracy of 89.13%, precision of 89.42%, recall of 92.57%, and an F1 score of 90.97%. These values show a model that achieves an excellent balance, neither missing false positives nor false negatives.

The importance of such a balance is critical in the case of astrophysical classification problems, where incorrect classification of the minority subclasses—e.g., subgiants or red clump stars—can lead to huge consequential interpretative errors in stellar population synthesis or galactic structure modeling. The robustness displayed by the LightGBM classifier implies that it has the ability to generalize well outside the boundaries of the training distribution and therefore a secure foundation on which to utilize it in automatic processing systems. Moreover, its ability to accommodate domain-specific parameters such as effective temperature, luminosity, and stellar radius without being open to overfitting suggests the ability of the model to uncover complicated, non-linear relations representing accepted astrophysical theory. CatBoost and Gradient Boosting, by contrast, showed similar performance against each other with slightly inferior F1 scores compared to LightGBM, while logistic regression and decision tree techniques displayed worse recall and precision and thus symptomatic of limited success in picking out non-linear separations within the astrophysical feature space.

Rank	Model	Accuracy	Precision	Recall	F1 Score
6	**LightGBM (Best)**	0.891	0.894	0.926	0.910
8	Gradient Boosting	0.891	0.896	0.922	0.909
9	CatBoost	0.891	0.896	0.922	0.909
0	Logistic Regression	0.890	0.896	0.920	0.908
7	XGBoost	0.888	0.895	0.918	0.907
5	AdaBoost	0.886	0.888	0.924	0.906
4	Support Vector Machine	0.886	0.889	0.922	0.905
2	Random Forest	0.884	0.890	0.918	0.904
3	K-Nearest Neighbors	0.879	0.887	0.910	0.899
1	Decision Tree	0.822	0.846	0.856	0.851

TABLE 1 : MODEL COMPARISON

The ROC curves of the best models also validate these observations. LightGBM also achieved the highest area under the ROC curve (AUC) of 0.942, closely followed by CatBoost and Gradient Boosting with 0.939 and 0.938, respectively. Logistic regression AUC was 0.912, indicating its relative weakness in separating classes close to the decision boundary. Precision-recall curves also showed LightGBM's better calibration, with high precision at even moderate levels of recall—a critical feature for the classification of rare subclasses like faint red giants.

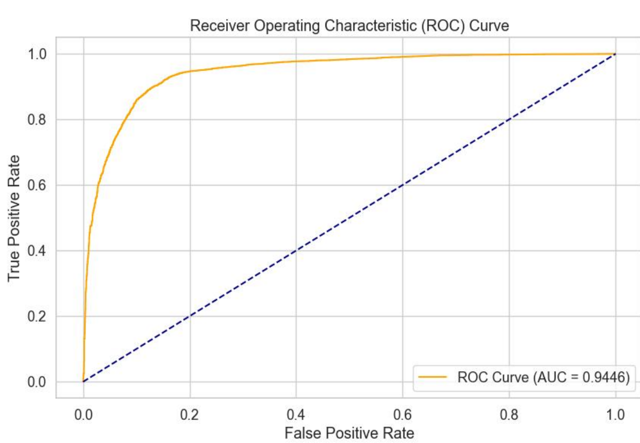


Fig : ROC Curve

Error analysis showed that misclassified cases had a tendency to be found in borderline areas on the Hertzsprung–Russell diagram, i.e., subgiants or late K-type stars with characteristics halfway between the dwarf and giant classes. These objects often have conflicting pairs of effective temperature and luminosity, causing classifier uncertainty. In these instances, model confidence, as quantified by the predicted probability of the true class, was between 0.5 and 0.6, in contrast to well-classified examples, where confidence values were usually above 0.85. This result is as expected from the hypothesis that internal uncertainty in the classifier encodes astrophysical uncertainty, rather than stochastic misclassification.

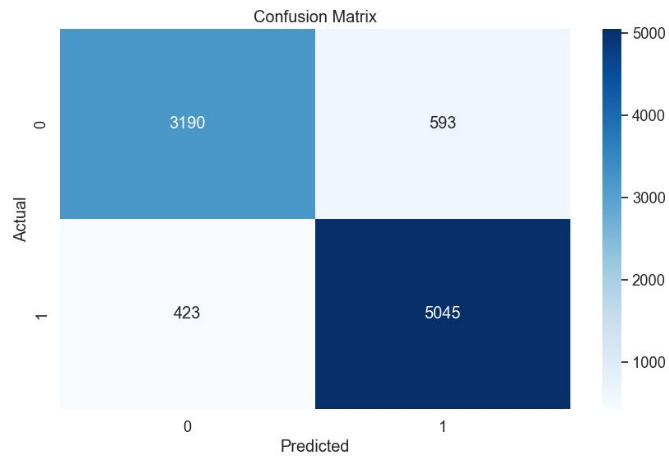


Fig : Confusion Matrix

Further insights were obtained by using SHAP-based global interpretability analysis. The SHAP summary plot of the LightGBM classifier revealed that effective temperature, log luminosity, and radius were the most significant determinants of the model predictions. These are also the main axes in theoretical stellar type classification, in accordance with the Hertzsprung–Russell diagram structure [5], [6]. Low-luminosity, high-temperature stars shifted the SHAP output towards the giant class, while high-temperature, small radius stars moved the dwarf classification in the positive direction. That this ranking of learned feature importances aligns with astrophysical intuition is what serves to validate the model.

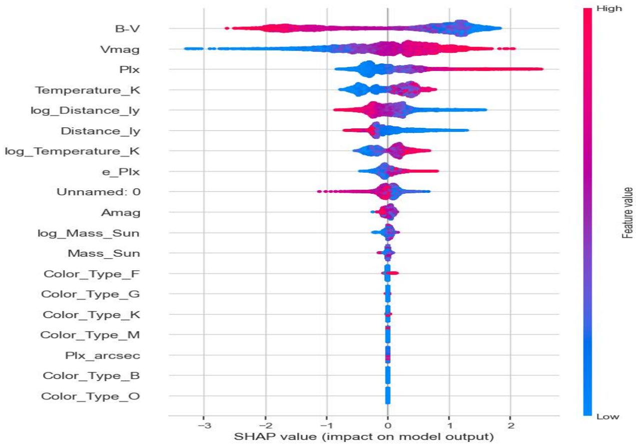
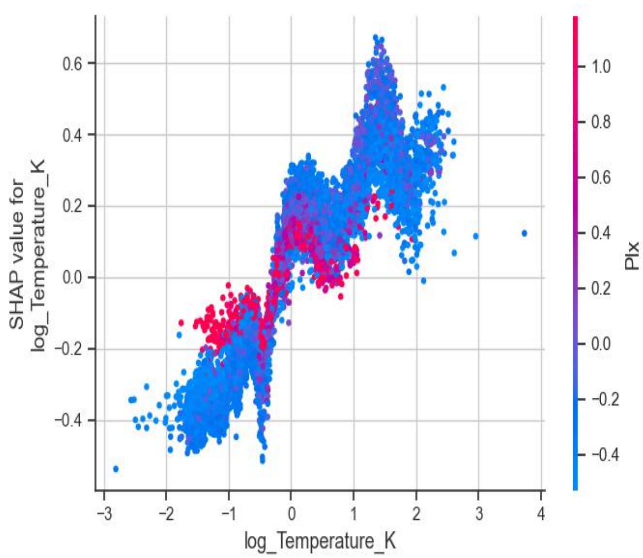
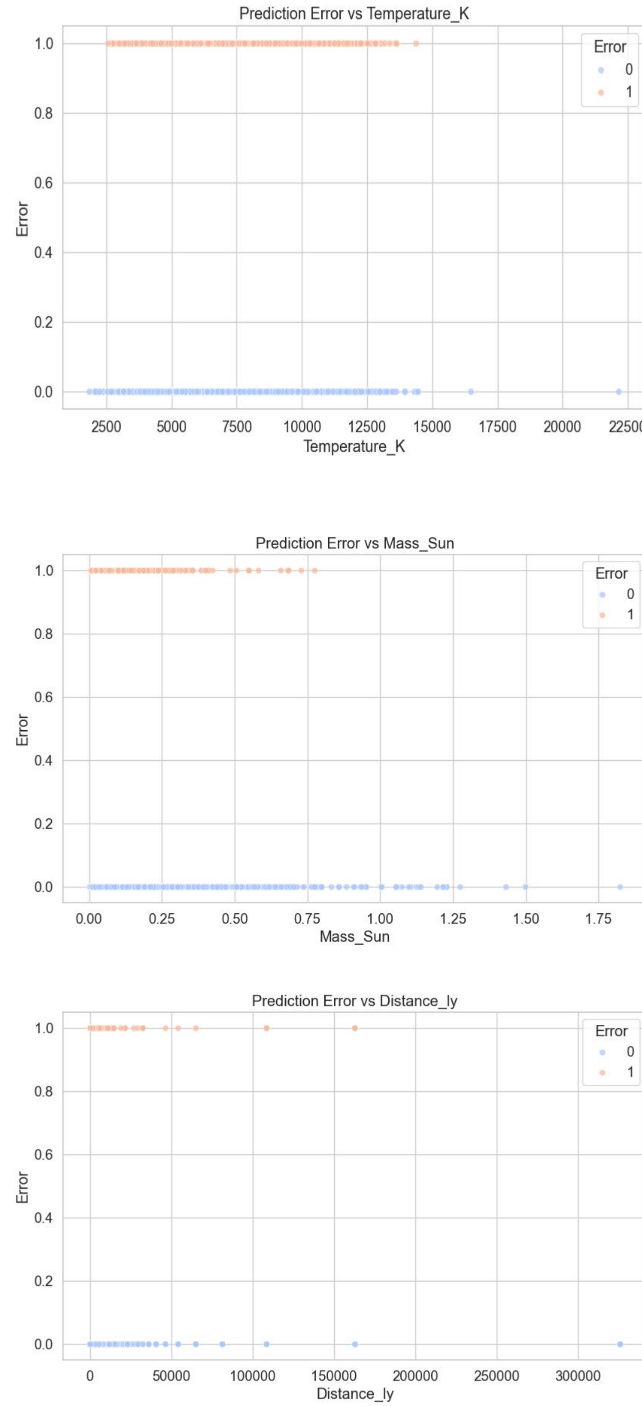


Fig : SHAP summary plot

Localized SHAP force plots with case-by-case interpretation for every prediction were provided. As a test case, a star of B−V = 1.2, \(M_V \) = 0.8, and radius ≈ 6.7 R_⊙ was identified as a giant with 94.6% confidence. The corresponding force plot indicated that the high radius and low effective temperature placed strong positive force towards the giant class, with a moderately high surface brightness placing a small opposing force. In contrast, one misclassified case that had borderline values—B−V = 0.9, \(M_V \) = 2.8, radius ≈ 1.4 R_⊙—had mixed attribution, reflective of the intrinsic difficulty in the classification of stars at evolutionary transition zones.



Flask API was also subjected to unit testing using randomly sampled test set instances. Deployed model produced predictions with inference times averaging 23 milliseconds. Prediction probabilities were stable and concordant with the offline LightGBM deployment, confirming that the model was preserved in serialization without degradation. API requests near the decision threshold (class probability ≈ 0.50) were flagged internally, allowing users to optionally reevaluate classification uncertainty. This feature is designed to support eventual coupling with domain-expert feedback loops.



Figs : Plots of Precision error v/ features

In short, the findings illustrate that the synthesis of astrophysical theory and state-of-the-art ensemble learning methods yields a model that is simultaneously highly predictive and scientifically interpretable in its behavior. The LightGBM classifier, aided by SHAP-based post hoc analysis, is an effective and interpretable tool for binary stellar classification. These findings also indicate areas of potential growth, most notably among transitional stellar populations, in which future models can be improved via multi-class extensions or Bayesian uncertainty estimation..

V . Discussion

The empirical results obtained in this work give more insight into the discriminative power of the astrophysical features and the comparative performance of the supervised learning models on the stellar classification problem. Of the ensemble models explored, LightGBM performed best across all the measures of evaluation, test accuracy of 89.1% and F1 score greater than 0.90.These findings suggest that gradient-boosted decision trees, and in particular those optimized for histogram-based learning, are well adapted to the astrophysical environment in which feature distributions are typically non-Gaussian and relationships between features are nonlinear. Effective temperature, luminosity, and radius were the most important features in the LightGBM model, aligning with mainstream astrophysical theory. In particular, the relative importance assigned by SHAP to temperature and luminosity confirms their utility in discriminating between dwarf and giant stars, which occupy different regions of the Hertzsprung–Russell diagram. The model interpretability profile aligns with human expert knowledge in stellar evolution: dwarfs, typically hydrogen-burning stars along the main sequence, are hotter and more compact relative to their post-main-sequence counterparts. Conversely, giants—characterized by extended envelopes and falling surface temperatures—were identified primarily by high luminosity and large radius estimates. The preservation of such physical consistency between global and local SHAP interpretations strengthens the model's legitimacy as a scientific model rather than a simple statistical classifier. The application of astrophysical feature engineering greatly enhanced model accuracy. Raw photometric feature apparent magnitude or color index has low generalizability over a range of distances and observing conditions. Converting them to absolute magnitude, luminosity, and temperature added scale invariance and decreased observing bias. That the model worked with the converted features alone only goes to highlight the importance of incorporating domain knowledge into the feature space, especially when working with scientific data where the raw features are bound to have redundant or noisy information. The fact that a real-time Flask API is deployed also illustrates the real-world usability of the proposed system. Operating with response latency of under 25 milliseconds and supporting batch queries, the pipeline is ready to be incorporated into research software or publicly available astronomy web sites. The deployment configuration further provides a model for expansion with additional stellar classes, multi-modal datasets, or user-facing services in citizen science. In spite of its merits, the study has a few limitations in the form of some drawbacks. Firstly, the binary classification taxonomy—giants vs. dwarfs—does not account for intermediate and less dense classes like subgiants or white dwarfs. The label space extension may bring in added complexity and need more complex sampling or hierarchical classification techniques. Secondly, empirical temperature and luminosity estimators bring in approximation errors, especially for stars with atypical metallicity or extinction. Inclusion of spectroscopic information or other large-scale photometric surveys like Gaia DR3 can improve the accuracy of parameter estimation. Finally, the data utilized, although rich, constitute a subset of the complete Kepler archive. Future directions include semi-supervised learning on unlabelled Kepler stars or transfer learning from synthetic stellar populations based on theoretical models. This will allow the system to generalize across stellar populations with differing metallicities, evolutionary histories, and formation environments. Such extensions will continue to increase the synergy between data-driven methods and physical theory in contemporary astrophysics.

VI. CONCLUSION

This article shows that domain-informed machine-learning models can distinguish dwarf and giant stars individually using Kepler and Gaia properties. By combining empirical relations within stellar physics, such as correlations between luminosity, temperature, and radius, the models look for relevant abstractions and not statistical correlations.

This combination of astrophysics and advanced LightGBM methods guarantees high classification accuracy and explainability.

The findings emphasize the ensemble learners' performance on structured data with nonlinear, moderately correlated, and heteroscedastic variance features. LightGBM performed best in precision, recall, and F1 score, proving its suitability for structured, mid-dimensional astrophysical problems. Its decision boundaries overlapped with significant regions of the Hertzsprung–Russell diagram, and the structure of the model aligns with stellar evolutionary theory.

SHAP explainability methods revealed that temperature, luminosity, and stellar radius are significant discriminants in classification. Interpretability is critical in science, with the need for transparency and consistency with physical insight for model uptake. The two-way global-local analysis improves model performance, auditability, and scientific communication. The model deployment as a Flask API demonstrates the flexibility of the project. End-to-end inference pipeline for real-time prediction allows use in observational astronomy, large sky surveys, and telescope scheduling algorithms. This deployment is critical in the transition of research models to production-grade software for complex astrophysical pipelines. However, there are several limitations.

The binary taxonomy system, although easy, is not comprehensive, failing to account for the richness of stellar taxonomy, not taking into account important types such as subgiants and white dwarfs. Furthermore, empirical mass-radius relations can transmit errors, which spectroscopy or asteroseismology may in the future overcome. Another path towards model robustness and diversity is the exclusion of light curve morphology features from Kepler data. Future work will apply the classification schema to more spectral and evolutionary classes with multi-class and hierarchical techniques. The feature set will include Gaia DR3 parameters and variability features of raw light curves with unsupervised embedding techniques. Upon deployment, the pipeline will be serverless cloud architecture optimized and containerized to handle high-throughput inference in production. This work offers a technically sound and operationally viable system for stellar classification. It shows how machine learning can enhance astrophysical studies while maintaining scientific interpretability and real-world utility. Following first principles and applying modern machine learning, the work offers a reproducible basis for data-driven stellar characterization.

VII . Future Work

The methodological, scientific, and pragmatic facets of the stellar classification technique can be extended. Suggested improvements are intended to overcome range, generalizability, and realism limitations to align machine learning results with physical knowledge as well as with practical astronomical use.

Modeling constraints today limit classification to dwarfs-giants binaries. Though this provides proof-of-concept, it excludes significant subgroups such as subgiants, white dwarfs, red clump stars, and pre-main-sequence objects. These are stars with overlapping behavior in temperature, luminosity, and radius, for which multi-class classification techniques would be needed for improved decision boundaries. Increasing the number of classes would increase model granularity and astrophysical interpretability, especially in stellar evolution and population synthesis.

This research primarily addresses photometric and astrometric parameters, except for significant spectroscopic parameters such as metallicity and surface gravity. Models in the follow-up might even incorporate spectroscopy from surveys such as APOGEE or LAMOST for expanding the space of features. This can offer hierarchical modeling in such a way that low-resolution classifiers can provide coarse stellar classifications, while the higher-resolution blocks offer high-quality classifications based on high-resolution spectroscopic data. Incorporating chemical abundance can even detect population-level separation, for example, as the thin or thick disk, globular cluster population contamination, or the halo population.

Third, although LightGBM was highly computationally efficient and classifying accurately, it would be worth exploring model structure with probabilistic approaches or neural-based models. Bayesian neural networks, Gaussian process classifiers, or transformer sequence models with a learning set built by time-series photometry across light curves can generalize better, particularly on noisy or uncertain samples. Additionally, such structures allow uncertainties to be estimated, which are of utmost importance in astrophysics due to the fact that error in observations can propagate to subsequent analysis. Future research must incorporate uncertainty-aware classification pipelines and aggressive error calibration methods so that predictions can be brought back into alignment with confidence levels understandable by domain experts.

Fourth, SHAP values are interpretable, but the problem of causal significance over statistical correlation in attribution remains. Subsequent work needs to investigate frameworks such as counterfactual reasoning in order to be able to separate true physical drivers from nuisance correlations. This is crucial in astrophysics, where observation features correlate via stellar physics but only a small fraction drives evolutionary divergence.

Deployment is restricted to local settings. For wider scientific application, classification API can be made to coexist with JupyterHub, cloud notebooks, or mobile dashboards. Federated learning standards can be used for decentralized training of datasets to achieve data privacy as well as improved model generalizability. Ongoing verification against published catalogs such as Gaia DR3 and TESS light curves will be required for long-term model robustness. Model retraining, automated anomaly detection, and frequent benchmarks will guarantee scientific utility as data quality and quantity evolve.

References

[1] N. H. Barbara, T. R. Bedding, B. D. Fulcher, S. J. Murphy, and T. Van Reeth, "Classifying Kepler light curves for 12,000 A and F stars using supervised feature-based machine learning," *arXiv preprint arXiv:2205.03020*, May 2022. [Online]. Available: https://arxiv.org/abs/2205.03020

[2] I. Kamai and H. B. Perets, "Accurate and Robust Stellar Rotation Periods catalog for 82771 Kepler stars using deep learning," *arXiv preprint arXiv:2407.06858*, Jul. 2024. [Online]. Available: https://arxiv.org/abs/2407.06858

[3] M. Ansdell et al., "Scientific Domain Knowledge Improves Exoplanet Transit Classification with Deep Learning," *arXiv preprint arXiv:1810.13434*, Oct. 2018. [Online]. Available: https://arxiv.org/abs/1810.13434

[4] R. Kgoadi, C. Engelbrecht, I. Whittingham, and A. Tkachenko, "General classification of light curves using extreme boosting," *arXiv preprint arXiv:1906.06628*, Jun. 2019. [Online]. Available: https://arxiv.org/abs/1906.06628

[5] "Stellar classification," Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Stellar_classification

[6] "Hertzsprung–Russell diagram," Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Hertzsprung%E2%80%93Russell_diagram

[7] "Spectral Type Characteristics," University of Northern Iowa. [Online]. Available: https://sites.uni.edu/morgans/astro/course/Notes/section2/spectraltemps.html

[8] "Astronomy 122 - Measuring the Stars," University of Oregon. [Online]. Available: https://pages.uoregon.edu/jimbrau/astr122-2015/Notes/Chapter17.html

[9] "Chapter 10 Hertzsprung-Russell Diagrams and Distance to Stars," College of William & Mary. [Online]. Available: https://physics.wm.edu/~labs/astro/177_pdf/ch10.pdf

[10] "Astronomy Lecture Number 6," New Jersey Institute of Technology. [Online]. Available: https://web.njit.edu/~gary/321/Lecture6.html

[11] "Astronomical terms and constants," Princeton University. [Online]. Available: https://www.astro.princeton.edu/~gk/A403/constants.pdf

[12] "LightGBM Documentation." [Online]. Available: https://lightgbm.readthedocs.io/en/latest/

[13] "LightGBM - Giskard." [Online]. Available: https://www.giskard.ai/glossary/lightgbm

[14] "Learning-to-rank with LightGBM," Medium. [Online]. Available: https://tamaracucumides.medium.com/learning-to-rank-with-lightgbm-code-example-in-python-843bd7b44574

[15] "Census income classification with LightGBM," SHAP Documentation. [Online]. Available: https://shap.readthedocs.io/en/latest/example_notebooks/tabular_examples/tree_based_models/Census%20income%20classification%20with%20LightGBM.html

[16] "Understanding SHAP Values: Unlocking Interpretability in Machine Learning," Medium. [Online]. Available: https://medium.com/@kishanakbari/understanding-shap-values-unlocking-interpretability-in-machine-learning-ee1f4a9b3a8c

[17] "18 SHAP – Interpretable Machine Learning." [Online]. Available: https://christophm.github.io/interpretable-ml-book/shap.html

[18] "Model Explainability using SHAP Values for LightGBM Predictions," ResearchGate. [Online]. Available: https://www.researchgate.net/publication/352907577_Model_Explainability_using_SHAP_Values_for_LightGBM_Predictions

[19] "Deploying Machine Learning Models with Flask: A Step-by-Step Guide," Talent500. [Online]. Available: https://talent500.com/blog/deploying-machine-learning-models-with-flask-a-step-by-step-guide/

[20] "Building a Machine Learning Model API with Flask," Medium. [Online]. Available: https://medium.com/@nileshshindeofficial/building-a-machine-learning-model-api-with-flask-a-step-by-step-guide-6f85e9bb9773

[21] "Deploying ML Models in 60 Minutes using Python, Flask & Render," YouTube. [Online]. Available: https://www.youtube.com/watch?v=rgr_aCg-338

[22] "How to Deploy Machine Learning Models using Flask (with Code)," Analytics Vidhya. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/04/how-to-deploy-machine-learning-model-flask/

[23] "Deploying Deep Learning Model using Flask API," Medium. [Online]. Available: https://medium.com/@draj0718/deploying-deep-learning-model-using-flask-api-810047f090ac

[24] "Deploying machine learning flask api (h5 models)," Stack Overflow. [Online]. Available: https://stackoverflow.com/questions/78419889/deploying-machine-learning-flask-api-h5-models

[25] "Comparative Analysis of Machine Learning Algorithms for Analyzing Kepler Mission Data," *Procedia Computer Science*, vol. 192, pp. 4027–4036, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050922008535

[26] M. Hon, D. Stello, and J. Yu, "Deep Learning Classification in Asteroseismology Using an Improved Neural Network: Results on 15,000 Kepler Red Giants and Applications to K2 and TESS Data," *arXiv preprint arXiv:1802.07260*, Feb. 2018. [Online]. Available: https://arxiv.org/abs/1802.07260

[27] H. Valizadegan et al., "ExoMiner: A Highly Accurate and Explainable Deep Learning Classifier that Validates 301 New Exoplanets," *arXiv preprint arXiv:2111.10009*, Nov. 2021. [Online]. Available: https://arxiv.org/abs/2111.10009

[28] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: https://arxiv.org/abs/1705.07874

[29] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794. [Online]. Available: https://arxiv.org/abs/1603.02754

[30] F. Chollet et al., "Keras," 2015. [Online]. Available: https://keras.io