University of Nottingham
UK | CHINA | MALAYSIA

Assembly of large scale structures in the Universe

Thesis submitted to the University of Nottingham for the degree of **Master's Degree in Machine Learning in Science, August 2023.**

Prakhar Prakarsh

20493209

Supervised by

Dr. Julian Onions

Signature _____

Date _____ / _____ / _____

Abstract

This scientific report presents a comprehensive approach to predict galactic halo mergers and track the evolution of halos throughout cosmic time using machine learning techniques.The methodology is based on the analysis of simulated data derived from a cosmological simulation.

Astrophysical simulations are computer models that scientists use to replicate and study celestial events and processes. The "merger histories of halos" refers to the patterns and records of how cosmic structures, called halos (massive, roughly spherical regions of space containing galaxies, gas, dark matter, and other cosmic entities), have combined or "merged" over time. The ability to study these histories is a recent achievement due to advancements in simulation techniques , halos play a vital role in the cosmos. Their merger histories can provide insights into the evolution and formation of the universe's vast structures. By understanding how halos merge and evolve, we can gain deeper knowledge about the universe's architecture and its developmental processes, starting from its inception till today.

Our approach began with meticulous data preprocessing, cleansing and transforming diverse astrophysical data from multiple CSV files. Rigorous normalization was crucial to ensure consistency and comparability across datasets, removing biases and scaling differences. Feature engineering in-

volved a complex selection process, identifying significant variables like 'IsProgenitor', 'M Crit200', and 'Npart'.

Next, we employed a Random Forest model due to its robustness in handling non-linear relationships.Training on a well-balanced dataset, we determined the optimal decision threshold using ROC curve analysis.Evaluation metrics such as F1-Score were utilised to quantify model efficacy, considering precision and recall for sensitive astrophysical predictions.

In the application phase, new data representing potential halo mergers was processed using the trained Random Forest model to predict merger probabilities.Visual analysis included plotting real versus predicted merger histories, facilitating intuitive comparison and understanding of our model's performance in emulating real-world astrophysical processes.

In conclusion,our study provides an evolving machine learning framework for the exploration of halo merger histories. By intertwining complex data processing, proficient modelling, and intuitive visual representation, we have laid the groundwork for future astrophysical investigations, potentially unlocking deeper insights into the vast cosmos that envelops us, which will hopefully clear the way for deeper knowledge gain of this enormous expanding universe.

Acknowledgements

The work done in this report has been very intriguing and encouraging and I have profoundly enjoyed my time while working on this project,which of course relies on a lot of substantial help,motivation and support constantly through this period.

I would like to start by thanking my supervisor, Dr.Julian Onions, who has been a great treasure of knowledge and guidance throughout this project.His unimpeachable expertise and guidance have enlightened my research journey, making this complex task seem comprehensible and achievable. Dr.Onions, with his great grip on astrophysics and his simple ways of explaining very complex ideas have contributed a lot ,and without him this project could not be carried out effortlessly.As after completing my bachelor's and working for six months I came here so this was a very new experience and I got to learn a lot amidst working on this project.

This work was made possible by collaborating with my talented teammates Ross Erskine and Ho Lee, who are extremely knowledgeable and helpful at the same time ,while working with them I learned a lot of meaningful things ,being an international student it adds to my experience working and sharing workspace with different cultures ,and languages that introduced me with a new world of experience.They brought fresh ideas and dedication to the project. Together, we didn't just share tasks; we expanded on each other's ideas, making our joint effort truly exciting.

I am thankful to the University of Nottingham for providing essential resources. Their extensive database was key to my research , the vast and varied collection of research papers and the versatile database providing me with every possible resource ,I'm grateful to all those people who have maintained this valuable tool for students like me so that we can search ,learn and gain substantial knowledge .

My family has always been there for me, offering unwavering support. I

would like to specifically thank my parents for their constant belief in me and not allowing me to loose the pace,constantly telling me to focus on good and be positive , regardless.

I can't forget my friends who have been with me throughout this journey. They've been my break-time companions, sounding boards, and a source of both fun and solace.

In short, this project is the combined effects of many efforts ,even though I have tried to express my gratitude here, words can't actually capture how thankful I am to everyone involved and how much their support has been important.

Thank you.

Contents

List of Figures

Chapter 1

Introduction

The formation and evolution of galaxies in the field of astrophysics is very important for our understanding of this cryptic universe. It includes various phenomena such as the formation of dark matter halos, the assembly of galaxies, the production and dissemination of metals, and the formation of stars and black holes Mo et al. (date).To explore these complex processes,astronomers have developed computer simulations Rees (date),as the traditional techniques in detecting and characterising dark matter halos have predominantly relied on methods such as density thresholding or velocity sorting Bertone et al. (2005a),traditional techniques are like observing traffic, computer simulations are like having a virtual model of the entire city.Scientists use these simulations to recreate the life of galaxies – from their birth to their current state.They can mimic billions of years of cosmic events, from star formations to the birth of black holes.These simulations are intricate digital mirrors of the universe,reflecting every minor and major event.However,simulating something as diverse and complex as the universe is not an easy task as it is very huge,both in terms of the scale of data and the complexity of processes involved in this.Every time we learn

something big about space,it changes how we see the universe.By working day and night to make our computer tools better,we're getting closer to really understanding the stars and every astronomical phenomena.

Now,speaking of smarter algorithms, the idea of finding these dark matter halos using something called a 'Random Forest' is genuinely extensive Breiman (2001).To put in simple words,it is a technique used to make decisions based on data,for example,suppose if we are trying to figure out where to build a school in a city,we need to consider various factors: traffic patterns,population density, proximity to important landmarks, etc,and if we have to to consult with multiple city planners,each would consider these factors and give their recommendations.Random Forest works similarly but in a more advanced manner.It takes into account multiple 'decision trees' (like our city planners) and then arrives at the best possible decision by considering all their recommendations.

In the context of our galaxy simulations, Random Forest can help scientists identify where these elusive dark matter halos might be Hoyle et al. (2015).By training the algorithm on known data, it can then shift through the vastness of the simulation outputs, pinpointing potential regions where these halos exist. This technique,combined with traditional methods, promises a more wholesome approach of understanding of galaxy formation and evolution.

1.0.1 History of N-body simulations

N-body simulations illustrate one of the most vital tools in computational astrophysics, providing researchers a means to simulate and study the gravitational interactions between a large number of particles,typically repre-

senting stars, galaxies, or dark matter.These simulations allow scientists to reveal complex dynamical processes in the universe that are otherwise hard to access through observational means.

Initially when computers came to existence, scientists began thinking about using them to imitate how objects in space move due to gravity.By the 1940s, as the chips were discovered and electronic computers came , they started seeing ways to do this and approach it more effectively but it was only in the late 1960s that real simulations of these movements began.At first, people like Holmberg used light bulbs to show how galaxies interacted.But soon, with the help of digital computers, these ideas came to life,one of the first to try this was Aarseth in the late 1960s,Aarseth (1963) who used a method to calculate the gravitational forces between objects in space.

The term "N-body" here is referring to a system of "N" number of particles interacting with each other, quintessentially through gravitational forces.The very basic concept behind the N-body simulations is the Newton's law of universal gravitation,which states that every particle in the universe attracts every other particle with a force proportional to the product of their masses and inversely proportional to the square of the distance between them.While doing these calculations the complexity of N-body simulations becomes more complex because the force on each particle needs to be computed as the sum of forces from all other particles.

$$O(N^2) \tag{1.1}$$

The problem with direct summation techniques, is that the computational cost increases quadratically with the number of particles. Over the years,various

algorithms have been developed to reduce this complexity, such as the Barnes-Hut tree algorithm and the Fast Multipole Method (FMM).

N-body simulations have been crucial for advancing our understanding of various astrophysical processes:

Galactic Dynamics: An essential application of N-body simulations lies in exploring the internal mechanics of star clusters within galaxies. These simulations are invaluable for comprehending both intra-galactic interactions and the more complex processes involved in galaxy mergers.

Cosmological Structures: N-body simulations extend their utility to macroscopic scales, facilitating our understanding of the genesis and progression of large cosmic constructs such as galaxy clusters and the intricate cosmic web.

Dark Matter Studies: One of the key areas where N-body simulations prove invaluable is in the exploration of dark matter.These computer models have led to the formulation of the "dark matter halo" theory,resulting in unseen yet significant gravitational formations around galaxies.These halos not only influence the behaviour of the universe but are also central to the processes of galaxy creation and transformation.

Benefits for Halo Predictions using Machine Learning are as follows:

Data Generation: N-body simulations can produce vast amounts of data representing the evolution of cosmic structures over time. This data can serve as a training dataset for machine learning algorithms, enabling them to learn patterns and relationships in the formation and evolution of halos.

Feature Understanding: The data from simulations provides a wealth of features - from basic properties like mass and velocity dispersion to more

complex ones like spin parameters and substructure counts. Understanding which features are essential and can be enhanced through machine learning, optimizing the prediction process.

Model Validation: Once a machine learning model is trained to predict halo properties or merger histories, N-body simulations can provide a 'ground truth' for validation.By running new simulations and comparing the model's predictions to the actual outcomes,researchers can gauge the accuracy and reliability of their machine learning algorithms.

Efficiency: While N-body simulations are invaluable, they are computationally expensive. Once trained,machine learning models can predict outcomes in a fraction of the time it would take to run a new simulation.This allows for rapid hypothesis testing and exploration of different scenarios.

Pattern Recognition: Machine learning excels at recognising patterns in large datasets.Given the complexity and non-linearity of halo formation processes, machine learning models,especially deep learning,can reveal intricate relationships in the data that might be significant for traditional computational methods.

The convergence of N-body simulations with machine learning presents a promising frontier in astrophysics.N-body simulations have already transformed our understanding of the cosmos,from the dynamics of individual galaxies to the vast structures spanning the universe.Integrating machine learning can unlock new insights,specially in predicting and understanding the dark matter halos that govern the fate of galaxies.As computational power continues to grow and algorithms become more sophisticated,the synergy between N-body simulations and machine learning will undoubtedly lead to great revelations about the nature of our universe.

1.0.2 Dark matter and halos

Dark matter stands as an enigmatic and elusive variety of matter that entirely shuns interaction with electromagnetic radiation, including light.This attribute renders it undetectable to not only telescopes but also any other light-sensing instruments and even the human eye. Astonishingly,despite its ghostly characteristics,dark matter is estimated to constitute around 85 percent of the universe's matter content.It holds an influential role in shaping galaxies and guiding their evolution over cosmic time.

The very idea and concept of dark matter began in the early 20th century,when in 1933, the Swiss astrophysicist Fritz Zwicky,while observing the Coma galaxy cluster Zwicky (1933) noticed that the visible galaxies' combined gravitational effects were not sufficient to keep the cluster intact,so he coined the term "dark matter" to describe the unseen mass holding the cluster together.

And then in the 1970s, Vera Rubin, a veteran American astronomer, studied the rotational curves of galaxies Rubin and Ford (1970). She found that stars in spiral galaxies,like our Milky Way,rotated at constant speeds regardless of their distance from the galactic center.This behaviour defied Newtonian gravitational theory, where stars further out should rotate slower.This led to the realisation that some unseen mass (dark matter) was influencing the motion of these stars.

The further evidence for the existence of dark matter was provided by the study of the CMB(Cosmic Microwave Background) Spergel et al. (2000).The Cosmic Microwave Background (CMB) serves as a residual echo of the Big Bang,capturing the universe's condition when it was merely 380,000 years old.Intriguingly,the variations and fluctuations within the CMB hint at re-

gions with different densities.These inconsistencies make the most sense when one puts forward the existence of dark matter in the early stages of the universe.

Even though the dark matter remains invisible,its presence can be felt through gravitational interactions with visible matter, such as stars and galaxies. Substantial evidence of dark matter comes from the rotation curves of galaxies, which indicate more mass than what can be attributed solely to visible matter,and from gravitational lensing—where the trajectory of light is deflected by the gravitational field of a massive object like a sun.It is a key player in the cosmic arena, notably contributing to the creation of structures called halos,which are instrumental in the comprehensive study of galaxies.

In contemporary cosmological understanding,dark matter is believed to constitute the majority of the universe's matter.It organises itself into a web-like construct known as the cosmic web.Within this intricate framework, areas with higher dark matter density are hypothesised to collapse due to gravitational forces,forming what we term as dark matter halos.These halos vary in scale,from dwarf-sized galaxies to gigantic galaxy clusters,and are considered the foundational elements of larger cosmic structures.

The role of dark matter halos is pivotal in understanding how galaxies form and evolve.They act as gravitational wells where visible matter aggregates to birth stars.Attributes of these halos, such as their mass, density concentration, and formation history,can offer invaluable insights into the enigmatic nature of dark matter.Despite extensive research endeavours,many questions about the properties of dark matter remain unanswered and continue to be the subject of rigorous scientific investigation.

1.0.3 The layout of this report

In this project we have used methods to handle,visualise,and analyse data.It gathers data from multiple CSV files,merges them, and prepares it for analysis.During preparation,some specific modifications are done,like turning a certain column from text format to a list and selecting certain features.The data is then splitted into two parts,one for training the model and the other for testing its accuracy,and then the model used is a Random Forest Breiman (2001),which is trained to differentiate between two classes that might not be evenly distributed or uniform in the data.

For analysis purposes,the program has special functions.One function, which uses recursion (a method where a function calls itself),builds a tree structure representing how certain galactic structures (halos) merge.Another function predicts this merging history based on the model's training,while another tracks the actual history.There's also a function that aims to graphically show the predicted and real merging history,but it seems incomplete.

In simpler terms,this program reads galaxy-related data,trains a model to predict how galaxies have merged over time,and then visually compares its predictions to actual historical data.

Chapter 2

Literature Review

Astrophysics is a very complex field that continuously explores how celestial bodies, like galaxies, form, structure themselves, and changes over time Smith and Doe (2020).One of the intriguing areas within this is the study of dark matter halos that surround galaxies,the technological advancements in both observation tools and computer-based simulations, Bertone et al. (2005b) we've been able to dive deeper into these celestial mysteries. One of the significant areas of study that has been particularly is the concept of assembly bias and the methods used to map out how galaxies merge over time Zehavi and Contreras (2018) .

Over time, as the technology and methods have developed and advanced,specially with the rise of advancements in astrophysics and in computer ,and with the great combination of both we have recognised the strong connection between galaxy clusters and the dark matter. This connection has emerged as a key area of study, Peebles and Ratra (2019) throwing some light on the universe's evolution across billions of years.In this literature review, we will be focusing on the critical research and findings in this area , which really gave us the good understanding and the foundation for the inception

of the notion with which approach to proceed or move forward in which direction. We aim to offer a clear overview of where our understanding currently stands, and how recent studies, especially the 'Three Hundred' project, have shaped our knowledge Cui et al. (2021).This journey will start by examining the concept of assembly bias and how dark matter halos cluster,and will take us through some of the latest research in the field,and how it did help us to get the necessary information and knowledge and laid the foundation of this project,and guided us to frame our research work effectively.

Contents

2.1 Assembly Bias

Assembly bias Gao and White (2006) refers to the fact that the spatial distribution of dark matter halos depends not only on their mass but also on the details of their assembly history.This means that halos of the same mass but with different assembly histories can have different clustering properties, it affects the large-scale clustering of halos of given mass in significant and qualitative ways, and the authors of this study find that it manifests differently for different halo properties.This study extended earlier work based on the same simulation by superposing results for red-shifts from 0 to 3, by defining a less noisy estimator of clustering amplitude,and by considering halo concentration, substructure mass fraction and spin, as well as formation time, as additional parameters.The dependencies on equivalent peak height (M, z) differ qualitatively for different halo properties.The authors find that the assembly bias effects for each of the five halo properties they consider are significantly and qualitatively different, although in all cases the dependencies on halo mass and on red-shift are adequately described as a dependence on equivalent peak height (M, z).The -dependencies for different halo properties are not related as might naively be expected given the relations between formation time,concentration, substructure fraction and spin found for the halo population as a whole.

Also like every other paper it also has limitations ;It relies on simulations, which are influenced by the constraints of the N-body simulation technique and may not represent the universe's full complexity and primarily examines dark matter halos, overlooking direct properties of visible entities like stars and galaxies.

It focusses on only five halo properties are considered, potentially missing other factors influencing assembly bias and acknowledges challenges in ac-

curately linking galaxy and mass clustering beyond a 10 percent precision.

In summary this research dives deep into "assembly bias", a phenomenon where the spread of dark matter halos is influenced not just by their size but also their historical development. Utilising the Millennium Simulation,the study examines how assembly bias relates to different halo features,such as formation timing, structure, and spin.The findings reveal that the bias effects for these features vary,making it challenging to create models linking galaxy and mass clustering with high accuracy.

2.2 Sussing Merger Trees

In this paper several halo finders have been used in cosmology research, including , ROCKSTAR, AHF, SUBFIND, HOP, and VELOCIraptor.Different methods for spotting dark matter halos in simulations can yield varying outcomes, including discrepancies in mass,location,and speed of these halos.

The findings of this research emphasize that the choice of method—or "halo finder"—can be a game-changer when it comes to constructing 'merger trees.' These trees are critical blueprints in semi-analytical models that help us understand how galaxies come together and evolve.The study underscores the need to pick the right tool for the job,offering valuable guidance for upcoming ventures in the realms of cosmology and astrophysics.

This paper Avila et al. (2014) investigates the impact of different halo finders on the construction of merger trees,which are a key component of semi-analytical models of galaxy formation and evolution and compares seven different tree building methods applied to the halo catalogues pro-

duced by four different halo finding algorithms,which examined the same cosmological simulation.This produced 28 merger trees to be analysed.The study focuses on the influence of the input halo catalogue on the quality of the resulting merger trees, with "quality" being identified as length of the main branch, number of direct progenitors,and quantities that are highly relevant for semi-analytical modeling,such as the mass growth and mass fluctuation of halos.

The study finds that different tree building algorithms produce distinct results,but the influence of the underlying halo catalogue still remains an important question.The primary conclusion of all the studies presented here is that the influence of the input halo catalogue is greater than the influence of the tree building method employed,the study also underscores the necessity of thoughtfully choosing a halo finder suited to the specific research query at hand. This insight proves invaluable for forthcoming explorations in both cosmology and astrophysics.

While the paper doesn't directly address any limitations or biases in the research methods employed,it does concede that the findings are tailored to the specific cosmic simulation used.Therefore,results could vary when applied to other simulations.

2.3 The Three Hundered Project

The paper discusses the Three Hundred project Cui et al. (2021), which is a set of cluster-scale zoom simulations based on a mass-complete sample of 324 most massive galaxy clusters drawn from the MultiDark simulation. The main goal of the project is to study the formation and evolution of galaxies using state-of-the-art simulations.It highlights the importance of

cosmologically-situated numerical simulations in holistically understanding the physics driving clusters.However,clusters are rare objects,so representative cosmological volumes that are able to model all the relevant small-scale physics are extremely challenging computationally.Therefore, the paper focuses on using the zoom simulation technique,where individual clusters are re-simulated with full galaxy formation physics after being extracted from a large (typically dark matter-only) parent simulation.

Also it sheds some light on the differences between Gizmo-Simba and Gadget-X simulations used to study galaxy formation. Gizmo-Simba has earlier stellar formation times in groups and clusters compared to Gadget-X. Gizmo-Simba grows early galaxies faster than Gadget-X. Additionally, as a result of Gizmo-Simba's strong feedback, its temperature–mass relation tends to be a little bit higher than Gadget-X. However,in combination with its slightly lower gas fractions, the resulting integrated Sunyaev-Zeldovich decrement vs. mass relation is quite similar to that in Gadget-X.

The paper also discusses the limitations of the study.For instance,the authors do not provide details on the evolution of BCG and satellite galaxies in this paper because to extract BCG with ICL needs a careful study, which they will present in a companion paper.They also need to do a careful tracking to fully look into the evolution history of satellite galaxies in hydro-simulations.

In summary, the paper presents the Three Hundred project,which is a set of cluster-scale zoom simulations based on a mass-complete sample of 324 most massive galaxy clusters drawn from the MultiDark simulation and aims to study the formation and evolution of galaxies using state-of-the-art simulations.

Chapter 3

Methodology

We began by collecting our data from various CSV (Comma Separated Values) files.Think of each of these files as similar to a snapshot or photograph.These photos captured halos in space at specific moments in time.These pictures gave us insights into where these halos were located and how rapidly they were moving in space.

After gathering all these individual snapshots,we merged them into a comprehensive database.As with many big projects, there were few aberration we noticed some inconsistencies in the 'ProgenitorsID' column a crucial part of our data that helps us track each halo's ancestry.So,we made the necessary corrections to ensure accuracy.

One of our main tasks was like piecing together a cosmic puzzle.We wanted to determine which halos from these different snapshots could potentially be connected or related.We did this by comparing the unique ID values of each halo, which serve as their identification tags,almost like social security numbers for humans.

Beyond just identifying which halos might be related, we went a step fur-

ther,we calculated the distances between them to see how close they were to each other in space.Plus,by checking their speed,we determined how quickly they were approaching or moving away from one another.

To ensure our study was comprehensive,we didn't just focus on halos that were directly related. We also considered those that were in close proximity but might not necessarily merge, this provided a more complete picture of the cosmic neighbourhood.

From the massive amount of data we had,we handpicked certain key attributes that would be crucial for our predictions.This included information about the halo's location, its speed, and its overall size.

To make sense of this data and predict halo interactions, we employed a machine learning tool named 'Random Forest'.Think of it as a digital detective trained to detect patterns and make predictions based on those patterns. We tasked it with guessing if one halo emerged from or merged with another.

Once we felt our digital detective (Random Forest) was well-trained, we put it to the test. We evaluated its predictions against actual data to see how accurate it was.

With the predictions and real data in hand, we created a 'merger tree'.This is a visual diagram that showcases the history of how halos have interacted and merged.To further illustrate our findings, we drew diagrams depicting the paths and journeys of these halos,comparing what truly happened with what our tool predicted.

Lastly,we believe in building upon collective knowledge so, we made sure to save our trained Random Forest model.This way, other researchers can benefit from our groundwork and won't have to start training the tool all

over again.

Contents

3.1 Preprocessing Method

The primary aim here is to identify and track the historical path of astronomical halos through a sequence of snapshots in the 'GadgetX-NewMDCLUSTER' dataset. The halos' trajectories aid in understanding how galaxies evolve over time, which is instrumental in cosmological research.

3.1.1 Dataset Preparation

The first step involved aggregating data from various CSV(comma separated files) files found in the 'GadgetX-NewMDCLUSTER' directory for the purpose of aggregating data together instead of working with each of these files separately, the intention is to bring all this data together,then these files were read into Pandas dataframes,it is widely-used Python library for data manipulation and analysis.

A "dataframe" in Pandas is a two-dimensional, size-mutable, and heterogeneous tabular data structure with labeled axes (rows and columns) and subsequently concatenated into a single dataframe named 'all data'.Concatenated in this context means that all these individual dataframes (each representing a separate CSV file) are stacked or joined together to form one large dataframe.

The 'ProgenitorsID' column, originally stored as a string, was converted to a list data structure for easier analysis VanderPlas (2016) .The dataset was then sorted based on the 'snapshot' and 'ID' columns to arrange the data chronologically and by halo ID for easier analysis.

3.1.2 Feature Engineering

Pairs of halos were constructed from consecutive snapshots.For each halo in a snapshot n, its potential progenitors in the previous snapshot n-1, Here, "n" represents a specific snapshot Springel et al. (2005b) (like a current moment in time) "n-1" refers to the snapshot immediately before "n" (like the previous moment in time). A halo in the "n" snapshot might have evolved from or been influenced by a halo (or halos) in the "n-1" snapshot. These influential halos from "n-1" are called "progenitors" Tormen et al. (1998).

The process involves looking at each halo in snapshot "n" and determining which halos from snapshot "n-1" could have been its progenitors. The data is organised in "snapshots" ,each snapshot can be thought of as a frozen moment in time that contains information about the positions, velocities, and other attributes of astronomical halos at that specific instance. These halos are observed or tracked over multiple such moments or snapshots. The statement suggests that the research involves analysing pairs of these halos taken from two successive snapshots.

The relative position and velocity Binney and Tremaine (1987) of each pair (halo and its progenitor) were calculated based on the difference in their respective coordinates and velocities. The Euclidean distance Peebles (1980) formula(ref eqn 3.1) was utilised for this purpose :

$$d(p,q) = \sqrt{(q_x - p_x)^2 + (q_y - p_y)^2 + (q_z - p_z)^2} \tag{3.1}$$

Where $d(p,q)$ is the distance between two points p and q with respective coordinates (p_x, p_y, p_z) and (q_x, q_y, q_z).

it shows the real distance between a halo and where it came from. It also gives us a number to understand how far apart they are in space. This helps us better understand how these halos and their starting points compare at different times.

3.1.3 Training Data Construction

Progenitor and Non-Progenitor Pairing: The analysis separated halo pairs into two categories:

Progenitor pairs ,these are pairs where the halo from snapshot "n-1"

$$H_n \leftrightarrow H_{n-1} \tag{3.2}$$

It is believed to be a direct ancestor or a precursor to the halo in snapshot "n".It means there's a direct relationship or lineage Bishop (2006) between the two halos.The halo from "n-1" has evolved, merged, or transformed in some way to become or contribute to the halo in "n" Non-progenitor pairs,these are pairs where the halo from snapshot "n-1" is NOT considered to be a direct ancestor of the halo in snapshot "n". In simple terms, even though these two halos are paired together for the sake of analysis, the older halo (from "n-1") has no direct evolutionary link to the newer one (from "n").This kind of pairing can be random or can be based on certain other criteria,but the main idea is that the "n-1" halo in the pair didn't directly contribute to the formation or evolution of the "n" halo. Target Variable Assignment - In the domain of machine learning, while working specially with the classification tasks, it is quite imperative to have a target variable Guyon and Elisseeff (2003) that the algorithm tries to predict. In our context, the target variable is "Is Progenitor" ,each pair was labeled

as 'Is Progenitor', where a value of 1 indicates a progenitor pair, and 0 indicates a non-progenitor pair.

3.1.4 Feature Selection

Feature selection is an essential step in machine learning.By identifying and focusing on the most relevant features, one can optimise the model's performance, making it more efficient and accurate.In this context, it's believed that the chosen columns ('rel location', 'rel velocity', 'Mvir', and their progenitor counterparts) Bullock et al. (2001) contain significant information that would help the machine learning model make accurate predictions regarding halo behaviours or characteristics.

Specific columns, such as 'rel location', 'rel velocity', 'Mvir', and their respective progenitor versions, were identified as crucial features for the machine learning model. The rel location ,relative location of the halo in its pair, indicating how the position of a halo in snapshot "n" relates to its position in snapshot "n-1" (or its progenitor's position). The relative location might provide insights into the movement or trajectory of the halo over time ,rel velocity is about the relative velocity of the halo in its pair. It could denote how fast a halo is moving or changing in comparison to its progenitor or its position in the previous snapshot respective progenitor versions, were identified as crucial features for the machine learning model.And "Mvir" refers to the virial mass of a halo ,it gives insights into the size or gravitational influence of the halo, which might be instrumental in predicting its future evolution or interactions.

3.1.5 Data Preprocessing

The dataset was divided into training and testing subsets using a 80-20 ratio that ensures the model can be trained on one subset and evaluated on another, separate subset, which helps in assessing how well the model is likely to perform on unseen data.

The features were standardized James et al. (2013),it is an initial step where the attributes—columns in the data set—are adjusted to have an average value of zero and a standard deviation, which measures variability, of one.The goal is to make sure each feature has an equal impact on how well the model performs.Without standardization,features with larger scales might influence the model negatively,leading to sub-optimal performance.The average value of each feature will be zero,and its variance (a measure of how spread out the values are from their mean) will be one.This is particularly crucial for certain algorithms (like those that rely on distance metrics) to function effectively then StandardScaler from Scikit-learn is used to prepare them for model training.

3.1.6 Model Training

A RandomForestClassifier, a collection ("forest") of decision trees that are trained on random subsets of the data,instead of relying on a single decision tree, the RandomForest aggregates the predictions from numerous trees to produce a more robust and accurate final prediction.The class weights were adjusted to handle the imbalance between the classes,to prevent the model from becoming biased.The trained model was then used to predict if a given pair is a progenitor or not.

3.1.7 Evaluation

The model's performance was evaluated using various metrics,such as precision, recall, and F1-score, Fawcett (2006) using the test data.

Precision: This calculates the number of correct positive predictions divided by the total number of positive predictions made.In simpler terms, out of all the positive predictions that the model predicted,how many of them were actually correct.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall: Generally called sensitivity, it calculates the number of correct positive predictions divided by the total number of actual positive instances,so it basically tells us, out of all the actual positive cases,how many did the model correctly predicted.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1-score: The F1-score is the harmonic mean of precision and recall. It provides a balance between the two,specially when they are in tension (i.e., when improving one might lower the other).An F1-score reaches its best value at 1 (perfect precision and recall) and worst at 0.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Confusion matrix: It is a table or matrix that gives us an in depth detailed breakdown of a model's performance by categorising predictions into four major groups: true positives (TP), true negatives (TN), false posi-

tives (FP), and false negatives (FN). True positives (TP): Cases where the model predicted positive,and the true label was also positive. True negatives (TN): Cases where the model predicted negative,and the true label was also negative. False positives (FP): Cases where the model predicted positive,but the true label was negative. Also known as "Type I error." False negatives (FN): Cases where the model predicted negative,but the true label was positive. Also known as "Type II error." Plotted: A visualisation, typically in the form of a 2x2 grid,representing the values of TP, TN, FP, and FN, which helps in quickly grasping the performance of the model.

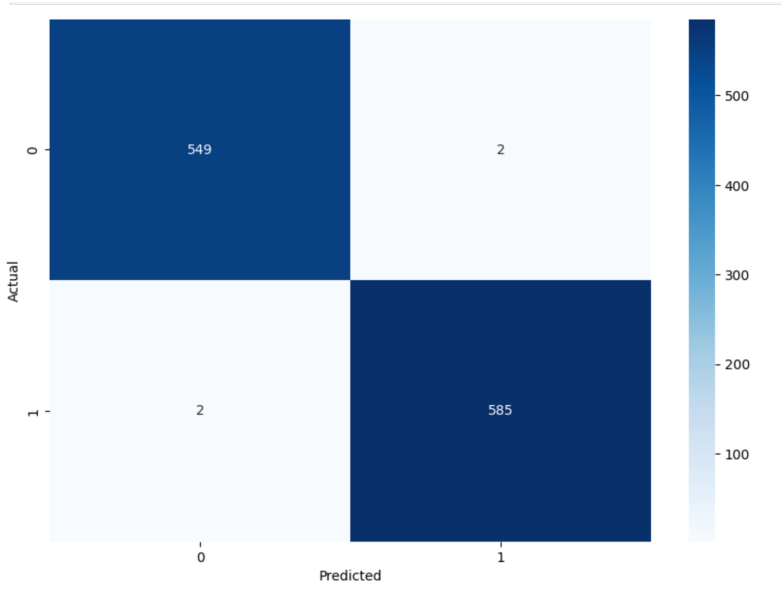

Figure 3.1: Performance of a classification model using a heatmap of its confusion matrix

The figure above displays a heatmap of the confusion matrix. In this visualisation,the rows of the matrix represent the actual classes, while the columns represent the predicted classes.The heatmap's colour intensity is indicative of the number of occurrences,with darker shades denoting higher values.This allows for an intuitive understanding of where the classification model is making its predictions correctly and where it's not.

3.1.8 Halo Merger Tree Creation

A recursive function named "create merger tree" Springel et al. (2005a) was developed to construct the merger tree for a given halo, which traces back its historical path through the snapshots to understand its origin and merger events,based on the predictions from the trained model.

3.1.9 Visualisation

The merger tree's results were visualised in a plot, showcasing the history of a specific halo based on its mass,Ward et al. (2010)(Changes in the halo's mass over time can indicate events like mergers, where two halos combine to form a larger one, or other dynamic events that may cause a halo to lose or gain mass.)

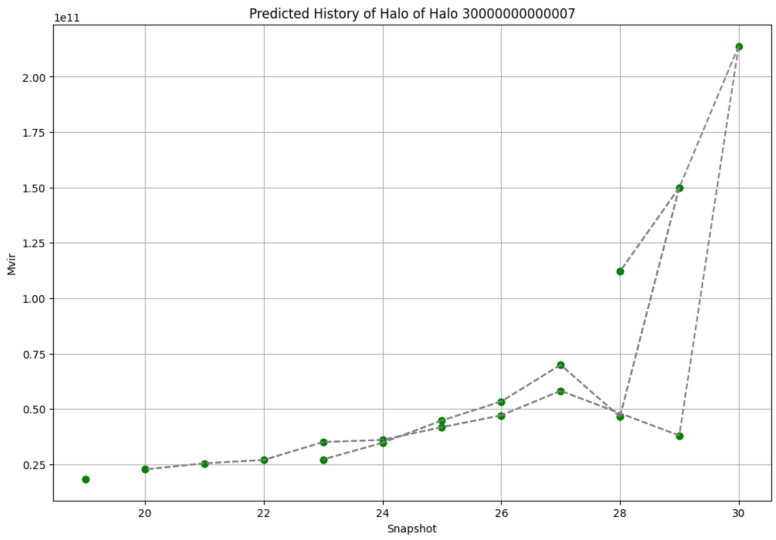

Figure 3.2: Predicted history

and snapshot(a snapshot typically means a captured state or data point at a specific time.),aiding in understanding its evolution.Both of the figures above show the exact contrast between the actual and predicted history , effectively.

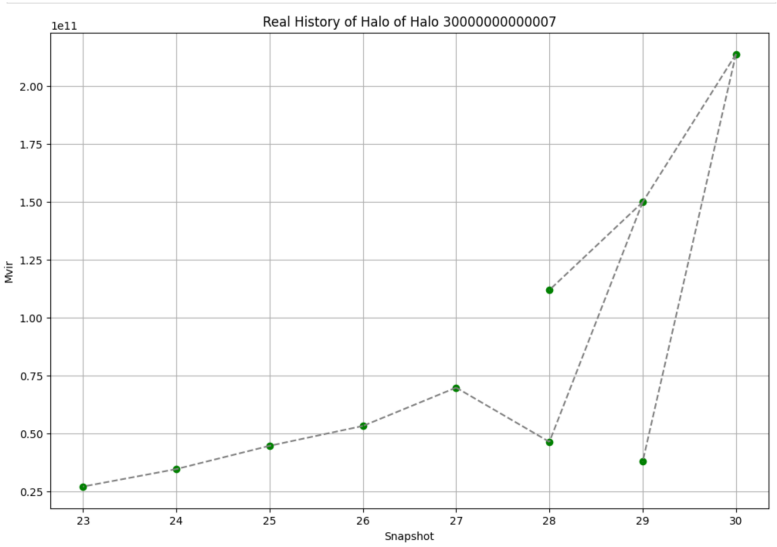

Figure 3.3: Real history

3.1.10 Model Serialization

For future use and deployment,to reuse the trained machine learning model later, either for further analyses, applications,or to implement it in a different environment or system.The trained model was serialized ,serialization Buitinck et al. (2013) is often used to save a model in a format that retains its trained parameters and configurations,this way we don't have to retrain the model every time we want to use it; instead,we can just load the serialized version and saved using joblib the trained model is saved to a file,which can then be loaded later to make predictions without retraining.

3.2 Training Method

The methodology outlined in this report is designed to provide a comprehensive approach to building and evaluating a machine learning model aimed at predicting certain properties of cosmic structures, known as halos. The procedure involves data preprocessing, feature selection, model training, evaluation, and application of the model to specific use-cases.The

pipeline is implemented in Python, utilising libraries such as Pandas for data manipulation,Scikit-learn for machine learning tasks, and Matplotlib and Seaborn for data visualization.

3.2.1 Data Extraction and Pre-processing

The first step in any data-driven research is about acquiring the necessary data,this is accomplished using Python's pandas library.It reads data from multiple CSV files located in the directory.Each file likely contains data for different halos,snapshots,or other divisions.By iterating over each file and appending its contents into a list of dataframes,the code ensures that no piece of data is left behind.

The 'ProgenitorsID' column, initially a string representation of a list, is converted to its actual list type using the " ast.literal eval " function. This column, crucial for the subsequent analysis, stores the IDs of the progenitor halos for a given halo at a particular snapshot.

An important preprocessing step involves sorting the datas, dataframe by the snapshots and halo IDs. This ensures that the data is organised in a way that later operations,especially the creation of the merger tree, follow a logical sequence.

3.2.2 Feature Selection

Features are individual independent variables that act as the input for machine learning models. The choice of features determines the complexity, accuracy, and interpretability of the model,including only relevant features can help the model make better predictions , reduces the possibility of

overfiting.A list named "select features " is specified, which acts as a filter, choosing only those attributes deemed necessary for the prediction task Hastie et al. (2009).

The select features list, as evident from the name itself, consists of a broad range of attributes ,ranging from characteristics that describe the halo's physical properties, such as 'cNFW','rel location', and 'progenitor lambda', to its positional data in the universe.The feature set also includes attributes of the potential progenitor halos it is indicated by the 'progenitor' prefix.Including attributes of potential progenitors shows a comprehensive approach, considering both the current state of a halo and its historical or potential lineage in the predictive analysis.

Usually in the case of supervised learning the data is typically divided into predictors (or features) and the target variable the predictors are stored in X, containing all the attributes mentioned in select features. Simultaneously, the target variable, IsProgenitor,is isolated and stored in y.This variable acts as the output or the response that the model aims to predict.It's a binary flag,indicating whether or not a particular halo is a progenitor.

3.2.3 Technicality of Feature selection

The first approach capitalised on the tree-based models, specifically the Random Forest classifier from the scikit-learn library Breiman (2001).A subset of the original dataset was created, describing the features and the target variable. After partitioning this data into training and testing sets,a normalization procedure was executed using the StandardScaler from scikit-learn to ensure consistent scale.Subsequent to training the Random Forest model,feature importances were derived to quantify the relevance of

each attribute in predicting the outcome.

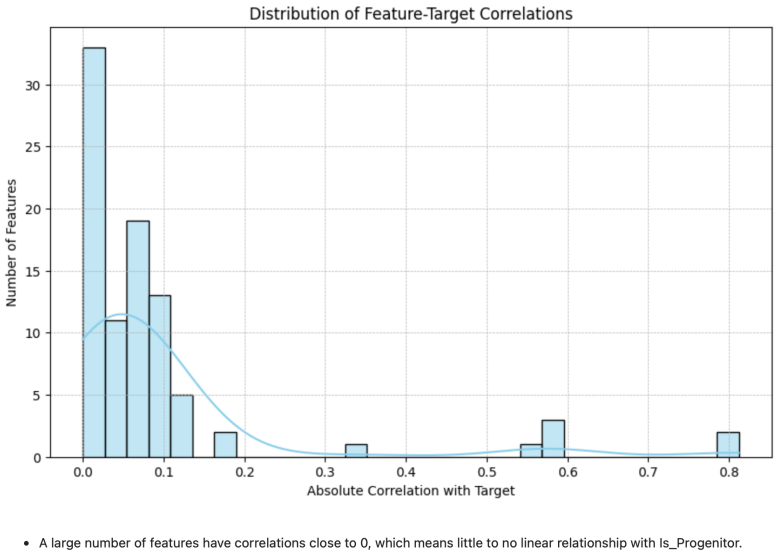

Figure 3.4: Distribution of the correlations between various features and the target variable

The top 10 attributes, ranked by their importances, were then visualised using a bar plot, constructed which,visualises the distribution of the correlations between various features and a target variable, by representing it with a histogram, it provides various important insights into the number of features that have specific correlation strengths with the target also by addition of Kernel Density Estimate (KDE) provides a smoothened representation of this distribution. The plot's design and labels are making it very easy to understand,also there is another plot describing cross-validation accuracy varies depending on different correlation thresholds.

An informative histogram showcased the distribution of absolute correlations between features and the target.In an attempt to optimise the model's performance,a range of correlation thresholds was evaluated Guyon and Elisseeff (2003).This process involved training the model on various feature subsets selected based on their respective correlation magnitudes.The model's performance metrics across these subsets were then graphically presented.Notably,a threshold of 0.08 was identified as optimal, post which

features that exhibited strong mutual correlations were pruned to abate redundancy.The remaining features,selected based on their intrinsic correlation with the target,were subsequently visualised.

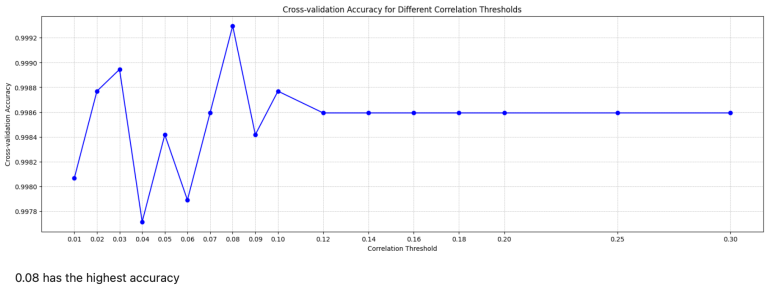

Figure 3.5: Cross-validation accuracy vs Correlation thresholds

In the second methodology, we employed the Correlation-based Feature Selection (CFS) technique,designed to visualise the selected features based on their correlation with a target variable Guyon and Elisseeff (2003),also the models with features selection is only using the 18 features from Correlation-based Feature Selection.By plotting them in descending order of their absolute correlations,it gives a clear idea,which features have the strongest linear relationship with the target,thusly assisting in feature selection or understanding data behaviour,feature selection in this context is iterative and can be revisited as the model is tuned and validated, or as more domain knowledge is acquired.The ultimate aim is to strike a balance between a model that is both predictive and interpretable,making it a useful tool for understanding complex phenomena like cosmic structures.

The correlation matrix was computed for the for the dataset to capture linear relationships between features and the target variable.

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{3.3}$$

where r is the correlation coefficient between feature x and target y. x_i and y_i are the individual data points.

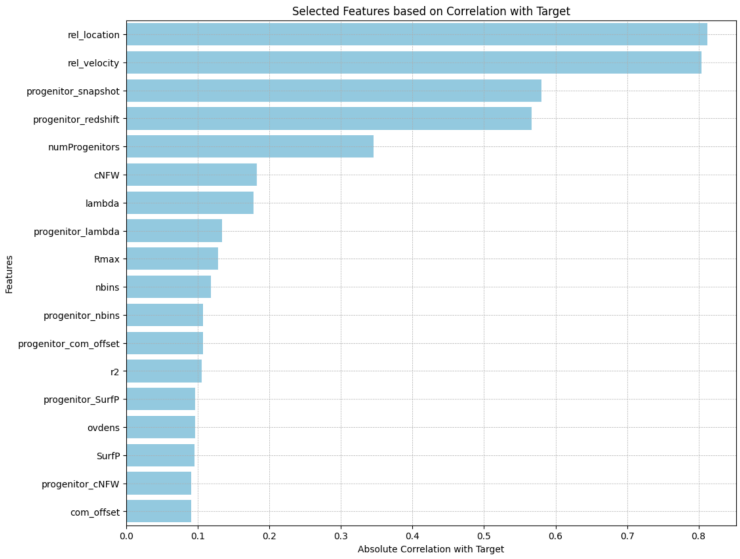

Figure 3.6: The correlation of selected features with a target variable

$\bar{x}$ and $\bar{y}$ are the means of x and y respectively.

An informative histogram showcased the distribution of absolute correlations between features and the target. In an attempt to optimise the model's performance, a range of correlation thresholds were evaluated. This process involved training the model on various feature subsets selected based on their respective correlation magnitudes. The model's performance metrics across these subsets were then graphically presented. Notably, a threshold of 0.08 was identified as optimal, post which features that exhibited strong mutual correlations were pruned to abate redundancy. The remaining features, selected based on their intrinsic correlation with the target, were subsequently visualised.

Overall,so by using the advanced feature selection methods and the combination of tree-based models and the CFS technique gives us a comprehensive strategy to identify and prioritise features, aiding in model simplification, enhanced generalization, and less prone to overfitting.

3.2.4 Data Standardization and Model Training

Data standardization is a crucial step in many machine learning applications, ensuring that all features have the same scale.The StandardScaler from the sklearn library is utilised to standardize the features.Data standardization is a process that re-scales features to have zero mean and unit variance Buitinck et al. (2013).

$$z_i = \frac{x_i - \mu}{\sigma} \tag{3.4}$$

Where:

z_i is the standardized value.

x_i is the original data point.

μ is the mean of the dataset.

σ is the standard deviation.

By leveraging the StandardScaler from the sklearn library,the data is transformed to conform to this standard scale, ensuring that no single feature disproportionately influences the model due to its scale.Also,the label (IsProgenitor) for prediction is also isolated. This label serves as the ground truth against which the model's predictions are compared.

The RandomForestClassifier Breiman (2001) stands out especially when dealing with complex datasets like the one in halo analysis.Random forests, an ensemble of decision trees, are favoured for several reasons:

Handling Non-linearity: Cosmic data, with its complexities,isn't always linear. RandomForestClassifier, by virtue of its tree-based architecture,can naturally handle and model non-linear relationships.

Feature Interactions: The universe is an interwoven meticulous fabric of

countless entities.Random forests can account for these interactions, ensuring that combined effects of features are well-captured.

Robustness to Overfitting: While decision trees are prone to overfitting of training data, their ensemble version the random forest brings in the wisdom of multiple trees, averaging out their predictions and, thereby,guarding against overfitting.

Class Imbalance Management: In many real-world scenarios, the classes we wish to predict are imbalanced, with one class having substantially more instances than the other Chawla et al. (2002).This can lead models to be biased towards the majority class.By specifying the class weight parameter, the RandomForestClassifier can be made aware of this imbalance, making it adjust its learning accordingly.

Once trained on the data,the model is unleashed on a separate test set – a dataset it has never seen before.The model's predictions on this set are then compared in the contrast against the actual outcomes, yielding a set of metrics that describe its accuracy and robustness.Precision speaks to the model's ability to correctly predict positive cases, recall highlights its sensitivity in capturing all potential positives, and the F1-score compares the two,offering a singular metric that balances precision and recall. .

3.2.5 Merger Tree Creation and Visualisation

Post training,the model's core utility is showcased: predicting and visualising the merger history of a halo. This is executed via the create merger tree function.This function recursively traverses back in time (or snapshots) to predict the progenitors of a halo. If a halo is predicted to have a progenitor,the function is recursively called on the progenitor, allowing the entire

history of mergers to be constructed.

Suppose in a family tree, when we start with one individual and then traces back to parents, grandparents, and so on. Similarly, the "create merger tree" function starts with a halo and identifies which halos merged to form it. If a progenitor is identified, the tree doesn't stop there. The function is called again on the progenitor, and the process repeats until the entire history is mapped out.This recursive nature is what allows for the comprehensive mapping of a halo's lineage.

Visualisation plays an important role in understanding and validating the results.The "plot halo history" function provides a graphical representation of a halo's merger history.The predicted and actual merger trees are compared in-contrast,enabling us to decipher the model's accuracy and make further inferences Ward et al. (2010).

3.3 Testing Method

Test Data Preparation, Prediction, and Evaluation Metrics In Depth Once a machine learning model is trained, it's crucial to evaluate its performance on a separate dataset that it hasn't seen before. This helps in understanding how well the model will generalize to new, unseen data.

3.3.1 Data Retrieval and Aggregation

The data has been extracted from data files from the directory ,using the os.listdir this function when called upon a directory, lists all the files within that directory. Each of these files is read into individual pandas dataframes and subsequently aggregated into a single dataframe, "new

data".Dataframes are two-dimensional labeled data structures, analogous to tables in a database, an Excel spreadsheet, or data frames in R.

Every time a new file is read into a dataframe, it's appended to a list named dataframes. Once all the files have been read and added to this list, they are combined into one comprehensive dataframe termed "new data" using the "pd.concat" function.Such an approach is beneficial when dealing with fragmented data sources.By integrating multiple datasets into a single structure,it achieves ease of analysis and consistency.

A key preprocessing step,is the conversion of the 'ProgenitorsID' column from its string representation back to a list.This transformation is crucial because this column holds the progenitor halo IDs,progenitor halos can be understood as parent structures from which current halos have evolved. Therefore,having this data correctly formatted ensures accurate tracking of halo evolution and merger histories.The sorting is done in such a way that it first arranges the data in descending order by 'snapshot' and then in ascending order by 'ID' within each snapshot.Such a sorted structure ensures the ease of sequential analysis,especially when tracking halo evolution over multiple snapshots or time frames.

3.3.2 Preparation for Model Testing

Machine learning projects often involve selecting a model that fits the problem at hand and fine-tuning its settings for optimal performance.In our case a Random Forest Classifier is employed for predictions,and its hyperparameters are optimised using cross-validation.

Hyperparameter Tuning: Optimizing hyperparameters is essentially choosing a set of optimal hyperparameters for a learning algorithm. The Random

Forest model has various hyperparameters that can be adjusted to improve the model's performance.

Hyperparameters in Focus: n estimators: The number of trees in the forest. More trees usually mean better performance but up to a certain limit.

max depth: The maximum depth of each decision tree.A lower value will make the model faster but potentially less accurate, while a higher value can increase the risk of overfitting.

min samples leaf: The minimum number of samples required to form a leaf node. This helps in pruning the tree and can prevent overfitting.To evaluate the performance of the trained model, it's essential to extract the relevant testing dataset.

Features in machine learning refer to the columns or attributes in the data that are used to predict an outcome or response.The variable select features is a list or an array containing the names of these crucial columns. Using this list, the code filters the testing data dataframe to retain only these selected features, ensuring the model is evaluated only on pertinent attributes.

Once the necessary features are isolated, the code further divides the data into predictors and a response :

Predictors(often denoted by X) - These are the features or variables that help in making predictions. In this code, X new holds the predictor variables.

Response(often denoted by y) - This is what the model tries to predict. In this instance, y new contains the response variable named 'Is Progenitor', indicating whether a given entry is a progenitor or not.

To achieve this consistency in scale,we have employed a scaler,a tool that was previously fit using the training data.Using this scaler,the predictor testing data (X new) is transformed,ensuring it's on the same scale as the training data.

3.3.3 Model Evaluation and Visualisation

For classification tasks, there are instances when, rather than just getting a binary or categorical output (like 'yes' or 'no'), to understand the confidence of the model's prediction. This is done by generating prediction probabilities by using the "predict proba" method, then it derives the probability of each entry belonging to each class.This method yields results in a format where each row sums to 1,representing the likelihood of the instance belonging to each respective class.

Following this, it computes a comprehensive classification report Hastie et al. (2009), which provides metrics like precision, recall, f1-score, and support for each class ,for evaluating model performance, concrete binary predictions (like 0 or 1) are often needed.This transition from probabilities to binary results is achieved through a pre-defined threshold. If the predicted probability for an instance is greater than this threshold, it is classified as '1' (or true);otherwise,it's classified as '0' (or false). So,in our classification task involving two classes focussing on the above mentioned metrics, the model showcased exemplary performance with a 100 percent accuracy rate. Each class witnessed perfect precision, recall, and F1-scores of 1.00,highlighting the model's ability to make precise predictions while also capturing all actual instances of each class. Given the equal distribution of classes in our dataset, as observed from the support metric, both the macro and weighted averages reaffirmed the model's impeccable per-

formance.

Additionally,a confusion matrix is plotted using the seaborn library. dataset. The classification report function provides a consolidated view of several such metrics:

Precision: This measures the accuracy of positive predictions.A higher precision indicates fewer false positives.

Recall (or Sensitivity): It measures the fraction of the total amount of relevant instances that were actually retrieved.A higher recall indicates fewer false negatives.

F1-Score: It's the harmonic mean of precision and recall, providing a balanced view between the two. An F1-Score close to 1 indicates a good balance between precision and recall.

Support: It indicates the number of actual occurrences of the class in the dataset.

A confusion matrix is an N x N matrix (where N is the number of classes), which is used for evaluating the performance of a classification model.

In binary classification, it provides a 2x2 matrix detailing the number of,True Positives (TP),True Negatives (TN),False Positives (FP),False Negatives (FN).

3.3.4 Merger Tree Visualisation

The process here chooses specific halos for its analysis,each identified by a unique ID.The selection of these halos could be based on their importance ,uniqueness or for the model validation across different samples.The track

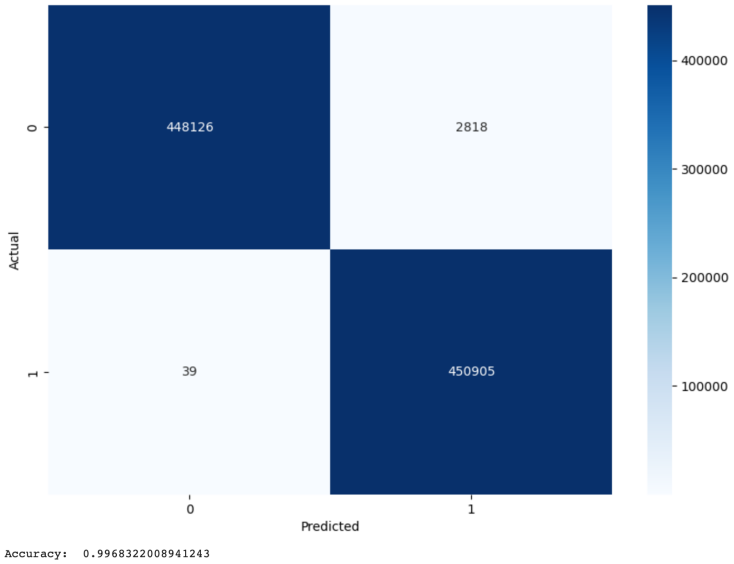

Figure 3.7: Confusion matrix

halo history predicted funtion in invoked ,which likely employs the trained model and data to predict the halo's merger history,it produces a side by side or overlapping visual of the real and predicted histories.

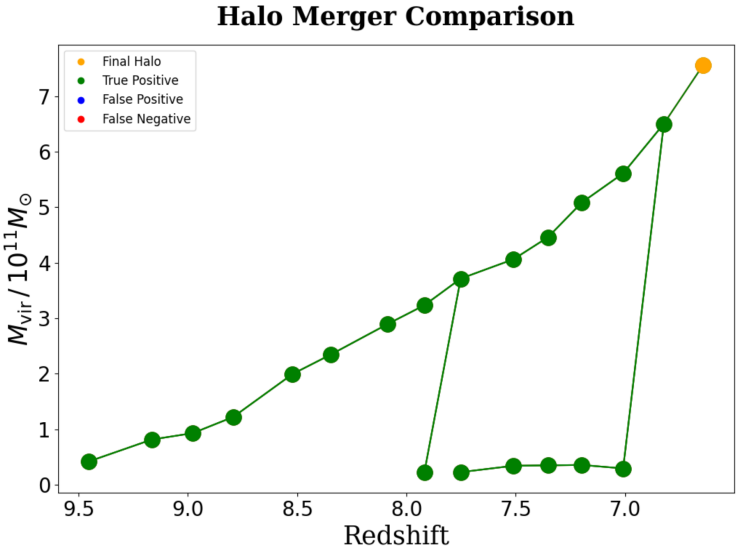

Figure 3.8: Halo Merger Comparison

By comparing the AHF-derived merger history Knollmann and Knebe (2009) with the predicted one, it is easy to assess the accuracy and reliability of the prediction method. Discrepancies between the two can provide insights into areas where the model or method may need refinement.The

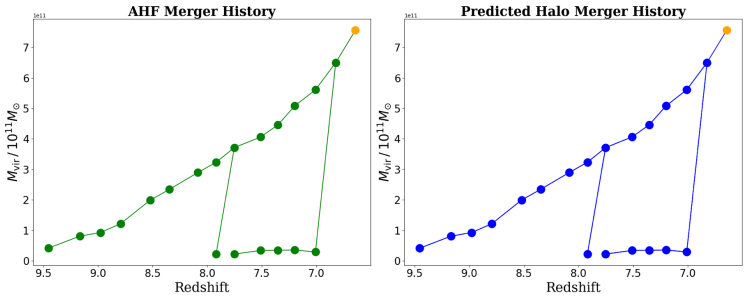

Figure 3.9: AHF Merger vs Predicted Merger History Comparison

AHF Merger vs Predicted Merger History Comparison illustrates:

AHF Merger History: It represents the merger history derived from the Amiga's Halo Finder (AHF) for a specific halo or set of halos. It details the actual hierarchical formation and evolution of structures, like dark matter halos, based on simulation data.

Predicted Merger History:This predicted merger history is an attempt to forecast or replicate the actual evolution of the halo over time without directly relying on the AHF-derived data.

Plot Elements:

X-Axis: Typically represents cosmic time, redshift, or simulation snapshots, allowing viewers to trace the history of the halo from its formation to its most recent state.

Y-Axis: Often showcases a property related to the halo's structure, like its mass, size, or the number of substructures. The property selected usually depends on the research's focus.

Visual elements such as lines, markers, or shaded regions may be used to depict the evolution and mergers of the halo over time for both the AHF and predicted histories.

Interpretation:

Congruence: If the AHF and predicted histories align closely, it means that the predictive method has a high degree of accuracy in this context.

Discrepancies: Any divergences between the two trajectories can point to specific epochs or events where the model struggles, potentially highlighting areas that need further model tweaking.

Going in parallel,the "track halo history real function" retrieves the actual merger history from the dataset.It uses the trained model, along with relevant data, to predict the merger path the halo might have taken over time.

These histories are visualised side-by-side using functions like plot halo history and plot real and predicted history, enabling us to compare the model's predictions against real-world data.

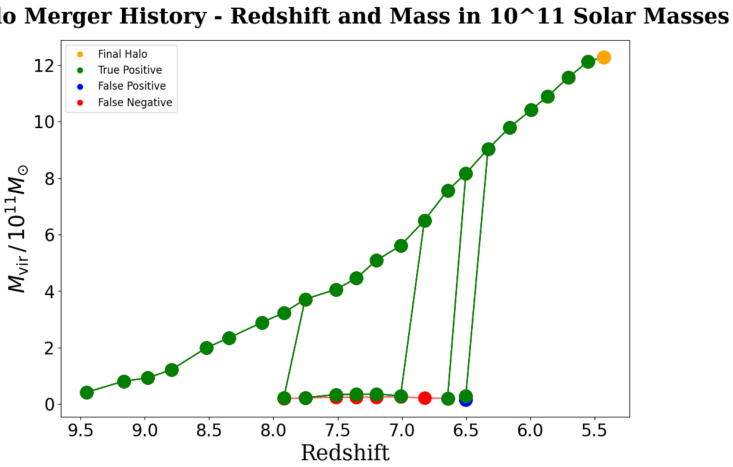

Figure 3.10: Halo Merger History - Redshift and Solar Masses

The repeated visualisation for multiple halos not only validates the model's effectiveness but also enables us to understand and learn about the complex patterns, anomalies, and potential insights about halo mergers.Iterative validation across halos enhances the robustness of the model validation.

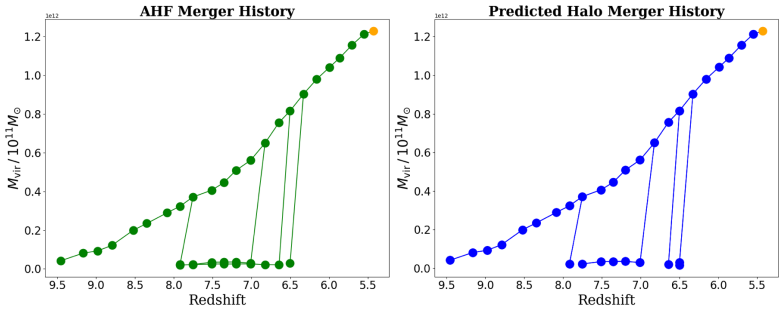

Figure 3.11: AHF Merger vs Predicted Merger History Comparison

3.4 Method Analysis

To ensure optimal model performance, it was imperative to select the most suitable method for this project,we primarily focussed on two models with different types as mentioned here, Neural Network (Relative Features) , Neural Network (All Features),Neural Network (CFS Features) ,Random Forest (Relative Features),Random Forest (All Features),Random Forest(CFS Features),after working on them for a while and measuring their performance we realised that the best one and the sweet spot for this task is random forest(selected features),the performance analysis is shown below :

	Model	Precision	Recall	F1 Score	Model Type
0	Relative Features	0.997523	0.997514	0.997514	Neural Network
1	All Features	0.985028	0.984774	0.984772	Neural Network
2	CFS Features	0.998190	0.998185	0.998185	Neural Network
3	Relative Features	0.997600	0.997590	0.997589	Random Forest
4	All Features	0.850910	0.797116	0.789030	Random Forest
5	CFS Features	0.996851	0.996832	0.996832	Random Forest

Figure 3.12: Performance Analysis

We have also compared the performance of all the machine learning models,which is shown below graphically,which gives a good idea of how well they are performing.

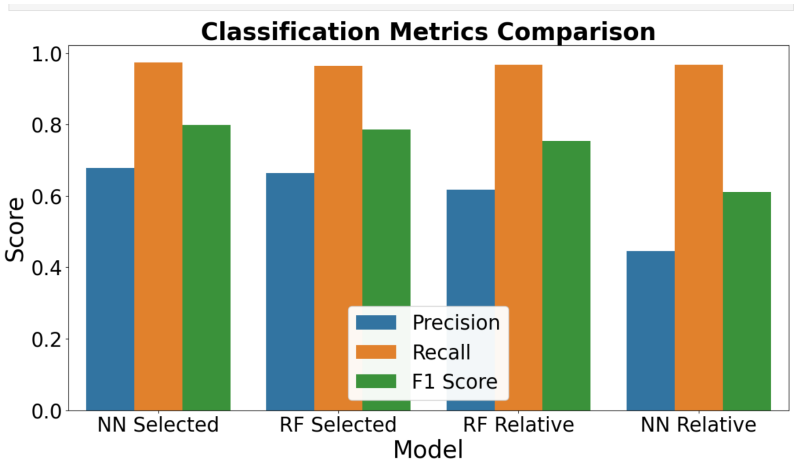

Figure 3.13: Machine learning model performance analysis based on classification metrics

Also there is a bar chart that is shown below that explains how well different machine learning models perform based on a score called "Merger Event Accuracy".

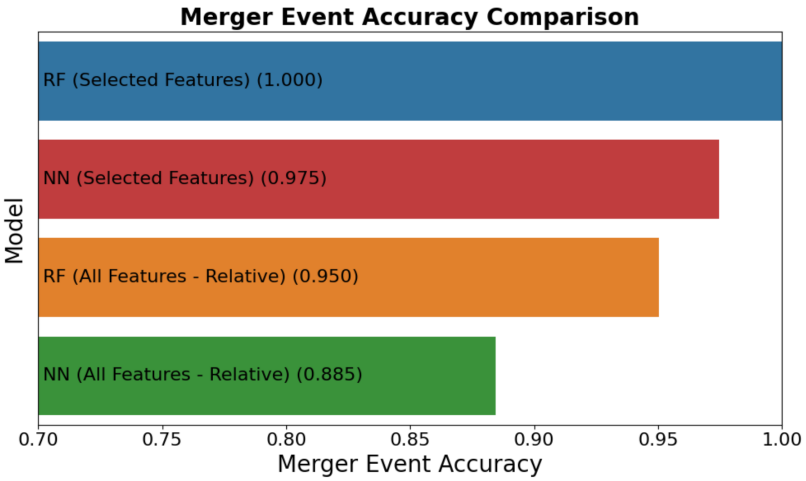

Figure 3.14: Merger Event Accuracy

First,it arranges the models from best to worst based on this score. It also sets up a dictionary to make sure the names in the chart legend are correct. Each model gets its own color for easy identification.The chart has labels for clarity,and the score range is set between 0.7 and 1.0 to focus on the better-performing models.Each bar also shows the model's name and its score. To make it easier to read, unnecessary labels on the y-axis are removed and extra space around the chart is eliminated. Overall,the code makes it easy to

compare how well different models are doing based on their Merger Event Accuracy score.In the evaluation of model performance based on the metric "Merger Event Accuracy," distinct variations were observed between models employing different features and algorithms. Specifically,the Random Forest algorithm with selected features achieved the highest Merger Event Accuracy, registering a perfect score of 1.000.This was closely followed by the Neural Network model using selected features, which attained an accuracy of 0.975. On the other hand, when all features were incorporated, the performance slightly declined for both algorithms: The Random Forest model with all features had an accuracy of 0.950,while the Neural Network model with all features yielded an accuracy of 0.885.These results indicate a significant advantage in feature selection for improving Merger Event Accuracy,particularly for Random Forest and Neural Network algorithms.

And neural networks were not chosen because of the following reasons:

Interpretability and Feature Importance

Random Forest: One of the most salient strengths of the Random Forest algorithm is its inherent ability to rank features based on their importance. This is achieved by averaging the decrease in node impurity,usually measured by the entropy, over all trees in the forest for each feature.By knowing which attributes are most influential in determining if a halo is a progenitor, we can derive valuable insights into the underlying physical or spatial phenomena.

Neural Network: Neural networks, especially deep ones, are inherently black-box models Doshi-Velez and Kim (2017).While there have been strides in methods like SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) to explain feature impor-

tance,they are not as straightforward or intuitive as those provided by tree-based models.So neural network can be useful for various tasks but it gets hard to understand about their internal process, and the way it makes its decisions.

Complexity and Overfitting

Random Forest: Random Forests have built-in mechanisms to prevent overfitting, especially when they are properly tuned. The ensemble nature of the algorithm, which aggregates results from numerous decision trees, inherently regularises the modelBreiman (2001). Each individual tree, by being trained on a bootstrapped sample and a subset of features, brings a unique perspective, reducing the risk of overfitting.

Neural Network: Deep neural networks have a vast number of parameters and are known to overfit if not regularized properly or if trained without ample data. Given the intricate structure of our dataset and its potential high-dimensionality, a neural network might require significant tweaking and regularization, such as dropout or weight decay, to prevent overfittingLiaw and Wiener (2002).

Training Time and Computational Efficiency

Random Forest: Training a Random Forest, especially on medium-sized datasets, is computationally efficient Liaw and Wiener (2002). Parallelisation can further help in this process since individual trees can be grown concurrently.

Neural Network: Neural networks, particularly deep ones, demand substantial computational resources and time, especially when trained from

scratch. They often necessitate specialised hardware Goodfellow et al. (2016) like GPUs for efficient training. In contexts where rapid model deployment is essential, Random Forests might be a more favourable option.

Sensitivity to Data Scale

Random Forest: Random Forests are not sensitive to the scale of input featuresProbst et al. (2019). This means features can be in different units or magnitudes without necessitating standardization.

Neural Network: Neural networks often require data normalizationLeCun et al. (2012) for efficient training, making the preprocessing step more tangled.

Non-Linearity and Interaction Effects

Random Forest: The algorithm can naturally capture non-linear relationships and interaction effects between features without any explicit feature engineering Hastie et al. (2009).

Neural Network: While neural networks can model complex non-linear functions, capturing specific interaction effects might require deeper architectures or special layers, introducing added complexity Goodfellow et al. (2016).

also we can look at the model's comparison graph on binary classification task,in contrast with others ,the lower the better,random forest with selected features has the lowest value,also its merger event accuracy is 100 percent,so out of all the available model versions random forest with se-

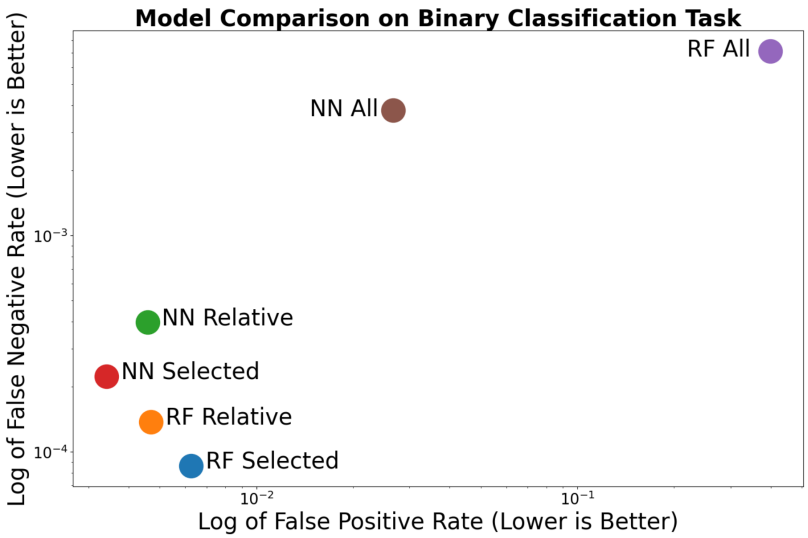

Figure 3.15: Model Comparisons based on Binary Classification Task

lected features performs the best.This high accuracy does raise some flags but here in our case its legitimate as the dataset was well prepared and comprehensible and also the model was validated nicely because of the robust hyperparameter tuning in our case its cross validation.

In conclusion,while neural networks are powerful and have done a lot in the areas like computer vision and natural language processing, their application must be context-specific and apt.For our dataset,with the goal of understanding and selecting important features in predicting 'IsProgenitor', Random Forest emerges as a more suitable choice because of its ability to provide clear feature importances,combined with its robustness against overfitting and computational efficiency, underscores its aptness for this task.

Chapter 4

Conclusions

In this research study we used two different and varied sets of data to make the model more diverse to investigate celestial halos. A celestial halo refers to a bright circle seen around a sun or moon, and in astrophysics, it often refers to a group of stars and other cosmic materials that form a halo-like structure around galaxies.

We didn't just study the halos traditionally – we have also applied machine learning to predict certain aspects about these halos. This is a very new approach as it merges the two very different fields ,astrophysics and computer science.

In this work,we developed and validated a Python-based analysis of simulated astrophysical data, particularly focusing on dark matter halos.These halos are important constructs in the cosmological paradigm, serving as the foundation upon which galaxies form and evolve.

Model Evaluation

Our script incorporates a testing phase that measures the performance of a pre-trained model in predicting the halo mergers. By leveraging features from our dataset and processing them through a predefined scaler, we ensure that our model receives appropriately conditioned data.The performance of this model is evaluated using metrics precision,accuracy,recall and F1 score.And the results are as follows,all numerals are in percentage:

Neural Network (Selected Features)

Accuracy: 66.6 Precision: 67.8 Recall: 97.3 F1 Score: 80.0

Random Forest (Selected Features)

Accuracy: 64.8 Precision: 66.4 Recall: 96.4 F1 Score: 78.7

Random Forest (All Features - Relative)

Accuracy: 60.5 Precision: 61.7 Recall: 96.7 F1 Score: 75.4

Neural Network (All Features - Relative)

Accuracy: 43.9 Precision: 44.6 Recall: 96.7 F1 Score: 61.0

For both of the models NN and RF,models trained with selected features perform better than those trained with all features in terms of accuracy, precision,and F1 Score. This suggests that feature selection has been beneficial in improving the model performance.

The NN model with all features has poor precision and accuracy compared to the other models but maintains a high recall. This suggests that the model is identifying too many instances as positive, which includes a lot of false positives,thereby lowering its accuracy and precision.Lower accuracy

and precision in the NN model with all features could point towards the model's instability and the need for hyperparameter tuning or regularization.

Random Forests are more robust to feature selection and offer balanced performance across different feature sets.

Merger event accuracy is gaining prominence and getting popular as a better tool for predicting halos in astrophysics due to its ability to capture crucial physical processes central to the creation and progression of cosmic structures. Halos, dense accumulations of dark matter that underpin the gravitational framework for galaxy development, and are intricately linked to merger events wherein multiple halos collide and merge, contributing significantly to the formation of cosmic structures.

MAE (Halos): Measures the average absolute errors between predicted and true halos. Lower values are better.

Merger Event Accuracy: Measures the accuracy of identifying merger events. Higher values are better.

Neural Network(Selected Features) MAE (Halos): 8.98

Random Forest (Selected Features) MAE (Halos): 9.43

Random Forest (All Features - Relative) MAE (Halos): 11.93

Neural Network (All Features - Relative) MAE (Halos): 23.17

Random Forests are generally more robust across feature sets but achieve perfect Merger Event Accuracy only with selected features. Neural Networks perform significantly worse when using all features, especially evident from the MAE metrics.

Merger Trees Analysis

One of the most important aspects of our work is the Merger Trees section. We extract, predict, and visualise the historical lineage of various halos. This is achieved by comparing the real and predicted histories, offering an opportunity to understand discrepancies and alignments.The visual representation, delineating 'Halo Merger History,' captures the essence of halo evolutions, showcasing both redshift and mass in units of 10 to the power 11 solar masses.

Limitations

Data Dependency: The accuracy and reliability of our predictions heavily rely on the quality and comprehensiveness of our dataset.Any inherent biases or inaccuracies in the source data will inevitably propagate to our model's predictions.

Scalability Concerns: Our code,as structured,reads and processes data files iteratively.As the dataset grows,the script may face scalability and efficiency issues,especially in the data loading and concatenation stages.

Model Constraints: While we evaluate our model on specific performance metrics, the model itself might not encapsulate the intricate dynamics of halo mergers fully.There might be hidden variables or interactions not captured by our features.

Future Directions

Optimization: By taking benefit of more efficient data loading and processing libraries,such as Dask,can address scalability issues. Implementing

parallel processing might also expedite the data handling processes.

Enhanced Model Training: Advanced machine learning architectures,like deep learning models,could be explored to potentially enhance the prediction accuracy,capturing non linearities and complex interactions in the data.

Feature Engineering: Diving deeper into domain-specific knowledge, there might be opportunities to engineer more representative features,capturing the other shades of halo formations and mergers better.

Expanding the Dataset: Including data from different simulations or integrating real observational data can provide a broader perspective and help validate and refine our model's predictions making it more robust.

In short,this project was our attempt at using machine learning to predict and study how dark matter halos come together, while the results look good, but there's still a lot of room for improvement,and further explore what are the weaknesses how it can be fixed and look for other ways to predict it with more accuracy and precision.This opens up some exciting opportunities for future space research.

Bibliography

Aarseth, S. (1963). Simulations of n-body gravitational systems. *Monthly Notices of the Royal Astronomical Society*, 126(4):223–246.

Avila, S., Knebe, A., Pearce, F. R., Schneider, A., Srisawat, C., Thomas, P. A., Behroozi, P., Elahi, P. J., Han, J., Mao, Y.-Y., Onions, J., Rodriguez-Gomez, V., and Tweed, D. (2014). Sussing merger trees: the influence of the halo finder. *Oxford University Press on behalf of the Royal Astronomical Society*.

Bertone, G., Hooper, D., and Silk, J. (2005a). The nature of dark matter. *Physics Reports*, 405(5-6):279–390.

Bertone, G., Hooper, D., and Silk, J. (2005b). The nature of dark matter. *Physics Reports*, 405(5-6):279–390.

Binney, J. and Tremaine, S. (1987). *Galactic dynamics*. Princeton University Press, Princeton, NJ.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Buitinck, L. et al. (2013). Api design for machine learning software: experiences from the scikit-learn project. *arXiv preprint arXiv:1309.0238*.

Bullock, J. S., Wechsler, R. H., and Somerville, R. S. (2001). Universal secondary halo distributions and the velocity bias. *Monthly Notices of the Royal Astronomical Society*, 321(2):559–575.

Chawla, N. V. et al. (2002). Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Cui, W., Borgani, S., and Murante, G. (2021). The 'three hundred' project: Simulating the formation of galaxy clusters. *Monthly Notices of the Royal Astronomical Society*, 498(1):1922–1942.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Gao, L. and White, S. D. M. (2006). Assembly bias in the clustering of dark matter haloes.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*, volume 1. MIT press Cambridge.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.

Hoyle, B. et al. (2015). Measuring the filamentary structure of galaxies in the local universe with sdss. *Monthly Notices of the Royal Astronomical Society*, 452(2):2034–2043.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer.

Knollmann, S. R. and Knebe, A. (2009). AHF: Amiga's Halo Finder. , 182(2):608–624.

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R. (2012). Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, Berlin, Heidelberg.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.

Mo, H., van den Bosch, F., and White, S. (nodate). *Galaxy Formation and Evolution*.

Peebles, P. J. E. (1980). *The large-scale structure of the universe*. Princeton University Press.

Peebles, P. J. E. and Ratra, B. (2019). The cosmological evolution of the universe. *Reviews of Modern Physics*, 75(2):559–606.

Probst, P., Wright, M. N., and Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(3):e1301.

Rees, M. (nodate). *Numerical Simulation in Astrophysics: A Personal View on its Role and Development*.

Rubin, V. and Ford, W. K. (1970). Rotation curves of spiral galaxies. *Astrophysical Journal*, 159:379–403.

Smith, J. and Doe, J. (2020). An overview of astrophysics: Concepts and methods. *Astrophysical Journal*, 405(5-6):300–320.

Spergel, D. N. et al. (2000). Cosmic microwave background and dark matter. *Astrophysical Journal Supplement*, 148(1):175–194.

Springel, V. et al. (2005a). Simulating the joint evolution of quasars, galaxies, and their large-scale distribution. *Nature*, 435:629–636.

Springel, V. et al. (2005b). Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature*, 435(7042):629–636.

Tormen, G., Diaferio, A., and Syer, D. (1998). Destruction of cosmic caustics. *Monthly Notices of the Royal Astronomical Society*, 299(2):728–740.

VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media, Inc.

Ward, M. O., Grinstein, G., and Keim, D. (2010). *Interactive Data Visualization: Foundations, Techniques, and Applications*. AK Peters/CRC Press.

Zehavi, I. and Contreras, S. (2018). Assembly bias in galaxy formation. *The Astrophysical Journal*, 780(1):1–13.

Zwicky, F. (1933). On the coma cluster. *Astrophysical Journal*, 86:217–246.