JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Convolutional Supernovae Networks

A Deep Learning Approach to Supernova Classification

Dr.Pravin Shinde, Mr.Aryan Anchan, Mr.Shaun D'souza, Mr.Krish Sanghvi, Mr.Aryan Bhanushali
Guide and Head of Department, Student, Student, Student, Student
Department of Artificial Intelligence and Data Science,
University of Mumbai
Mumbai, India

Abstract : This research aims to create a deep learning classification model for supernova classification using light curve data. The model is trained to accurately distinguish different types of supernovae by leveraging observational data from known supernovae of diverse classes. The challenge lies in identifying unique characteristics and patterns in the light curve data that correspond to specific types of supernovae. The initial objective is to construct a robust model that can generalize from the training data and effectively categorize supernovae into their respective classes. The successful outcome of this project will enhance our understanding of supernovae and provide valuable insights to astronomers for their research and analysis of these celestial events. Supernovae come in various types, and our goal is to classify and identify the pertinent ones. This problem is well-suited for supervised machine learning. We began by extracting insightful features from the data, guided by scientific principles. Subsequently, we selected different machine learning algorithms for supernova classification and assessed the outcomes. We have chosen the PLASTiCC dataset. This project involves a multi-step approach to advance our knowledge of supernovae. The process begins with the extraction of scientifically relevant features from raw light curve data and augmenting them. These features are carefully decided to capture the intrinsic properties of supernovae, allowing the model to make informed classifications.

I. INTRODUCTION

The study of supernovae, which are massive star explosions, has always fascinated astronomers because they hold significant implications for our understanding of the universe. These cosmic explosions, though diverse, can be categorized into different types based on their observed traits, like their spectra (the different colors of light they give off) and how their brightness changes over time. Accurate classification of supernovae is crucial as it deepens our knowledge of these incredible astronomical events and contributes to fields like cosmology (the study of the universe's structure and origins), the life cycles of stars, and the dynamics of galaxies (how galaxies move and evolve). Traditionally, astronomers have classified supernovae by carefully examining them visually and doing it manually. But in recent times, machine learning algorithms have become extremely effective tools for classifying astronomical phenomena, and they could revolutionize the way that supernovae are categorized. Convolutional neural networks (CNNs), Random Forest, Gradient Boosting Decision Trees, K-Nearest Neighbor, and Multi-Layer Perceptrons are some of the techniques that have become well-known for their capacity to extract features—that is, significant details—and identify patterns. In simpler terms, they are like super-smart computer programs that can learn to tell different types of supernovae apart by looking at the data we collect from them. The forms of variable star light curves provide important insights into the physical mechanisms causing the variations in brightness. Supernova light curves can provide information about the type of supernova. While spectra are used to determine supernova kinds, each has a characteristic form for its light curve. While Type II supernovae have less sharp maxima, Type I supernovae have light curves that sharply peak and then gradually decrease. Light curves are useful in the sub-type classification and categorization of weak supernovae.

This research aims to investigate how Convolutional Neural Networks (CNNs), Random Forest, Gradient Boosting Decision Trees, K-Nearest Neighbour and Multi-Layer Perceptron can be applied to classify different types of supernovae using various data dimensions, including spectral details (the different colors of light they emit), the way their brightness changes over time, and photometric measurements (precise measurements of their light). This means we're trying to teach these computer programs to look at all the information we have about supernovae and figure out what type they are, which will help us learn more about the universe and its amazing phenomena.

Photometric redshift, a fundamental concept in the field of astronomy and astrophysics, plays a pivotal role in our supernova classification project. In essence, photometric redshift is a technique employed to estimate the distance or redshift of astronomical objects, such galaxies and supernovae, based on the observed light they emit across multiple photometric filters. This shift is manifested as a change in the color of the light, where objects moving away from us exhibit a "redshift," indicating an increase in their observed wavelength. The magnitude of this redshift is directly related to the velocity and distance of the object, making it a valuable tool for astronomers to determine the motion and position of celestial bodies in the vast expanse of the cosmos. In our project, redshift serves as a critical parameter, enabling us to estimate the cosmic distances of supernovae and classify them based on their spectral characteristics.

In the vast expanse of astrophysical research, the quest to understand supernovae—celestial explosions that illuminate the cosmos—continues to captivate scientists worldwide. Amidst this pursuit, the forthcoming research paper titled "Convolutional Supernovae Networks" represents a significant leap forward in the realm of computational astrophysics. Authored by Aryan Anchan, Shaun D'souza, Aryan Bhanushali and Krish Sanghvi , this paper introduces an innovative methodology that harnesses the power of convolutional neural networks (CNNs) to revolutionize the classification and analysis of supernovae events. By publishing this paper, we aim to contribute to the collective endeavor of unraveling the mysteries of the cosmos, one supernova at a time

II. Problem Statement

The main aim here is to teach the computer model by using information about different types of known supernovae (like Ia, Ib, II, etc.). We want to see how good it is at figuring out the type of new supernovae based on their light patterns. The tricky part is finding the unique things in the light patterns that tell us what type of supernova it is. Our goal is to create a model that learns from this information and can accurately tell us the type of a new supernova by looking at its light pattern. We want it to work well even if it hasn't seen that kind of supernova before.

We expect to have a strong computer model, thanks to "snmachine," which is a helpful tool for astronomy work in Python. This model will be good at telling us what kind of supernova we're looking at based on its light pattern. This helps us understand these cosmic events better and supports astronomers in their research and studies. It's like having a handy tool to speed up our progress toward our goal.

III. Research Methodology

Our study introduces a systematic approach to analyzing supernovae and their light curves using the Python program SN machine. Initially, we import data from the extensive PlasticcData dataset, a repository containing diverse supernova candidates. Subsequently, we select a subset of this dataset for further analysis, preserving it as a persistent instance (.pkl) to ensure accessibility. Preprocessing the light curves is pivotal, involving setting a maximum permissible gap duration to enhance data quality and continuity. This step aids in accentuating prominent features and improving analysis accuracy by ensuring smooth temporal transitions in the observations. Gaussian processes constitute a fundamental aspect of our methodology for modeling light curves. Leveraging Gaussian processes enables various applications, including feature extraction, anomaly detection, and supernova behavior modeling, facilitating more effective identification and characterization of different supernova types. Augmenting the dataset involves generating photometric redshifts for synthetic events, simulating real-world scenarios while considering uncertainties in photometric redshift measurements. This enriched dataset provides a robust foundation for categorization, incorporating both photometric and spectroscopic redshift data. Our analysis identifies three primary supernova categories, each exhibiting distinct characteristics and behavioral patterns. Utilizing the SN machine package streamlines the categorization process, allowing for rapid and automated classification of supernovae, exploration of their features, and extraction of valuable insights from astronomical data. These efforts collectively contribute to advancing our understanding of celestial phenomena.

3.1Population and Sample

As part of the LSST (Large Synoptic Survey Telescope) project, simulated astronomical time-series data were collected to construct the PLAsTiCC "(Photometric LSST Astronomical Time-series Classification Challenge)" dataset. It contains a variety of celestial objects, such as supernovae. The PLAsTiCC dataset often contains measurements of the object's brightness (flux) over time in various filters (bands), including u, g, r, i, and z. Every data point has a time stamp and a matching flux measurement inside a certain band. Light curves, or sequences of observations of the same item across time, are how the data is arranged. The brightness progression of the supernova is captured by these light curves, which is essential for differentiating and comprehending the characteristics of many supernovae.

3.2 Data and Sources of Data

Supernova datasets are publicly available on several governments backed space websites. These sites are: Open Supernova Catalog (OSC), Plasticc Photometric Dataset, "NASA/IPAC Extragalactic Database (NED), SIMBAD Astronomical Database" etc. For our project, we decided to use the plasticc database as it contains light curves of more than 100,000 supernovae, on the other hand the SPP dataset contains only 1000. As we are performing a comparative study of 4-5 different algorithms, we wanted to use a larger dataset. These datasets contain a variety of data on the supernovae such as their light curves, spectra, and positions. By using the different variables, we can create different predictive models for the same. Furthermore, we are using the snmachine library which is a Python package designed for astronomical applications, particularly in the field of supernova classification and light curve analysis. It offers methods and instruments for categorizing supernovae according to their light curves. This package is often used by astronomers and researchers to automatically categorize supernova candidates into various types, such as Type Ia, Type Ib, Type II, etc. The package offers a range of features for performing tasks related to supernova classification and analysis, including data loading, feature extraction, machine learning, and Gaussian process regression. Our project uses a subset of the large supernovae data to work upon and correctly classifies based on them.

3.3 Theoretical framework

Light curves, which represent the brightness of a supernova over time, contain valuable information about the underlying physical processes and characteristics of these cosmic phenomena. In this study, we leverage machine learning algorithms implemented in the snmachine library to classify supernovae based on their light curve features. The dataset utilized in this research has been preprocessed, featuring extracted features such as wavelet transforms to capture temporal patterns and augmentations to enhance the diversity and robustness of the training data. Additionally, Gaussian process modeling has been employed to refine the understanding of the data distribution and improve classification accuracy. We aim to use the set of machine learning algorithms and denoted in the snclassifier to classify the types of supernovae.

We investigate how measurements behave when it comes to classification with specific strengths and limitations. For every type of supernova, light curve simulations were performed. The program learns to associate the corresponding supernova kinds with the characteristics of the light curves during training. By contrasting their features with those of the training data, newly discovered light curves can be classified by the algorithm once it has been trained.

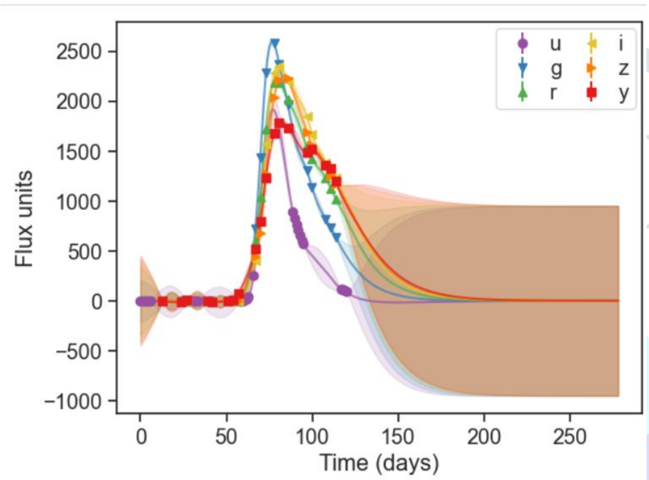

Figure 1. Depicts the light curves that starts as the death of the star (dim) and gets bright.

IV. Models

A performance metric, or a single scalar number that measures a classifier's suitability for a given job, is necessary to discern various classification techniques as best as possible. Thus, it makes sense that selecting a metric for PLASTICC would be linked to the objectives of the challenge. We take multiple classes of supernovae, with X variable being features and Y variable being Datalabels.

Convolutional Neural Networks:
We have developed a traditional convolutional neural network, consisting of a sequence of convolutional, pooling, flattening, dropout and fully-connected layers with a sigmoid activation function, which in our case is the ReLU activation function. The 'classification', function takes a combination of hyperparameters, training and testing data as input. We have used the "Plasticc_dataset", which is spilt into 3 parts; training, validation, and testing. It uses a spilt ratio of (70% train, 30% validation) if the 'og' flag is True (training on data generated by Normalizing Flows). It later on converts the data into torch sensors, the model uses a loss function which is binary cross entropy along with an adam optimizer.

Random Forest Classifier:
Using the RFClassifier class from the snclassifier module, we initialize a random forest classifier instance called classifier_instance. This classifier, which is based on the random forest algorithm, is appropriate for classification problems. Since the random number generator is started with a predefined seed, repeatability of results is ensured by setting the random_seed option to 42. The subsequent phase establishes an optimization parameter grid, indicating distinct values for two hyperparameters of the random forest classifier: the maximum depth of the trees (max_depth) and the number of estimators (n_estimators). For every hyperparameter, there is a list of potential values in the param_grid dictionary. During the optimization process, this grid will be utilized to determine the ideal set of hyperparameters.

```
array([[0.28231293, 0.13265306, 0.58503401],
       [0.11180124, 0.20496894, 0.68322981],
       [0.06623198, 0.07343857, 0.86032944]])
```

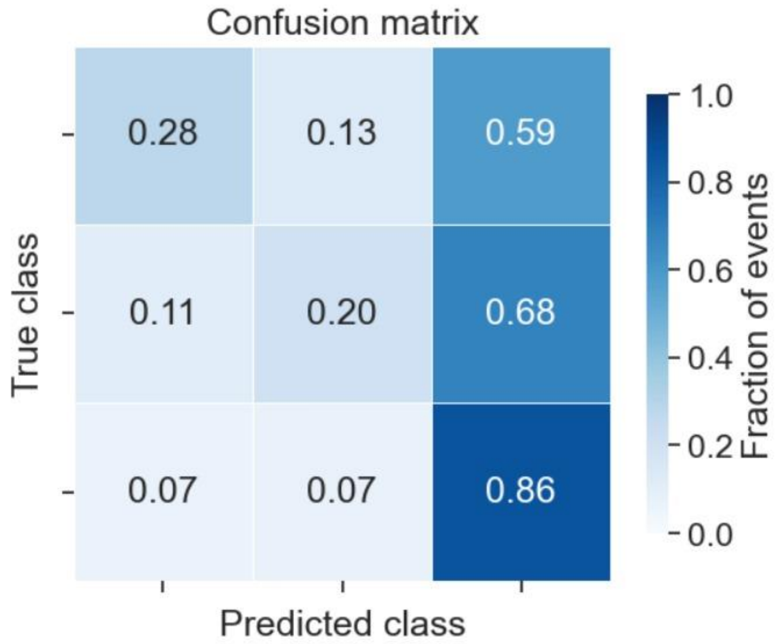

Figure 2. Confusion Matrix of Random Forest Classifier

K-Nearest Neighbours:

We set up a parameter grid for optimizing a K-Nearest Neighbors classifier. The param_grid dictionary contains hyperparameters to be tuned, such as the number of neighbors which is the (n_neighbors) and the weight function (weights). It specifies different values to try for each hyperparameter.

The classifier_instance.optimise() function optimizes and trains the following KNN classifier by using the provided dataset given (X and y). It uses cross-validation with 5 folds (number_cv_folds=5) to evaluate different parameter combinations. The optimization is based on minimizing the log loss (scoring='logloss'). Additionally, it seems like metadata might be some additional information or settings passed to the optimization process.

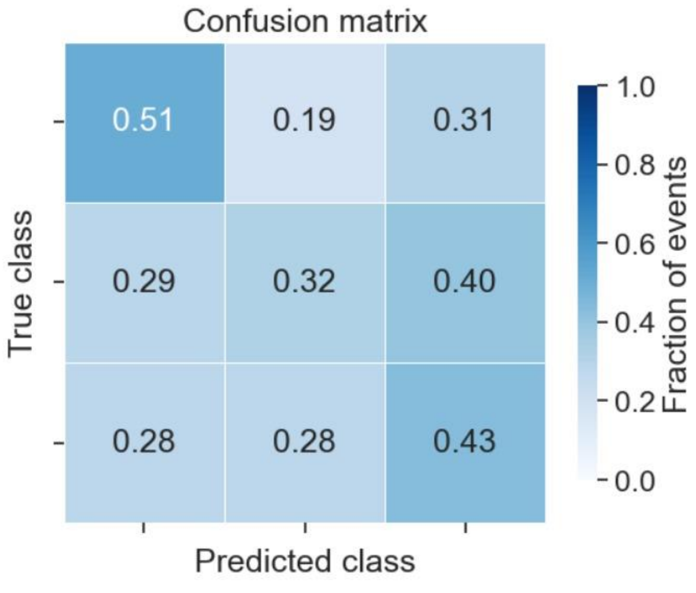

Figure 3. Confusion Matrix of K-Nearest Neighbor

Multi-layer Perceptron:

We initialize a neural network classifier, set up a parameter grid for hyperparameter optimization, convert column names to strings, and then optimizes the MLP classifier by applying the provided feature matrix X, labels y, and hyperparameter grid. Then we perform hyperparameter optimization using grid search cross-validation. We search through the specified hyperparameter grid (param_grid) to find the combination that minimizes the log loss, using 5-fold cross-validation. This evaluates the performance of the following MLP trained classifier on the test data. Finally we retrieve the best classifier and its best parameters found during the optimization process, set a specific attribute (which_column) of the classifier, calculate the Area Under the Curve (AUC) and log loss scores using the trained classifier and the test data, and finally print out the evaluation metrics in a formatted manner.

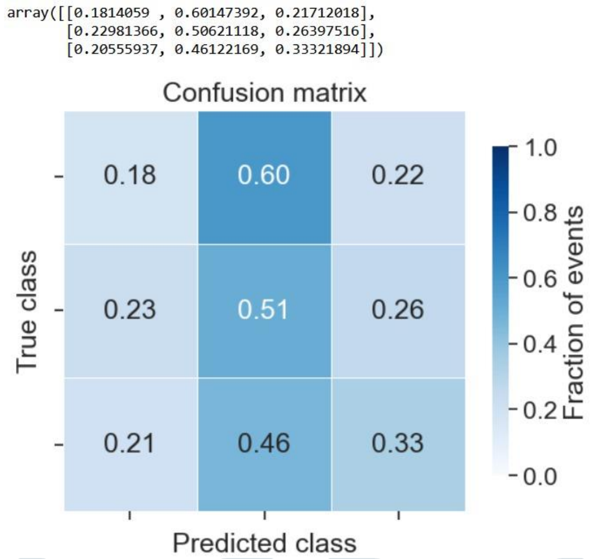

Figure 4. Confusion Matrix of Multi-Layer Perceptron

LightGBM:
The optimization process involves grid search over a parameter grid, specifically focusing on the learning rate hyperparameter. The grid contains predefined learning rate values of 0.1, 0.25, and 0.5. The classifier_instance's "optimise" method is called with the dataset features (X), labels (y), and additional parameters, including the parameter grid, scoring metric ('logloss'), the number of cross-validation folds (5), and the dataset's metadata.

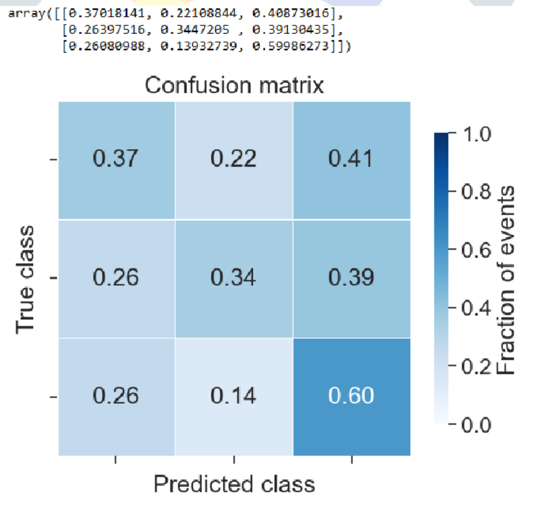

Figure 5. Confusion Matrix of LightGBM

V. RESULTS AND DISCUSSION

5.1 Metrics

We examine how the AUC score and logloss metrics react to the classifiers covered in Section 4 in the following sections, taking into account different weights on the impacted classes.

5.1.1 Log-loss
The information-theoretic log-loss is connected to entropy, represented as

$$Hn = -\sum_{m}^{M} p(m \mid d_n)\, ln[p(m \mid d)]$$

which quantifies the range of possible states that a system, in this case the light curve class, can occupy. When a classification posterior reduces classification to a deterministic result by giving one class a probability of 1 and all other classes a probability of 0, it achieves minimal entropy. This entropy metric, however, does not intrinsically correspond to the true class of the light curve; rather, it is based only on the classification probabilities.

We introduce the cross-entropy, which is represented as follows, to align the classification posterior with the true class determined by the challenge overseers:

$$L_n = Q_n^L = -\sum_{m}^{M} T_{n,m} \, ln[p(m \mid d)]$$

This can be understood as the artificially enlarged range of potential states (a rise in disorder) resulting from substituting the classification posterior for the indicator variable. While Hn reaches a minimum value of 0 with any deterministic classification, Ln only minimizes to 0 when τn and p(m | dn) are identical.

5.1.2 ROC_AUC

The "Receiver Operating Characteristic Area Under the Curve," or "ROC AUC," is a commonly used metric to assess how well binary classification algorithms perform. It offers insightful information about how well the classifier performs over a range of threshold values in differentiating between positive and negative classes. It shows the relationship between the true positive rate (sensitivity) and the false positive rate (1-specificity) across different thresholds, as shown visually by the ROC curve. Improved discriminating between the two classes is shown by a higher ROC AUC score, which is a number between 0 and 1. To be more precise, a score of 1 denotes perfect classification performance, meaning the model correctly detects all positive cases while completely avoiding false positives.

$$ROC\ AUC = \int_0^1 TPR(fpr)d(FPR)$$

Where:

- The TPR/Recall, which stands for True Positive Rate, is also referred to as Sensitivity or Recall. It's calculated as TP/(TP+FN).
- Similarly, the FPR/Precision, representing False Positive Rate, is calculated as FP/(FP+TN).
- Here, TP stands for True Positives, FP for False Positives, TN for True Negatives, and FN for False Negatives.

The region beneath the ROC curve, which ranges from 0 to 1, is calculated using this integral.

5.2 Results of Descriptive Statics of Study Variables

Table 5.2.1: Descriptive Statics

Models	AUC	Log Loss	Recall	Precision
Convolutional Neural Networks	0.8425	0.3489	0.9896	0.8617
K-Nearest Neighbour	0.623	-1.891	0.4182	0.4287
Random Forest	0.696	-1.063	0.522	0.532
LightGBM	0.634	-1.290	0.4586	0.4774
Multi-layer Perceptron	0.523	-1.102	0.3408	0.3721

Table 5.2.1 Higher ROC AUC values for a model (closer to 1) indicate better discrimination between classes, implying superior predictive performance. Lower log loss values for a model indicate better-calibrated probabilities and more accurate predictions.
Based on the ROC AUC and log loss values in the table, the Convolutional Neural Networks (CNN) model appears to have the highest performance, with the highest ROC AUC value and the lowest log loss value among the listed models. On the other hand, the Interest rate model seems to perform poorly based on these metrics, with a negative ROC AUC and comparatively high log loss.

VI. CONCLUSIONS

According to the table's data, CNN did better at categorizing supernovae than other machine learning methods. CNN outperformed other models, including K-Nearest Neighbor, Random Forest, LightGBM, Multi-layer Perceptron, and others, with an AUC of 0.8425 and a Log Loss of 0.3489. Of particular notice is CNN's remarkable recall of 0.9896, which shows that it can accurately discover real positive cases. This is important since precise event detection is critical in the classification of supernovae. CNN further demonstrated its effectiveness in separating real positives from false positives with an impressive precision of 0.8617. CNN's effectiveness may be traced back to its capacity to extract complex features from light curves. It does this by using the spatial correlations present in the data to identify subtle patterns that correspond to various supernova classes. Furthermore, CNN performs robustly in supernova classification due to its capacity to automatically extract pertinent characteristics from the data using many layers of convolution and pooling. All things considered, these findings highlight how deep learning methods—especially CNN—are useful for developing the fields of supernova classification and astronomy in general. With careful data preprocessing, augmentation, and model training, we have finished the supernova categorization project. In order to assure the dataset's integrity and applicability, we started our trip with thorough data pretreatment techniques. By utilizing sophisticated augmentation techniques, we improved the diversity of the data, enriching the training process and reducing the possibility of overfitting. We started model training after obtaining a well-preprocessed and enhanced dataset. During this process, we used a portion of the data to train and test different machine learning algorithms. We have determined which supernova categorization methods show the greatest promise by doing thorough testing and analysis.

REFERENCES

Artificial intelligence classifies supernova explosions with unprecedented accuracy. Center for Astrophysics. (n.d.). https://pweb.cfa.harvard.edu/news/artificial-intelligence-classifies- supernova-explosions-unprecedented-accuracy
Author links open overlay panel.

Dobryakov a, a, b, c. We propose a novel approach for a machine-learning-based detection of the type Ia supernovae using photometric information. Unlike other approaches, AbbottT.M.C., AbellP.A., BarbaryK., BellmE.C., BlondinS., BooneK., BreimanL., CabralJ., FriedmanJ.H., GuillochonJ., GuyJ., HunterJ.D., IshidaE.E.O., JonesD.O., … LochnerM. (2021, February 15). Photometric data-driven classification of type ia supernovae in The open supernova catalog. Astronomy and Computing. https://www.sciencedirect.com/science/article/abs/pii/S2213133721000056

Automated Supernova IA classification using Adaptive Learning Techniques. (n.d.-a).
https://www.researchgate.net/publication/313806700_Automated_superno
va_Ia_classification_using_adaptive_learning_techniques
Charnock, T., & Moss, A. (2017, May 5).

Deep recurrent neural networks for supernovae classification. arXiv.org. https://arxiv.org/abs/1606.07442
Demianenko1, M., Samorodova4, E., Sysak3, M., Shiriaev5, A., Malanchev6, K., Derkach1, D., & Hushchyn1, M. (2023a, February 1). IOPscience. Journal of Physics: Conference Series. https://iopscience.iop.org/article/10.1088/1742-6596/2438/1/012128/meta#:~:text=Recent%20studies%20have%20demonstrated%20the,significantly%20affects%20the%20final%20quality.

Demianenko1, M., Samorodova4, E., Sysak3, M., Shiriaev5, A., Malanchev6, K., Derkach1, D., & Hushchyn1, M. (2023b, February 1). IOPscience. Journal of Physics: Conference Series.
https://iopscience.iop.org/article/10.1088/1742-6596/2438/1/012128/meta#:~:text=Recent%20studies%20have%20demon strated%20the,significantly%20affects%20the%20final%20quality.

Dessa. (2021, May 27). How 3 engineers built a record-breaking Supernova Identification System with deep learning. Medium. https://medium.com/dessa-news/space-2-vec-fd900f5566

Guy, J., Astier, P., Baumont, S., Hardin, D., Pain, R., Regnault, N., Basa, S., Carlberg, R. G., Conley, A., Fabbro, S., Fouchez, D., Hook, I. M., Howell,
D. A., Perrett, K., Pritchet, C. J., Rich, J., Sullivan, M., Antilogus, P., Aubourg, E., … Ruhlmann-Kleider, V. (2007, January 29). Salt2: Using distant supernovae to improve the use of type Ia supernovae as distance indicators. arXiv.org. https://arxiv.org/abs/astro-ph/0701828

Lochner, M., McEwen, J. D., Peiris, H. V., Lahav, O., & Winter, M. K. (2016, September 7). Photometric Supernova Classification with machine learning. arXiv.org. https://arxiv.org/abs/1603.00882

Möller, A., & de Boissière, T. (2019, December 3). SuperNNova: An open- source framework for Bayesian, neural network based Supernova Classification. arXiv.org. https://arxiv.org/abs/1901.06384

Pelican: Deep architecture for the light curve analysis | astronomy ... (n.d.- b). https://www.aanda.org/articles/aa/full_html/2019/07/aa34473- 18/aa34473-18.html

P. V. Shinde, R. Deshmukh and P. S. Patil, "Design of Deep Learning Approach for Expression of Genes by Tumour RNA-Sequencing Data," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-7, doi: 10.1109/I2CT54291.2022.9825178. keywords: {Microprocessors;Lung;Computer architecture;Organizations;Streaming media;Feature extraction;Data mining;Cancer;RNA;CNN;DNN;Tumor;Gene Expression},

Qu, H., Sako, M., Möller, A., & Doux, C. (2021, June 8). Scone: Supernova classification with a Convolutional Neural Network. arXiv.org. https://arxiv.org/abs/2106.04370

Qu1, H., Sako1, M., Möller2, A., Doux1, C., https://orcid.org/0000-0003- 1899-9791, H. Q., https://orcid.org/0000-0003-2764-7093, M. S.,
https://orcid.org/0000-0001-8211-8608, A. M., & https://orcid.org/0000- 0003-4480-0096, C. D. (2021, July 20). IOPscience. The Astronomical Journal. https://iopscience.iop.org/article/10.3847/1538-
3881/ac0824/meta#:~:text=We%20present%20a%20novel%20method,nee d%20for%20accurate%20redshift%20data.
SNCOSMO¶. SNCosmo. (n.d.). https://sncosmo.readthedocs.io/en/stable/
SUPERNOVA CLASSIFICATION FROM SPECTRAL DATA USING MACHINE LEARNING. (n.d.-c).