Simulation-Pretrained Foundation Models for Domain-General Astronomical Time Series Tasks with Minimal Labeled Data

Anonymous Authors[1]

Abstract

Astronomical time series analysis faces a critical limitation: the scarcity of labeled real data. We present a foundation model approach that leverages physics-informed simulations as pretraining data, significantly reducing the need for labeled examples. Our models, trained on simulated data from multiple telescopes, learn generalizable representations that transfer effectively to downstream tasks. Using classifier-based architectures enhanced with novel contrastive and adversarial objectives, we create domain-agnostic models that recognize similar astronomical phenomena across different instrumental contexts. These models demonstrate substantial performance improvements over previous methods on classification, redshift estimation, and anomaly detection tasks when fine-tuned with minimal real data. Remarkably, our models exhibit effective zero-shot transfer capabilities, achieving comparable performance on future telescope (LSST) simulations when trained solely on existing telescope (ZTF) data. Furthermore, they generalize to entirely different astronomical phenomena (Kepler periodic variables) despite being trained on transient events, demonstrating cross-domain capabilities similar to large language models. Our approach provides a practical solution for building robust time series foundation models when labeled data is scarce but domain knowledge can be encoded in simulations.

1. Introduction

Time series analysis in astronomy often requires substantial labeled data for supervised learning approaches. Models have been developed to classify variable stars and transient events[1] (e.g. Narayan et al., 2018; Muthukrishna et al., 2019; Rehemtulla et al., 2024), detect anomalies (Perez-Carrasco et al., 2023; Muthukrishna et al., 2022; Villar et al., 2021), and estimate physical parameters like redshift (e.g Qu & Sako, 2023; Zhang et al., 2024). However, these models are typically trained for specific instruments or tasks, missing opportunities to leverage commonalities across astronomical datasets. Furthermore, they struggle with new telescopes where labeled data is initially scarce.

Foundation models (FMs) have transformed natural language processing and computer vision by learning generalizable representations from large quantities of data. However, astronomical data presents unique challenges: (1) limited publicly available labeled data compared to other domains; (2) instrument-specific characteristics that hinder cross-survey generalization; and (3) complex physical phenomena that require domain expertise to model properly.

Astronomy has a significant advantage over many domains: decades of physical understanding encoded in simulations. Astronomers have developed detailed models of astrophysical phenomena that generate synthetic light curves (e.g. PLAsTiCC Modelers, 2019). While these simulations cannot perfectly replace real observations (Gupta & Muthukrishna, 2025), they effectively encode domain knowledge that can bootstrap learning through the training of foundation models.

Motivated by the success of supervised classifier-based methods for anomaly detection (Gupta et al., 2024), we propose a novel approach to building foundation models for astronomical time series that:

1. Leverages the latent space of classifiers pretrained on physics-informed simulations
2. Develops domain-agnostic representations through adversarial and contrastive learning objectives that mirror how humans naturally view the same astronomical objects consistently across different telescopes
3. Enables effective downstream task performance with minimal labeled real data
4. Facilitates zero-shot transfer to new telescopes

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

[1]See Figure 1 for example time series

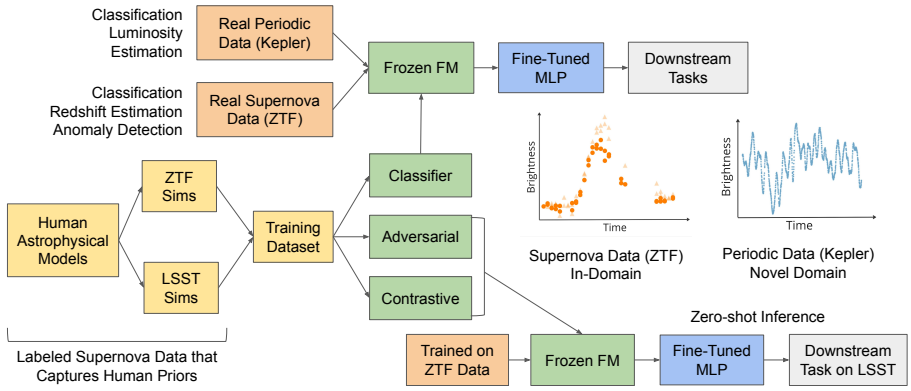

Figure 1. A visual summary of the methodology described in this work. We first train various classifier-based and domain-agnostic foundation models using human-generated simulated data. We then evaluate these models on various downstream tasks, including zero-shot estimation for new telescopes.

This approach is particularly valuable for upcoming surveys like the Vera C. Rubin Observatory's Legacy Survey of Space and Time (LSST), which will produce millions of time-series alerts nightly (Ivezić et al., 2019). Having models ready to analyze LSST data from day one—without requiring extensive new labeled datasets—would dramatically accelerate scientific discovery.[2]

2. Datasets and Benchmarks

Modern telescopes capture a wide range of astronomical data, including galaxy images and spectroscopic measurements. This time-series data spans numerous wavelength channels and is irregularly sampled. Further, instrumental differences make similar objects look vastly different across telescopes. Our work focuses on developing models for supernova discovery through time series, driven largely by the upcoming Legacy Survey of Space and Time (LSST). To pretrain our foundation models, we use simulated supernova data from both LSST and an existing telescope, the Zwicky Transient Facility (ZTF), which observes similar supernova to LSST but at a significantly smaller scale. The simulations have been designed to reflect the observational characteristics of each instrument. After our models are trained, we evaluate them on several downstream tasks using real observational data. We divide each task two scenarios (`Limited` with 128 labeled objects and `Full` with all the labeled data) reflecting the amount of labeled data available during training. To assess generalization and adaptability, we further evaluate our models on different types of astronomical events, specifically periodic time-series from Kepler. Real data is prioritized for evaluation as it represents the intended application domain of our models, whereas models are pretrained using simulations. In cases where real observations are unavailable (particularly for LSST[3]), we evaluate on simulations. Appendix A contains further information and a comprehensive list of benchmarks.

[2]The code used in this work is publicly available: https://anonymous.4open.science/r/astrofm-E4BB/ and https://anonymous.4open.science/r/Kepler-FM-450B/

3. Methods

To build effective foundation models, we propose building domain-agnostic classifiers using existing simulated data. Specifically, we build a classifier using simulated data from both ZTF and LSST and use this model for downstream tasks on real data. We also build domain-agnostic models by incorporating additional adversarial and contrastive loss components to ensure the model learns cross-survey relationships. More specific training and model information can be found in Appendix B.

3.1. Classifier

Neural network classifiers have demonstrated the ability to capture the underlying structure of astronomical transients and have been applied to various tasks beyond classification (Gupta et al., 2024). Motivated by these prior successes, we propose building a foundation model that is a classifier trained on data from both ZTF and LSST. To achieve this, we first ensure that the class labels across both datasets are identical and unified. Once the classifier has been trained, we discard the output classification layer, leaving the penultimate layer as the output of our FM. This penultimate layer, henceforth referred to as the latent space, provides an effective feature space because it captures meaningful structure and exhibits coherent clustering patterns (Gupta et al., 2024). Although the classifier successfully learns a structured representation of the transients (as shown in Figure 4 of Gupta et al., 2024), it does not produce a domain-agnostic latent

[3]LSST is expected to begin observations in 2026

space. Appendix C qualitatively discusses this issue with visualizations of the latent space (Figures 5 and 6).

3.2. Adversarial Training

To encourage domain-agnostic representations across observatories, we adopt an adversarial training framework. This method promotes the unification of feature representations for similar transients originating from different surveys, such as ZTF and LSST. Specifically, we jointly train a classifier $C(X_i)$, which outputs both class predictions and provides the intermediate latent representations, and a discriminator $D(L_i)$, which attempts to predict the observatory domain. The classifier is trained to both classify the transient correctly and to confuse the discriminator, thereby learning a latent space that is useful for classification yet agnostic to the survey domain. The discriminator, on the other hand, is trained to distinguish between observatories using the latent representations. The adversarial training process is detailed further in Appendix B.2.

3.3. Supervised Contrastive Training

While adversarial training encourages ZTF and LSST transients to be embedded more closely in the latent space, it does not explicitly enforce that objects of the same class should be clustered together. Since we know that class identity is a strong indicator of similarity, we propose using a supervised contrastive loss (Khosla et al., 2020). Rather than relying on data augmentations like traditional contrastive losses, we treat all samples belonging to the same class as positive pairs and apply the contrastive objective accordingly. This modification provides a more explicit signal for class-based alignment and naturally encourages unification across domains, as long as examples from the same class are drawn from both surveys. We use the contrastive loss proposed in Chen et al. (2020) for model pretraining. Further details can be found in Appendix B.3.

Our adversarial and contrastive models are trained to learn an explicit relationship between ZTF and LSST, and once these models are trained on ZTF, they can be directly applied to LSST. However, for downstream tasks on a single survey, classifiers perform as well as these domain-agnostic models.

3.4. Downstream Tasks

Fine Tuning Foundation Models To fine-tune our classifier-based foundation models for regression and classification tasks, we attach a multi-layer perceptron (MLP) to the penultimate neural network layers of our models. We then freeze the initial model and fine-tune the MLP for downstream tasks, however this does not work effectively for all tasks. Further details on our fine-tuning setups can be found in Appendix B.4 and in the released code.

Baselines and Our Models When evaluating our models for downstream tasks, we compare them with models trained directly on a downstream task (`No Pretraining`). This is largely how models for real data have been trained in past research. We propose three models for evaluation: a `Classifier`, an `Adversarial` model, and a `Contrastive` model, all trained on both ZTF and LSST data. We evaluate these three methods of pretraining on various downstream and zero-shot tasks. While our classifiers are built for ZTF and LSST, we also evaluate pretraining `Classifiers` on datasets individually. In Table 1, we clearly disambiguate which dataset the model used for pretraining. In Table 2, `Classifier` refers to our proposed foundation model trained on both ZTF and LSST data.

4. Results

We evaluate our foundation models on multiple downstream tasks using both real observational data and simulated data from different telescopes. Our experiments demonstrate that pretraining on physics-informed simulations provides substantial improvements over training from scratch, with particularly strong results for cross-survey generalization. Further analysis can be found in Appendix D.

4.1. Downstream Performance on Real Astronomical Data

As seen in Table 1, our classifier-based foundation models outperform previous baseline methods trained directly on numerous tasks. Overall, these models achieve better performance than SoTA no pretraining methods for astronomical tasks on real data. We also see performance improvements for tasks from the `Limited` to `Full` testing scenarios, which makes sense as more data for fine-tuning should result in better performance. This also indicates that the simulations used for pretraining do not perfectly reflect real data, which we further discuss in Appendix D.2.

Surprisingly, our models also improve performance on tasks on Kepler Data, even though Kepler data is not given to the model during pretraining. Kepler specifically looks for periodic transient events, in comparison to the supernova transients from ZTF and LSST. We find this to be similar to how LLMs perform well on tasks they were not trained on. We think this is because the physics systems behind both periodic and supernova objects are closely interconnected, just like how various language tasks are connected.

4.2. Cross-Survey Generalization: Zero-Shot Inference for LSST

Our contrastive and adversarially trained models are designed to be domain-agnostic and understand relationships between different telescopes in their latent space. To lever-

Model (Pretraining Data)	ZTF Real Data Classification		ZTF Real Data Redshift ($\times 10^2$)		ZTF Real Data AD		Kepler Real Data Classification	Kepler Real Data Luminosity	Simulations (Redshifting) ZTF	Simulations (Redshifting) LSST
	Limited	Full	Limited	Full	Limited	Full	Full	Full	Full	Full
Previous Work	0.637	0.853	0.602	0.385	0.498	0.527	0.901	0.57	0.079	0.289
No Pretraining	±0.005	±0.013	±0.005	±0.031	±0.013	±0.027	±0.003	±0.03	±0.022	±0.018
Classifier ZTF	0.875 ±0.020	0.904 ±0.012	0.491 ±0.010	0.387 ±0.036	0.605 ±0.025	0.596 ±0.008			0.028 ±0.003	0.252 ±0.004
Classifier ZTF and LSST	0.879 ±0.011	0.910 ±0.013	**0.479 ±0.006**	0.382 ±0.039	**0.622 ±0.018**	**0.616 ±0.036**	**0.968 ±0.006**	0.24 ±0.00	**0.026 ±0.002**	**0.177 ±0.008**
Contrastive	**0.886 ±0.026**	**0.914 ±0.005**	0.487 ±0.003	**0.373 ±0.017**	0.584 ±0.018	0.576 ±0.028	0.946 ±0.021	**0.22 ±0.02**	0.028 ±0.002	0.191 ±0.013
Adversarial	0.844 ±0.016	0.853 ±0.012	0.520 ±0.004	0.419 ±0.042	0.559 ±0.024	0.546 ±0.077	0.925 ±0.015	0.26 ±0.02	0.030 ±0.003	0.197 ±0.014

Table 1. Foundation model performance on various tasks. We train five different foundation models and fine-tune each of them five times. We report the mean and standard deviation of these recorded results. The final three model rows are the main contributions of this work and the first row is the current baseline and SoTA. The reported performance metrics are AUROC for classification an anomaly detection and MSE for other tasks. The best performance is bolded for each task.

Pretraining	Redshifting Data ZTF	Redshifting Data LSST	LSST MSE
No Pretraining	0%	100%	0.0750 ± 0.025
No Pretraining	100%	100%	0.0614 ± 0.003
Classifier	100%	100%	0.0579 ± 0.006
No Pretraining	100%	**0%**	0.1869 ± 0.012
Classifier	100%	**0%**	0.1035 ± 0.0081
Contrastive	100%	**0%**	0.0727 ± 0.0056
Adversarial	100%	**0%**	0.0744 ± 0.0063
Contrastive kNN	100%	**0%**	0.0854 ± 0.0040

Table 2. Performance of various models for LSST redshift estimation. Performance is reported as the mean and standard deviation of training five different foundation models and five different iterations of fine-tuning each of them. The first last 5 rows are zero-shot methods. Our zero-shot methods achieve similar performance to previous methods directly trained on redshifting LSST.

age this learned relationship, we first fine-tune our FM on a downstream task for ZTF and then evaluate this model's zero-shot performance on LSST. Importantly, we freeze the entire FM to preserve the learned relationship between the surveys during pretraining. If our FM is indeed encoding a unified latent space, the zero-shot performance on LSST should improve as we train on ZTF. When evaluating LSST redshifting in the zero-shot setting, we restrict the evaluation to LSST light-curves in the same redshift range as ZTF. This decision is discussed further in Appendix A.1.

As seen in Table 2, domain-agnostic FMs fine-tuned only ZTF redshifting data work exceptionally well when repurposed to redshift LSST transients. We compare these zero-shot methods to previous baseline methods that do not involve pretraining and our proposed method involving pretraining (similar to the fine-tuning done for Table 1). Models trained on both ZTF and LSST redshifting data (Rows 2 and 3 of Table 2) are first fine-tuned to redshift ZTF and then to redshift LSST. We describe the k Nearest Neighbors (kNN) zero-shot estimation method in Appendix E. Overall, our domain-agnostic models achieve the performance of baseline methods trained directly on LSST without any LSST data and significantly improve the performance of previous zero-shot methods.

Table 2 further reiterates that pretraining on adjacent domains and tasks produces SoTA models. The best model for this redshift estimation task outside the zero-shot scenario is a pretrained classifier fine-tuned on both ZTF and LSST redshifting data, essentially incorporating two tasks across two domains.

5. Conclusion

The shift from manual discovery to data driven discovery has motivated the development for machine learning in many scientific domains. Effective foundation models can expedite this process. To build such models for astronomy, we propose leveraging existing physics-informed simulations. Training specialized classifiers on human-generated simulated data proves to be an effective way to incorporate domain expertise into these models. Fine-tuning our models for tasks on real data achieves SoTA performance on numerous downstream tasks and has excellent zero-shot task performance. We believe that the development of foundation models for astronomy is the next major step in expediting discovery and we hope that this work facilitates future research in the development of FMs for science through supervised training.

We see numerous promising research directions for future work. Incorporating unlabeled data in model fine-tuning could yield better results by better exposing models to the structure of real data after being pretrained. Further, different methods of supervision could help models extract more meaningful information from physics-informed simulations. In conclusion, our work aims to bridge the gap between past research for machine learning for astronomy, with the current era of discovery necessitating the development of models that leverage all that we know for novel tasks.

References

Audenaert, J., Kuszlewicz, J. S., Handberg, R., Tkachenko, A., Armstrong, D. J., Hon, M., Kgoadi, R., Lund, M. N., Bell, K. J., Bugnet, L., Bowman, D. M., Johnston, C., García, R. A., Stello, D., Molnár, L., Plachy, E., Buzasi, D., and Aerts, C. Tess data for asteroseismology (t'da) stellar variability classification pipeline: Setup and application to the kepler q9 data. *The Astronomical Journal*, 162(5):209, October 2021a. ISSN 1538-3881. doi: 10.3847/1538-3881/ac166a. URL http://dx.doi.org/10.3847/1538-3881/ac166a.

Audenaert, J., Kuszlewicz, J. S., Handberg, R., Tkachenko, A., Armstrong, D. J., Hon, M., Kgoadi, R., Lund, M. N., Bell, K. J., Bugnet, L., et al. Tess data for asteroseismology (t'da) stellar variability classification pipeline: setup and application to the kepler q9 data. *The Astronomical Journal*, 162(5):209, 2021b.

Boone, K. Avocado: Photometric classification of astronomical transients with gaussian process augmentation. *The Astronomical Journal*, 158(6):257, dec 2019. doi: 10.3847/1538-3881/ab5182. URL https://doi.org/10.3847%2F1538-3881%2Fab5182.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations, 2020. URL https://arxiv.org/abs/2002.05709.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734. Association for Computational Linguistics, 2014. doi: 10.3115/v1/D14-1179.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

Gupta, R. and Muthukrishna, D. Transfer learning for transient classification: From simulations to real data and ztf to lsst, 2025. URL https://arxiv.org/abs/2502.18558.

Gupta, R., Muthukrishna, D., and Lochner, M. A classifier-based approach to multi-class anomaly detection for astronomical transients, 2024. URL https://arxiv.org/abs/2403.14742.

Huang, H., Muthukrishna, D., Nair, P., Zhang, Z., Fausnaugh, M., Majumder, T., Foley, R. J., and Ricker, G. R. Predicting the age of astronomical transients from real-time multivariate time series. *arXiv preprint arXiv:2311.17143*, 2023.

Ivezić, Ž., Kahn, S. M., Tyson, J. A., Abel, B., Acosta, E., Allsman, R., Alonso, D., AlSayyad, Y., Anderson, S. F., Andrew, J., and et al. LSST: From Science Drivers to Reference Design and Anticipated Data Products. *The Astrophysical Journal*, 873:111, March 2019. doi: 10.3847/1538-4357/ab042c.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2017.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Liu, F. T., Ting, K. M., and Zhou, Z.-H. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008.

McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

Muthukrishna, D., Narayan, G., Mandel, K. S., Biswas, R., and Hložek, R. RAPID: Early classification of explosive transients using deep learning. *Publications of the Astronomical Society of the Pacific*, 131(1005):118002, sep 2019. doi: 10.1088/1538-3873/ab1609. URL https://doi.org/10.1088%2F1538-3873%2Fab1609.

Muthukrishna, D., Mandel, K. S., Lochner, M., Webb, S., and Narayan, G. Real-time detection of anomalies in large-scale transient surveys. *Monthly Notices of the Royal Astronomical Society*, 517(1):393–419, November 2022. doi: 10.1093/mnras/stac2582.

Muthukrishna, D., Mandel, K. S., Lochner, M., Webb, S., and Narayan, G. Real-time detection of anomalies in large-scale transient surveys. *Monthly Notices of the Royal Astronomical Society*, 517(1):393–419, sep 2022. doi: 10.1093/mnras/stac2582. URL https://doi.org/10.1093%2Fmnras%2Fstac2582.

Narayan, G. and ELAsTiCC Team. The Extended LSST Astronomical Time-series Classification Challenge (ELAsTiCC). In *American Astronomical Society Meeting Abstracts*, volume 241 of *American Astronomical Society Meeting Abstracts*, pp. 117.01, January 2023.

Narayan, G., Zaidi, T., Soraisam, M. D., Wang, Z., Lochner, M., Matheson, T., Saha, A., Yang, S., Zhao, Z., Kececioglu, J., Scheidegger, C., Snodgrass, R. T., Axelrod, T., Jenness, T., Maier, R. S., Ridgway, S. T., Seaman, R. L., Evans, E. M., Singh, N., Taylor, C., Toeniskoetter, J., Welch, E., Zhu, S., and ANTARES Collaboration. Machine-learning-based Brokers for Real-time Classification of the LSST Alert Stream. *The Astrophysical Journals*, 236:9, May 2018. doi: 10.3847/1538-4365/aab781.

Parker, L., Lanusse, F., Golkar, S., Sarra, L., Cranmer, M., Bietti, A., Eickenberg, M., Krawezik, G., McCabe, M., Morel, R., Ohana, R., Pettee, M., Régaldo-Saint Blancard, B., Cho, K., and Ho, S. Astroclip: a cross-modal foundation model for galaxies. *Monthly Notices of the Royal Astronomical Society*, 531(4):4990–5011, June 2024. ISSN 1365-2966. doi: 10.1093/mnras/stae1450. URL http://dx.doi.org/10.1093/mnras/stae1450.

Perez-Carrasco, M., Cabrera-Vives, G., Hernandez-García, L., Förster, F., Sanchez-Saez, P., Arancibia, A. M. M., Arredondo, J., Astorga, N., Bauer, F. E., Bayo, A., Catelan, M., Dastidar, R., Estévez, P. A., Lira, P., and Pignata, G. Alert classification for the alerce broker system: The anomaly detector. *The Astronomical Journal*, 166(4):151, sep 2023. doi: 10.3847/1538-3881/ace0c1. URL https://dx.doi.org/10.3847/1538-3881/ace0c1.

PLAsTiCC Modelers. Libraries & Recommended Citations for using PLAsTiCC Models, March 2019. URL https://doi.org/10.5281/zenodo.2612896.

Qu, H. and Sako, M. Photo-zsnthesis: Converting type ia supernova lightcurves to redshift estimates via deep learning, 2023. URL https://arxiv.org/abs/2305.11869.

Rehemtulla, N., Miller, A. A., Laz, T. J. D., Coughlin, M. W., Fremling, C., Perley, D. A., Qin, Y.-J., Sollerman, J., Mahabal, A. A., Laher, R. R., Riddle, R., Rusholme, B., and Kulkarni, S. R. The zwicky transient facility bright transient survey. iii. btsbot: Automated identification and follow-up of bright transients with deep learning. *The Astrophysical Journal*, 972(1):7, aug 2024. doi: 10.3847/1538-4357/ad5666. URL https://dx.doi.org/10.3847/1538-4357/ad5666.

Villar, V. A., Cranmer, M., Berger, E., Contardo, G., Ho, S., Hosseinzadeh, G., and Lin, J. Y.-Y. A deep-learning approach for live anomaly detection of extragalactic transients. *The Astrophysical Journal Supplement Series*, 255 (2):24, 2021.

Zhang, G., Helfer, T., Gagliano, A. T., Mishra-Sharma, S., and Villar, V. A. Maven: A multimodal foundation model for supernova science, 2024. URL https://arxiv.org/abs/2408.16829.

A. Task Descriptions

In this section, we describe each downstream task in detail.

1. Training Data (Supernova data)
   (a) Classification (Sims): **Simulated time series** with corresponding class labels from ZTF (Muthukrishna et al., 2019; PLAsTiCC Modelers, 2019) and LSST (Narayan & ELAsTiCC Team, 2023). This task is only used for training our FMs.
2. Supernova Data Downstream Tasks (in domain)
   (a) Classification (Real Data): **Real time series from ZTF** data with corresponding class labels (Rehemtulla et al., 2024). This task is evaluated using the macro-averaged Area under the ROC Curve (AUROC).
   (b) Anomaly Detection: **Real time series from ZTF** with anomalous objects labeled (Rehemtulla et al., 2024). This task is also evaluated using the macro-averaged AUROC metric, treating anomaly detection (AD) as a binary classification task. Models are not provided any anomalous data during training to emulate real-world AD.
   (c) Redshift Estimation (Real Data): **Real time series from ZTF** with corresponding spectroscopic redshifts (Rehemtulla et al., 2024). This task is evaluated using the mean squared error (MSE).
   (d) Redshift Estimation (Sims): Simulated time series from ZTF (Muthukrishna et al., 2019; PLAsTiCC Modelers, 2019) and LSST (Narayan & ELAsTiCC Team, 2023) with corresponding spectroscopic redshifts. This task is evaluated similarly to the last one using the MSE. We note that this task does not share any data with the training task that also uses simulated data.
   (e) Zero-shot Redshift Estimation (Sims): Special testing scenario of the above task. The training data includes ZTF and evaluation is performed **zero-shot on LSST data**. This task is evaluated using the MSE.
3. Periodic Data Downstream Tasks (out of domain)
   (a) Periodic Classification (e.g. Audenaert et al., 2021a): **Real time series from Kepler** (Audenaert et al., 2021b) with corresponding class labels. This task is evaluated using the macro-averaged AUROC metric.
   (b) Luminosity Estimation: **Real time series from Kepler** (Audenaert et al., 2021b) with corresponding luminosity estimates. This task is evaluated using the MSE.

For most benchmarks on real data, we propose two scenarios. The `Limited` scenario limits the amount of data available for fine-tuning to simulate the development process for new telescopes where labeled data will not be readily available. In this scenario, the rest of the data is provided without labels. We choose to limit this scenario to 100 labeled objects for each task which is the amount of data that can be labeled in a few months on new telescopes. Due to the high variability in selecting such a small number of samples, we evaluate over 5 randomly selected `Limited` data samples. Incorporating unlabeled data into the model fine-tuning process is beyond the scope of this work, but we hope that future researchers explore ways to accomplish this. The `Full` mode includes all data. The zero-shot redshift and Kepler benchmarks do not include the `Limited` testing scenario, the latter is an out-of-domain benchmark that is not a direct target for our FMs. Appendix A contains more information about each task.

A.1. Simulated Tasks

Table 3 shows the number of objects per supernova class in our datasets. The supernova classes are described further in previous work (Gupta et al., 2024; Muthukrishna et al., 2022). The classification tasks are used for pretraining and redshift task is used for downstream zero-shot evaluation. Because classification is used for pretraining, we ensure that there is no overlap between the data for both tasks. The LSST zero-shot redshift estimation task restricts evaluation to light curves that are in the ZTF redshift range. We do this because, as seen in Figure 2, LSST will observe a far greater range of redshifts. The goal of this work is to leverage existing data for future models, and the current prior only exists for data from ZTF. Thus, we find it reasonable to perform zero-shot evaluation solely on LSST. Evaluating in this manner also seems to limit the applicability of these zero-shot models, however astronomers may know the rough redshift range of an object and thus can choose to use these models when they see fit.

A.2. ZTF Real Data

Table 4 shows the number of light curves from each class in our dataset of real ZTF light curves. We define anomalies to be transients from any of the following classes: TDE, Ca-rich, ILRT, LBV, LRN, SLSN-I, SLSN-II, SN Ia-91T, SN Ia-91bg, SN Ibn, SN Ic-BL, SN Icn, and TDE. These objects are specifically chosen because of their low observation rates and limited human understanding.

Figure 3 shows the redshift distribution in our dataset of real ZTF objects. This redshift range is similar to that of the simulated ZTF data as seen in 2 [top]. Unlike with the simulated redshift estimation task, we do not use any real light curves for pretraining and reserve them solely for

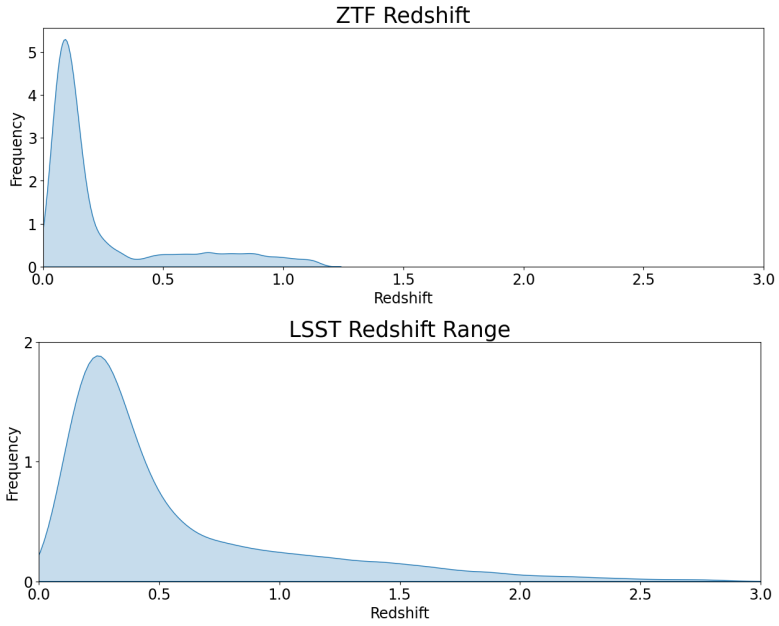

Figure 2. The range of redshifts for objects in our datasets. After fine-tuning a domain-agnostic FM to estimate redshifts for ZTF objects, we evaluate its zero-shot LSST performance only using transients in the ZTF range.

Task	SNIa	SNIa-91bg	SNIax	SNIb/c	SNII	TDE	SLSN-I	AGN	Total
ZTF Classification	9436	10663	10681	6769	31193	9260	10451	8627	87080
LSST Classification	8427	6079	8298	7664	9465	9686	6947	7822	64388
ZTF Redshifting	967	1140	1124	702	3213	932	1053	869	10000
LSST Redshifting	1236	1238	1226	1286	1248	1279	1253	1234	9990
Zero-shot LSST	265	273	535	258	333	215	446	271	2596

Table 3. The amount of simulated labeled data pairs available for the different tasks on simulated data. There is no overlap between the data in the classification tasks and the redshifting tasks. The zero-shot scenario for LSST only includes transients from LSST which are in the redshift range of ZTF.

Task	SNIa	SNIb/c	SNII	Anomaly	Total
Classification	771 (107)	2828 (350)	148 (12)	0	3747
Redshift Estimation	771 (107)	2828 (350)	148 (12)	0	3747
Anomaly Detection	771 (107)	2828 (350)	148 (12)	0 (38)	3785

Table 4. The amount of real labeled data pairs available for the different LSST and ZTF tasks used in this work. The `Limited` versions of these tasks use a random sample of `128` objects. The number in parenthesis represents the amount of data provided in the evaluation set.

evaluation. Thus, all tasks on ZTF real data share a roughly identical pool of observed objects. Further information about the real ZTF data used in this work can be found in (Rehemtulla et al., 2024).

A.3. Zero-Shot LSST Redshifting

The motivation behind zero-shot prediction is to directly reuse our understanding of ZTF to train models for LSST. Thus, when we evaluate zero-shot FM performance, we restrict the evaluation to LSST light-curves in the same redshift range as ZTF. LSST will observe a much broader redshift range than ZTF (as seen in Figure 2) and its imperative to maximize discovery from all new transients. However, it is unreasonable to expect that zero-shot models will be able to perform well on data that is outside the range of prior telescopes; to build more general models, fine-tuning a pretrained FM or using transfer learning (Gupta & Muthukrishna, 2025) is a better option (as done in Table 1).

A.4. Kepler Tasks

The Kepler telescope is an older telescope which has been observing periodic astronomical events for a long time. Kepler observes objects in different passbands and has a regular cadence for observations. Each object Kepler observes has 1024 roughly evenly spaced brightness measurements at roughly a 30 minute cadence. ZTF and LSST takes measurements at a significantly slower cadence and has much fewer observations per object. Thus, to repurpose our FMs for Kepler, we bin Kepler objects by taking the mean of every five consecutive observations, resulting in 205 measurements per object. This is more reasonable because it more closely matches the size of the objects in the supernova training sample.

Our dataset of Kepler objects contains 8 different classes of objects, and their frequencies are shown in 5. Human-annotated luminosity estimates only exist for a subset of observed objects from certain classes. The distribution of luminosities is shown in Figure 3. When fine-tuning, we scale the luminosity by dividing by the mean value in the dataset.

B. Model and Training Details

B.1. Classifier Training

Our neural network classifier is trained to take a vector of telescope observations as input, with each row being a distinct observation in the form $[\lambda_p, t_i, f_i]$, where λ_p represents the median passband wavelength of the observation, t_i represents the time since the first observation in days, and f_i represents the flux (measured brightness) of the event. We scale the flux by dividing the measured fluxes by 500, a value chosen close to the mean of our various datasets. This limited preprocessing allows for the usage of our models in real time, however real-time evaluation is beyond the scope of this work. This input method (Huang et al., 2023; Gupta et al., 2024; Gupta & Muthukrishna, 2025) and specifically allows for the usage of the same model across surveys, something not facilitated by many previous input methods.

B.2. Adversarial Training Algorithm

Our adversarial pretraining is summarized in Algorithm 1. Here, X_i denotes the input light curve, c_i is its class label, and $O_i \in \{\text{ZTF}, \text{LSST}\}$ indicates the observatory. The latent representation is extracted as $L_i = C_L(X_i)$, where C_L is the penultimate layer of the classifier. The categorical cross-entropy loss is denoted by $H(p, q)$, where p is a predicted distribution and q is a target one-hot vector.

Algorithm 1 Adversarial Training

Require: Dataset $\{(X_i, c_i, O_i)\}_{i=1}^{N}$: light curves, class labels, and observatory labels
Require: Classifier C, Discriminator D
Initialize C and D with random weights
repeat
  // Step 1: Train the discriminator
  Freeze the classifier C
  For each sample, compute latent representation $L_i = C_L(X_i)$
  Compute discriminator loss:

$$\mathcal{L}_D = H(D(L_i), O_i)$$

  Update D to minimize $\mathcal{L}_D$
  // Step 2: Train the classifier
  Freeze the discriminator D
  Compute classifier loss with adversarial objective:

$$\mathcal{L}_C = H(C(X_i), c_i) - H(D(C_L(X_i)), O_i)$$

  Update C to minimize $\mathcal{L}_C$
until convergence

B.3. Contrastive Training Algorithm

For our supervised contrastive loss, we use the contrastive objective proposed in Chen et al. (2020) for model pretraining. It is formally defined as follows:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\mathbf{L}_i, \mathbf{L}_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\text{sim}\left(\mathbf{L}_i, \mathbf{L}_k\right)/\tau\right)} \quad (1)$$

where: $\mathbf{L}_i, \mathbf{L}_j$ are the latent representations from the classifier, $\text{sim}(\mathbf{L}_i, \mathbf{L}_j)$ denotes cosine similarity: $\text{sim}(\mathbf{a}, \mathbf{b}) =$

Task	Aperiodic	Constant	Contact	DSCT_BCEP	Eclipse	GDOR_SPB	Instr	RR_CEP	Solar
Kepler Classification	831	1000	2260	772	974	630	1171	63	1800
Kepler Luminosity	–	–	–	38	–	788	–	–	–

Table 5. The number of labeled data pairs available for each Kepler task used in this work. Luminosity labels are available for a subset of classes.

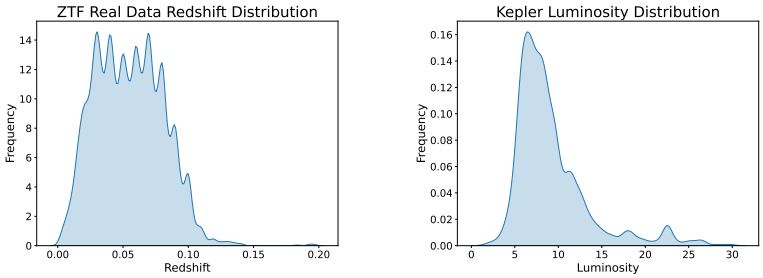

Figure 3. Redshift distribution [left] for our dataset of real ZTF objects and luminosity distribution [right] for our dataset of real Kepler objects.

$\frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|}$, $\tau > 0$ is a temperature parameter that scales the similarity scores, and the loss is computed for all pairs (i, j) where X_i and X_j share the same class label.

The total supervised contrastive loss is computed by summing $\ell_{i,j}$ over all valid positive pairs in a batch. This encourages latent vectors from the same class to be close together, while implicitly pushing apart representations from other classes.

Our contrastive pretraining is summarized in Algorithm 2. We set $\tau = 0.5$ similar to the default set in previous work (Chen et al., 2020).

Algorithm 2 Supervised Contrastive Training (with Mean Contrastive Loss)

Require: Training set $\{(X_i, c_i)\}_{i=1}^{N}$; light curves, class labels
Require: Classifier C, temperature parameter τ
Initialize C with random weights
repeat
  Compute classification loss

$$\mathcal{L}_C = H(C(X_i), c_i)$$

  Compute latent representations $\mathbf{L}_i = C_L(X_i)$
  Initialize total contrastive loss $\mathcal{L}_{\text{SCL}} \leftarrow 0$, counter $M \leftarrow 0$
  for each anchor sample $i \in \{1, \ldots, N\}$ **do**
    Let $P(i) = \{j \neq i : c_j = c_i\}$
    for each $j \in P(i)$ **do**
      Compute pairwise contrastive loss:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\mathbf{L}_i, \mathbf{L}_j\right)/\tau\right)}{\sum_{k \neq i} \exp\left(\text{sim}\left(\mathbf{L}_i, \mathbf{L}_k\right)/\tau\right)}$$

      Accumulate loss: $\mathcal{L}_{\text{SCL}} \leftarrow \mathcal{L}_{\text{SCL}} + \ell_{i,j}$
      Increment counter: $M \leftarrow M + 1$
    end for
  end for
  Compute mean contrastive loss: $\mathcal{L}_{\text{SCL}} \leftarrow \mathcal{L}_{\text{SCL}}/M$
  Compute total loss: $\mathcal{L} \leftarrow \mathcal{L}_{\text{SCL}} + \mathcal{L}_C$
  Update C to minimize $\mathcal{L}$
until convergence

B.4. Fine-Tuning

B.4.1. General Tasks

We usually freeze the entire model when fine-tuning for downstream tasks. However, freezing the initial model does not work effectively for novel domains (most notably Kepler). Thus, when applying our foundation model to novel data, we unfreeze the initial neural network layers to assist in domain generalization. These techniques are motivated by past research done in transfer learning and they have been shown to improve performance (Gupta & Muthukrishna, 2025).

B.4.2. Anomaly Detection

For anomaly detection, which is neither a regression or classification task, we use a classifier-based approach (Gupta et al., 2024), where a classifier is trained on a set of *normal* data. The penultimate layer of this classifier is then used as a latent space for anomaly detection and an isolation forest (Liu et al., 2008) is trained using this latent space to detect anomalies. This method has state-of-the-art (SoTA) performance for anomaly detection on real data[4] (Perez-Carrasco et al., 2023; Gupta et al., 2024).

B.5. Architecture Details

Our FMs are built with a recurrent neural network architecture containing Gated Recurrent Units (GRU; Cho et al., 2014). We chose to use GRUs because they are shown to be more effective than RNNs and have quicker training times than LSTMs (Chung et al., 2014). Further, neural network and GRU-based models have worked effectively in training past models for time-domain astronomy (e.g. Boone, 2019; Gupta et al., 2024; Muthukrishna et al., 2019). The provided code has more details describing the exact architecture.

When fine-tuning our FMs on downstream tasks for real data, we freeze the foundation model and leave the MLP unfrozen. When training on tasks from new observatories, we further unfreeze the initial layers. These decisions were based on rough hyperparameter searches. More precise tuning is beyond the scope of this work and the main contributions stand as long we remain consistent across different model types. The provided code also has more details on which layers are frozen.

We train and fine-tune our models using the `Adam` optimizer (Kingma & Ba, 2017) and stop training when the validation loss has not decreased for 5 epochs. Our classifier-based, adversarial, and contrastive models take roughly 10, 20, and 45 minutes to converge on a standard V100 GPU respectively. The final experiments required an estimated 15-25 GPU hours, however ideation and experimentation required considerably more.

Figure 4 shows the training loss as a function of epoch for our models. Contrastive and adversarial models end at a worse cross-entropy loss but are able to simultaneously optimize for more complex loss functions. All models perform well on downstream tasks, as seen in Table 1, which shows that there is no clear metric to predict how well an FM will perform on downstream tasks. This opens up future research to use different techniques to achieve better performance.

[4]This dataset uses human-defined features extracted from light curves. However, they are not standardized across telescopes and thus we opt to use raw time series for our models.

C. Qualitative Latent Space Analysis

Figures 5 and 6 show a UMAP (McInnes et al., 2020) visualization of the penultimate layer of our neural network classifiers. As seen, contrastive and adversarial models help unify the distributions of ZTF and LSST in the latent space down to the class level.

D. Further Analysis

D.1. Adversarial vs. Contrastive Loss

Incorporating the specialized training techniques proposed in this work does not improve model performance on downstream tasks (Table 1), which makes sense because the specialized loss functions are not designed for direct downstream tasks. In the case of the adversarial loss, however, we see a significant decrease in performance. We believe this is because the adversarial model lacks direct supervision and is forced to learn an implicit relationship between the two telescopes while training, whereas the contrastive model is given an explicit relationship through the supervised contrastive loss.

On zero-shot tasks, the adversarial and contrastive models outperform a classifier, showing that the unification in the latent space is indeed meaningful. The contrastive model slightly outperforms the adversarial model, similar to Table 1, and we think this is again due to the increased supervision provided during model training. Training with both losses simultaneously also does not improve performance in comparison to a purely contrastive model.

D.2. Performance on Simulations

We observe that fine-tuned performance on real-data lags the performance on the same tasks for simulations. In the redshift task for real data, our best model achieves an R^2 score of 0.431 ± 0.053, while the same metric for simulations is 0.580 ± 0.011. Models can leverage these physics-based simulations as effective starting points but still require labeled real data to perform well. This gap between real data and simulations is why there is a significant performance

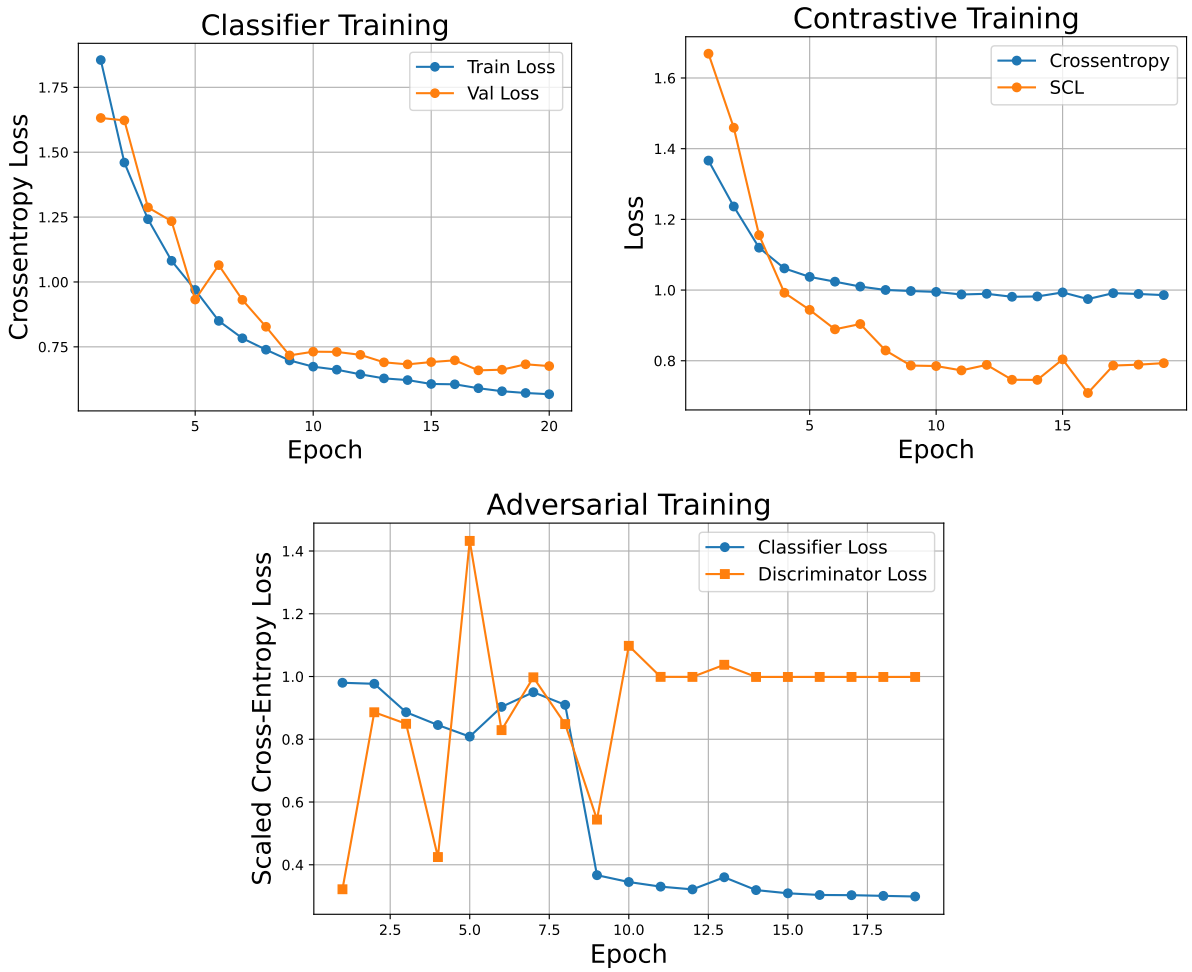

Figure 4. Loss as a function of training epoch for each FM in this work.

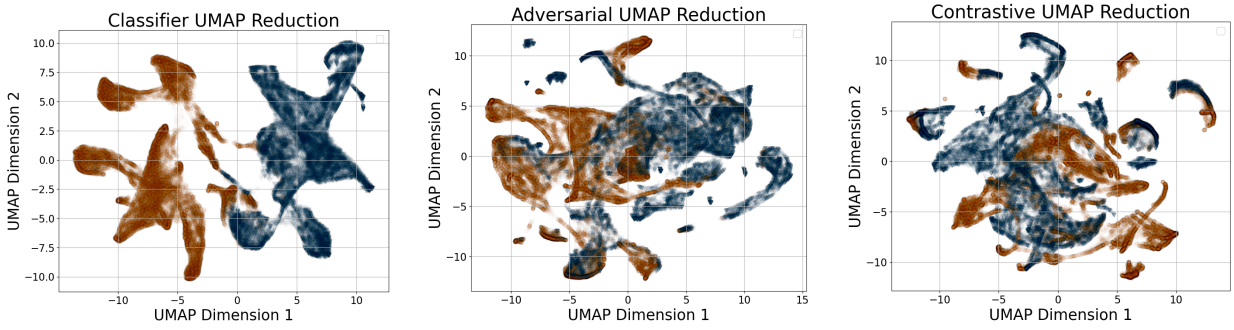

Figure 5. UMAP representations of the final layer of each foundation model. Brown circles represent ZTF while blue triangles represent LSST. As seen, both the contrastive and adversarially trained models are able to unify LSST and ZTF transients into the same latent space, unlike a classifier which has distinct clusters.

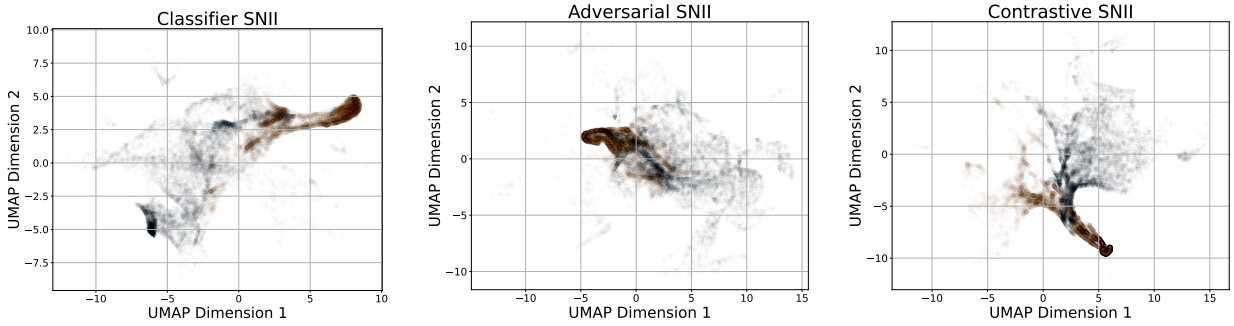

Figure 6. UMAP representations of the final layer of each foundation model only for a single class of supernova (Type II). Brown circles represent ZTF while blue triangles represent LSST. As seen, both the contrastive and adversarially trained models are able to unify LSST and ZTF transients even at a class level.

gap between the `Limited` and `Full` evaluation scenarios. In other words, models trained on simulations need to be fine-tuned on real data to work well.

D.3. Anomaly Detection

Anomaly detection is the only task in which we do not see a performance improvement from the `Limited` to `Full` settings (Table 1). By definition, anomalies are objects that astronomers find interesting. By using human-defined simulations to pretrain FMs, they are naturally equipped to detect specifically what humans find interesting. The gap between simulations and real data is what anomaly detection pipelines are trying to fill. Further analysis of anomaly detection specifically is out of the scope of this work, however we hope that future researchers analyze the nature of this task and how domain expertise can be incorporated into it.

D.4. Are Classifiers the Best Foundation Models?

For our `Classifier` foundation models, we find that classification performance (as AUROC or crossentropy) is an effective metric for model selection, i.e. better performing classifiers perform better on downstream tasks. However, this method of model selection does not generalize to our domain-agnostic FMs (most notably the contrastive model). These models perform worse on the classification objective because they optimize for a more complex loss function (as seen in Figure 4). However, they still perform well on downstream tasks, as noted in Table 1. This result shows that there is significant room for growth and that novel methods could incorporate information not captured by the models described in this work. Ultimately, this line of research does not end with classifiers.

For example, while we believe that supervised training is a promising direction for foundation models, at some point the complexity of physics-informed simulations may render it difficult to directly incorporate this information into deep learning models. Thus, using unsupervised methods alongside class-based supervision in model pretraining is also an important research direction. This is one of the many ways we see scientific foundation models expanding beyond classifiers with more complex pretraining methods.

E. kNN Zero-Shot Estimation

Aside from fine-tuning an MLP, we also evaluate using a k Nearest Neighbors approach for zero-shot estimation (Zhang et al., 2024; Parker et al., 2024). To perform zero-shot redshift estimation for an LSST object, we first find the $k = 100$ closest ZTF embeddings to the LSST light curve embedding in the latent space. Then, we use the distance-weighted average of the corresponding redshifts to estimate the final redshift of the LSST object. As seen in Table 2, this zero-shot estimation method performs worse than using a directly trained MLP.