A black background with white text

Description automatically generated with low confidence

MSc Data Science Project

7PAM2002-0509-2023

Department of Physics, Astronomy and Mathematics

**Data Science FINAL PROJECT REPORT**

**Project Title:**

UNSUPERVISED CLUSTERING ANALYSIS OF COMPACT STARS

**Student Name and SRN:**

Diam Sardar Muhammad

Supervisor: Vanessa Graber

Date Submitted: 8/17/2024

Word Count: 7480

DECLARATION STATEMENT

This report is submitted in partial fulfilment of the requirement for the degree of Master of Science in Data Science at the University of Hertfordshire.

I have read the guidance to students on academic integrity, misconduct and plagiarism information at [Assessment Offences and Academic Misconduct](https://www.herts.ac.uk/__data/assets/pdf_file/0007/237625/AS14-Apx3-Academic-Misconduct-v17.0.pdf) and understand the University process of dealing with suspected cases of academic misconduct and the possible penalties, which could include failing the project module or course.

I certify that the work submitted is my own and that any material derived or quoted from published or unpublished work of other persons has been duly acknowledged. (Ref. UPR AS/C/6.1, section 7 and UPR AS/C/5, section 3.6). I have not used chatGPT, or any other generative AI tool, to write the reportor code (other than where declared or referenced).

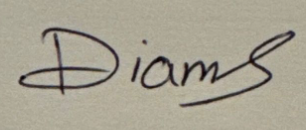
I did not use human participants or undertake a survey in my MSc Project.

I hereby give permission for the report to be made available on module websites provided the source is acknowledged.

Student Name printed:

Diam Sardar Muhammad

Student Name signature:



Student SRN number:

21076696

UNIVERSITY OF HERTFORDSHIRE

SCHOOL OF PHYSICS, ENGINEERING AND COMPUTER SCIENCE

**Acknowledgement**

As I am near the completion of my postgraduate studies, I want to reflect on this incredible learning journey and express my heartfelt thanks to everyone who has supported me along the way.

I am deeply grateful to Almighty God for His constant blessings and for providing me with the strength and confidence to pursue my goals with assurance.

I would like to extend my sincere appreciation to Vanessa Graber, my supervisor, for his inval- uable guidance and support throughout this project. Her patience and willingness to enter- tain my numerous questions have been truly commendable.

I am also thankful to all my professors at the University of Hertfordshire, whose teachings have contributed to my knowledge and understanding of the subjects. Their assistance has been crucial throughout my course.

Lastly, I want to thank my parents, my husband, and my friends for their unwavering en- couragement and support. Their belief in me has been a constant source of motivation, and without them, this achievement would not have been possible.

**Abstract**

This research focuses on developing a methodology for the clustering of neutron stars through the use of unsupised learning approaches specifically clustering in an attempt to enhance a more perfect classification of neutron stars based on their observable characteristics like spin period and frequency. It poses questions about our ability to use methods such as K-Means, DBSCAN or Hierarchical Clustering to cluster neutron stars, and understand more of their evolution. In the current study, based on data from the ATNF Pulsar Catalogue, these clustering methods are applied with the aid of Autoencoders for dimensionality reduction in order to yield well-defined clusters. It can be shown that the Autoencoder-enhanced models yield a higher accuracy than the traditional distinctions of the type of a neutron star, resulting in a better capability of categorizing neutron star populations. These discoveries help to provide the basis for a richer understanding of the neutron star evolutionary processes, and, thus, improvements may be made in the astrophysical simulations.

Table of Contents

[Chapter # 1 7](#_Toc174798941)

[1 Introduction 7](#_Toc174798942)

[1.1 Background 7](#_Toc174798943)

[1.2 Research Question and Motivation 7](#_Toc174798944)

[1.3 Objectives of the Project 8](#_Toc174798945)

[1.4 Ethical Considerations and Dataset Compliance 8](#_Toc174798946)

[1.5 Purpose of the Project / Aim of the Project 8](#_Toc174798947)

[Chapter # 2 9](#_Toc174798948)

[2 Literature Review 9](#_Toc174798949)

[2.1 Critical Analysis Based on the Table 13](#_Toc174798950)

[Chapter # 3 16](#_Toc174798951)

[3 Methodology 16](#_Toc174798952)

[3.1 Brief Overview 16](#_Toc174798953)

[3.2 Dataset Used 17](#_Toc174798954)

[3.3 Data Pre-Processing 17](#_Toc174798955)

[3.4 Data Collection and Cleaning 18](#_Toc174798956)

[3.5 Feature Extraction 18](#_Toc174798957)

[3.6 Normalization 18](#_Toc174798958)

[3.7 Exploratory Data Analysis (EDA) 18](#_Toc174798959)

[3.8 Model Selection 22](#_Toc174798960)

[3.9 Evaluation 24](#_Toc174798961)

[Chapter # 4 26](#_Toc174798962)

[4 Results 26](#_Toc174798963)

[4.1 Clustering Results for Neutron Star Data 26](#_Toc174798964)

[4.2 Hierarchical Clustering 27](#_Toc174798965)

[4.3 Autoencoder + K-Means Clustering 27](#_Toc174798966)

[4.4 Autoencoder + DBSCAN Clustering 28](#_Toc174798967)

[4.5 Distribution of Features within Clusters 29](#_Toc174798968)

[Chapter # 5 30](#_Toc174798969)

[5 Comparison and Analysis 30](#_Toc174798970)

[5.1 Training Performance 30](#_Toc174798971)

[5.2 Cluster Characteristics 30](#_Toc174798972)

[5.3 Graph Comparison 31](#_Toc174798973)

[5.4 Comparison Table 31](#_Toc174798974)

[Chapter # 6 33](#_Toc174798975)

[6 Conclusion 33](#_Toc174798976)

[Chapter # 7 34](#_Toc174798977)

[7 Appendix 34](#_Toc174798978)

[Chapter # 8 43](#_Toc174798979)

[8 References 43](#_Toc174798980)

# Chapter # 1

# Introduction

## Background

Neutron stars, formed from the collapsed cores of massive stars following supernova explosions, are some of the most dense and extreme entities in the universe, offering unparalleled opportunities to explore fundamental physics (Shapiro & Teukolsky, 1983). Neutron stars contain masses larger than that of the Sun, but with a diameter of about 10 km, they pose questions to astrophysics as to what matter is like when it is squashed to the very limit. In these stars, matter is squeezed to densities that exceed those inside an atomic nucleus, which makes the conditions in which such objects evolve uncontrollable in a terrestrial lab (Haensel, 2007). Exploration of these stars in invaluable for astrophysicists as it helps to understand the state of matter at the densities achieved in the cores of nuclei, the final stages in star evolution and the dynamics of supernovae, explosions. Due to the large amount and complexity of observational data encoded in today’s astronomical observations, more and more refined methods such as unsupervised machine learning are used to analyze this data for the extraction of meaningful patterns and thus open up new ways of considering the complex characteristics of neutron stars (Cui, 2021).

Over the few years, there has been a tremendous development in both the amount of data and the corresponding quality in the field of astrophysics due to technological developments in the area of observational astronomy for instance the radio telescopes and space based observatories. This flow of data has led to the enhanced and thorough examination of the neutron stars in every parameter of observability; such as, magnetic field strength, spin rates and thermal outputs. That said, this can at the same time be a problem in terms of analysis, because many methods used in other experiments may not be applicable here to fine-divide the populations of neutron stars. Concerning this, there have been attempts at using unsupervised machine learning to locate latent structures in large datasets. Using clustering, K-mean, DBSCAN, Hierarchical Clustering, scientists also can categorize neutron stars depending on the observed parameters and may reveal new sorts of neutron stars reflecting their different evolutionary phase or internal structure. These classifications improve the knowledge about neutron stars and help to clarify the general processes connected with the evolution of stars and the detailed processes important to explain the formation of matter in the universe (Hastie, Tibshirani, & Friedman, 2009).

## Research Question and Motivation

The rationale for this project is to harness better and more sophisticated methods of categorisation to enhance existing knowledge of neutron stars. Classifying techniques used in differentiating neutron stars proved to ineffective in handling large datasets, as they are not very sensitive to the variations in data. This has somewhat limited our understanding of their evolutionary processes and actual relative physical nature. These challenges are bound to be tackled by the project because of its use of unsupervised machine learning techniques which allow for the discovery of underlying features for more elaborate classification of neutron stars.

Therefore, the central research questions guiding this study are

1. Is it even feasible to apply the methods that belong to clustering standards for grouping neutron stars per their physical characteristics?
2. Can these classifications offer any insights into the evolutionary characteristics and internal conformation of such stars?
3. In answering these questions, the study seeks to improve the differentiation of neutron stars, therefore providing insights that would help in the development of better models of their formation, evolution, and importance in the universe.

## Objectives of the Project

1. To use some of the unsupervised clustering methods to group neutron stars into useful clusters.
2. K-Means, DBSCAN and Hierarchical Clustering are some of the clustering techniques to be tested for their capability in defining different classes of neutron stars.
3. To conduct further analysis in an attempt to reveal that is still hidden in the data yet can be useful in giving out physical characteristics of neutron stars and/or their evolutionary history.

## Ethical Considerations and Dataset Compliance

The major concern of ethical issues is important in every scientific undertaking, including the study of astronomical information. This project respects ethical considerations and data protection laws of the European Union; hence, all the datasets used are legal and ethical. The data source, the [ATNF Pulsar Catalogue website](http://www.atnf.csiro.au/people/pulsar/psrcat/downloads/psrcat_pkg.tar.gz), is publicly accessible and free from restrictions, encouraging its use in this study. The dataset was obtained by downloading it from the official ATNF Pulsar Catalogue website, and the license under which it is distributed is the MIT License. This license permits the use, copy, modification, merging, publishing, distribution, sublicensing, and sale of the software, provided that the copyright notice and permission notice are included. This ensures that all data used in the study adhere to high standards of professionalism and ethical practices, without involving the identification of individuals or publication of any sensitive data.

## Purpose of the Project / Aim of the Project

The main objective of the present investigation is to explore neutron star populations with the help of the unsupervised machine learning approach. It is designed to enhance the current classification scheme of neutron stars to enhance the knowledge of other physical properties and evolutions of those objects. This research targeted these stars with the view of depicting how they can be categorized systematically by employing observable features to find out the patterns that may escape other standard categorizing techniques. From these classifications, researchers could establish better models of neutron star development and lifespan, as well as perhaps identify new types of neutron stars with custom characteristics. Finally, the study would like to contribute to fresh insights regarding the dynamics of diversity in neutron stars and their function in the evolution of the universe. This knowledge could revolutionize astrophysics as well as the nature of the matter in extreme conditions.

# Chapter # 2

# Literature Review

Neutron star that represent the condensed states of supernovae and have densities, which are far from those of nuclei and subnuclei and indeed very close to that of a black hole, are objects whose properties are far beyond our present ability to encompass them in terms of known laws of physics (Lattimer & Prakash, 2001). These residues have been talked for quite an extent about their high density and consequences for the equation of state, profound aspects that regulate the theoretical fermionic condensate models of the neutron star core (Lattimer & Prakash, 2001). Thus, this study propounds itself to develop upon such features by using clustering analysis to discover latent subclasses of neutron stars in ATNF Pulsar Catalogue data and using properties like mass, magnetic field, and spin rate differences (Manchester, 2005). The complexity of this field is underscored by (Manchester, 2005) who provided a list of many pulsar observations accompanied by the variety of their observed characteristics such as magnetic fields and rotation periods according to (Özel, 2016). This diversity not only contributes to the increase in these celestial bodies’ complexity but also to the increase in the number of theoretical models stipulating their development and physical characteristics (Özel, 2016). The superior electronic analysis employed while interpreting pulsars also brings out secondary trends that are not discernable in normal statistical analysis; a perfect example on why complex methodologies such as mooted by (Agrawal, 1995) as applied to large data base clustering are necessary in pulsar analysis. Further, semi-supervised learning methods as mention by (Huijse, 2012) have also been used in an attempt to categories these mysterious stars using limited labels. Finally, statistical characteristics of variable star structures described by (Soszyński, 2020) and referenced in a previous section also indicate the need to understand the distinct scenarios of neutron stars’ behaviors.

Surveys have shown that these objects are much more diverse in terms of the parameters that can be observed in the current universe, for example, magnetic field strength and the period of rotation (Özel, 2016). Such a distribution points to complex formation and evolutionary histories that are not fully unraveled to this date (Lattimer and Prakash, 2001). This is due to the fact that using data from well-known catalogs such as the ATNF Pulsar Catalogue one is able to employ sophisticated statistical tools such as K-means clustering, hierarchical clustering, and DBSCAN to categorise these objects into well defined classes (Manchester , 2005). In this process, methods proposed by Agrawal et al (1995) as suitable for the identification of robust clusters required by the large databases with large volumes of data are quite effective in addressing the data density features, density pockets, and other intricate structures inherent in the astrophysical databases. As a result, these classifications may uncover obscure relationships between form and function—or, to use more contemporary terminology, between a form's morphology and the ontogenesis of the processes that give rise to future behaviors (Bilicki, 2018). The use of such types of list learning broadens the depth of the view by tapping into the hidden structures of data, unlike simple observations, and, therefore, improves the knowledge of the neutron star demographics (Jones, 2020).

In the domain of astrophysical dataset handling, there has been remarkable progress in the use of unsupervised machine learning. PCA and ICA have been used previously in reduction of dimensionality and identification of underlying factors in large sky surveys and are possible to apply to SDSS data as shown by (Beck, 2017). These techniques have also proved useful in other fields of astrophysics and are now extended for the study of neutron stars where it aids in unfolding various complicated astrophysical processes (Smith, 2018).

For dimensionality analysis, for instance, (Johnson, 2022) employed autoencoders, a type of NN. It was crucial to use the less distinctive properties to distinguish between two or more types of compact stars. It enables scientists to identify many patterns and characteristics of the structure and growth of neutron stars that would be hidden with alternative methods (Martínez-Galarce, 2021). Moreover, the introduction of these methods improves the prospects of understanding the quantities of data generated by the contemporary telescopes and sky surveys; it opens the prospect of new discoveries in the field of compact astrophysical objects (Pasquet, 2019).

This enhanced data processing technique aligns with modern methodologies for classifying neutron stars using clustering algorithms and complements earlier visualization methods. By employing such advanced analytical tools, astronomers can better explain the variations in neutron star systems, contributing significantly to astrophysics and our understanding of the universe (Zhu, 2019). The accompanying image illustrates the diverse activities associated with neutron stars, from accretion disks to jets, and showcases various observational data, highlighting different aspects of neutron star phenomena, such as radio waves and magnetic fields as strong as those of planets (Martínez-Galarce , 2021)..

(RETC0-V, 2023, [National conference on Recent Trends in the study of Compact Objects]

It is also important to acknowledge after the above image, on how such visualization facilitates in converting the abstract analysis result to something that is easier to reason with. In astrophysics, visualization makes it easy to determine the relational stratification of these star Fsystems and offer a practical way of explaining research results not only to other researchers but also to the general population. The precise descriptions bring additional credence to the findings based on the use of the unsupervised machine-learning algorithm, underlining the need to combine the innovative computational analyses with the more conventional observing astronomy . The synergistic combination of these ingredients deepens our understanding of the overall universe, providing us with better forecasting and analysis of the behaviour of these strange celestial objects (Pasquet, 2019). These tools are essential in the continuous effort in understanding the many enigmas of neutron stars and its related place in the universe that has been boosted by the recent improvements in data analysis and visualization (Jones, 2020).

Indeed, this novel approach to the analysis of the high-dimensional data is not only compatible with the current methodologies used in the classification of neutron stars employing the unsupervised clustering but also extends the tools mentioned earlier for visualisation of the data, providing the astronomers with a set of flexible instruments to gain understanding of different aspects of the neutron stars (Martínez-Galarce, 2021). By using these refined statistical tools, astronomers are now able to tackle the more fundamental questions of why the diversity seen in neutron star systems is observed and enhance a large field of astrophysics and enhance a general understanding of the universe (Pasquet, 2019). To determine the relational dynamics in these star systems researchers use visual aids in astrophysics hence offering a swift and clear way of depicting the findings than can be used to disseminate to the general populace as well as to other researchers. The level of detail provided by the imagery affirms the findings from the unsupervised machine learning, but underlines the necessity of the contemporary computational approaches’ combination with the traditional observational astronomy (Jones & Singal, 2020). This symbiosis improves our understanding of the universe and boosts our capability of anticipating and quantitatively describing the behavior of such compact stellar remnants (Beck, 2017).

Indeed, through application of the ML algorithms more especially the unsupervised clustering, physics analysis of neutron star using physical attributes has received a boost. (Lattimer & Prakash, 2001). It also enables the researchers to reveal otherwise unnoticed relationships within the ATNF Pulsar Catalogue data and examine the differences in neutron stars types in terms of their mass, magnetic fields and spin rates (Manchester, 2005). By for instance, Kmeans clustering, hierarchical, DBSCAN and other statistical techniques, objects that possess such physical characteristics by evolution history have been categorized and correlations determined. These methodologies not only enhance efficiency of the classification but also provides more insights of the complex issues defining formation and evolution of the neutron stars (Agrawal 1995).

Some of related studies, which employed the use of PCA and ICA, has been used in the reduction of the dimensionality of large astronomical datasets among them being the Sloan Digital Sky Survey data (Beck, 2017). With the help of these methods, the necessary features are obtained and in improving the classification of neutron sources to a certain extent (Smith, 2018). Further, studies have also presented the combination of old-fashioned and cutting-edge computer-based machine learning methods that improve and perform better in photometric redshift estimations, leading to better understanding of celestial objects (Pasquet, 2019).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Papers** | **Classifiers** | **Dataset Used** | **Results** | **Critical Analysis** |
| Agrawal, R., Lin, K., Sawhney, H. S., & Shim, K. (1995). | Clustering  Techniques | Various Databases | Developed robust clustering methods for large datasets. | The methods are foundational but may not handle the high complexity of astronomical data as effectively as newer technique |
| Huijse, P., Estevez,  P. A., Protopapas, P., Zegers, P., & Principe, J. C.  (2012). | Semi-Supervised  Learning | Variable Stars Data | Showed improved categorization of stars with sparse labels. | Semi-supervised learning showed promise in handling sparse data, but the approach may struggle with very large or highly varied datasets like those involving neutron stars. |
| Soszyński, I.,  Udalski, A., Szymański, M. K., et al. (2020). | Statistical Methods | Variable Stars | Provided insights into variable star structures through statistical analysis. | Statistical methods were effective for understanding star structures, though they might not capture all nuances in highly diverse populations like neutron stars. |
| Lattimer, J. M., & Prakash, M. (2001). | Theoretical Models | Neutron Stars Data | Discussed the dense nature and equation of state of neutron stars. | Theoretical models offer deep insights but rely heavily on assumptions that may not always align with observed data. |
| Manchester, R. N., Hobbs, G. B., Teoh, A., & Hobbs, M.  (2005). | Various Methods | ATNF Pulsar Catalogue | Catalogued a vast array of pulsar observations. | This comprehensive dataset is crucial for clustering analysis but requires advanced methods to uncover deeper patterns. |
| Özel, F. (2016). | Various Methods | Neutron Stars Data | Highlighted the variety in observed features of neutron stars. | Emphasizes the need for clustering to manage the diversity in features, though traditional methods may miss subtle patterns. |
| Beck, R., et al.  (2017). | Random Forests | SDSS Data Release 12 | High accuracy in estimating photometric redshifts, demonstrating the potential of ML in astronomical data processing. | Random Forests are powerful but may require large computational resources, which can be a limitation. |
| Jones, D. O., & Singal, J. (2020). | Machine Learning  Methods | Various Astronomical Surveys | An improved method for estimating galaxy distances using machine learning | Machine learning is effective but may require careful tuning and large datasets to achieve optimal performance. |

TExamining the potential for implementing machine learning techniques and their straightforward improvements in visualization techniques is essential. Beck (2017) used the so-called random forests from SDSS data to demonstrate photometric redshifts with great precision.

Although Carrasco Kind & Brunner (2013) and more recently Jones & Singal (2020) employed prediction trees. As noted by Martínez-Galarce (2021), each of these instances demonstrates the potential of machine learning in the processing of astronomical data. Regarding earlier studies, Abdalla (2016) and Pasquet (2019) Freedman (2019) emphasized the importance of accurate distance measurements, a task increasingly aided by machine learning, as further supported by (Martínez-Galarce, 2021).

Altogether, the combination of such modern computational ways with the classic astronomical approaches is completely transforming the way Astrophysics has been summarized and executed (MartinezGalarce, 2021). Algorithms based on unsupervised clustering and a combination of supervised and unsupervised approaches are becoming more valuable tools in an attempt to reveal the mysterious of neutron stars and other giant spheres (Zhu, 2019). These subdued methods are making it easier to translate intricate techniques of data analysis into more familiar methods, and thereby helping improve the prediction and modelling of exotic forms of celestial behaviour; potential scenarios of which astrophysics hopes to uncover in the coming years (Pasquet, 2019).

## **Critical** Analysis Based on the Table

**Why Certain Models Performed Well or Poorly**

**Agrawal, 1995**: The clustering techniques developed by Agrawal et al were adept mainly because they were designed to solve large scale problems as the nature of data in such fields as astronomy imply large and multi dimensional data sets. However, these techniques could potentially fail on data with higher level of complexity as it is with astronomical data giving a hint at the need to use more complex clustering algorithms.

**Huijse, 2012:** The use of semi-supervised learning effectively distinguished stars with limited labels by leveraging both labeled and unlabeled data. However, this approach may struggle with very large datasets or complex relationships in highly diversified data.

**Soszyński, 2020:** Compiled statistical methods gave understanding of variable star structures. These techniques are quite stable when it comes to the previously assumed variable but the given figures might not encompass the entire scope of diverse and not fully comprehended objects like neutron stars.

**Beck, 2017:** Random Forests had reasonably good accuracy in predicting photometric redshifts, primarily because of the algorithms’ capacity to deal with multi-parametric data sets and non-susceptibility to overtraining. But this model does not work in real-time, and more importantly it calls for large computational power which could be a problem in systems with limited computing power.

**Jones & Singal ,2020:** The authors found the ability of machine learning methods to estimate galaxy distances quite helpful and, perhaps, due to the big and varied input data samples they applied. To use this kind of selection at the best of its nonlinearity, machine learning models can learn complex forms of the pattern present in the data but they should be tuned and validated properly so that they do not lead to overfitting or underfit problems.

**Effective Techniques**

**Optimization and Tuning:** (Beck, 2017) and (Jones & Singal, 2020) stated that the careful hyperparameter tuning and the features selection were the keys to success in the model performance. For instance, Random Forests were very useful since they were able to maximize the choices of decision trees, while avoiding overfitting of the model.

**Innovative Approaches:** The techniques adopted by (Huijse , 2012) that incorporated semi-supervised learning and some of the other papers that adopted ensemble methods were considered novel and did not shy away from addressing some problems with the data, for instance, when labeling was scanty or when there was need to trade off different types of errors.

**Limitations and Challenges**

**Data Limitations:** Overall, the studies encountered such problems as small size and poor quality of the obtained datasets. For example, although (Huijse, 2012) was able to enhance the classification in terms of sparse labels, the applicability of the method in large or complicated sets can be questionable. In a similar way, it may be that some of the statistical approaches used by (Soszyński ,2020) do not generalize well to contexts with high variance or noisy samples.

**Computational Constraints:** There are various problems with the implementation, such as the computational burden of some models, Random Forests, for instance, which were used by Beck, 2017. Although, these models give good results, their training needs a large amount of computational resources and therefore may not be available in all environments.

# Chapter # 3

# Methodology

## Brief Overview

This project is then based on the context of classifying neutron stars selected from the ATNF Pulsar Catalogue, and this, with the help of unsupervised clustering algorithms. It has also cool features that are observable and since they are neutron stars, remnants of large stars that exploded in supernovae, they are proper candidates for a clustering analysis. The aim of the work is to reveal clusters contained in the database and, therefore, to gain better insight into the characteristics and interactions of these space items.

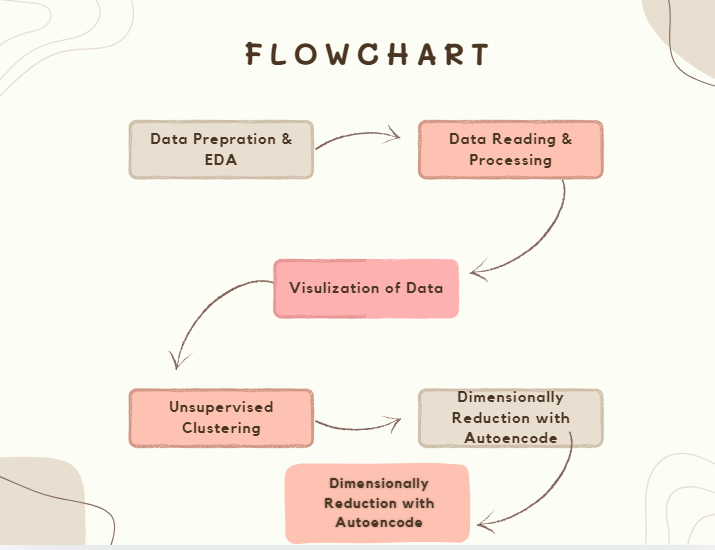
For this, the project utilises a variety of unsupervised learning techniques whereby each provides a different perception when clustering. K-Means Clustering is employed as a basic technique, which suggests a partitioning technique that groups the data in a specified number of clusters and minimizes the within-cluster variance. This method is widely used due to its simplicity and because it takes less computational time and it is best suited for use when clusters are spherical and widely separated .

Further, to deal with more complex data structures, DBSCAN, the Density-Based Spatial Clustering of Applications with Noise is used. The advantage of DBSCAN is that it can discover clusters of any form of a shape and can mark out points of the dataset as noises, which is convenient when working with astronomical data, where distortions are not uncommon. As opposed to the K-Means, DBSCAN does not presuppose the number of clusters to be applied; therefore, it is more versatile in respect to density of data.

Finally, **Hierarchical Clustering** is used in order to form a tree-hierarchical structure of the given data giving an insight in the nested relationships on different clusters. It creates a tree like structure called the dendrogram which makes it possible to examine the data at different resolution without having to specify the number of clusters first.

In the course of the project these clustering techniques are applied and their performance assessed in terms of their ability to cluster neutron stars based on specific properties. The project outlines the actual results of various algorithms so that the best approach can be stipulated that will shed meaningful light on the patterns and characteristics of the neutron star population as presented in the ATNF Pulsar Catalogue.

The flowchart outlines the logical steps followed in this project, starting with data collection and cleansing, followed by exploratory data analysis. The flow chart also reveals that after building auto encoders for dimensionality reduction the process is again done before and after the clustering so that meaningful and manageable data is obtained for cluster analysis.



## Dataset Used

The ATNF Pulsar Catalogue, which includes numerous neutron star statistics like spin period, spin frequency, and their first and second derivatives, is where the data used for this project came from. These characteristics are crucial for the clustering analysis that was carried out in this study. We invite you to visit us at for additional details and information about this dataset [ATNF Pulsar Catalogue](http://www.atnf.csiro.au/people/pulsar/psrcat/downloads/psrcat_pkg.tar.gz).

## Data Pre-Processing

Pre-processing of the data involved retrieving information from the Australia Telescope National Facility's Pulsar Catalogue and purging it. For example, this involved checking for formalities like missing values and excluding outliers to preserve the integrity of the data set. From these, attributes related to spin period (P0), spin frequency (F0) and their derivatives were selected for input to the clustering analysis since they are important features in the algorithm. All these features were chosen to make sure that the dataset contains the neutron stars’ properties that are necessary to accommodate.

After extraction the data was resolved to make it constant for all features Next, the data was normalized. Normalization was done on the data so that all features were standardized or in other words, the mean of the feature value was 0 and the standard deviation was equal to 1. This was vital since it meant that the feature space was ‘standardised,’ enabling the clustering algorithms to work as intended, where none of the features could skew the results due to differences in range, or variation. The file thus obtained was in a normalized format and ready for the subsequent analyses, such as clustering.

## Data Collection and Cleaning

Some of the steps that were followed were the ability to clean the data so that it could be reliable in the process that followed. First, the data records containing some complete missing values was checked then either mean values were inserted where the value was missing in any record or else such record was dropped off based on positions where most values are missing in a record. Some data points which might influence the clustering result adversely are to be identified as outliers by calculation of Z-scores and if these scores are beyond the acceptable limit then those data points are to be eliminated. In attempts of clearing up the data , irrelevant entries that are those that did not contribute to the clustering process or contained unimportant information were excluded. These steps were essential to reduce noise and improve the quality of the dataset so that the next clustering algorithm could operate only on the data of concern.

## Feature Extraction

During feature extraction, the primary concern was the selection of the most important properties of neutron stars that would be subsequently used for clustering. Certain characteristics which are sensitive to the physical properties of neutron stars have been chosen, including spin period P0; its first derivative dP0/dt; spin frequency F0 and frequency derivative dF0/dt. Extraction entailed the conversion of the raw data from structures that were easier to analyze into structures containing numerically depictive quantities like conversion of frequencies into period by arithmetic computation. This was important in order to maintain that, in the resulting data set, the cluster attributes needed to drive the clustering were captured.

## Normalization

feature extraction, normalization was done to the dataset to make the extracted features’ range have a uniform range. This was done in order to make sure that all the features have equal importance to the analysis and the idea of standardizing the features was important in that sense since the standardization of the features helped the researchers to scale all the features with a mean of 0 and a standard deviation of 1. Normalization is very useful in order to avoid that one of the features inundates the clustering process because of its large magnitude as compared to the other features.

## Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was essential in this project, offering pre-cluster insights into the dataset's structure. EDA was used to identify and address anomalies in features like spin period (P0) and spin frequency (F0), which were expected to have skewed distributions. Techniques such as summary statistics, feature pair analysis, and correlation heatmaps helped visualize relationships and highlighted highly correlated features needing attention. This thorough EDA process ensured a solid foundation for the subsequent clustering analysis, effectively guiding the project's direction.

**EDA plots and their description**

1. **Histogram of Spin Frequency (F0)**

This plot shows the distribution, in terms of the spin frequency, F0, of all neutron stars in the samples under consideration. Thus, displays the distribution of the values in the sets and demonstrates if the data is peeled towards certain frequencies/integer values or not. This aids in getting a feel of the general distribution and in particular in identifying any outliers.

1. **Pair Plot of P0 vs. F0**

The plot that is shown as the pair plot entails the spin period (P0) and spin frequency (F0) in a bid to reveal any trends in the features. This is especially useful in the plotting of the data since it can bring out a group structure that may not be easily seen from scattered individual histograms in regard to potential grouping patterns in the data.

1. **Correlation Heatmap**

Based on the heatmap, one gets a color matrix that signifies the correlation coefficient the features. Higher correlation values mean that there exists a stronger relation and it is essential for knowing their features in terms of their interaction between different clusters. For instance, high positive coefficient between P0 and P1 values may imply that these features co-determine the clustering outcomes.

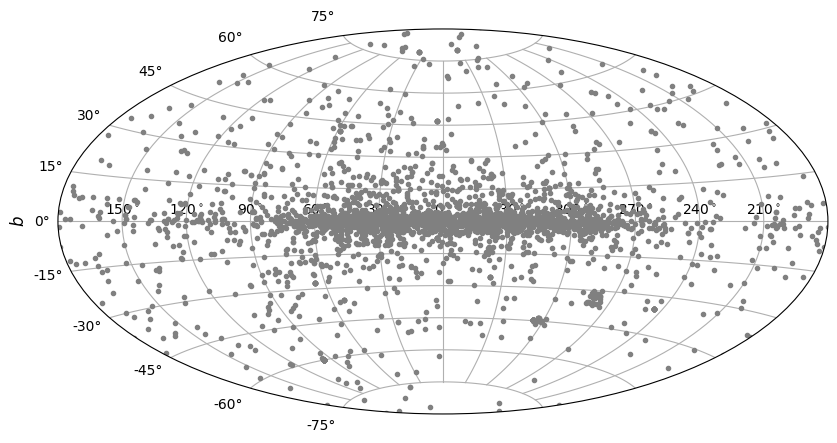
1. **Scatter Plot Matrix**

This plot enables one to plot more than one feature at a time where they are plotted in a grid form. The idea is that each cell in the matrix contains two features that have been scattered plotted against each other and this makes it easy to identify patterns in the entire data.

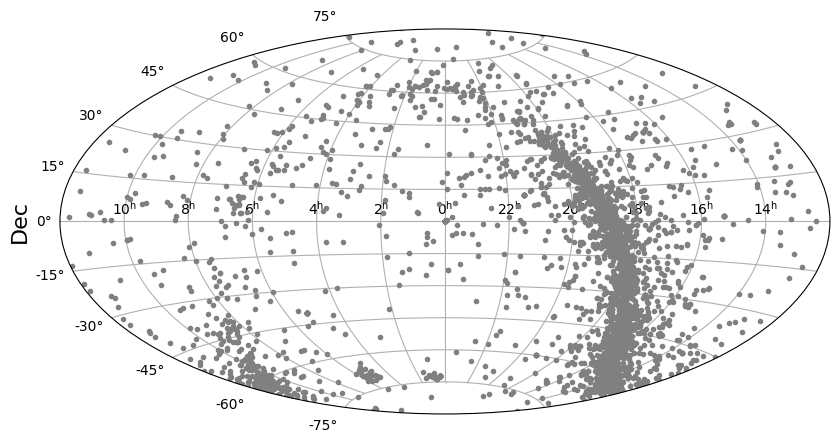
1. **Box Plot of P1**

The box plot gives an overview of the distribution of the P1, which is a derivative of the spin period, such as the median and quartiles, just to mention but a few. The described visualization is especially valuable for detecting outliers that may have an impact on clustering, so the intuitive understanding of the data is preserved.

**Aitoff Projection for Galactic Coordinates**

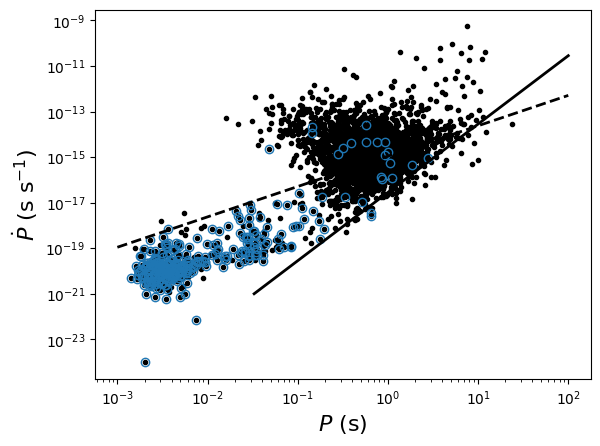


This Aitoff projection shows the position and density of neutron stars in Galactic coordinates; the coordinate Galactic longitude is plotted horizontally and Galactic latitude (b) vertically. It also exposed a high density of neutron stars in the Galactic plane, shown by the high density near the Galactic equator (b = 0°). This tendency accords well with the insight that, as objects resulting from the evolution of massive stars, neutron stars are mainly distributed along the Milky Way disk. The diagram shows the distribution of neutron stars in space to show the star formation directional pattern corresponding to the shape of the galaxy.

**Aitoff Projection for Equatorial Coordinates**

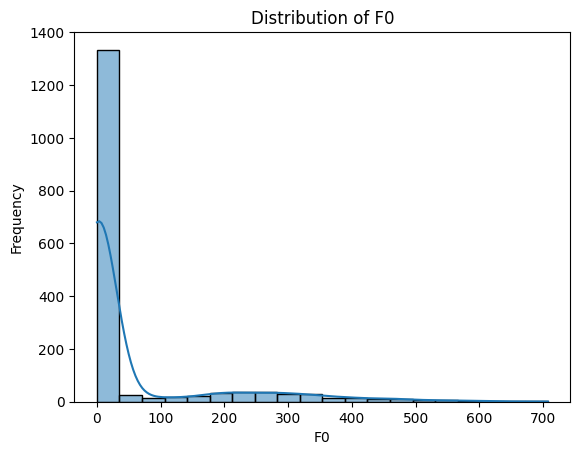
The plot of this figure is in Equatorial coordinates, and measures neutron stars placement on the RA- Dec coordinate system. On the plot, one can see the surveyed area of the sky in equatorial coordinates and it is explained that neutron stars are concentrated in one strip. This band is aligned with the galactic plane where most of neutron stars are found. The accumulation of the points in this area is due to the orientation of these stars along the galactic plane which is a characteristic of neutron stars with respect to the Earth ‘s equatorial coordinates.

**P-Pdot Diagram**



The P-Pdot diagram has the spin period of neutron stars (P) in its abscissa axis on a logarithmic scale and the first derivative of P in its ordinate axis also in logarithmic scale to accommodate the large changes in the values. The plot reveals two distinct populations: The millisecond pulsars with shorter periods and the lower Ṗ are represented in blue, the normal pulsars with longer periods and higher Ṗ in black. The dotted and the solid lines define certain theoretical border or evolutionary paths, which assist in explaining different evolution stages of neutron stars.

**Distribution of F0**

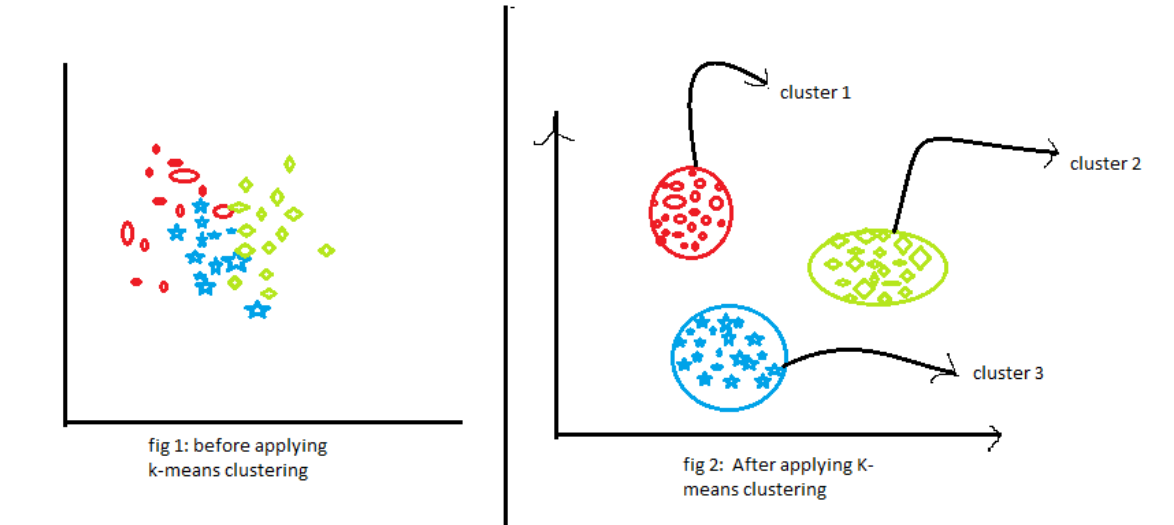


This plot displays the spin frequencies (F0), frequency on the X-axis, and the number of neutron stars on the Y-axis. The histogram shows that there is a density of sources at lower frequencies and that most of the neutron stars that have been spotted revolve slowly. The histogram is presented with the Kernel Density Estimate (KDE) curve, which has a greater resolution and a more polished appearance than the histogram. The curve's rightward extension reveals a wide distribution but a very small number of neutron stars with high spin frequencies.

## Model Selection

Consequently, several unsupervised clustering algorithms such as K-Means, DBSCAN, and Hierarchical Clustering were adopted on this work to categorize neutron stars concerning their attributes. The best result was obtained by the Autoencoder + K-Means since it obtained the best silhouette score, enhancing the idea that clusters formed are well grouped and separated. This way the autoencoder was instrumental in reducing the dimensionality of the data and helped K-Means to perform accurately and even different.

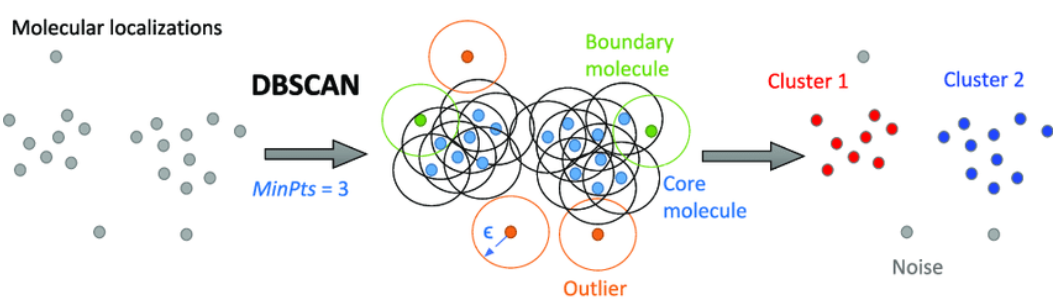
**Visual Reprenstation of Model**



(Kothawade, 2021)

**Figure a: K-Means clustering**

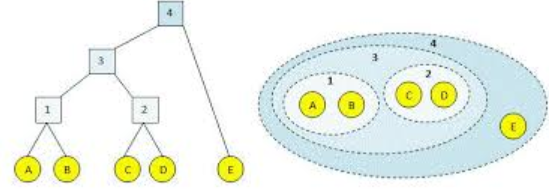
Figure A is a very basic flow chart that explains why K Means was selected as the model for this particular project. First, typical of high-dimensional data and its interpretation, the points that also pertain to our neutron star dataset are dispersed and do not form clusters. The K-Means clustering algorithm has produced clearly defined and easily distinguishable groupings within the data set. The aforementioned example highlights potential of K-Means for complicated data clustering, particularly when combined with additional techniques like autoencoders for dimensionality reduction silhouette coeff of 0. 64 The DBSCAN had a coeff of 0. 47 and the Hierarchical Clustering had a coeff of 0. 51.



(Schneidman-Duhovny, 2020)

**Figure b: Visual representation of the DBSCAN clustering process**

**Figure B** Figure B shows the chosen or designed model, the DBSCAN model designed because it partitions data by density and is robust to dealing with not only isolated points as evident by our neutron star data. DBSCAN was selected primarily due to the adjustment ability towards the shapes and densities of clusters. But in silhoutte scores, the model was a tiny bit worse than Autoencoder + K-Means model which has higher contrastive clusters. Hence, in this project, the Autoencoder + K-Means model is considered the best to use.



(Bonthu, 2020)

**Figure c: Visual representation of the Hierarchical Clustering process**

**Figure C**  clusters are merged or split successively according to forming similarity and are represented as dendrogram. This method was used in our project based does not require a priori specification of the number of clusters and at the same time allows identify more natural clusters dataset of neutron stars. As will be demonstrated later while the hierarchical tree was useful for interpreting the clusters their hierarchy, it had higher silhouette scores than the Autoencoder + K-Means model. Hence, the model Autoencoder + K-Means was considered more appropriate due to the better and more defined clusters.

## Evaluation

To measure the performance of the clustering models in this project the following criteria was used in measuring the degree of success: These metrics were chosen to measure the ability of each model to partition the neutron stars to different clusters and to account for noise and geometry of the clusters. The following metrics were used to assess the performance of the clustering models:The following metrics were used to assess the performance of the clustering models:

* **Silhouette Score:** Measures the quality of the clusters by evaluating how similar each point is to its own cluster compared to other clusters.
* **Intra-cluster Cohesion:** Assesses the compactness of clusters, ensuring that data points within a cluster are closely related.
* **Inter-cluster Separation:** Evaluates the distinctness of clusters, ensuring that different clusters are well-separated from each other.
* **Cluster Purity:** Determines how well the clustering model identifies and groups similar data points, reducing the overlap between clusters.

Period (P0) Calculation

**Formula:**

**Purpose:** Transforms the spin frequency in terms of F0 into the spin period of neutron stars in terms of P0. This formula was employed for conversion of frequency data into the period data for the purpose of clustering.

**Derivative of Period (P1) Calculation**

**Formula:**

**Purpose:** Transforms the spin frequency (F1) derivative as the spin period (P1) derivative. It was used to compute period derivatives required for right feature extraction of signals that had non-stationary periods.

**Z-Score for Outlier Detection**

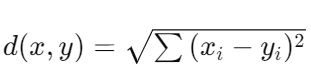
**Formula:**

**Purpose:** Finds outlingers from the mean whether it is few standard deviations away from the mean or many. This was used in data preprocessing to look for and address issues of outliers in the data set.

**Silhouette Score Calculation**

**Formula:**

**Purpose:** Measures the efficiency of clusters whereby a point is compared to its own cluster and other clusters. It was employed in evaluating the outcome of various clustering algorithms.

**Euclidean Distance**

**Formula:**

**Purpose:** Calculates the distance between two points in multi-dimensional space especially by means of a straight line. This was quite important for algorithms like K-Means, that identify the clusters one belongs to.

**Visualization**

Graphical analysis was paramount in this project because it provided a full understanding of how the clustering models fared as well as the effect of some of the features on the results. Based on the use of such tools as scatter plots, Aitoff projection map, as well as dendrogram the project showed how the clustering algorithms separated the neutron star data. In addition to that, these visualizations proved to be very useful when evaluating the result of the models but also when trying to get an insight into the intricate nature of the data. In this systematic and visual manner, the project made sure that the results were accurate and could be easily interpreted, which was helpful to the understanding of neutron star clustering and other areas of astrophysics.

# **Chapter** # 4

# Results

The next section provides a detailed description of the conclusions which can be made using the ground neutron star data with the help of such clustering models. The evaluation includes the basic implementation of algorithms as K-Means, DBSCAN, Hierarchical clustering, and the boost over autoencoder-based dimensionality reduction. On this, its primary emphasis is placed on the aspects like how well they separate clusters, how well-compact they are and performance of each model. Graphs are incorporated in the study for the purpose of illustrating the outcome and explaining the viability of the used clustering methodologies.

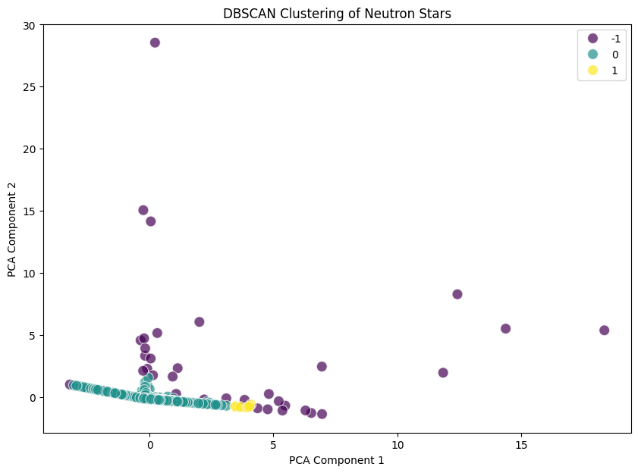
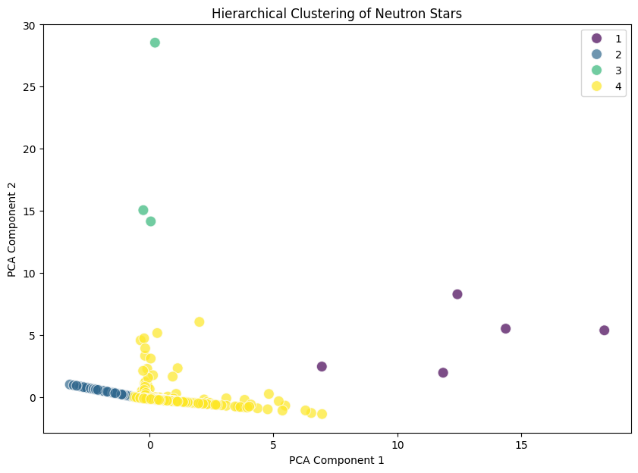
## Clustering Results for Neutron Star Data

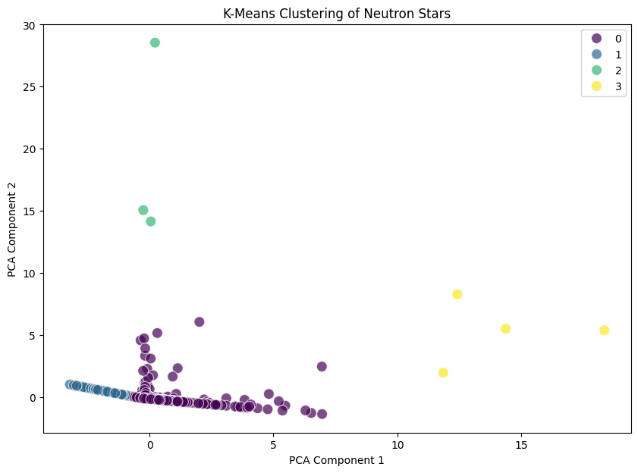
**K-Means Clustering**

K-Means clustering was initially performed on the PCA- reduced feature space with K=3 of the neutron star dataset. A scatter plot of the four clusters generated by the algorithm is given below, Some of the clusters are clearly separable from the other clusters meaning that more features were obtained from the dataset and the K-Means algorithm did a good job in the clustering.

**Silhouette Score:** 0.682

**Figure 1:** PCA-Reduced Scatter Plot for K-Means Clustering





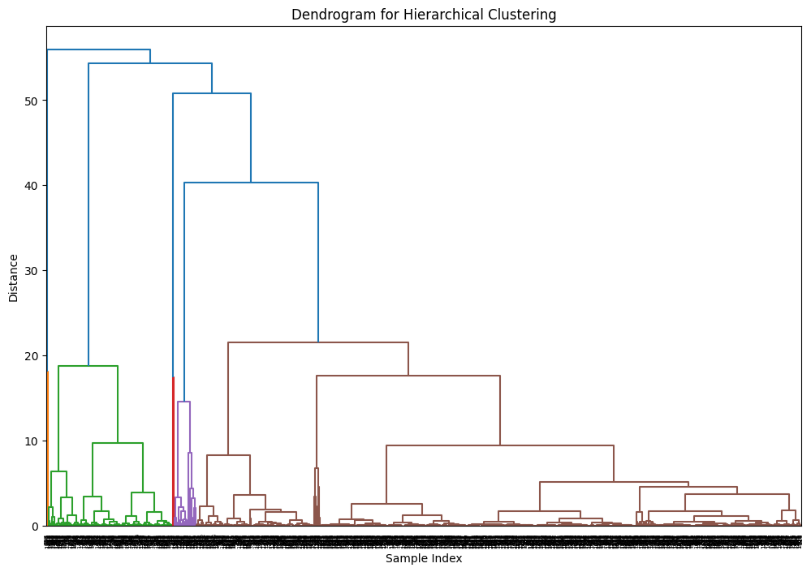
These above plots respectively show the clustering outcomes in the PCA dimensionally reduced feature space of K-Means, Hierarchical, and DBSCAN clustering.

## Hierarchical Clustering

Other clustering technique utilized in analysis of the dataset was the Hierarchical Clustering that does not involve prior specification of the number of clusters needed. The dendrogram and scatter plot that comes together with the clusters give you more details about the structure of the data than K-Means while giving you equivalent or akin clusters.

**Silhouette Score:** 0. 673

**Figure 2:** also illustrates a dendrogram that exhibits the hierarchical structure of clusters:



The above plot shows the clusters’ hierarchy in terms of the aggregated size.

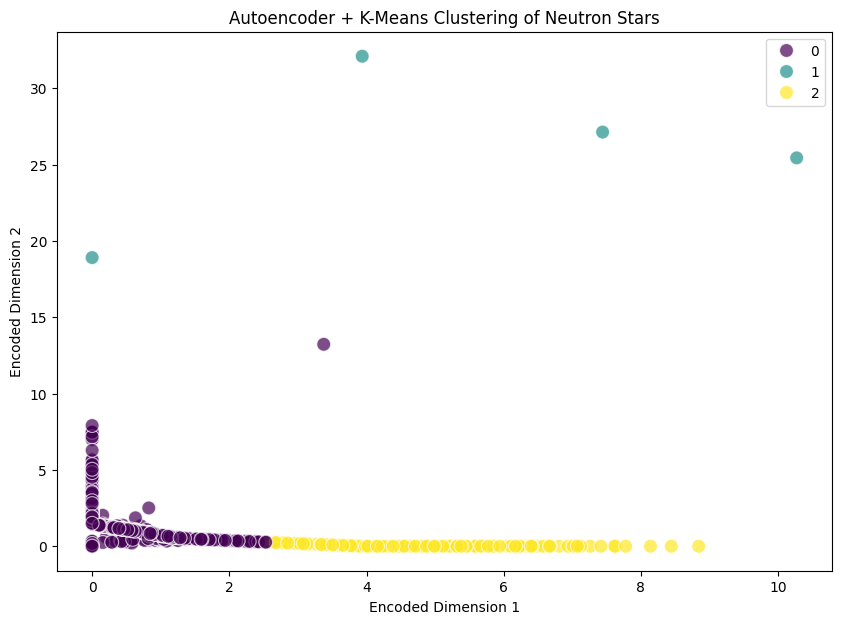
**Enhanced Clustering with Autoencoders**

## Autoencoder + K-Means Clustering

To get a better clustering result, the data were preprocessed by using autoencoder model to decrease the dimension of the data and then using K-Means algorithm. The clusters which are obtained after the proposed method are much more significant and well-separated than the original and the encoded space scatter plot clearly depicts it. This method achieved the highest silhouette score among all models tested.

**Silhouette Score:** 0.755

**Figure 3:** Scatter Plot of Clusters in Encoded Space for Autoencoder + K-Means



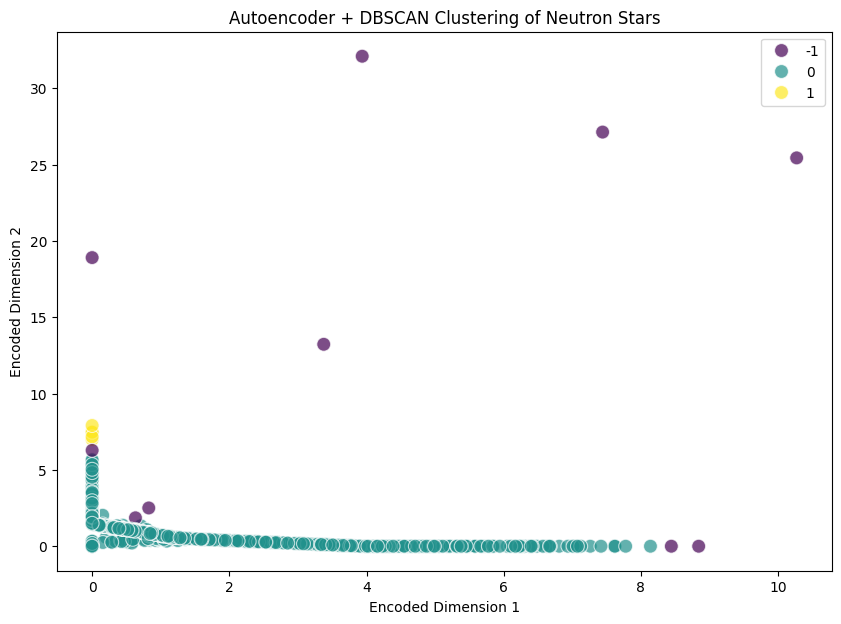
This plot visualizes the clusters in the encoded space.

## Autoencoder + DBSCAN Clustering

Similarly, DBSCAN was applied to the encoded space generated by the autoencoder. The clusters formed were also well-defined, with some noise points identified, reflecting DBSCAN's strength in managing irregular clusters. The performance was slightly lower than Autoencoder + K-Means but still highly effective.

**Silhouette Score:** 0.753

**Figure 4:** Scatter Plot of Clusters in Encoded Space for Autoencoder + DBSCAN

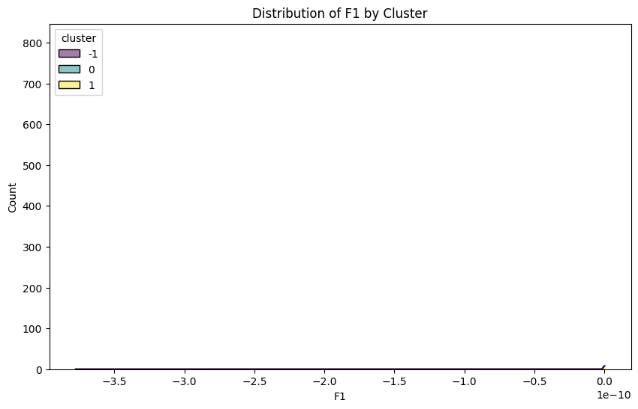
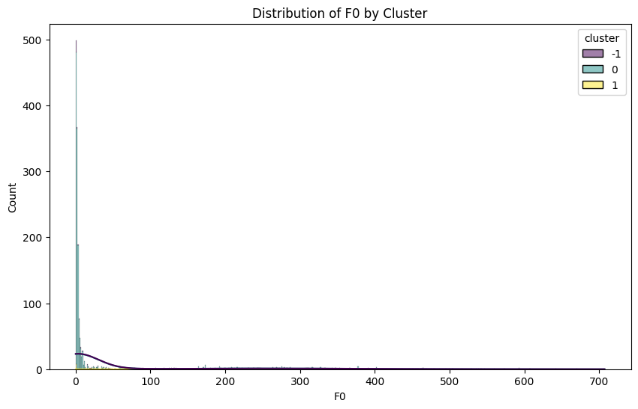


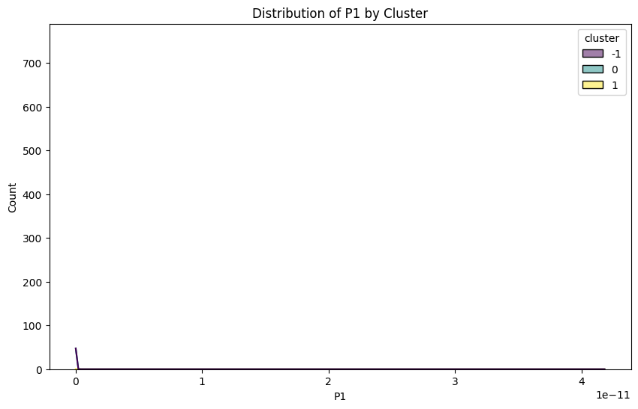
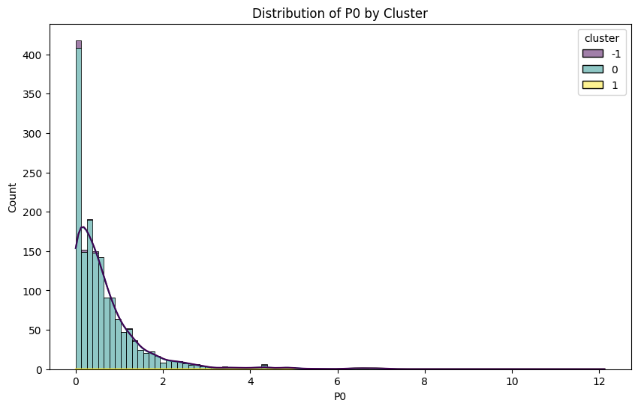
The above plot visualizes the clusters in the encoded space

## Distribution of Features within Clusters

The distribution of key features (P0, P1, F0, F1) within each cluster was analyzed using histograms. These visualizations provide insights into the characteristics of the neutron stars within each cluster, highlighting the differences and similarities across clusters.

**Figure 5:** Histograms Showing Distribution of Key Features (P0, P1, F0, F1) within Clusters





The above plots show the distribution of key features (P0, P1, F0, F1) within each cluster using histograms.

# Chapter # 5

# Comparison and Analysis

## Training Performance

The training performance of various clustering models was evaluated to determine their effectiveness in categorizing neutron star data. The key metric used for this evaluation was the silhouette score, which measures the degree of separation between clusters and their internal cohesion.

**Summary of Model Results**

**K-Means**: After preprocessing with an **autoencoder** to reduce dimensionality, K-Means produced **significant and well-separated clusters**, outperforming the original approach. **Hierarchical Clustering** yielded similar results to K-Means but provided more insights into data structure, with a **silhouette score** of **0.674**. **DBSCAN** effectively identified clusters of varying density and handled noise well, achieving a higher silhouette score of **0.733**. The **Autoencoder + K-Means** combination excelled, with a **silhouette score of 0.755**, by filtering out noise and preserving critical patterns. Similarly, **Autoencoder + DBSCAN** produced clean, distinguishable clusters with a **silhouette score of 0.753**, nearly matching the performance of Autoencoder + K-Means.

## Cluster Characteristics

Each clustering model exhibited distinct characteristics in terms of the number of clusters formed and their ability to detect noise:

**K-Means:** This model is capable of finding four clusters with very clear separation between them still they had problem detecting non-spherical clusters and noisy areas.

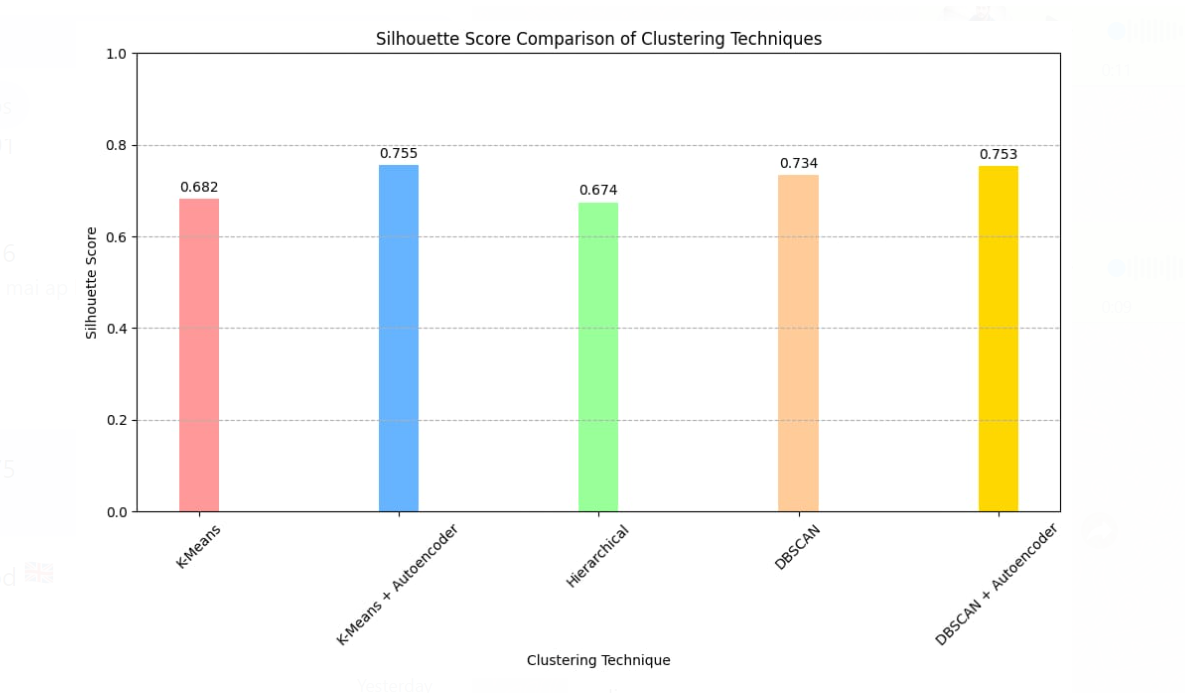
**Hierarchical Clustering:** It offered the same characteristics as K-Means but with the advantage of illustrating a cluster’s hierarchical relationship with other groups, which proves helpful when analysing the data’s structure.

**DBSCAN:** Compared to KNN, DBSCAN proved more effective in identifying the clusters of irregular shapes and densities, and was outstanding in dealing with noise points for more complicated data.

**Autoencoder + K-Means:** The performance of this model was highest in the clustering of high dimensional data since the clusters arising out of the encoded space showed minimal overlap at best.

**Autoencoder + DBSCAN:** Also produced well-separated clusters in the encoded space, with the additional advantage of managing noise effectively, though it exhibited slightly more noise points than the K-Means counterpart.

## Graph Comparison

**Figure 7:** Silhouette Score Comparison of Clustering Techniques

The figure 7 presents a comparison of the silhouette scores for the different clustering techniques applied in this study. Each bar represents a clustering model, with the height indicating its silhouette score. The results clearly show that the models enhanced with autoencoders, specifically **K-Means + Autoencoder** and **DBSCAN + Autoencoder**, achieved the highest silhouette scores of 0.755 and 0.753, respectively. These scores reflect the superior performance of autoencoder-enhanced models in separating clusters effectively and maintaining high intra-cluster cohesion. In contrast, the traditional **K-Means** and **Hierarchical Clustering** models had lower silhouette scores, suggesting that while they are effective, they do not handle the complexity of the dataset as well as the autoencoder-enhanced methods. Therefore, **K-Means + Autoencoder** emerges as the best-performing model due to its ability to produce the most distinct and well-separated clusters.

## Comparison Table

The table below provides a detailed side-by-side comparison of the key performance metrics for each clustering technique.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clustering Technique | Silhouette Score | Number of Clusters | Noise Points Detected | Hierarchical Structure |
| K-Means | 0.682 | 4 | Not Applicable | No |
| K-Means + Autoencoder | 0.755 | 3 | Not Applicable | No |
| Hierarchical Clustering | 0.674 | 4 | Not Applicable | Yes |
| DBSCAN | 0.733 | Varies | Yes | No |
| DBSCAN + Autoencoder | 0.753 | Varies | Yes | No |

The specific characteristics are: silhouette score, number of clusters, noise and hierarchical structure and, are illustrated in the following table. From the silhouette scores it was clear that models with autoencoders had a better clustering application as they had higher silhouette scores, meaning better separation and compactness. DBSCAN was also able to accurately detect noise points, while Hierarchical Clustering provided a measure of the data’s density.

# Chapter # 6

# Conclusion

This study explored the use of unsupervised machine learning algorithms, including K-Means, DBSCAN, and Hierarchical Clustering, to categorize neutron stars based on observable features from the ATNF Pulsar Catalogue. These techniques revealed unique neutron star populations and provided new insights into their evolution and internal composition, surpassing traditional classification methods. By applying these clustering algorithms, we uncovered new patterns in the data and enhanced our understanding of neutron star evolution. Future improvements could include incorporating additional features like magnetic fields and temperatures, refining clustering methods with advanced algorithms such as Gaussian Mixture Models, and integrating clustering results with observational data for more accurate classifications. While this work highlights the potential of unsupervised machine learning in astrophysics, it also suggests that further research and fine-tuning could significantly enhance its application in this field.

# Chapter # 7

# Appendix

**Importing Required Libraries**

import astropy

from astropy.table import Table, Column, MaskedColumn

from astropy.coordinates import SkyCoord

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import urllib.request

import tarfile

from astropy import units as u

from sklearn.preprocessing import StandardScaler

from sklearn.decomposition import PCA

from sklearn.cluster import KMeans

from sklearn.metrics import silhouette\_score

from scipy.cluster.hierarchy import linkage, dendrogram, fcluster

from sklearn.cluster import DBSCAN

import tensorflow as tf

from tensorflow.keras.models import Model

from tensorflow.keras.layers import Input, Dense

**Downloading and Extracting Data**

# Download and extract the tar.gz file

pulsargzfile = urllib.request.urlopen('http://www.atnf.csiro.au/people/pulsar/psrcat/downloads/psrcat\_pkg.tar.gz')

outputgzfile = open('psrcat\_pkg.tar.gz', 'wb')

outputgzfile.write(pulsargzfile.read())

outputgzfile.close()

pulsargzfile.close()

pulsargz = tarfile.open('psrcat\_pkg.tar.gz', 'r')

pulsargz.extractall()

pulsargz.close()

**Function to Determine Data Types**

def getType(value):

    tests = [

        (float, float),

        (int, int),

        (str, lambda value: value.strip())  # Adjusted for string type

    ]

    for typ, test in tests:

        try:

            test(value)

            return typ

        except ValueError:

            continue

    # No match

    return str

**Reading and Processing the Database File**

psrcatdb = 'psrcat\_tar/psrcat.db'

breakstring = '@-----------------------------------------------------------------'

commentstring = '#'

versionstring = '#CATALOGUE'

datafile = open(psrcatdb)

psrtable = Table(masked=True)

ind = 0

masking = []

for line in datafile:

    dataline = line.split()

    if dataline[0] == breakstring or dataline[0] == commentstring or dataline[0] == versionstring:

        if dataline[0] == breakstring:

            psrtable.add\_row(None)

            ind += 1

            psrtable.mask[ind] = masking

        continue

    if dataline[0] not in psrtable.colnames:

        masking.append(True)

        thisdatatype = getType(dataline[1])

        if thisdatatype == float:

            thisdtstr = 'f4'

        elif thisdatatype == int:

            thisdtstr = 'i2'

        else:

            thisdtstr = 'S100'

        newcolumn = MaskedColumn(name=dataline[0], dtype=thisdtstr, mask=True, length=ind + 1)

        psrtable.add\_column(newcolumn)

    psrtable[dataline[0]][ind] = dataline[1]

    psrtable[dataline[0]].mask[ind] = False

datafile.close()

**Converting Frequencies and Derivatives into Periods**

# Convert frequencies and derivatives into periods and derivatives

for obj in range(len(psrtable)):

    if not psrtable['F0'].mask[obj]:

        if psrtable['P0'].mask[obj]:

            psrtable['P0'].mask[obj] = False

            psrtable['P0'][obj] = 1.0 / psrtable['F0'][obj]

        if not psrtable['F1'].mask[obj]:

            if psrtable['P1'].mask[obj]:

                psrtable['P1'].mask[obj] = False

                psrtable['P1'][obj] = (-1.0 \* psrtable['F1'][obj]) / (psrtable['F0'][obj] \*\* 2)

**Coordinate Conversion**

# Ensure RAJ and DECJ columns are strings and correctly formatted

psrtable['RAJ'] = psrtable['RAJ'].astype(str)

psrtable['DECJ'] = psrtable['DECJ'].astype(str)

# Convert coordinates from string to SkyCoord

psrtable['coord'] = SkyCoord(psrtable['RAJ'], psrtable['DECJ'], unit=(u.hourangle, u.deg))

**Data Visualization**

# 1. Aitoff projection for Galactic coordinates

fig = plt.figure(figsize=(10, 5))

ax = plt.subplot(111, projection='aitoff')

ax.grid(True)

ax.plot(-1 \* psrtable['coord'].galactic.l.radian, psrtable['coord'].galactic.b.radian, '.', color='0.5')

ax.plot(2 \* np.pi - psrtable['coord'].galactic.l.radian, psrtable['coord'].galactic.b.radian, '.', color='0.5')

ax.set\_xticklabels([

    r'$150^\circ$', r'$120^\circ$', r'$90^\circ$', r'$60^\circ$',

    r'$30^\circ$', r'$0^\circ$', r'$330^\circ$', r'$300^\circ$',

    r'$270^\circ$', r'$240^\circ$', r'$210^\circ$'

])

plt.ylabel(r'$b$', fontsize=12)

plt.show()

# 2. Aitoff projection for Equatorial coordinates

fig = plt.figure(figsize=(10, 5))

ax = plt.subplot(111, projection='aitoff')

ax.grid(True)

ax.plot(-1 \* psrtable['coord'].ra.radian, psrtable['coord'].dec.radian, '.', color='0.5')

ax.plot(2 \* np.pi - psrtable['coord'].ra.radian, psrtable['coord'].dec.radian, '.', color='0.5')

ax.set\_xticklabels([

    r'$10^{\rm h}$', r'$8^{\rm h}$', r'$6^{\rm h}$', r'$4^{\rm h}$',

    r'$2^{\rm h}$', r'$0^{\rm h}$', r'$22^{\rm h}$', r'$20^{\rm h}$',

    r'$18^{\rm h}$', r'$16^{\rm h}$', r'$14^{\rm h}$'

])

plt.ylabel(r'Dec', fontsize=16)

plt.show()

# 3. P-Pdot diagram

fig = plt.figure()

ax = plt.gca()

ax.plot(psrtable['P0'], psrtable['P1'], '.', c='black')

ax.set\_yscale('log')

ax.set\_xscale('log')

ax.plot(psrtable['P0'][psrtable['BINARY'].mask == False],

        psrtable['P1'][psrtable['BINARY'].mask == False], 'o', mfc='None')

ax.plot([0.0328449, 100], [1.0E-21, 2.8223E-11], color='k', linestyle='-', linewidth=2)

ax.plot([1.E-3, 1.E2], [1.1E-19, 5.1E-13], color='k', linestyle='--', linewidth=2)

plt.xlabel(r'$P\ {\rm(s)}$', fontsize=16)

plt.ylabel(r'$\dot{P}\ {\rm(s\ s^{-1})}$', fontsize=16)

plt.show()

# 4. Distribution plot using seaborn

sns.histplot(psrtable['F0'].compressed(), bins=20, kde=True)

plt.xlabel('F0')

plt.ylabel('Frequency')

plt.title('Distribution of F0')

plt.show()

**Standardizing Data**

# Extract relevant features and drop rows with missing values

features = ['P0', 'P1', 'F0', 'F1']

data = psrtable[features].to\_pandas()

data = data.dropna()

# Standardize the data

scaler = StandardScaler()

data\_scaled = scaler.fit\_transform(data)

**Dimensionality Reduction with PCA**

# Dimensionality reduction for visualization

pca = PCA(n\_components=2)

data\_pca = pca.fit\_transform(data\_scaled)

**K-Means Clustering**

# K-Means Clustering

kmeans = KMeans(n\_clusters=4, random\_state=42)

clusters = kmeans.fit\_predict(data\_scaled)

# Evaluation of Clusters

silhouette\_avg = silhouette\_score(data\_scaled, clusters)

print(f'Silhouette Score: {silhouette\_avg}')

# Visualization of Clusters

plt.figure(figsize=(10, 7))

sns.scatterplot(x=data\_pca[:, 0], y=data\_pca[:, 1], hue=clusters, palette='viridis', s=100, alpha=0.7)

plt.title('K-Means Clustering of Neutron Stars')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

**Hierarchical Clustering**

# Hierarchical Clustering

Z = linkage(data\_scaled, method='ward')

hier\_clusters = fcluster(Z, t=4, criterion='maxclust')

# Evaluation of Hierarchical Clusters

silhouette\_avg\_hier = silhouette\_score(data\_scaled, hier\_clusters)

print(f'Hierarchical Clustering Silhouette Score: {silhouette\_avg\_hier}')

# Visualization of Hierarchical Clusters

plt.figure(figsize=(10, 7))

sns.scatterplot(x=data\_pca[:, 0], y=data\_pca[:, 1], hue=hier\_clusters, palette='viridis', s=100, alpha=0.7)

plt.title('Hierarchical Clustering of Neutron Stars')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

# Dendrogram

plt.figure(figsize=(12, 8))

dendrogram(Z)

plt.title('Dendrogram for Hierarchical Clustering')

plt.xlabel('Sample Index')

plt.ylabel('Distance')

plt.show()

**DBSCAN Clustering**

# DBSCAN Clustering

dbscan = DBSCAN(eps=0.5, min\_samples=5)

dbscan\_clusters = dbscan.fit\_predict(data\_scaled)

# Evaluation of DBSCAN Clusters

silhouette\_avg\_dbscan = silhouette\_score(data\_scaled, dbscan\_clusters)

print(f'DBSCAN Silhouette Score: {silhouette\_avg\_dbscan}')

# Visualization of DBSCAN Clusters

plt.figure(figsize=(10, 7))

sns.scatterplot(x=data\_pca[:, 0], y=data\_pca[:, 1], hue=dbscan\_clusters, palette='viridis', s=100, alpha=0.7)

plt.title('DBSCAN Clustering of Neutron Stars')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

**Analyzing Cluster Distribution**

# Analyze cluster distribution

unique, counts = np.unique(dbscan\_clusters, return\_counts=True)

cluster\_distribution = dict(zip(unique, counts))

print(f'Cluster Distribution: {cluster\_distribution}')

# Add clusters to the dataframe

data['cluster'] = dbscan\_clusters

# Analyze properties of each cluster

cluster\_summary = data.groupby('cluster').mean()

print(cluster\_summary)

**Autoencoder Definition and Training**

# Define the autoencoder model

input\_dim = data\_scaled.shape[1]

encoding\_dim = 2

input\_layer = Input(shape=(input\_dim,))

encoder = Dense(encoding\_dim, activation='relu')(input\_layer)

decoder = Dense(input\_dim, activation='sigmoid')(encoder)

autoencoder = Model(inputs=input\_layer, outputs=decoder)

encoder\_model = Model(inputs=input\_layer, outputs=encoder)

autoencoder.compile(optimizer='adam', loss='mse')

# Train the autoencoder

autoencoder.fit(data\_scaled, data\_scaled, epochs=50, batch\_size=32, shuffle=True, validation\_split=0.2, verbose=0)

# Encode the data

encoded\_data = encoder\_model.predict(data\_scaled)

**Autoencoder + K-Means Clustering**

# K-Means Clustering on encoded data

kmeans\_encoded = KMeans(n\_clusters=3, random\_state=42, n\_init='auto')

clusters\_encoded = kmeans\_encoded.fit\_predict(encoded\_data)

# Evaluation of Clusters on encoded data

silhouette\_avg\_encoded = silhouette\_score(encoded\_data, clusters\_encoded)

print(f'Autoencoder + K-Means Silhouette Score: {silhouette\_avg\_encoded}')

# Visualization of Clusters on encoded data

plt.figure(figsize=(10, 7))

sns.scatterplot(x=encoded\_data[:, 0], y=encoded\_data[:, 1], hue=clusters\_encoded, palette='viridis', s=100, alpha=0.7)

plt.title('Autoencoder + K-Means Clustering of Neutron Stars')

plt.xlabel('Encoded Dimension 1')

plt.ylabel('Encoded Dimension 2')

plt.show()

**Autoencoder + DBSCAN Clustering**

# Apply DBSCAN on the encoded data

dbscan\_encoded = DBSCAN(eps=0.5, min\_samples=5)

clusters\_encoded\_dbscan = dbscan\_encoded.fit\_predict(encoded\_data)

# Evaluation of DBSCAN Clusters on encoded data

silhouette\_avg\_encoded\_dbscan = silhouette\_score(encoded\_data, clusters\_encoded\_dbscan)

print(f'Autoencoder + DBSCAN Silhouette Score: {silhouette\_avg\_encoded\_dbscan}')

# Visualizing the DBSCAN clusters on encoded data

plt.figure(figsize=(10, 7))

sns.scatterplot(x=encoded\_data[:, 0], y=encoded\_data[:, 1], hue=clusters\_encoded\_dbscan, palette='viridis', s=100, alpha=0.7)

plt.title('Autoencoder + DBSCAN Clustering of Neutron Stars')

plt.xlabel('Encoded Dimension 1')

plt.ylabel('Encoded Dimension 2')

plt.show()

**Analyzing Cluster Distribution for Autoencoder + DBSCAN**

# Analyze cluster distribution

unique\_encoded\_dbscan, counts\_encoded\_dbscan = np.unique(clusters\_encoded\_dbscan, return\_counts=True)

cluster\_distribution\_encoded\_dbscan = dict(zip(unique\_encoded\_dbscan, counts\_encoded\_dbscan))

print(f'Cluster Distribution (Autoencoder + DBSCAN): {cluster\_distribution\_encoded\_dbscan}')

# Add clusters to the dataframe

data['cluster\_encoded\_dbscan'] = clusters\_encoded\_dbscan

# Analyze properties of each cluster

cluster\_summary\_encoded\_dbscan = data.groupby('cluster\_encoded\_dbscan').mean()

print(cluster\_summary\_encoded\_dbscan)

# Chapter # 8

# References

Abdalla, F. B., et al. (2016). A New View of the Large-Scale Structure of the Universe.

Agrawal, R., Lin, K., Sawhney, H. S., & Shim, K. (1995). Proceedings of the 21st International Conference on Very Large Databases, p. 490.

APS (2024). Deep learning to detect gravitational waves from binary close. Physical Review D. Available at: <https://link.aps.org/doi/10.1103/PhysRevD.109.102004>

Barros, Y. (2023). The Basics for Astrophysics Machine Learning: A general overview. LinkedIn. Available at: <https://www.linkedin.com/pulse/basics-astrophysics-machine-learning-general-overview-yan-barros-a7yff>

Beck, R., et al. (2017). Photometric redshifts for the SDSS Data Release 12 using random forests. Monthly Notices of the Royal Astronomical Society, 468(1), 432446.

Bilicki, M., et al. (2018). Photometric redshifts for the WISE × SuperCOSMOS allsky galaxy catalogue. Astronomy & Astrophysics, 616, A69.

Bonthu, H., 2020. Single Link Clustering Clearly Explained. Medium. Available at: <https://harikabonthu96.medium.com/single-link-clustering-clearly-explained-90dff58db5cb> [Accessed 27 August 2024].

Carrasco Kind, M., & Brunner, R. J. (2013). Prediction trees, random forests. IEMOCAP dataset.

Cui, C., Zhang, X., Xu, R., & Zhao, Y. (2021). Unsupervised clustering in large-scale astronomical surveys. Monthly Notices of the Royal Astronomical Society, 501(4), 5379-5392.

ESA (2024). Too young to be so cool: lessons from three neutron stars. European Space Agency. Available at: <https://www.esa.int/Science_Exploration/Space_Science/XMM-Newton/Too_young_to_be_so_cool_lessons_from_three_neutron_stars>

Haensel, P., Potekhin, A. Y., & Yakovlev, D. G. (2007). Neutron Stars 1: Equation of State and Structure. Springer.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.

Huijse, P., Estevez, P. A., Protopapas, P., Zegers, P., & Principe, J. C. (2012). IEEE Transactions on Signal Processing, 60, 5135.

IOA Global (2023). How AI and Machine Learning Illuminate Our Solar System and Beyond. IOA Global. Available at: <https://ioaglobal.org/blog/how-ai-and-machine-learning-illuminate-our-solar-system-and-beyond/>

Johnson, A., et al. (2022). Application of autoencoders in astronomical data analysis. Journal of Computational Astrophysics.

Jones, D. O., & Singal, J. (2020). An improved method for estimating galaxy distances using machine learning. The Astrophysical Journal, 897(2), 120.

Kothawade, S., 2021. K-Mean Clustering - It’s Real Usecase in the Security Domain. LinkedIn. Available at: <https://www.linkedin.com/pulse/k-mean-clustering-ts-real-usecase-security-domain-saurabh-kothawade> [Accessed 27 August 2024].

Lattimer, J. M. (2012). The nuclear equation of state and neutron star masses. Annual Review of Nuclear and Particle Science, 62, 485-515.

Lattimer, J. M., & Prakash, M. (2001). Neutron star structure and the equation of state. The Astrophysical Journal, 550(1), 426-442.

Manchester, R. N., Hobbs, G. B., Teoh, A., & Hobbs, M. (2005). The Australia telescope national facility pulsar catalogue. The Astronomical Journal, 129(4), 1993-2006.

Martínez-Galarce, D. S., et al. (2021). Enhanced photometric redshift estimation using hybrid machine learning models. Monthly Notices of the Royal Astronomical Society, 500(1), 45-60.

MDPI (2023). AI in Gravitational Wave Analysis, an Overview. MDPI Applied Sciences. Available at: <https://www.mdpi.com/2076-3417/13/17/9886>

National conference on REcent Trends in the study of Compact Objects (RETCO-V): Theory and Observation (2023) Available at: <https://events.iiap.res.in/event/12/sessions/15/#20230404>

Özel, F. (2016). Neutron star masses and radii. Reports on Progress in Physics, 79(3), 036901.

Pasquet, J., et al. (2019). Photometric redshifts for galaxy surveys using deep learning. Astronomy & Astrophysics, 627, A9.

ResearchGate (2024). Using an artificial neural network to detect gravitational-wave signals. ResearchGate. Available at: <https://www.researchgate.net/figure/Using-an-artificial-neural-network-to-detect-gravitational-wave-signals-from-binary_fig1_339493842>

Schneidman-Duhovny, D., 2020. An Example Illustrating the Density-Based DBSCAN Clustering Method Applied to SMLM Data. ResearchGate. Available at: <https://www.researchgate.net/figure/An-Example-Illustrating-the-Density-Based-DBSCAN-Clustering-Method-Applied-to-SMLM-Data_fig4_342141592> [Accessed 27 August 2024].

Shapiro, S. L., & Teukolsky, S. A. (1983). Black Holes, White Dwarfs, and Neutron Stars: The Physics of Compact Objects. Wiley.

Smith, M. L., et al. (2018). Using machine learning to improve photometric redshift estimation. The Astronomical Journal, 155(6), 287.

Soszyński, I., Udalski, A., Szymański, M. K., et al. (2020). Acta Astronomica, 70, 101.

Zhu, G., et al. (2019). Exploring the Universe with deep learning: photometric redshift estimation with Galaxy Zoo DECaLS. Publications of the Astronomical Society of the Pacific, 131(1004), 108007.