

Study on Separating Objects in Λ CDM Cosmological Simulations Using Machine Learning Algorithms

A Thesis

submitted to

Indian Institute of Science Education and Research Pune

in partial fulfillment of the requirements for the

BS-MS Dual Degree Programme

by

Soorya Narayan R



Indian Institute of Science Education and Research Pune

Dr. Homi Bhabha Road,

Pashan, Pune 411008, INDIA.

April, 2024

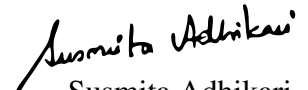
Supervisor: Susmita Adhikari

© Soorya Narayan R 2024

All rights reserved

Certificate

This is to certify that this dissertation entitled Study on Separating Objects in Λ CDM Cosmological Simulations Using Machine Learning Algorithms towards the partial fulfilment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research, Pune represents study/work carried out by Soorya Narayan R at Indian Institute of Science Education and Research under the supervision of Susmita Adhikari , Assistant Professor, Department of Physics , during the academic year 2023-2024.



Susmita Adhikari

Committee:

Susmita Adhikari

Sourabh Dube

This thesis is dedicated to the hours of sleep I lost in the past year

Declaration

I hereby declare that the matter embodied in the report entitled Study on Separating Objects in Λ CDM Cosmological Simulations Using Machine Learning Algorithms are the results of the work carried out by me at the Department of Physics, Indian Institute of Science Education and Research, Pune, under the supervision of Susmita Adhikari and the same has not been submitted elsewhere for any other degree. Wherever others contribute, every effort is made to indicate this clearly, with due reference to the literature and acknowledgement of collaborative research and discussions.



Soorya Narayan R

20191027

Acknowledgments

I would like to thank Dr Susmita Adhikari for the opportunity to work on this project. Your mentorship and guidance were one of the, if not the most important, driving factors for the progress I made. I would also like to thank Dr Susmita's group members for the engrossing discussions about their own projects and mine.

Thank you, Sanjana, for always being there for me. Going into the years of support would leave me no space for any other acknowledgements, so I will stick to this thesis year. There were times when your calls alone were motivating enough for me to get off my depressed ass. Thank you so much for listening to me blab about my thesis even when you did not have the slightest inclination for physics (you don't have to *cope* anymore). Thank you for sharing overseas *tea* about your labmates, which was my main source of entertainment. I wouldn't have started cooking if it weren't for you. The very same cooking saved me multiple days of digestive issues and pain. Thank you so much for keeping me company while I wrote my thesis.

I would like to thank my parents. Your daily calls and updates made me feel like I was at home, not missing a single thing. For reminding me to take it easy with work and chill out every now and then. For keeping me company in all my walks. Thank you, Mom, for all the psychology lessons and for always checking up on how I was doing.

I would also like to thank my sister for sharing her artwork, which kept me motivated enough to expand my creative outlets outside of work and stop me from burning out.

Abstract

Dark Matter, the most abundant matter in the universe, has eluded our understanding for decades. From rotation curves, CMB, and gravitational lensing, we see that structure formation in the universe is driven by dark matter, not baryons. Dark matter halos are some of the densest structures in the universe, making them the best objects for studying the microphysics of dark matter, like annihilation and reaction rates, which depend on phase space number density. Simulations go a long way in helping ascertain the best dark matter models and their properties. Comparing simulations to real data lets us constrain various parameters of these models. While halo finders do an amazing job of finding structures like halos and subhalos in simulations, they fall short when it comes to finding elongated structures like streams, which occupy a distinct phase space region when compared to halos and subhalos. To identify such structures, especially the elongated structures, in simulations, we use data from a Λ CDM zoom-in simulation and implement a non-linear dimension reduction algorithm, namely UMAP. We focus on a $1\text{Mpc } h^{-1}$ box around the MW. We use 6D phase space information of all the particles in the box as our input data. We reduce the 6D information to a 2D representation using UMAP. UMAP separates the largest halos in the box, MW and four massive infalling halos in output space. Within the virial boundary of the MW, particles are segregated based on velocity and dynamics. Infalling streams are separated from the intact core of infalling subhalos. Infalling subhalo particles at their pericentre are separated from the rest of the subhalo. We can use these separations to identify streams and other substructures within the virial boundaries of halos. Which in turn will help us constrain various microphysical properties. This also shows that topological methods like UMAP and GNNs are viable options for data analysis in cosmology and simulations.

Contents

Abstract	xi
1 Introduction	5
1.1 Evidence of Dark Matter	5
1.2 Λ Cold Dark Matter	9
1.3 Structure Formation in Λ CDM	11
1.4 Simulations	12
1.5 The Current State-of-the-Art	14
2 Data & Methods	23
2.1 Data	23
2.2 Selecting Subhalos and Identifying Tidal Disruptions	24
2.3 Machine Learning Algorithms to Identify Substructure	28
3 Results	41
3.1 Preliminary Results	41
3.2 Separation of Dynamic Particles	43
3.3 Effects of UMAP Parameters	45
3.4 DBSCAN & HDBSCAN	48

3.5 Dense Neural Networks	50
3.6 Discussion	52
4 Conclusion	69

List of Figures

1.1	DM - Rotation Curves	7
1.2	DM - Bullet cluster	8
1.3	ROCKSTAR and Consistent Trees	22
2.1	Why <i>our</i> subhalo selection criteria?	26
2.2	Difference between ROCKSTAR and hand-selection of subhalos	27
2.3	Streams	28
2.4	Illustration of DBSCAN algorithm and the difference between KMeans and DBSCAN	31
2.5	Difference between HDBSCAN and DBSCAN	33
2.6	UMAP (theory) - Uniform data	35
2.7	UMAP (theory) - Non-uniform data	35
2.8	UMAP (theory) - Point-wise metrics for non-uniform data	36
2.9	UMAP (theory) - Weighted graphs	36
3.1	Density plot of data in real and phase space alongside halo mass distribution and distribution of subhalos	55
3.2	UMAP 2D Output	56
3.3	Most massive clusters in UMAP space	56

3.4	Virial, real and phase space distribution of the UMAP <i>ellipse</i>	57
3.5	UMAP - separation of different parts of a tidally disrupted/infalling subhalo	58
3.6	UMAP - distribution of fully phase-mixed subhalo particles	59
3.7	UMAP - effect of <code>n_neighoburs</code> parameter	60
3.8	UMAP - effect of <code>min_dist</code> parameter	61
3.9	UMAP - <code>n_componenets = 3</code>	62
3.10	UMAP - <code>n_componenets = 6</code>	62
3.11	UMAP - effect of <code>metric</code> parameter	63
3.12	UMAP - Failure of Canberra Metric	64
3.13	2D iterative DBSCAN	65
3.14	3D iterative DBSCAN	66
3.15	6D iterative DBSCAN	67
3.16	DNN architecture, ROC and F1 score	68

List of Tables

1.1	Early Universe Timeline and Important Events	9
-----	--	---

Chapter 1

Introduction

There is a daunting amount of evidence for the existence of dark matter in the Universe that comes from galaxy rotation curves, gravitational lensing observations and large-scale structures. According to current estimates, dark matter constitutes 85% of matter density and 26% of the energy density of the universe.

1.1 Evidence of Dark Matter

In 1932, when Jan Oort was observing stars in galaxies and their rotation curves, he observed that the stars towards the galaxy's edge were moving at suspiciously high velocities for the amount of visible matter. Therefore, there must be invisible matter providing the gravitational potential required to explain the stellar velocities.

Similarly, in 1933, Fritz Zwicky was observing the Coma Cluster and the velocities of its galaxies on the edge of the cluster[1]. The coma cluster had roughly a thousand galaxies. The cluster was approximately spherical with a radius of $\approx 10^6 \text{ ly}$. The average *stellar* mass of each galaxy was $\approx 10^9 M_{\odot}$, estimated from the mass-luminosity ratio determined from our local neighbourhood. Using this information and the virial theorem, Zwicky estimated a velocity dispersion of $\approx 10^5 \text{ m/s}$, which differed from the observed dispersion of $\approx 10^6 \text{ m/s}$ by a whole order of magnitude. Upon reverse-engineering from the observation, Zwicky found the cluster 400 times more massive than what was visible and concluded that

the bulk of the matter had to be invisible. Thus the label - *dunkle materie* (dark matter)[1].

In the late 1970s, Vera Rubin and Kent Ford observed spiral galaxies and estimated rotation curves[2]. A particularly famous one was the rotation curve of Galaxy M31 shown in Figure 1.1a. The rotation curve of a galaxy (defined for disc galaxies) plots the radial velocity of visible matter (stars or gas) to the radial position from the galactic centre. The velocities are calculated using the observed Doppler shifts. These curves are now a major tool for estimating the mass distribution of the galaxy and understanding its formation and evolution.

A key result from Vera Rubin’s study is that the rotation velocity of M31 remains high at very large radii ($r > 20$ kpc). This was counter-intuitive at that time because the visible matter was concentrated towards the centre. Therefore, larger radii would not experience enough gravitational potential to warrant such high orbital velocities. The notion of unseen matter in and around the galaxies became a necessity. According to Rubin’s calculations, the unseen matter amounted to roughly ten times the mass of the visible matter[2]. This was compelling evidence for Zwicky’s *Dark Matter*. A series of papers in the late 1970s ([3–7]) and early 1980s ([8–10]) confirmed the flat-rotation curves. Works like Carignan et al.[11], shown in Figure 1.1b, demonstrate the high rotation curves beyond the radii Vera Rubin worked with, using sensitive H(I) measurements. This suggests that the stellar mass and gas account for only a small fraction (15%) of the mass in spiral galaxies[12].

More recently, over the past decade, there has been overwhelming evidence for dark matter from gravitational lensing[13] and, most importantly, from the Cosmic Microwave Background[14], whose fluctuation spectrum requires a significant fraction of the energy budget in the universe to be in the form of dark matter.

The path of light bends in the presence of gravity. This underlying physics of gravitational lensing provides an alternate independent estimate of the total mass and its distribution in the universe. Since the bending of light can be explained and predicted using general relativity, the distorted images from galaxy surveys allow us to estimate the mass of various objects in the sky. In 2006, Clowe et al.[15] studied the Bullet Cluster (1E 0657-558) at a redshift of $z = 0.296$, a pair of galaxy clusters that merged ≈ 150 million years ago. Due to the merger, the dissipationless stellar component and the X-ray-emitting plasma component of the galaxies were spatially segregated. Comparison of the distribution of the visible components like the plasma and the stars to the gravitational potential estimated

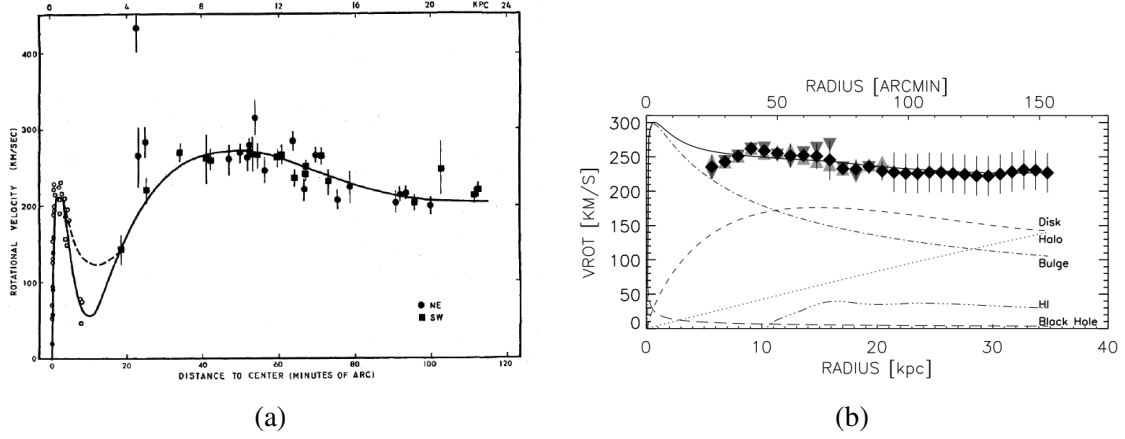


Figure 1.1: Figure 1.1a is the rotation curve of M31 from [2]. Rotational velocities for OB associations in M31 with respect to radial distance. For $r < 12'$, the solid curve is a fifth-order polynomial, and for $r > 12'$, the solid curve is a fourth-order polynomial. The dashed curve near $r = 10'$ shows a second rotation curve with a higher inner minimum. Figure 1.1b is the extended rotation curve of M31 from [11]. The rotation velocities for $R > 21 \text{ kpc}$ come from the Effelsberg and GBT 100-m observations. The velocities for $r \leq 21 \text{ kpc}$ are recomputed from the Unwin (1983) HI data. Light grey upward-pointing triangles show the receding side, while the dark grey downward-pointing triangles show the approaching side as obtained from a tilted-ring model. The solid line is the best fit for the data.

from weak lensing shows a mismatch between the profiles of the potential and that of the visible, baryonic matter. It was concluded that upon collision, the baryonic components heated up while the *invisible* component barely interacted with anything. This acts as a piece of strong evidence for non-baryonic dark matter.

The Cosmic Microwave Background (CMB) is a photon gas permeating the universe. The CMB decoupled from the baryonic matter $\approx 380,000$ years after The Big Bang [16] (redshift $z = 1100$). CMB is also regularly used as a time indicator. From the time protons and neutrons (baryons) came into existence (refer Table 1.1) to the CMB, because of strong Compton scattering, the plasma of photons, protons, and electrons (photon-baryon fluid) shared similar spatial patterns of density. Shortly after recombination, because of the drastic decrease in the number density of free electrons, the photons decoupled from the baryonic matter, and we now observe them as the relics that provide information about the photon-baryon fluid at the *surface of last scattering*. The CMB, albeit perfectly matching a black body, has relatively minute fluctuations in temperature and polarization, which give us information on the spatial distribution of baryons (anything

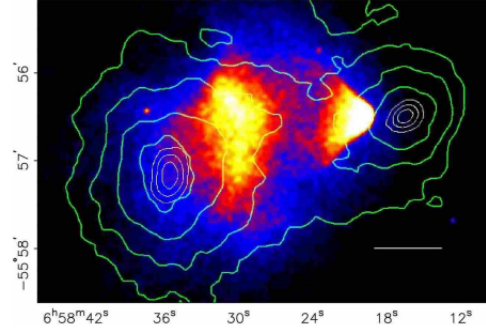


Figure 1.2: This image from [15] shows the gravitation potential as calculated from gravitational lensing and the X-ray emission from the baryon distribution. The white bar indicates 200 kpc at the distance of the cluster. This is a 500 ks Chandra image of the cluster. With an outer contour of $\kappa = 0.16$ and steps of 0.07, green shows κ reconstruction contours for weak lensing. $1-\sigma$, $2-\sigma$, $3-\sigma$ levels in the positional errors of κ are depicted by the white contours.

photons interact with, really) at the CMB.

General relativity predicts spatial fluctuations of matter to grow linearly with the expansion of the universe. For the above-mentioned baryonic distribution at the time of recombination to produce the distribution of galaxies we see today, the fluctuations in the CMB have to be of the order of 10^{-2} , but the CMB is uniform to almost below 10^{-5} [17]. Therefore, the baryonic distribution from the epoch of recombination could not have led to the galaxy distribution we see today.

This conundrum can be solved if we postulate the existence of a large amount of matter that gravitationally interacts with baryons and starts structure formation much before the CMB. This matter will certainly have to be non-baryonic. That way, the baryons and electrons could pass through “dark matter” almost freely before the CMB, leading to the almost uniform spatial distribution in CMB observed today.

Today, we estimate visible matter to occupy $\approx 10 - 20\%$ of the total mass budget of the universe and the rest to be constituted by dark matter.

Universe Timeline		
Event	Temperature (K)	Age of Universe
Inflation	10^{28}	10–34 s
Baryons form	?	?
Dark Matter Decouples	?	?
EW Phase Transition	10^{15}	10^{-11} s
Hadrons Form	10^{12}	10^{-5} s
Neutrinos Decouple	10^{10}	1 s
Big Bang Nucleosynthesis	10^9	200 s
Recombination	3400	260,000 yrs
Photons Decouple (CMB)	2900	380,000 yrs
First stars	50	100 Myr
First Galaxies	20	1 Byr

Table 1.1: The major events and their times in the first 1 billion years of the universe. This table is taken from Daniel Baumann’s Cosmology[18]. The “?” denotes a lack of information.

1.2 Λ Cold Dark Matter

The present, most intriguing questions in the fields of cosmology, astrophysics, and even particle physics surround the nature of dark matter. While we understand today that there must be some form of matter that is “dark”, the exact microphysical nature of dark matter is still unknown. For example, questions of the sort: i) What is the mass of the dark matter particle, ii) what is the nature of its interactions, beyond gravity, with particles of the standard model? iii) What are the means of its production in the universe iv) Are there interactions within in the dark sector or between dark matter particles themselves?

As things stand today, on large scales, a simple, non-relativistic, collisionless model of dark matter called Lambda Cold Dark Matter (Λ CDM) explains almost all observations. Going by the explanation in the CMB part of the previous section, we can infer some constraints that govern our postulated dark matter[19] -

1. **DM has to be non-baryonic** - Since the decoupling of photons from baryons at the epoch of recombination, baryons have not had enough time to form the structures we see today in the universe.
2. **No colour or charge** - Since we have had no direct electromagnetic observation of DM, we can safely say that DM is electrically neutral. Otherwise, it would interact

with photons. The same can be said about having colour and taking part in strong interactions.

3. **Relic abundance should agree with observation, and DM cannot be hot** - If DM were hot (relativistic at decoupling), then all structures below the free streaming length scale would be washed out by Silk Damping[20]. However, the two-point correlation functions of galaxies indicate a large power on small scales.

Λ CDM successfully fits all these conditions and explains our current observations very well except for small-scale problems. We have modifications of Λ CDM that can better explain some of these issues. Some of the most pressing issues with Λ CDM are [21] -

1. **The cusp/core problem (CC)** - Flores & Primack[22] and Moore[23] ruled out the cuspy profile from the rotation curves of David Dunlap Observatory catalogue's (DDO) galaxies. They showed the rotation curve is well-explained by a cored-isothermal density profile. This is in disagreement with the cuspy profiles produced by dissipationless CDM simulations. Although this problem is evident in low-surface brightness galaxies and dwarf galaxies, when it comes to high-surface brightness galaxies, the estimation of density profile nearer the centre of the halo becomes a non-trivial task.
2. **Too Big To Fail (TBTf)** - Simulations predict a larger number of dense and massive subhalos ($> 10^{10} M_{\odot}$) than what is observed. These subhalos are massive enough to have formed larger satellite galaxies than what is observed today. Boylan-Kolchin et al. [24] found six MW-analogue dark matter simulations from the Aquarius Project[25] to predict a population of subhalos that are too massive and dense to host the observed satellite galaxies[26].

Not mentioning the Missing Satellite Problem (MSP) feels like a crime, but MSP is not a problem anymore[27]. The MSP states (stated) - N-body simulations predict a much larger number of satellite galaxies than what is observed. Every CDM simulation of a MW-mass halo predicts $\mathcal{O}(100)$ while we observe a whole order lesser.

Accounting for the detector efficiency of our current devices solves the tension between the luminous satellite count from Λ CDM simulations and that from real data. With the deeper

(long-exposure) images from projects like LSST[28], we have slowly started to observe smaller and fainter satellites, which again contributes to resolving the MSP.

Astrophysics and Cosmology are particularly poised to answer some of these questions. In particular, the mass, production mechanism and interactions among dark matter particles. Interactions of dark matter particles with SM are going to affect the evolution of structure in the universe. Some of the questions that involve interactions among dark matter particles themselves, like annihilation and self-interaction cross-sections, can only be addressed astrophysically.

1.3 Structure Formation in Λ CDM

The formation of structure is driven by two counteracting phenomena – the gravitational collapse of high-density regions and the expansion of the Universe. In Λ CDM, structure formation takes place in a bottom-up fashion. Where smaller structures merge to form bigger structures. Dark matter halos are one of the building blocks of structure formation. Halos are defined as spheres in which the mean matter density is some factor of a reference density, such as the critical density or the matter density of the Universe (e.g., [29]). The choice of this factor is motivated by the simple physics of virialization, which states that if a region encloses ~ 200 times the background density, it is likely to be self-bound and in virial equilibrium, this model is called the spherical collapse model. Other definitions involve splashback radius, the radius where particles reach the apocentre of their first orbit or a fraction of it. There is also an increasing recognition when it comes to definitions based on the dynamic nature of the halos. Garc et al.[30] defines halos as the collection of orbiting particles. This nature becomes apparent in our work as well.

Dark matter halos are some of the densest regions of dark matter in every DM model. Due to the high densities, we can use these regions to put constraints on certain properties of dark matter, like annihilation in the Λ CDM paradigm. Large halos usually comprise a smooth background of diffused particles (at least ~ 200 times its background density) and subhalos that are denser than the diffused background of the host halo. Tidally disrupted subhalos leave behind elongated structures with densities that lie in the range between that of subhalos and host halo. These structures are called tidal streams or simply streams. Iden-

tifying these structures will go a long way in ascertaining the microphysical properties of dark matter. For example, assuming a $\gamma\gamma$ final state, the photon flux from DM annihilation is directly proportional to the square of the local DM density. The subhalos and streams, being the region of the higher densities, will dominate all annihilation signals from large halos.

CDM follows hierarchical structure formation, which means subhalos are constantly merging with the MW. This leaves a number of subhalos fully intact/partially disrupted within the virial radius of the MW at any point in time. A significant fraction of the dark matter mass of a halo is associated with smaller objects like merger remnants, subhalo and streams. Since these smaller objects (substructures, from here on out) are distinctly different from the diffused host halo background particles, they constitute local density fluctuations in phase space. The current algorithms identify substructures by searching for these phase-space density fluctuations in simulations.

1.4 Simulations

In this work, we use a Cosmological, N -body simulation of Cold Dark Matter. A typical N -body method simulates the evolution of N cold dark matter particles in a given volume of space using cosmological initial conditions. Dark matter particles evolve under the collective gravitational field in an expanding universe. Modern simulations have been extremely successful in defining the Universe's large-scale structure and the small-scale physics in the interiors of halos. Simulations like The Millenium Simulation[31] (a Λ CDM simulation) seem to statistically agree very well with surveys like SDSS[32], CfA2[33] and 2dFGRS[34]. Further solidifying the existence of dark matter. Comparison of cosmological simulations to real data helps us analyse and test various predictions from different models of dark matter like Cold Dark Matter (CDM), Warm Dark Matter (WDM), Self Interacting Dark Matter (SIDM), etc.

Cosmological parameters -

1. Flat Universe - As the name suggests, a flat universe has no curvature. κ is set to 0

in the first Friedmann Equation.

$$H^2 = \left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho + \frac{\Lambda}{3} - \frac{\kappa}{R_0^2 a^2} \quad (1.1)$$

Setting κ to be 0 and combining the rest of the terms in the RHS gives us an expression for the critical density

$$H^2 = \frac{8\pi G}{3}\rho_{CR} \quad (1.2)$$

Therefore, critical density is defined as the density required for the Universe to be flat.

2. Ω_M is the ratio of the matter density to the critical density of the Universe. $\Omega_M = 0.286$ in the simulation used in this study.
3. Ω_Λ is the ratio of the dark energy density to the critical density of the Universe. $\Omega_\Lambda = 0.714$ in the simulation used in this study.
4. h is defined as the ratio of H_0 (in $\text{Km}^{-1} \text{Mpc}^{-1}$) to $100 \text{ Km}^{-1} \text{Mpc}^{-1}$. $h = 0.7$ in the simulation used in this study.
5. σ_8 is defined as the r.m.s. density variation when smoothed with a top hat filter of a radius of $8\text{Mpc } h^{-1}$. $\sigma_8 = 0.82$ in the simulation used in this study
6. n_s is the spectral index of the primordial power spectrum. $n_s = 0.96$ in the simulation used in this study

Dark matter-only cosmological simulations are run in cubical boxes of sizes ranging from a few $100\text{Mpc } h^{-1}$ to a few $1000\text{Mpc } h^{-1}$ with the number of particles extending upwards of a few billion. If one were to use atomic-scale masses for particles in a box of $1000\text{Mpc } h^{-1}$, no computational facility in the world could even run simulations. For this reason, the mass resolution (mass of the smallest particle) of these huge-volume simulations is typically $\mathcal{O}(10^7 M_\odot - 10^{10} M_\odot)$. The MW halo mass is $\mathcal{O}(10^{12} M_\odot)$ [35, 36] and the subhalo masses range from $10^6 - 10^{10} M_\odot$. From a typical cosmological simulation, we can identify halos with a few $10s$ to a few $1000s$ particles and label them as the MW. This is not enough resolution to study the small-scale structures of the universe, substructure evolution in halos or microphysical properties like annihilation. Therefore, cosmologists find interesting structures like the MW halo from a typical cosmological simulation, zoom into

the object, and re-simulate them with higher mass and time resolution. These simulations are called zoom-in simulations.

Zoom-in simulations simulate the object with a larger number of particles and take snapshots more frequently than the full cosmological simulation. Snapshots, as the name suggests, are information about the particle positions and velocities at a given time. Zoom-In simulations, they need not span the entire time window of the parent cosmological box; they can span any window of interest. To ensure the physics remains the same, the initial conditions are taken from the full cosmological box at the start of the zoom-in simulation. This allows for studies of the small-scale structures of the universe with finer resolutions.

1.5 The Current State-of-the-Art

Simulations provide a means to test the various dark matter models, properties and theories floating around in the community. However, analysing simulations and extracting usable and useful information is a non-trivial task. In order to test out various dark matter models, we need to be able to compare the results from simulations to real data. Therefore, information like halo mass profile and halo mass distribution (number of halos in each mass bin) can be tested using gravitational lensing data, CMB analysis, rotation curve estimates and large-scale structures of the universe.

All of this can be broken down to the simple task of identifying halos, subhalos and their properties, like mass, radius, position and velocity of the centre. The rest can be calculated. This is precisely what halo finders are designed to do. Early on, most halo finders used only 3D position information to cluster particles, but recently, more algorithms have started using 6D phase-space information. Presently, the popular halo finders either follow a grid system to identify density peaks, search for spherical overdensities or follow the Friends-of-Friends(FoF) algorithm or some modification of FoF.

An example of a halo finder that uses the grid system is AHF[37]. AHF identifies local overdensities in the density field by recursively refining the grid. These peaks act as the centres of prospective halos. The hierarchy of grids is then used to ascertain halo-subhalo relationships. Friends-of-Friends, on the other hand, uses the concept of linking lengths and neighbourhoods to ascertain particles belonging to a cluster. The initial FoF

algorithms used 3D information. The latest modification of the FoF algorithm is called ROCKSTAR, which uses 6D information and implements a hierarchical FoF algorithm with varying linking lengths.

1.5.1 ROCKSTAR

ROCKSTAR is a halo finder developed by Peter Behroozi et al.[38]. ROCKSTAR is based on an adaptive hierarchical refinement of friend-of-friend (FoF) groups in six phase-space dimensions and the time dimension. The design of ROCKSTAR was motivated by the requirement for consistent accuracy across multiple timesteps.

To understand ROCKSTAR, we first need to understand the friend-of-friend algorithm[39]. The algorithm goes from particle to particle, assigning particles to already existing or new groups. The assignment is based on a parameter called linking length (l). If a distance $\leq l$ separates two particles, then the two particles are said to be part of the same group. All particles with a distance of l from a given particle are called *friends*, and all particles that are indirectly connected to the reference particle are called *friend-of-friend*. The size of a group depends on the number of particles that constitute the group. Usually, groups smaller than a threshold are discarded from the final catalogue.

For a quick overview of the algorithm please refer to Figure 1.3a. ROCKSTAR algorithm goes as follows -

1. **Identify overdense regions using a rapid 3D FoF algorithm** - The rapid 3D FoF algorithm is roughly an order of magnitude faster than the traditional FoF explained above. Particles separated by a distance of l are assigned to the same group as friends. For a given particle, if the number of such friends rises above a certain threshold (16 in ROCKSTAR), the neighbour-finding process for the neighbours is skipped. Instead, neighbours for the original particle out to *twice* l are calculated. If there are particles that belong to two different groups then the groups are combined.
2. **Build a hierarchy of FoF subgroups in phase space by progressively and adaptively reducing the 6D linking length** - Starting with the base 3D FoF groups from the previous step, a 6D linking length is calculated using the standard deviation of

particles in position and velocity space. The metric is defined as

$$d(p_1, p_2) = \left(\frac{|\vec{x}_1 - \vec{x}_2|^2}{\sigma_x^2} + \frac{|\vec{v}_1 - \vec{v}_2|^2}{\sigma_v^2} \right)^{1/2} \quad (1.3)$$

Up to 10,000 particles are chosen randomly from a group (based on the group size), and the nearest neighbour distances are calculated. The choice of the 6D linking length is decided such that a tunable fraction f (70% is the ROCKSTAR default value) of particles form groups of at least two (the particle and one other).

For deeper subgroups, the metric is re-evaluated, and a 6D linking length is chosen the same way as before, but with respect to the subgroup particles.

3. Converting the Hierarchical FoF groups into particle membership for halos -

A seed halo is generated for each subgroup at the deepest level. ROCKSTAR then recursively analyses groups in the higher hierarchy levels until all the particles in the original FoF (the 3D FoF) are assigned to halos.

If there is only a single seed halo for a group at a higher level (parent group), then the entire group is assigned to the corresponding halo. In a parent group with multiple seed halos, the particles are assigned to the seed halo that is closest in phase space (6D). Distance between a particle p and seed halo h is given by -

$$d(h, p) = \left(\frac{|\vec{x}_h - \vec{x}_p|^2}{r_{dyn,vir}^2} + \frac{|\vec{v}_h - \vec{v}_p|^2}{\sigma_v^2} \right)^{1/2} \quad (1.4)$$

$$r_{dyn,vir} = v_{max} t_{dyn,vir} = \frac{v_{max}}{\sqrt{\frac{4}{3}\pi G \rho_{vir}}} \quad (1.5)$$

where σ_v is the seed halo's current velocity dispersion, v_{max} is its current maximum circular velocity, and “vir” specifies the virial overdensity. The virial overdensity used in ROCKSTAR is defined using ρ_{vir} from Bryan & Norman (1998)[40], which corresponds to 360 times the background density at $z = 0$.

On a separate note, the reason for using $r_{dyn,vir}$, as opposed to σ_x like in the previous step, is to produce intuitive and stable results. Using σ_x leads to the mis-assignment of particles in the outskirts of the halo to subhalos.

4. Calculate host halo/subhalo relationships among halos. ROCKSTAR incorporates the time dimension to verify these relations -

Until the previous step, ROCK-

STAR calculated everything at the particle level. In order to calculate the relationships at a halo level, ROCKSTAR uses the most basic definition of subhalo, which is a bound halo contained within another, larger halo. ROCKSTAR assigns satellite membership based on phase-space distances. Using Eq. 1.4 as a metric to calculate distances, ROCKSTAR calculates the distance of a halo centre to all other halos with more assigned particles. The satellite halo of interest is then assigned to the nearest *larger* halo within the same 3D FoF group.

If multiple time steps are available, or a halo catalogue from a *earlier* timestep is available, then the subhalo-host halo relationships are modified to remain consistent with the *earlier* timestep.

5. Calculating halo properties and generating merger trees

- (a) Halo positions - ROCKSTAR chooses a set of x particles closest to the density peak (a proxy for the centre), which best minimises the Poisson error ($\sigma_x \sqrt{N}$), to calculate the position of the centre of the halo. For a halo of 10^6 particles, the innermost 10^3 particles are chosen to calculate the position of the halo centre.
- (b) Halo velocities - A fact to note when it comes to the velocity of halo centres is that the halo centres can have substantial velocity offsets from the bulk. Under the assumption that the galaxy hosted by the halo will best track the centre of the halo, particles up to 10% of the virial radius are averaged to estimate the halo centre velocity.
- (c) Halo masses - ROCKSTAR calculates halo densities according to multiple user-specified density thresholds, for example, the virial threshold, density threshold relative to the background or one that is relative to the critical density. All the particles assigned to the halo are used to calculate said overdensity condition.

In this work, we use virial mass.

Halo masses and velocities are calculated after performing an unbinding algorithm. A single pass through the modified Barnes-Hut (original Barnes-Hut algorithm - [41]) method calculates particle potentials.

- (d) V_{cmax} and R_{vir} - V_{cmax} is taken as the maximum of $\sqrt{GM(r)r^{-1}}$. R_{vir} is calculated as the extent to which particles follow the virial threshold overdensity conditions.
- (e) ROCKSTAR also calculates many other halo properties, but the ones mentioned

above are the ones used in this work. You can find the full list of calculated properties in [38]

ROCKSTAR generates particle-based merger trees. A descendant halo is one that has the highest number of common particles to its progenitor in the previous timestep (excluding particles from subhalos). It is recommended to use an advanced merger tree algorithm called Consistent Trees to correct for mistakes inherent to particle-based merger trees.

1.5.2 Consistent Trees

Particle-based halo catalogues and merger trees work well, but they have their own set of consistency issues and, therefore, cannot be used in places that require high-precision halo catalogues and merger trees. The following points summarise the consistency issues in particle-based merger trees -

1. A subhalo passing through its pericentre can easily be confused with the highly dense host halo core. This leads to the subhalo disappearing in one timestep and then reappearing a few timesteps later when it is sufficiently far from the host halo centre. This would generate an entry for a subhalo within the host halo without any progenitors.
2. Subhalos identified close to the host halo centre can cause miss-assignment of the subhalo particles to the host halo particles. This could lead to a record of a merger event when the merger has not yet taken place. This can also cause inconsistencies in the halo properties (halo mass, V_{cmax} , etc.) for both the subhalo and the host halo for multiple time steps until the subhalo is far enough from the host halo to regain all of its miss-assigned particles.
3. The opposite might occur - Some of the particles from a larger halo can be accidentally assigned to a smaller subhalo passing very close to the centre of the larger halo. This can create records of spuriously massive subhalo. This can also lead to duplicate entries of the host halo.
4. Super low-mass halos that are at the identification threshold may appear and disappear in multiple timesteps without descendants and progenitors. This causes records of false mergers and/or a bias against the low-mass halos.

To tackle and overcome the above-stated problems, Consistent Trees[42] makes use of gravitational evolution. Using the knowledge of positions, velocities and mass profiles of halos from one timestep, Consistent Trees predicts its properties in adjacent timesteps using gravity and inertia. Comparing the predictions to actual halo catalogues lets Consistent Trees fill in missing information and correct the record.

In predicting the halo motion between successive timesteps, Consistent Trees assume two things - The position and mass profiles of all dark matter halos are the *only* factors that control the kinematics of halos in the simulation, and individual halo mass distributions are approximated by fitting spherical NFW profiles. While these assumptions might seem drastic and physically wrong, halo motion is tracked accurately between timesteps even with these assumptions.

Consistent Trees is broken into two broad stages. Figure 1.3b illustrates the steps in the first stage of Consistent Trees. The **first stage** of Consistent Trees is very straightforward. The underlying approach for repairing merger trees is the observation of a bottom-up halo formation in Λ CDM. Every halo has at least one progenitor at the previous timestep, even if the halo mass is below the threshold of the halo finder. If tracing a halo back in time to its expected location does not yield a progenitor, then the halo catalogue is said to be incomplete. If going back a few more steps produces a progenitor at the expected location, then the intrinsic properties of the halo are interpolated between timesteps based on the best estimates of the gravitational evolution algorithm. If there are no matches even after going back a few timesteps, the halo is either labelled as spurious and removed from the catalogue or considered to have just formed. Halo properties like V_{cmax} , M_{vir} , R_{vir} and angular momentum are expected to change slowly, and properties like position and velocity are expected to change predictably across timesteps. τ_x , τ_v , (τ_{vmax} are the characteristic errors in predicting position, velocity, and V_{cmax} , respectively. These quantities are used to construct a distance metric that can be used to rank the candidate progenitors. The expected progenitor properties are denoted by e , and the candidate properties are denoted by c . The distance metric is given by -

$$d(e, c) = \sqrt{\frac{|\vec{x}_e - \vec{x}_c|^2}{2\tau_x^2} + \frac{|\vec{v}_e - \vec{v}_c|^2}{2\tau_v^2} + \frac{\log_{10}\left(\frac{v_{\text{max},e}}{v_{\text{max},c}}\right)^2}{2\tau_{v\text{max}}^2}} \quad (1.6)$$

The **second stage** deals with full halo tracks, i.e., the lineage of the most massive progenitors for a given halo. Checking for consistencies in these tracks relates the phase-mass checks from stage one to temporal checks. This allows for removing halos that appear for a few timesteps in the catalogues. Three types of problematic halo tracks are removed -

1. Halos whose lineage of most-massive progenitors contains more than a fraction f_{phant} of phantom halos. Phantom halos are placeholders created at time t_{n-1} for each halo in time t_n with zero progenitors. Halos with a high fraction of phantom progenitors are usually very close to the detection threshold. More massive halos with a high fraction of phantom progenitors are, more likely than not, an invalid detection. For example, subhalos miss-assigned host halo particles when moving close to the centre.
2. Halos, which are tracked for a very short period of time (span of timesteps). For massive halos, the same reasoning can be applied to say they are invalid detections. For smaller halos and earlier redshifts, this condition might remove a few legitimate halos, but the value for $t_{tracked}$ is set after some trial and error, so it doesn't harm the catalogue as much.
3. Subhalos whose tracks do not extend outside the virial radius of the host and are tracked for fewer than $t_{tracked,sub}$ timesteps. These subhalos might seem obviously spurious, but in cases where the interval between timesteps is large, it could be that a subhalo formed outside the virial radius of the host but was detected for the first time within the host.

Here is short summary of all that Consistent Trees accomplishes aside from solving the problems mentioned at the start of this section -

1. Missing halos are completely reconstructed with all of their properties with quantifiable errors.
2. Merger tree links are assigned a natural likelihood estimate. If the particle-based links are unphysical, then the links are cut and reconnected with more plausible candidates.

3. The resolution limit of the simulation becomes explicitly quantifiable in terms of errors induced in the position and velocity of the halos.
4. Distinguishes between tidally disrupted subhalos at the next timestep and subhalos that are lost by halo finders.

Most halo finders, including ROCKSTAR, fail when it comes to picking out particles that belong to a tidal stream. Tidal streams comprise particles that occupy a unique region in phase space. These particles are distinct from the subhalo core that is left behind but they are also different compared to the MW phase-mixed particles. If we were to think of subhalos as bound structures floating around in the MW with the MW phase-mixed particles as their background, then we can think of subhalo cores as really dense spheres, just like a normal halo. Streams can be thought of as dense structures (relative to the MW) comprising loosely bound particles. Therefore, streams become a valuable substructure to study microphysical properties that depend on the velocity distribution and spatial density of particles. Streams might not be as dense as bound subhalos, therefore being a sub-optimal source for constraining microphysical properties of dark matter, but they can act as an independent source to verify, and even further constrain the microphysics of dark matter. In this work, we use Milky Way-size high-resolution halos to identify substructures and segregate particles based on their dynamic activity.

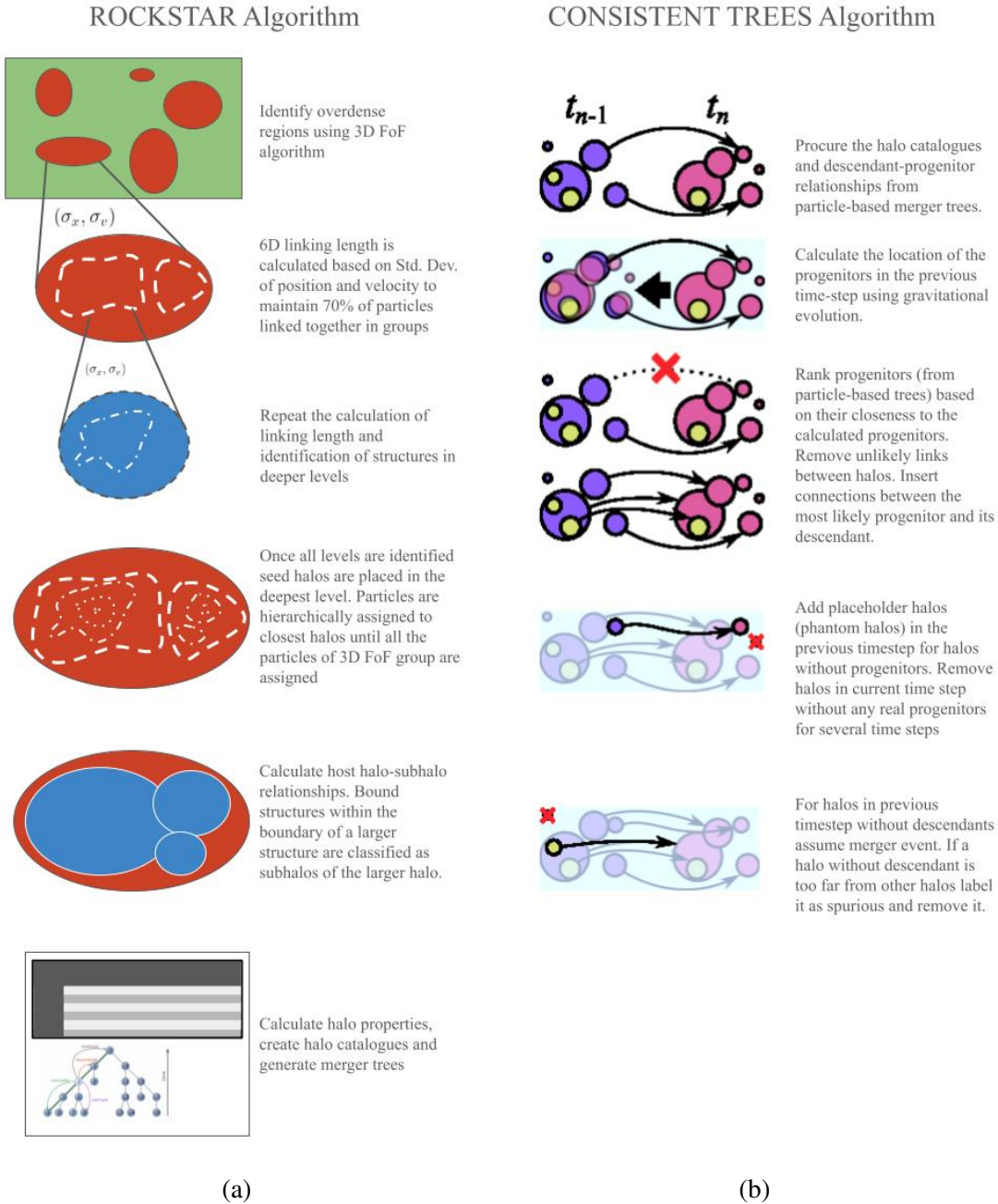


Figure 1.3: Illustration of the ROCKSTAR algorithm and the first stage of the Consistent Trees algorithm. The illustration of the first stage of Consistent Trees is taken from [42]. [Link to go back to ROCKSTAR \(1.5.1\)](#) and [Consistent Trees \(1.5.2\)](#).

Chapter 2

Data & Methods

2.1 Data

This study uses zoom-in simulations of Milky Way-mass halos from the c125-2048 box[43]. c125-2048 box is a dark matter-only cosmological simulation run with L-Gadget (based on Gadget-2[44, 45]). The properties are as follows -

1. Initial conditions are generated by 2LPT_{IC}¹[46] at $z=199$, with the power spectrum generated by \mathcal{C}_{AMB} ²
2. $125\text{Mpc } h^{-1}$ box
3. 2048^3 particles of mass $1.8^7 M_{\odot} h^{-1}$.
4. Softening length is $0.5\text{kpc } h^{-1}$
5. $\Omega_m = 0.286$. Ω_m is the ratio of matter density to the critical density of the Universe.
6. $h = 0.7$. h is defined as the ratio of H_0 (in $\text{Km}^{-1}\text{Mpc}^{-1}$) to $100 \text{ Km}^{-1}\text{Mpc}^{-1}$.
7. $\Omega_{\Lambda} = 0.714$. Ω_{Λ} is the ratio of the dark energy density to the critical density of the Universe.

¹<http://cosmo.nyu.edu/roman/2LPT/>

²<http://camb.info/>

8. $n_s = 0.96$. n_s is the spectral index of the primordial power spectrum.
9. $\sigma_8 = 0.82$. σ_8 is defined as the r.m.s. density variation when smoothed with a top hat filter of a radius of $8\text{Mpc } h^{-1}$.

The zoom-in simulations are selected from the c125-1024 box, which is a low-resolution version of the c125-2048 box. MUSIC code³[47] is used to generate the initial conditions of the zoom-in simulations, which are matched to the cosmological box to the 1024^3 scale. The zoom-in simulation starts from $z = 19$ and consists of 236 snapshots. The lowest mass (highest resolution) in the zoom-in simulation is $\approx 3 \times 10^5 M_\odot h^{-1}$. The softening length in the highest-resolution region is $170\text{pc } h^{-1}$ comoving.

2.2 Selecting Subhalos and Identifying Tidal Disruptions

After running ROCKSTAR and Consistent Trees on the particle data from all timesteps (referred to as snapshots in this section) of the simulation, we identify the most massive halo at redshift $z = 0$, referred to as The Milky Way (MW), from here on. All information about the MW halo is mentioned in Section 3.1.1. To identify the subhalos and other substructures, we track particles belonging to halos that fall into the MW through all the time steps.

2.2.1 Selecting Subhalos

From the halo catalogue at $z = 0$, we identify all the halos up to $2R_{\text{vir,MW}}$. $2R_{\text{vir,MW}}$ is a liberal limit to select any and all halos affected by the tidal forces of the MW. From this list, we follow all the halos back in time to the point where the centres of these halos are separated from the MW centre by a distance equal to $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$. These values change from snapshot to snapshot and from halo to halo as halos grow and shrink depending on their surroundings. The choice of this criterion is motivated by the phase space plot of all the particles in the box (3.1b) and testing multiple different criteria. A good metric to test such criteria would be to see what fraction of the peak mass of the subhalo is captured.

³<https://bitbucket.org/ohahn/music>

This is a good metric because once a subhalo enters the MW, the subhalo stops accreting mass and starts losing mass because of tidal forces. The proof of the decreasing mass for various subhalos is shown in Figure 2.1a. The X-axis is redshift, so time increases from right to left. The vertical lines depict the time at which the selection for the particular subhalo was made according to the criterion mentioned above. One can see that the mass of the halo, as estimated by ROCKSTAR, decreases after the selection. This is because the tidal forces of the MW strip the subhalo particles slowly. Therefore, the validity of various selection criteria can be compared using how close to the peak mass we can select subhalos using them.

Figures 2.1b and 2.1c show the comparison of a few different criteria we studied. As mentioned above, the logic for finding and selecting these subhalos is to follow all the subhalos from $z = 0$ backwards in time till they are separated from the MW centre by some criteria involving $R_{\text{vir,MW}}$ and $R_{\text{vir,sub}}$. Any subhalo that does not go beyond the criteria is removed from the list. This results in lists with varying numbers of subhalos depending on the criteria we are working with. Some numbers that would help the reader understand this - There are a total of **3710** subhalos within $2R_{\text{vir,MW}}$. Number of subhalos lost because they do not meet the criteria upon backtracing -

1. $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$ - 204 subhalos
2. $1.5R_{\text{vir,MW}} + 1.5R_{\text{vir,sub}}$ - 200 subhalos
3. $1.5R_{\text{vir,MW}} + R_{\text{vir,sub}}$ - 200 subhalos
4. $1R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$ - 65 subhalos
5. $1R_{\text{vir,MW}} + 1.5R_{\text{vir,sub}}$ - 62 subhalos

Looking at these numbers, one might think, shouldn't the choice be obviously $1R_{\text{vir,MW}} + 1.5R_{\text{vir,sub}}$, as it retains most of the subhalos. But this is where Figure 2.1b comes in. Figure 2.1b shows that all the criteria work relatively well, but the selected criterion has an edge over the others. The figure shows the fraction of all identified halos (particular to the selection criteria) which follow $\frac{M_{\text{selection}}}{M_{\text{peak}}} > \text{threshold}$. One can see that the criteria with $1R_{\text{vir,MW}}$ perform poorly. These criteria capture the subhalos too close to the MW which results in

the subhalos already being tidally disrupted by the time the selection is performed. So, out of the three $1.5R_{\text{vir,MW}}$ criteria, which one is the best choice? This can be explained using Figure 2.1c, which shows the fractions of subhalos following the $\frac{M_{\text{selection}}}{M_{\text{peak}}}$ condition with respect to the fraction obtained using $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,MW}}$ for the same threshold. Therefore, positive values indicate better performance than $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,MW}}$. $1.5R_{\text{vir,MW}} + 1.5R_{\text{vir,sub}}$ and $1.5R_{\text{vir,MW}} + 1R_{\text{vir,sub}}$ seem to keep up for low values of the threshold, but after a threshold of 0.7 $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$ performs better than both of the other criteria.

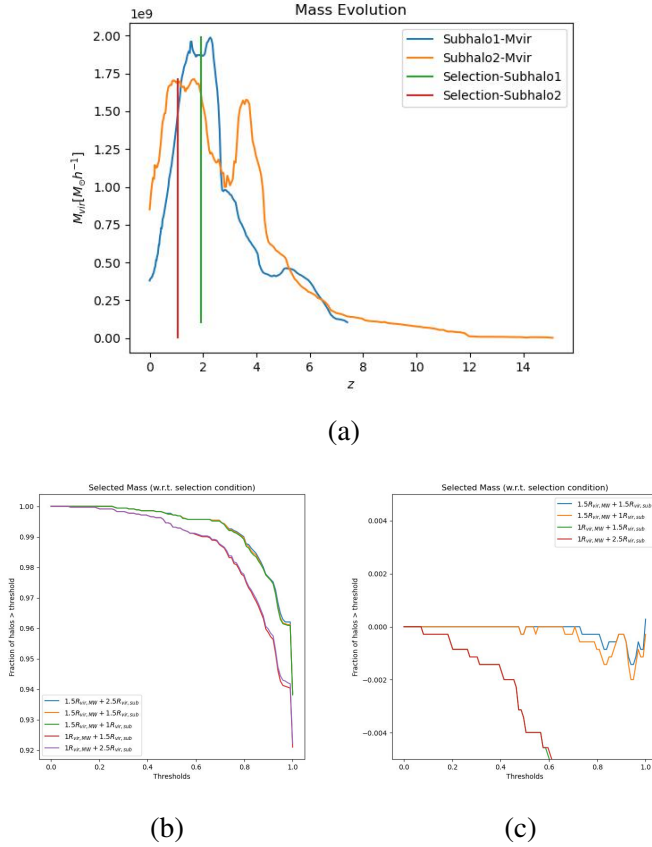


Figure 2.1: Figure 2.1a shows the evolution of the virial mass of two subhalos selected randomly from the same mass bin. The vertical lines corresponding to each subhalo show the time when the subhalo and MW were separated by roughly $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$. Figure 2.1b shows the fraction of subhalos that follow $\frac{M_{\text{selection}}}{M_{\text{peak}}} > \text{threshold}$. There is a bit of nuance to this that can be found in Section 2.2.1. Figure 2.1c shows the same values but with respect to the corresponding value of $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$. If one of the criteria had $y = 0.95$ for a threshold of 0.8 and $1.5R_{\text{vir,MW}} + 2.5R_{\text{vir,sub}}$ had a value of $y_{\text{ref}} = 0.97$ for the same threshold, then we plot $y - y_{\text{ref}}$ as a function of threshold.

2.2.2 Selecting Subhalo Particles

Once we have the information of when a subhalo is just outside the MW (according to the abovementioned condition), we mark all the particles up to $1.5R_{\text{vir,sub}}$ as belonging to that particular subhalo. This selection is again motivated by the phase space plot of the MW in (3.1b). ROCKSTAR has a very strict boundedness condition in place to mark the R_{vir} of halos. we chose a value greater than the R_{vir} measured by ROCKSTAR because the loosely bound particles of an infalling subhalo are tidally ripped away first. The underlying assumption behind selecting all the particles up to a radius of $1.5R_{\text{vir,sub}}$ is that the halo is isolated.

Figure 2.2 shows the selected particles of two subhalos at $z = 0$ in comparison to the corresponding descendant halo particles as estimated by ROCKSTAR. One can see that for a particular subhalo, the two plots have the same skeleton, but the ROCKSTAR one has fewer particles compared to the selection. Since the selection, by construction, only selects particles that are tidally undisturbed by the MW, the *extra* particles in the selection plots have not phase mixed enough to be part of the MW (directly). In short, ROCKSTAR does an “OK” job of identifying streams as well as subhalos, but our goal is to create an algorithm that does it better.

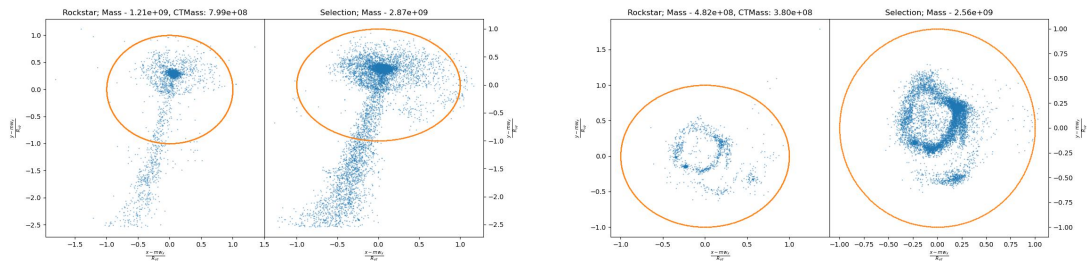


Figure 2.2: The mass estimated by ROCKSTAR and Consistent Trees is mentioned in the title. The ROCKSTAR mass is calculated as the product of the number of particles and the mass of each particle, while the mass from Consistent Trees is the from the halo catalogues after running Consistent Trees. The selection mass is calculated the same way ROCKSTAR mass is calculated. *Left* - shows a subhalo of mass $\mathcal{O}(10^9 M_\odot)$. The left-hand side shows the XY plot of particles assigned to the halo by ROCKSTAR, and the right-hand side shows the particles selected before entering MW ($z=?$) after following them till $z = 0$. The orange circle shows the MW boundary. *Right* - shows the same thing for another halo of mass $\mathcal{O}(10^9 M_\odot)$.

2.2.3 Tidal Disruption

The focus of the project is to be able to identify streams and other elongated substructures given a single snapshot. we have focused on the $z = 0$ snapshot. To identify tidal disruptions in the subhalos that fall into the MW, we follow *the particles* of the subhalo through all the snapshots till $z = 0$. Since no literature quantifies the properties of streams down to the particle level, identifying streams is a very subjective and hand-wavey task. To give the reader an idea of what we consider as *streams* in this work, we have shown a few tidally disrupted subhalos with streams in figure 2.3.

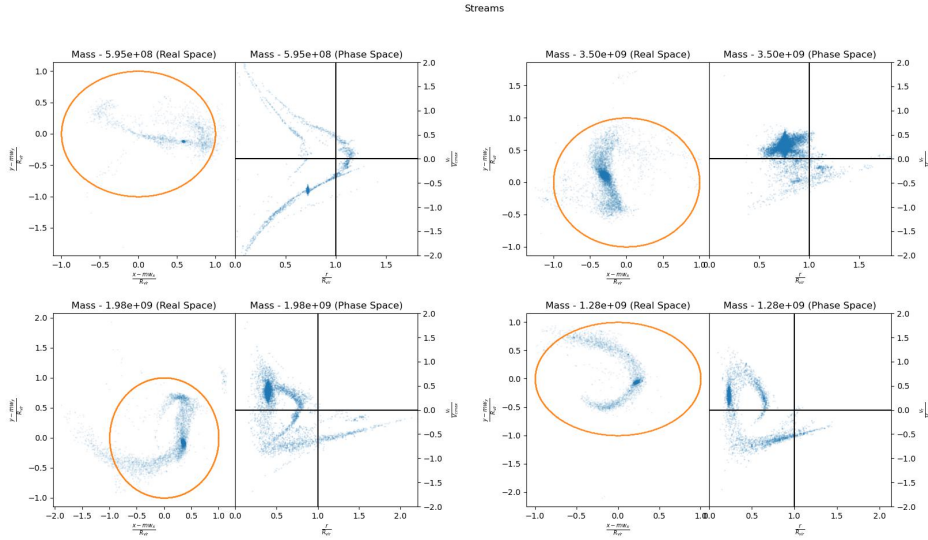


Figure 2.3: A few examples of subhalos that have formed streams at $z = 0$

2.3 Machine Learning Algorithms to Identify Substructure

2.3.1 DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN)[48] is an unsupervised machine-learning technique that uses the notion of *clusters* and *noise*. The underlying assumption for DBSCAN to work is that clusters are dense regions separated by

low-density noise regions. Clustering of such dense regions is dictated by the combination of two parameters set by the user - `eps`, epsilon or radius, and `min_samples`, the minimum number of points to lie in an "eps" sphere around a single point.

This allows for clustering in arbitrary shapes instead of spheres in centroid-based methods like K-means. Figure 2.4b illustrates this point quite well. DBSCAN successfully separates the two clusters, while K-Means fails to separate the points belonging to the two clusters. Figure 2.4a visualises the algorithm. The DBSCAN algorithm -

1. Find all the points within a `eps` (all user-defined parameters will be boldface in this section) radius and identify other points among them that are either core points or points with `min_samples` of points in their `eps` sphere.
2. For each core point, if not assigned to a cluster, assign a new cluster.
3. Recursively find all the density-connected points and assign them to the same cluster.
4. Visit all the points. The points that do not belong to any cluster are labelled as noise.

2.3.2 Iterative Hierarchical DBSCAN

This technique makes use of DBSCAN's density-based clustering algorithm to find clusters or varying densities. Since the main objective of this work is to be able to identify subhalos and streams in simulations, and the densities of subhalo, streams and host halo follow a particular order, we decided to implement this technique to be able to extract different types of substructures.

The density of subhalos is greater than those of their corresponding streams, and the density of the host halo particles is lesser than any streams created from subhalos via tidal disruptions. Therefore, searching for structures of different densities, starting from the highest to the lowest density, should separate out different structures. The algorithm is -

1. Select a `min_samples` value for DBSCAN. In this work, we select the value that matches the peak of the halo mass distribution at $z = 0$.
2. Generate the `min_samples` - *th* nearest neighbour distribution.

3. Select your DBSCAN `eps` value as some multiple ($f > 1$) of the minimum of the nearest neighbour distribution. For this work, we used $f = 1.15$.
4. Perform DBSCAN with selected `eps` and `min_samples`. This will produce a set of clusters of the densest objects. In the initial few runs of this process, these clusters will be subhalos or the inner regions of subhalos.
5. Remove the clustered points from the dataset and repeat from the first step. The later iterations will produce intermediate-density and low-density objects like streams and host halos. At least, that is the idea. You can choose to leave `min_samples` fixed and change `eps` from iteration to iteration.

2.3.3 HDBSCAN

Hierarchical Density-Based Spatial Clustering of Applications and Noise (HDBSCAN)[49] is an unsupervised machine-learning algorithm akin to DBSCAN mentioned in the previous section (2.3.1). Like DBSCAN, HDBSCAN distinguishes between low-density noise regions and high-density clusters and finds clusters of varying shapes. One can think of HDBSCAN as performing DBSCAN over a range of values of `eps` (for a fixed `min_samples`) instead of a single `eps`. The difference is depicted in 2.5 where DBSCAN and HDBSCAN are run using the same parameters on a dataset with clusters of 2 different densities. DBSCAN fails to pick up the low-density cluster, admittedly because of the choice of hyperparameters. Still, HDBSCAN successfully picks out the low-density cluster despite running with the same hyperparameters.

The HDBSCAN algorithm is -

1. **Transform the space according to the density/sparsity** - The k th (`min_cluster_size`; analogous to `min_samples` from DBSCAN) nearest neighbour distances are used to estimate density. For a pair of points, a and b , the distance to their respective k th neighbours is called core distance. Let's denote them as $core_k(a)$ and $core_k(b)$ respectively. The space is transformed under a new metric called mutual reachability distance. Mutual reachability distance is defined as -

$$d_{mreach-k}(a, b) = \max\{core_k(a), core_k(b), d(a, b)\} \quad (2.1)$$

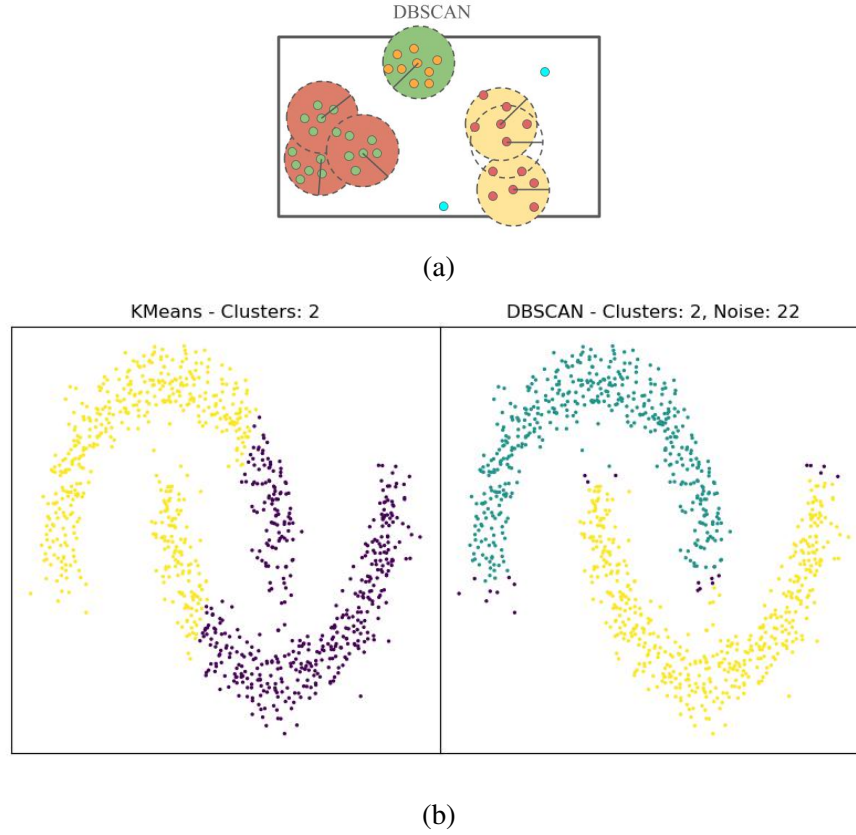


Figure 2.4: Figure 2.4a illustrates the DBSCAN algorithm. The dotted circles are made of the same radius ϵ s. The `min_samples` value used for this demonstration is 6. All points in circles of the same colour belong to a single cluster. The green circle is a singular cluster, the yellow circles are a combination of two individual clusters, and the red circles are a combination of 3 individual clusters. The two cyan-coloured points that do not belong to any cluster are the noise points.

Figure 2.4b shows the difference between KMeans and DBSCAN when it comes to clustering points in shapes that are not an n -dimensional sphere. The data is generated using `make_moon()` from `sci-kit`. *Left* shows the clustering produced by Kmeans for 2 clusters. *Right* shows the clusters identified by DBSCAN.

where $d(a,b)$ is the distance in the metric set by the user.

2. **Construct minimum spanning tree of distance weighted graph** - With the data points as vertices and the mutual reachability distance as the edges of the graph, a minimum spanning tree is generated. High-value edges are dropped, which leads to disconnections, which in turn generate a hierarchy of components at different thresholds. This ensures the preservation of the density structure of the data points.

3. **Construct cluster hierarchy and condense based on minimum cluster size** - Sort the edges in ascending order and then iterate through with decreasing density constraint (edge distance), creating new merged clusters for each new edge. This constructs and merges clusters at different thresholds, i.e., a hierarchy. To condense this hierarchy into clusters, HDBSCAN uses a user-defined parameter, **min_cluster_size**. As you go from low density to high density, clusters "lose points" (edges in terms of the hierarchy) or can split into two or more clusters with multiple points. If, at a particular branch, a cluster loses a point, then the algorithm checks if the remaining cluster has points greater than **min_cluster_size**. If no, then the parent retains the identity of "cluster". If, at a particular branch, a cluster splits into two or more clusters, then to decide if the parent is "the" cluster or if the progenitors are clusters, **min_cluster_size** is used. This is iteratively performed until the entire tree is traversed.
4. **Selecting stable clusters** - Use persistence and stability to identify clusters. To measure persistence and stability HDBSCAN uses λ , where $\lambda = \frac{1}{curr_threshold_distance}$. Persistence is defined as $\lambda_{birth} - \lambda_{death}$, where λ_{birth} is when a cluster splits off and becomes its own cluster and λ_{death} (if any) is when the cluster splits off into smaller clusters. Stability is defined as $\sum_{p \in cluster} (\lambda_p - \lambda_{birth})$, where λ_p is when the point falls out of the cluster.
Clusters with low persistence are discarded, and high persistence is preserved. λ_p gives us a measure of the probability of the point belonging to a cluster.

2.3.4 UMAP

Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP)[50] is a manifold learning technique based on Riemannian geometry and algebraic topology. The core theory of UMAP requires the reader to possess a basic understanding of category theory. Therefore, the description provided here may not bring out the mathematical beauty of the theory to its fullest. UMAP assumes these three conditions to be axiomatically true[50]

1. There exists a manifold on which the data would be uniformly distributed

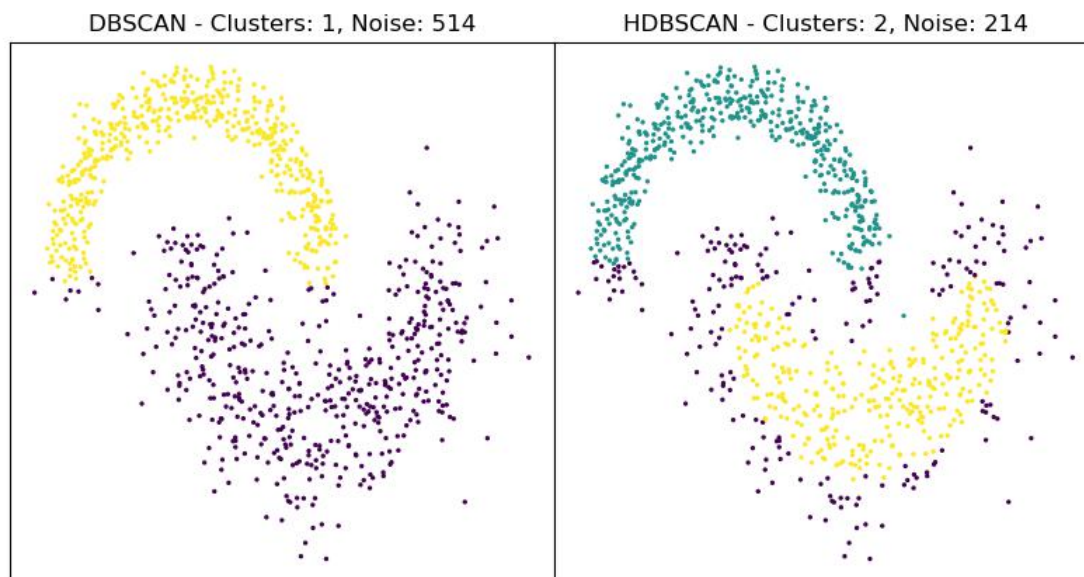


Figure 2.5: The data is generated using `make_moon()` from `sci-kit` for 2 different noise parameters (determines the density of the moons). *Left* shows the result of clustering using DBSCAN with `min_samples = 90` and `eps = 0.32`. *Right* shows the result of clustering using HDBSCAN with the same parameters as DBSCAN.

2. The underlying manifold of interest is locally connected
3. Preserving the topological structure of the manifold is the primary goal

UMAP: Theory

To skip the theory start with the computation part in section 2.3.4. Terms required to understand the theory of UMAP (all of these are abstract terms) -

1. Topological Space - A set X and the collection of its subsets T are said for a topological space if
 - (a) ϕ (empty set) $\in T$
 - (b) $X \in T$
 - (c) $\cap_{i=\{1,2,\dots,n\}} t_i \in T$
 - (d) Union of an arbitrary number of $t_i \in T$

2. Manifolds - Manifolds are topological spaces that are locally Euclidean.
3. Simplices (k -simplex) - The convex hull of $k + 1$ independent points gives a k -simplex. Therefore, a 0-simplex is a point, a 1-simplex is a line, a 2-simplex is a triangle (with three 1-simplex as *faces*) and so on. Simplices are used to build k -dimensional objects.
4. Simplicial Complex - is a set of simplicies glued together along their *faces*. This can be used to construct topological spaces.
5. Simplicial Set - are higher-dimensional generalisation of multigraphs.
6. Open Cover - is a family of sets whose union gives the entire space. In the case of finite data points, we can get an approximation of a true open cover.
7. Čech Complex - is a combinatorial way to convert topological spaces into a simplicial complex. Every set in the open cover of a topological space is converted into a 0-simplex. A 1-simplex is created between pairs of sets that have a non-empty intersection. A 2-simplex is created between three sets if the triple intersection is non-empty, and so on.

Assuming the data we provide is *uniformly* drawn from some topological space, UMAP tries to construct a representation of the said topological space. The first step is to produce a reasonable approximation of a true open cover. If provided with a metric, this can be done by simply making fixed radius balls around each point. This is illustrated in figure 2.6. Considering the intersections of these balls as intersections of sets from the open cover, UMAP implements the Čech Complex. This creates a meaningful representation of the underlying topological space as backed by the Nerve Theorem[51].

In reality, assuming real data is uniformly drawn from some topological space is naive. So, what changes if the data is not uniformly distributed? A fixed radius ball around each point fails to *cover* the entire manifold and capture the properties of the underlying space. This is depicted in figure 2.7. In regions with very few points, there will be too many disconnected 0-simplex, and in regions with a lot of points, there will be high k valued k -simplex (more than ideal). So, what does UMAP do? UMAP assumes the data to be uniformly distributed in *some* manifold and proceeds to ask *what does this tell us about the manifold itself?*

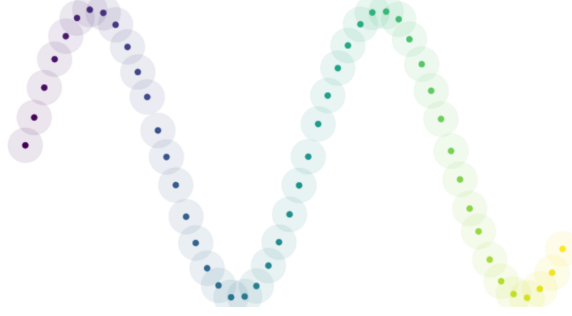


Figure 2.6: Constant radius balls around a uniformly drawn dataset. Image taken from [52].

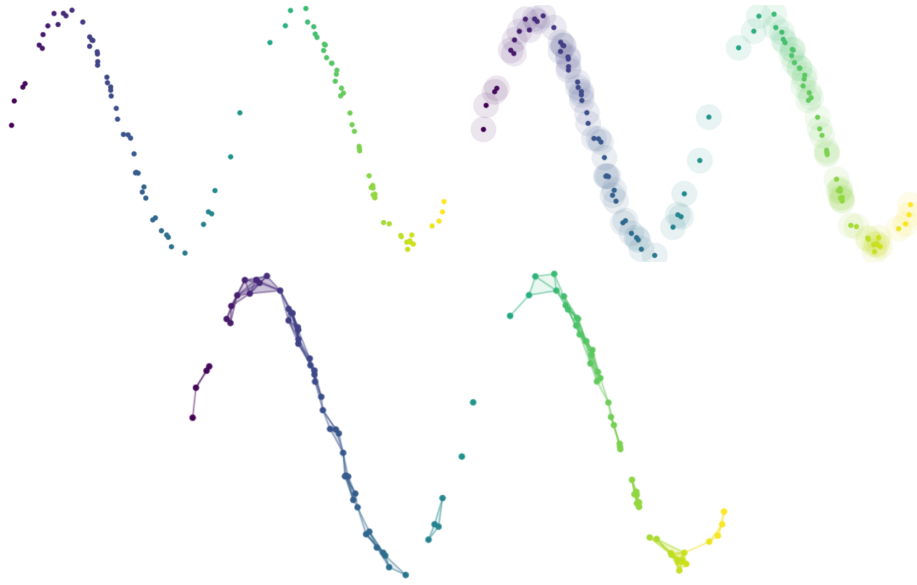


Figure 2.7: Balls of constant radius do not work on non-uniform datasets. Too many high k k -simplex and too many lonely 0-simplex(es). Lack of complete coverage of the space. Images taken from [52].

This is where Riemannian geometry and the user come in. A *unit ball* about a point stretches to the p -th nearest neighbour (not using the conventional term k -NN because k is used to represent the simplices) of the point, where p is the `n_neighoburs` provided by the user. A metric is calculated for each point, and unit balls are created in accordance with the calculated metric. UMAP takes this a step further by redefining the “intersection of sets” from a binary yes-or-no to a weight depending on the separation between the points in the local metric. This can be thought of as working in a *fuzzy topology*. This is illustrated in figure 2.8.

Local metrics solve the coverage problem caused by using fixed radius balls, but this

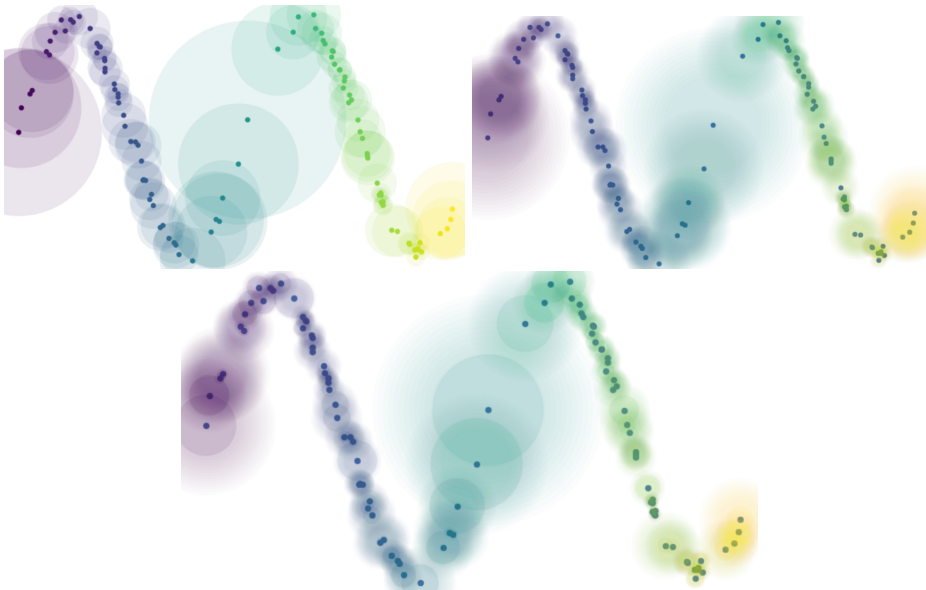


Figure 2.8: Applying Riemannian geometry to estimate *unit balls* that can cover the entire manifold. Changing intersection from binary to continuous values. Depiction of *fuzzy topology*. Images are taken from [52].

brings up a new problem. The local metrics change from point to point. For a given pair of points, the edge connecting them can have different *distance* or *weight* depending on the reference point. To overcome this, for an edge with two weights (depending on the reference points) a and b , UMAP merges the two edges into one with a weight of $a + b - a \cdot b$. The weights are probabilities that an edge (1-simplex) exists, and the combination of weights, in the above fashion, is the probability that *at least* one of the edges exists. Applying this to union all the fuzzy simplicial sets gives a single fuzzy simplicial complex, which is basically just a weighted graph! Phew, that was tedious.

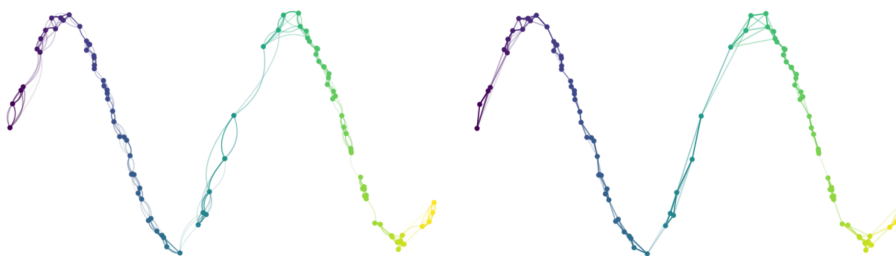


Figure 2.9: Point-wise local metrics cause multiple edges, with different weights, between a given pair of points. Weighting the edges to obtain a single weight for an edge connecting two points. Images taken from [52]

Once the topological representation of the input data is calculated, UMAP assumes a random low-dimensional representation in \mathbb{R}^d (this manifold can be changed). In reality, for computational purposes, the initial representation is not random. Regardless, without the need to estimate a manifold, as in the case of the input, UMAP simply jumps to computing the fuzzy topological representation. Once the representation of both input and target spaces are calculated, UMAP simply minimises the fuzzy set cross-entropy loss between the two representations (considering only the 1-skeleton of the fuzzy simplicial sets). This is better explained from the computational perspective.

In summary, at a high level, UMAP creates a topological representation for the input data using fuzzy simplicial sets of an approximate manifold. Create a low-dimensional representation in \mathbb{R}^d (known manifold) and find its topological representation. Optimisation of the low-dimensional representation is done by simply minimising the fuzzy set cross-entropy.

UMAP: Computational Perspective

From a computational point of view, UMAP simply constructs and manipulates weighted graphs. This puts UMAP in the category of k-neighbour-based graph learning algorithms. UMAP can be broken down into four broad steps.

1. Generate a weighted graph. This will be the source graph.
2. Initialize a low dimensional (target dimension, provided by the user) graph using spectral embedding. A random initialization works in theory, but spectral embedding converges better and faster.
3. Generate a weighted graph for the low-dimensional embedding.
4. Use a force-directed graph layout algorithm to optimize the low-dimensional weighted graph to resemble the source graph as closely as practically possible. This, in a sense, preserves the topology.

Low-Dimensional Graph Construction

Let $X = \{x_1, x_2, \dots, x_N\}$ be the input dataset and d be the metric. Given an input parameter k , for each x_i UMAP computes the set $\{x_{i_1}, x_{i_2}, \dots, x_{i_k}\}$ of the k nearest neighbours of x_i under the metric d . Once this is obtained, UMAP calculates ρ_i and σ_i for each x_i according to the following equations [50]

$$\rho_i = \min\{d(x_i, x_{i_j}) | 1 \leq j \leq k, d(x_i, x_{i_j}) > 0\}$$

$$\sum_{j=1}^k \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right) = \log_2(k)$$

UMAP then defines the weighted graph using $\tilde{G} = (V, E, w)$ where V of \tilde{G} is simply X , $E = \{(x_i, x_{i_j}) | 1 \leq j \leq k, 1 \leq i \leq N\}$ and

$$w((x_i, x_{i_j})) = \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right)$$

This would imply that the edge between two fixed points x_i and x_j would have two different weights for the two directions depending on the distribution of points in the neighbourhood of each of the points. To combine them to form a unified topological representation, let's look at A , the weighted adjacent matrix of \tilde{G} , and consider the symmetric matrix

$$B = A + A^T - A \circ A^T$$

where \circ is the Hadamard (or pointwise) product. Then, a graph G is an undirected weighted graph whose adjacency matrix is given by B .

Graph Optimisation

A force-directed approach uses attractive forces applied along edges and repulsive forces applied on vertices. The attractive force between two vertices i and j at low-dimensional coordinates y_i and y_j , respectively, is determined by

$$\frac{-2ab||y_i - y_j||_2^{2(b-1)}}{1 + ||y_i - y_j||_2^2} w((x_i, x_j))(y_i - y_j)$$

where a and b are hyper-parameters. Repulsive forces are computed via sampling due to computational constraints. Thus, whenever an attractive force is applied between two vertices i and j , one is repulsed from another vertex k , chosen via sampling. The repulsive force is given by

$$\frac{2b}{(\epsilon + \|y_i - y_j\|_2^2)(1 + a\|y_i - y_j\|_2^{2b})}(1 - w((x_i, x_j)))(y_i - y_j)$$

ϵ is a small number (0.001) to avoid division by zero.

These forces are derived gradients optimising the edge-wise cross-entropy between the weighted source graph G and an equivalent weighted graph H generated using $\{y_i\}_{i=1..N}$. Therefore, $\{y_i\}$ is transformed such that the cross entropy loss between H and G is minimum, i.e., topology is conserved in the low-dimension representation.

Implementation

If one wants simply to run UMAP as a dimension reduction algorithm, one can forego understanding the theory behind it and focus on the inputs, outputs and parameters. UMAP offers a set of four basic parameters - `n_neighbors`, `min_dist`, `n_components` and `metric`.

1. `n_neighbors` - decides the scale at which structure is extracted from the input data. Low values of `n_neighbors` look at very small scales, usually at the cost of structure at larger scales, while high values of `n_neighbors` look at global structure at the cost of structure at finer scales.
2. `min_dist` - controls the compactness of clusters in the low-dimensional representation (output). While the clusters are ascertained in the high dimensional data (input) based on `n_neighbors` and `metric`, how packed these clusters are in the low-dimensional representation is decided by `min_dist`. `min_dist` is the minimum distance apart the points ought to be in the low-dimensional representation.
3. `n_components` - is the output dimensions.
4. `metric` - is the metric used to calculate distances between points in both high-dimensional input and low-dimensional output. UMAP offers a variety of pre-defined

metrics and allows users to define their own metrics.

Chapter 3

Results

This section compiles and lays out the observations from all the algorithms and tests we performed to ascertain the best possible way to separate out dynamically distinct objects and particles belonging to said objects.

3.1 Preliminary Results

We will present the preliminary results and argue the choice of input data used for all subsequent techniques. All the preliminary results are from ROCKSTAR and Consistent Trees.

3.1.1 MW Halo Properties

At $z = 0$, the virial radius of the MW halo is $0.22\text{Mpc } h^{-1}$, and the mass contained within the virial radius of the MW halo is $8.741 \times 10^{11} M_{\odot}$. Figure 3.1a shows the XY scatter density plot of all the particles inside our $1\text{Mpc } h^{-1}$ box of interest centred at the MW (referred to as "box" from here on), and Figure 3.1b shows the phase space density plot (V_r vs r) with the MW centre's position and velocity as the reference. The phase space plot shows particles that extend beyond the $1\text{Mpc } h^{-1}$ box to emphasize the infall. Particles with negative V_r are falling towards the centre, while those with positive V_r are moving

away from it. The triangular region (with one side along the y-axis) shows particles that are in orbits. Far outside, one can see that there are only infalling particles. It is clear from the phase space plot that particles are in orbit around the MW centre up to ≈ 1.5 times the predicted virial radius of the MW, as indicated by the positive values of V_r at values of $r > 1$. [30, 53–55] define halo boundaries by the extent of orbits. Figure 3.1c shows the spatial distribution and mass of halos in the box. There are a total of **3711** halos (including MW and its subhalos) ranging from a mass of $5.64 \times 10^5 M_\odot$ to $8.741 \times 10^{11} M_\odot$ (MW) as shown in Figure 3.1d. The peak of the distribution corresponds to $\approx 1 \times 10^7 M_\odot$, or ≈ 30 particles.

3.1.2 UMAP

We run UMAP to reduce the 6D data at $z = 0$ to 2D. The output space will be referred to as *UMAP space* from here on out. We use the normalised 6D information from the MW frame of reference. All the particle positions and velocities are taken relative to the MW centre position and velocity and normalised by the virial radius and maximum circular velocity of the MW. We chose an output dimension (`n_components`) of 2 for all our analyses for ease of visualisation. As you will see in section 3.3.3, higher output dimensions can produce similar results. Since we want our clusters to be as compact and separated from other clusters, we decided to use `min_dist = 0`. Section 3.3.2 elaborates on this choice. For the `n_neighbors` parameter, we chose the value corresponding to the peak of the mass distribution of halos in our box. The peak in figure 3.1d corresponds to ≈ 30 particles since each particle weighs $\approx 3 \times 10^5$. Figure 3.2 shows the scatter plot and the density plot of the resultant UMAP space. The most eye-catching elements of the plot are the big ellipse-like structure (referred to as *ellipse* from here on), the many blobs far away from the ellipse and a plethora of streaks and various other shapes in between the ellipse and the blobs. A note to keep in mind: UMAP is a non-linear dimension reduction algorithm. Therefore, the axes in the UMAP space do not have any physical analogues.

3.2 Separation of Dynamic Particles

3.2.1 The Major Elements in UMAP Space

As mentioned in the previous section, the most eye-catching elements in UMAP space are the big ellipse and the many blobs far away from the ellipse. We could select these elements by hand, but that could introduce some selection bias. The reader and the author may not agree on what should be the boundary of each selection. So, we perform a DBSCAN in UMAP space to obtain a more objective, density-based selection. We run DBSCAN with $\text{eps} = 23$ and $\text{min_samples} = 0.027$. DBSCAN finds **5647** clusters, and $\approx 10.5\%$ of the points are classified as noise. The most massive clusters are shown in Figure 3.3.

From the XY plot, we can see that the ellipse is completely contained in the virial boundary of the MW, and the blobs are outside the boundary. If we take a look at the $V_r - r$ space, we see that the blobs are, in fact, infalling halos, and the ellipse is an almost uniform set of orbiting particles. The proximity of the dark green and the light green blobs in real space and phase space is also picked up by UMAP, evident from the relative positions of the blobs.

3.2.2 Deeper Look at the Ellipse

We have seen that the ellipse in UMAP space is completely within the virial boundary of the MW. The phase space distribution of the particles of the ellipse leads us to believe it is a set of particles in orbit. ROCKSTAR and [57] use a metric to estimate how *relaxed* a halo is. n particles picked randomly from the halo can measure the *relaxedness* (virialisation) based on the fraction of particles that follow $\frac{2KE}{|U|} < 1.35$. We use this same metric to evaluate the dynamics of the particles in the ellipse. Particles that follow the above-mentioned condition are called *virialised particles* from here on out. In Figure 3.4, we have put in selections by hand to split the ellipse into three elliptical shells. The innermost shell (practically a solid ellipse) is 0.35 times the entire ellipse. The second shell extends to 0.7 times the entire ellipse. The third ellipse extends to 1.05 times the ellipse. All the shells are mutually exclusive, i.e., they have no common particles.

From the virialisation plot, we see that the inner regions of the ellipse mainly comprise virialised particles, $\approx 98\%$ of the particles follow $\frac{2KE}{|U|} < 1.35$. As one goes outward to the boundary of the ellipse, the shells contain smaller fractions of virialised particles. $\approx 85\%$ of the particles in the second shell satisfy the virialisation condition, and only $\approx 37\%$ of the third shell satisfy said condition.

Another thing to note is the number of substructures in the different regions. In the XY plot of region 1 (purple), we observe small substructures in the form of solid dark circles. The *darker* diamond-like shapes seen close to the $V_r = 0$ reference line in the $V_r - r$ plot correspond to these substructures. Region 1 particles also have lower V_r values compared to the other two regions, indicated by the clustering of points closer to the $V_r = 0$ reference line and the smaller extent of the V_r axis compared to the other regions. Similarly, looking at regions 2 and 3, we can observe an increase in the fraction of particles with high V_r , region 3 with a higher fraction than region 2. We also see an increase in the number and size of substructures going from Region 1 to 2 to 3.

3.2.3 What else does UMAP do?

To understand how UMAP separates particles and if UMAP is capable of separating out coherent structures like streams, where the current halo finders fail, we decided to look at the distribution of tidally disrupted subhalo particles in UMAP space. The particles belonging to a particular subhalo are selected using the algorithm mentioned in section 2.2.2. Keep in mind that the selection of particles belonging to a subhalo has an error of its own.

Figure 3.5 shows three tidally disrupted subhalos that have not completely phase mixed with the MW at $z = 0$. We run DBSCAN on the particles in UMAP space to illustrate and identify any separation of physically coherent structures. These clusters are identified by the colours. Let's look at these images row by row. Figure 3.5a shows a subhalo of mass $6.35 \times 10^9 M_\odot$. From the clustering in UMAP space, DBSCAN identifies 3 clusters, one inside the ellipse and two outside. The blue cluster (outside the ellipse) is the infall stream, evident from the XY plot and the $V_r - r$ plot. The green cluster (inside the ellipse, close to the boundary) comprises particles that are turning around, again, evident from the XY and $V_r - r$ plots. Therefore, in this case, UMAP distinguishes between particles that are

infalling and those turning around (particles at the pericentre).

Figure 3.5b shows a subhalo of mass $5.3 \times 10^9 M_\odot$. DBSCAN identifies 4 clusters in UMAP space denoted by the colours of the points. The yellow and light green clusters within the ellipse are made of particles turning around (at pericentre), as observed from the XY and $V_r - r$ plots. These particles form what is known as the leading head. The trailing tail, the turquoise cluster, is behind the intact core (blue cluster). In UMAP space, the leading head, trailing tail, and core are well separated. Particles are put inside and outside the ellipse depending on what part of their trajectories they are in.

Figure 3.5c shows a subhalo of mass $3.5 \times 10^9 M_\odot$. DBSCAN identifies 2 clusters in UMAP space, each of which is coloured differently. From the XY plot, we can say that the subhalo is not disrupted and stretched much, unlike the other two subhalos, and correspondingly, from UMAP space, we recover almost all the particles belonging to the subhalo as a single cluster. The same is observed in the $V_r - r$ plot. Therefore, UMAP successfully put a relatively undisturbed subhalo outside the ellipse completely. Is this all? Does UMAP do anything else?

Figure 3.6 shows two subhalos that have almost fully phase-mixed with the MW. We can see this from the triangular distribution in $V_r - r$ space and the lack of high-density spots (relative to the rest of the map) in the XY plot. The UMAP distribution of these particles is also roughly uniform within the ellipse. So, UMAP uniformly distributes phase-mixed subhalos (cannot be classified as subhalos anymore) within the ellipse. The opposite is not true, that the entire ellipse is a set of fully phase-mixed particles. This is shown in Figure 3.4.

3.3 Effects of UMAP Parameters

3.3.1 Nearest Neighbours (`n_neighoburs`)

`n_neighoburs` dictates the resolution at which UMAP probes for structures in the input data. We tested out a range of `n_neighoburs` values for the particles in the box. For values

of `n_neighoburs` above and below, what is shown in this work leads to a memory crash when using 140GB of memory for ≈ 5 million particles. Figure 3.7 shows the 2D UMAP space for fixed `min_dist` and `metric`, but varying `n_neighoburs` values. For small values of `n_neighoburs`, like 5 and 10, the maps look very *grainy*, and there are very few, if not zero, elongated structures. When going from small values of `n_neighoburs` to larger values, we can see the drastic changes in the nature of visible structures. Going from 5 to 10, we can see a higher number of elongated structures. Going from 10 to 30, we see lesser *grains*. Going from 30 to 100 reduces the number of low-density regions (the intensity is a proxy for density), but overall, the graphs look similar. For very high values, like 100, 150 and 200, the maps look almost identical. But regardless of the `n_neighoburs` values, we observe UMAP to separate out the MW from the massive infalling halos.

3.3.2 Minimum Distance (`min_dist`)

`min_dist` dictates how compact the representation of clusters is in the output. `min_dist` does not affect the clustering, but it affects how closely points can be packed in the output. Figure 3.8 shows the variation in the output for three `min_dist` values for three different `n_neighoburs` values. We can observe the same pattern across the different `n_neighoburs` values. Going from a `min_dist` value of 0 to 1 makes the maps more diffuse. This makes it particularly difficult to separate clusters from each other visually or using algorithms like DBSCAN or HDBSCAN. For this reason, for the purpose of this work, `min_dist` is set to 0 to obtain dense and well-separated clusters.

3.3.3 Dimensions (`n_componenets`)

`n_componenets` is the dimension of the output. we have tried running UMAP for 3 different `n_componenets` settings. However, the effects of this parameter in the context of physically coherent structures and substructures are not completely understood. Figure 3.9 shows the output for `n_componenets = 3, n_neighoburs = 30, min_dist = 0`. The large-scale separation of particles into clusters (blobs) corresponding to the infalling massive subhalos and to the *background MW particles* is still observable. Understanding the small-scale features requires some more time. Figure 3.10 shows the output for

$n_components = 6, n_neighbors = 30, min_dist = 0$. The clusters in this space appear closer than those in the other spaces with lower dimensions. Further investigation is required to assess the viability of higher $n_components$ values. Theoretically speaking, the high-dimensional maps (3D and 6D) should have more information stored in the form of independent axes. But nothing can be said without further investigation.

3.3.4 Metric (`metric`)

In simple words, UMAP minimises the cross entropy between the weighted graph of the input data and the graph of the output. The weighting of edges is done using the distance information between points. This is exactly why different metrics lead to different results. The topography of the input data is dependent on the distance metric. In this work, the choice of metric seems to make very little difference, provided the metrics fall under the Minkowski formalism. Figure 3.11 shows exactly this. Metrics like the Manhattan, Chebyshev, or a general Minkowski seem to make little difference to the output. However, non-minkowski-like metrics like Canberra produce drastic changes to the output map. The metrics used for this study are -

1. Canberra metric - $d(\vec{p}, \vec{q}) = \sum_{i=1}^n \frac{|p_i - q_i|}{|p_i| + |q_i|}$
2. Chebyshev metric - $d(\vec{p}, \vec{q}) = \max_i (|p_i - q_i|)$
3. Manhattan metric - $d(\vec{p}, \vec{q}) = \sum_{i=1}^n |p_i - q_i|$
4. Minkowski metric - $d(\vec{p}, \vec{q}) = (\sum_{i=1}^n |p_i - q_i|^k)^{1/k}$ ($k = 2$ in this work)

Whether we choose Minkowski-like metrics or special metrics like the Canberra metric depends on the physics of the problem. Figure 3.3 shows the clusters obtained using the Euclidean metric (Minkowski metric with $k = 2$), and Figure 3.12 shows the 5 most massive clusters from the Canberra map. The solid division between the 4 quadrants of the XY space shows the unphysical nature of the clusters identified via the Canberra metric. This can be attributed to the grid-dependent nature of the metric. This is a strong reason to move away from using metrics like Canberra.

3.4 DBSCAN & HDBSCAN

DBSCAN and HDBSCAN are clustering algorithms that rely on the difference in densities between noisy regions and clusters. In simulations, we have objects of a wide range of densities ranging from very dense subhalos to relatively diffused host halo. Fine-tuning the parameters of DBSCAN and HDBSCAN to find all of these structures in a single run is nearly impossible. It will also change from simulation to simulation or region to region within a single simulation snapshot. Especially with DBSCAN only looking for structures with a density greater than the user-defined density. For this reason, we tried the iterative hierarchical DBSCAN technique mentioned in Section 2.3.2.

We perform the iterative process for different input data, as you will see in the subsequent sections. All data is normalised with the virial radius and maximum circular velocity of the MW.

- For 2D phase space $(V_r - r)$, the values for V_r and r are calculated with respect to the MW centre and normalised over the maximum circular velocity and $R_{\text{vir,MW}}$, respectively.
- For 3D position space (x, y, z) , the particle positions are $(x', y', z') = \frac{\vec{x}_p - \vec{x}_{\text{MW}}}{R_{\text{vir,MW}}}$ where \vec{x}_{MW} is the position of the centre of the MW and $R_{\text{vir,MW}}$ is the virial radius of the MW.
- For 6D phase space (x, y, z, v_x, v_y, v_z) , the particle positions and velocities are modified as $(x', y', z') = \frac{\vec{x}_p - \vec{x}_{\text{MW}}}{R_{\text{vir,MW}}}$ and $(v'_x, v'_y, v'_z) = \frac{\vec{v}_p - \vec{v}_{\text{MW}}}{V_{\text{cmax}}}$ where \vec{v}_{MW} is the velocity of the centre of the MW and V_{cmax} is the maximum circular velocity of the MW.

3.4.1 Phase Space (2D)

Figure 3.13 shows the iterative process applied in the 2D phase space for a fixed `min_samples` value of 30 and for a varying `eps` values calculated as $f \times \min(30 - NN \text{ distribution})$ where $f = 1.15$. Fixing both the fraction f and the `min_samples` is definitely not the most optimal configuration for this technique. However, doing so can still help us gauge the capabilities

of this technique.

Figure 3.13 shows the first 10 iterations of the process. The most noticeable feature when clustering in this space is the circular rings in real space (XY plot). For the initial steps, the particles in phase space are clustered with very small extents in V_r and r . From the later steps, one can see a minutely larger extent in the r space, but overall, the spherical shell structure of clusters in real space is preserved throughout the steps.

3.4.2 Position Space (3D)

Figure 3.14 shows the iterative process applied in 3D position space for a fixed `min_samples` value of 30 and for a varying `eps` values calculated as $f \times \min(30 - NNdistribution)$ where $f = 1.15$. Again, fixing both the fraction f and `min_samples` is definitely not the most efficient approach, but it is a good trial to test the technique.

Figure 3.14 shows iterations 7 through 16 of the process. The initial iterations do not find big structures because of the stringent density requirements. A total of 26,373 (0.52% of the total particles) were identified as part of clusters. Therefore, to make better use of real estate, we show iterations 7 through 16. As shown in Figure 3.14, the algorithm starts picking up small spherical clusters in real space (circular in XY plot). In the later iterations, we see the algorithm pick up the cores of the MW (light green cluster at the centre) and the massive infalling subhalos (purple clusters).

3.4.3 Phase Space (6D)

Figure 3.15 shows the iterative process applied in 6D phase space for a fixed `min_samples` value of 30 and for a varying `eps` values calculated as $f \times \min(30 - NNdistribution)$ where $f = 1.15$. Again, fixing the fraction f and `min_samples` is an inefficient approach but a good preliminary run.

As expected, this input data gives the best result for this iterative DBSCAN technique. From Figure 3.15, we can see that clusters are not always spherical or densely packed in real space. We get large but diffuse clusters in real space, which when compared with their phase space counterparts, turn out to be substructures. We also observe the identification of the massive infalling halos (near the bottom of the XY plot) in multiple iterations, indicating the incomplete identification of the halo in each step. Initially, the algorithm picks up the dense cores as shown by the blue clusters in figures 3.15f and 3.15g (around $(-0.5, -1.5)$ in XY plot). In the subsequent iterations, the algorithm picks up less dense clusters (comprised of different particles) in the same region. However, compared to the previous applications of this iterative technique, we see the identification of more physically sound structures.

3.5 Dense Neural Networks

From the 6D phase space iterative DBSCAN, which required human intervention in each step, Dense Neural Network (aka neural network) is one step closer to automating the entire process. For training and testing, we need labelled and disjoint datasets from the same distribution. The idea is for the binary classifier to identify the underlying relationships among particles of a particular kind (substructure particles) and what separates them from the rest (host halo particles). The dataset used is 6D phase space information taken with respect to the MW centre and normalised over the virial radius and maximum circular velocity of the MW. This is the same as in the case of the 6D phase space DBSCAN technique (refer 3.4).

Training is done on all the particles in the box at $z = 0$. There are a total of $\approx 5 \times 10^6$ particles, which are split into training-validation-test as $\approx 90 : 5 : 5$. To ascertain the robustness of the network, it is tested on a $1\text{Mpc } h^{-1}$ box from around the MW at $z = 0.5$. After numerous experiments with architecture and hyper-parameters, we settled on the following -

1. The architecture is shown in Figure 3.16a. The network has 27,753 parameters, with 12 non-trainable parameters and 27,741 trainable parameters.
2. The box at $z = 0$ has $\approx 5M$ particles split into a training sample of $4.45M$ particles, a validation sample of $270k$ particles and a test sample of $270k$ particles. There are

$\approx 3.71M$ particles belonging to the negative class (not-substructure) and $\approx 1.27M$ particles belonging to the positive class (substructure).

3. The box at $z = 0.5$ has a total of $\approx 3.96M$ particles, with $\approx 2.95M$ particles belonging to the negative class and $\approx 1M$ particles belonging to the positive class.
4. Training batch size is 1000
5. Trained with Early Stopping with a patience of 4 and **min_delta** of 10^{-3} .

The particles are labelled according to Sections 2.2.1 and 2.2.2. All the particles labelled as subhalo particles form the positive class for the network, and all the particles left unlabelled form the negative class of the network (host halo particles). The basic results from training and testing are as follows -

1. Technical results - the network trains for 28 epochs, with each epoch taking 26s.
2. The training and testing ROCs¹ are shown in Figure 3.16c. The performance of the network on particles from the same snapshot as training is remarkable. However, the performance across snapshots is poor. This is depicted by the high value of *area under the curve* (AUC) for the training and testing sets at $z = 0$ and the low AUC value for the testing set from $z = 0.5$.
3. Since there exists a class imbalance in both snapshots and ROCs are insensitive to class imbalances[58], F1 scores² is a better metric for judging the network's performance and deciding what threshold to use. Figure 3.16b shows the F1 score plot for the network performance on the various datasets as a function of threshold. The F1 scores agree with the conclusions from the ROC curves. The network's performance for the dataset from $z = 0.5$, for the threshold where the datasets from $z = 0$ peak, is very poor.

¹Receiver Operating Characteristic curve (ROC curve) is a plot of TPR vs FPR for varying thresholds. The area under the ROC curve is labelled AUC. AUC is 0 for a model with 100% wrong predictions and is 1 for a model with 100% correct predictions. $TPR = \frac{TP}{TP+FN}$ is the true positive rate, and $FPR = \frac{FP}{FP+TN}$ is the false positive rate. TP is the true positives, i.e., positive class objects that are predicted to be positive by the network. TN is the true negatives, i.e., negative class objects that are classified to belong to the negative class. FP is the false positives, i.e., negative class objects predicted to belong to the positive class. FN is false negatives, i.e., positive class objects predicted to belong to the negative class.

² $F1 = 2 \left(\frac{P \times R}{P+R} \right)$ where $P = \frac{TP}{TP+FP}$ is the precision and $R = \frac{TP}{TP+FN}$ is the recall (also known as TPR). An excellent model (with a particular threshold) has an F1 score of 1, while a subpar model has a low F1 score. Explanation for each individual term is provided in the previous footnote ¹

3.6 Discussion

Let us summarize all the results of this work and some of their possible explanations. We will begin with the iterative DBSCAN technique, then move on to DNN and finally to UMAP. We performed an iterative DBSCAN from a bottom-up fashion to identify the densest to least dense objects in 2D phase space ($V_r - r$), 3D position space and 6D phase space. The 2D phase space seemed to find clusters distributed in bins of constant radial distance. The initial steps find the densest regions in $V_r - r$ space, which tend to have a small spread in V_r and r . The small extent in r translates to spherical shells in the real space, as seen in Figure 3.13. This is an artefact of clustering in the 2D phase space. In reality, substructures are unlikely to have the form of spherical shells. Therefore, this technique is unsuitable for separating out subhalos or other spherical substructures, let alone elongated structures like streams which extend into a large number of radial bins.

In the case of 3D position space, the algorithm finds spherical structures like the current halo finders. However, since the input data has only position information, the clusters include both host halo and subhalo particles as long as they occupy volumes that are either overlapping or close enough in real space. There is no distinction between host halo particles and subhalo halo particles. The algorithm fails to identify any elongated substructures like streams. The algorithm is unable to distinguish between structures that have not completely merged but are in the process of merging. Any overlap between distinct structures renders the algorithm unable to distinguish between the structures. In most cases, these structures can be distinguished with velocity information, which was not used in this version. Therefore, while this technique is useful to identify regions of varying density, which can be used to distinguish substructure from the MW, the assignment of particles to said substructures is incorrect. This will lead to inaccurate estimations for any phase space distribution-dependent process like reaction and annihilation rates.

In the case of 6D phase space, the algorithm performs much better than the previous two versions. As the algorithm has information on the velocity distribution as well as spatial distribution, it is capable of separating out two merging objects, unlike the 3D version. One of the biggest problems with this technique is the fact that it is incapable of producing usable results without human intervention in each step. Manual verification of the clusters and what they represent is required in each step. Therefore, instead of making lives easier, this makes it more difficult, which is against the objective of this work. Another thing

to keep in mind is since the protocol is to find the densest structures, remove them from the next step and repeat this cycle, very dense halo cores are identified first instead of the entire halo. This is depicted in figures 3.15f, 3.15g, 3.15h and 3.15i, where the infalling massive halos (located around $(-0.5, -1.5)$ in XY plot) are picked up in multiple iterations starting from their dense cores. Therefore, this protocol, by construction, is sub-optimal. An improvement to this is what is implemented in ROCKSTAR, where one finds the largest structures first and then goes down to deeper and deeper levels.

Another failed technique was the use of DNNs to classify particles as belonging to the MW phase-mixed set of particles or subhalo particles. The subhalo particles could be tidally disrupted streams of intact cores. This network (figure 3.5) was trained on the particles at $z = 0$ and tested on the particles at $z = 0.5$. The performance on the test set from $z = 0$ was incredible, but that on the $z = 0.5$ set was sub-par. In conclusion, the trained DNN managed to separate out particles belonging to substructures in the same snapshot it was trained but failed to match the same performance in a different snapshot (within the same simulation). This leads us to believe the network was not learning the underlying distributions of substructure particles in phase space, or the difference in the phase space regions occupied by substructure particles and host halo particles. The network only has two classes, while in reality, there could be n number of physically motivated classes of particles. This could be a cause for the poor performance. It could also be that we failed to provide all relevant information to the network in the context of the physics of the simulation. Maybe DNNs just cannot be used for this task. Plenty of explanations and workarounds to explore.

These failures brought us to UMAP, a non-linear dimension reduction algorithm. UMAP is run using `n_neighbors` equal to roughly the peak of the halo mass distribution obtained from ROCKSTAR and Consistent Trees. At first glance (figure 3.2), UMAP seems to separate out different halos. Notice how it says halos and not subhalos. This is because separating the subhalos using UMAP is not as straightforward. The 2D UMAP plot comfortably separates out tidally undisturbed massive infalling halos from the MW (figure 3.3). Upon further investigation, one finds that the biggest feature of the UMAP plot, the ellipse, is in itself a segregation plot where the particles are separated into different elliptical shells based on their virialisation status (figure 3.4). Upon careful inspection of the various $V_r - r$ plots for the different elliptical shells, we can see a pattern. The innermost shell mostly

consists of low V_r particles as indicated by the smaller extent of the V_r axis and the packing of points close to the $V_r = 0$ reference line. The various elliptical shells also have different numbers of substructures. Therefore, the segregation of particles is not simply based on their virial status but also on their V_r values and whether or not the particles are part of some substructure. This is good! We are able to distinguish between massive structures and also segregate the MW particles based on their dynamics. Can we make this better? Can we learn something about the huge area of streaks and slashes? Figure 3.5 tells us that the *region of streaks* is the house for *parts* of many tidally disrupted subhalos, infalling streams, infalling cores, etc. Infalling particles that are very close to their pericentre are put within the ellipse, while infalling particles that have not crossed their pericentres are put outside the ellipse. The intact core of the infalling subhalo is a single dense cluster in UMAP space put outside the ellipse. In conclusion, UMAP separates out particles that are infalling and turning around (at pericentre). In some cases, UMAP is capable of separating out the leading head, trailing tail and the intact core of a tidally disrupted subhalo (figure 3.5b). UMAP also distributes completely phase-mixed subhalos (not subhalo anymore) uniformly in the ellipse (based on their current V_r values), as shown in figure 3.6.

So, are we able to identify streams? Not quite. Are we able to separate out particles with different dynamics? Very much so. We need to better understand what UMAP is doing in order to manipulate it to give us the results we want. Maybe a different loss function could give us better results? Maybe we need to pre-process our data differently? Maybe the higher dimensions for outputs could solve our problem (figures 3.9,3.10)?

Answering the question in the introduction section of this work - can we precisely find elongated structures or separate out particles based on their dynamics, will go a long way in helping us ascertain the microphysics of dark matter. Since the goal is to find *a* method to do so, Maybe UMAP might not be the answer to the question we are asking. Once we establish the capabilities of UMAP as a structure finder we can explore other types of ML algorithms like GNNs to solve this problem. There are a lot of questions to be answered and lots of paths to be explored. But regardless of what the next step is, UMAP, a topological algorithm, definitely manages to study the dynamics of the system to some extent. That should be the main takeaway from this work.

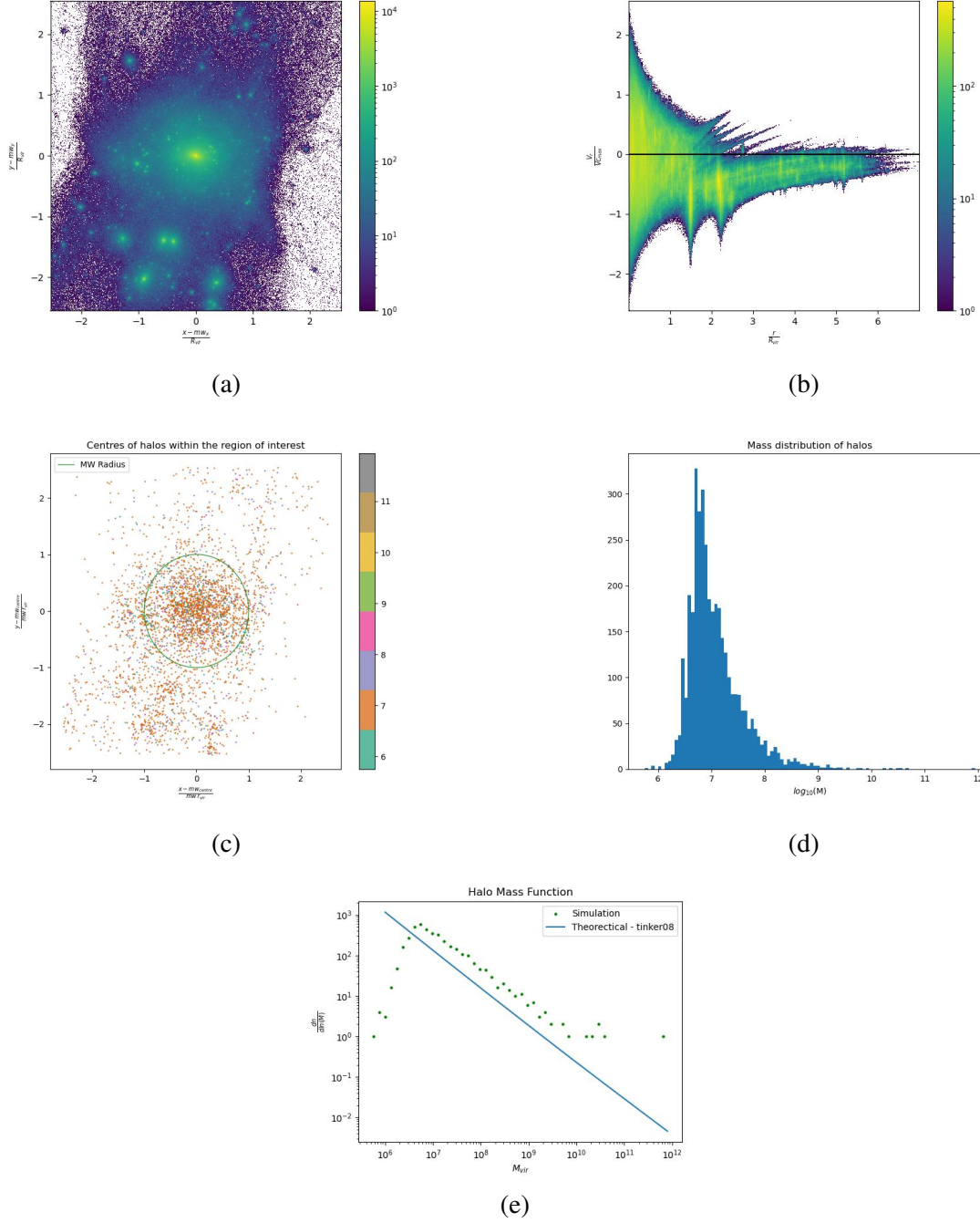


Figure 3.1: Figure 3.1a shows the distribution of all the particles at $z = 0$ in the box. The positions of the particles are taken relative to the centre of the MW (as estimated by ROCKSTAR) and normalized with the virial radius of MW ($R_{\text{vir},\text{MW}}$). The colours depict the density. Figure 3.1b shows the $V_r - r$ plot of the particles that extend outside the box (for visualisation purposes). V_r and r are estimated using the MW centre position and velocity. The colours depict the density. Figure 3.1c shows the positions of all the halos with respect to the MW centre and normalised over the virial radius of the MW. The colours represent the mass of the halo as powers of 10. The solid line shows the virial boundary of the MW. Figure 3.1d shows the mass distribution of the halos in the box. Figure 3.1e shows the halo mass function calculated from the simulation and the theoretical halo mass function calculated by Tinker et al.[56] for the same mass range.

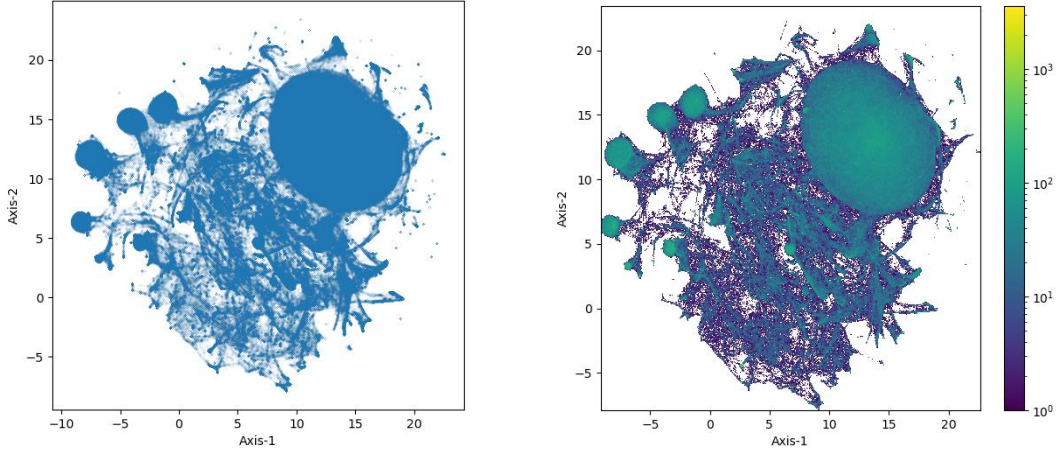


Figure 3.2: UMAP generated from $1\text{Mpc } h^{-1}$ centred at MW from $z = 0$ with UMAP parameters $n_neighbors = 30, min_dist = 0, n_components = 2, metric = euclidean$. The input data is the positions and velocities of all the particles relative to the MW centre normalised over the virial radius (for position dimensions) and maximum circular velocity (for velocity dimensions). *Left* - shows the scatter plot in UMAP space. *Right* - shows the density plot in UMAP space.

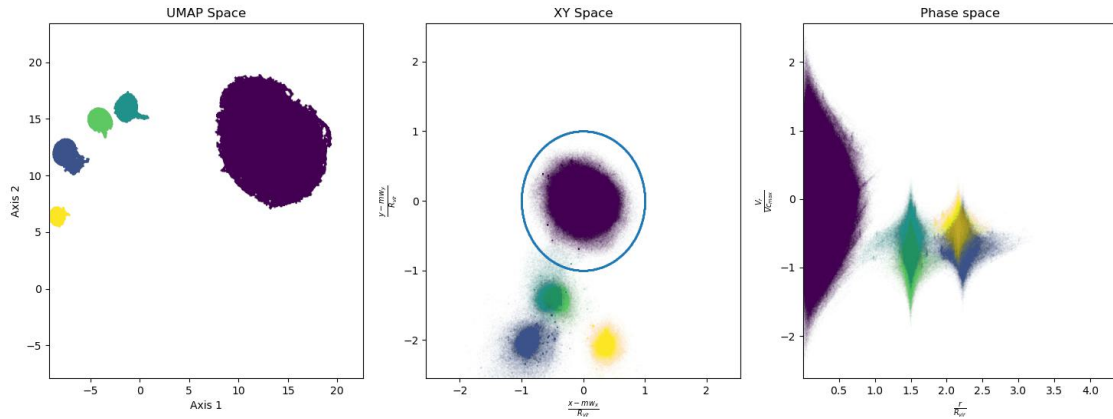


Figure 3.3: Most massive clusters from DBSCAN when run in UMAP space with $\approx 10.5\%$ particles classified as noise. DBSCAN finds 5647 clusters. *Left* - shows the clusters in UMAP space. *Centre* - shows the corresponding particles in real space (XY). The blue circle shows the virial boundary of the MW. *Right* - Shows the phase space ($V_r - r$) distribution of the particles. Colours remain constant throughout the plots.

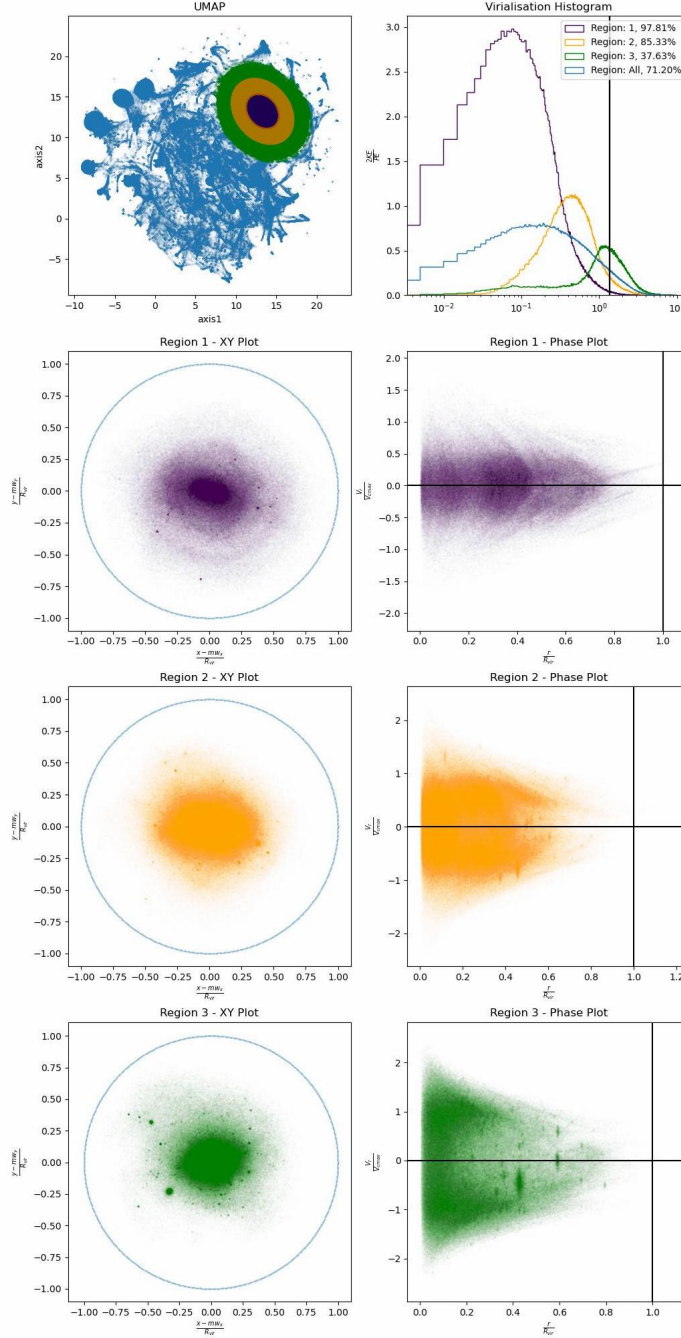


Figure 3.4: Three elliptical shells spanning the entire ellipse in UMAP space. The real space (XY) and phase space ($V_r - r$) plots of the particles belonging to each shell are shown in rows 2 to 4. The blue circle in the XY plot is the virial boundary of the MW. The black lines in the $V_r - r$ are the $V_r = 0$ and $R_{\text{vir,MW}}$ reference lines. The virialisation histogram in the top right shows the distribution of $\frac{2KE}{|U|}$ values of the particles belonging to the different regions. The vertical line in the virialisation plot is a reference line for 1.35 (condition used in ROCKSTAR[38] and estimate by [57]). The % values in the legend on the plot show the number of particles that follow $\frac{2KE}{|U|} < 1.35$.

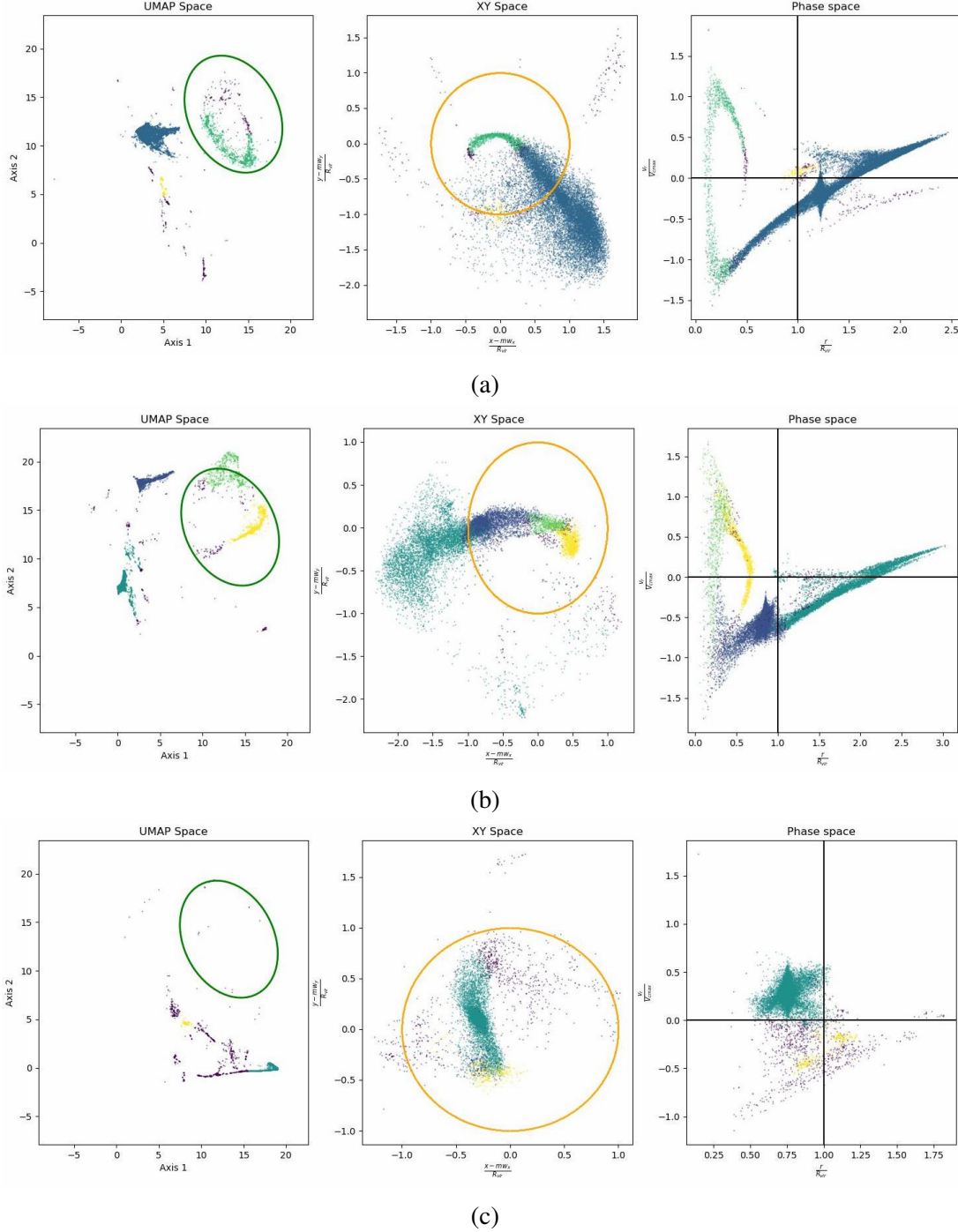


Figure 3.5: Three different subhalos at $z = 0$ in UMAP space, real space (XY) and phase space ($V_r - r$) space. The colours within a row are constant for all three plots. The colours are obtained using DBSCAN clustering in UMAP space. The green ellipse in the left column is the boundary of the ellipse in UMAP space (hand-selected for reference purposes). The orange circle in the centre column is the virial boundary of the MW. The black lines in the right column are the $V_r = 0$ and $r = 0$ reference lines.

Figure 3.5a shows a subhalo of mass $6.35 \times 10^9 M_\odot$. DBSCAN identified three clusters with 1.8% of the particles classified as noise (violet-coloured points). Figure 3.5b shows a subhalo of mass $5.3 \times 10^9 M_\odot$. DBSCAN identified 6 clusters with 2% of the particles classified as noise (violet-coloured points). Figure 3.5c shows a subhalo of mass $3.5 \times 10^9 M_\odot$. DBSCAN identified 2 clusters with 8.2% particles classified as noise (violet-coloured points).

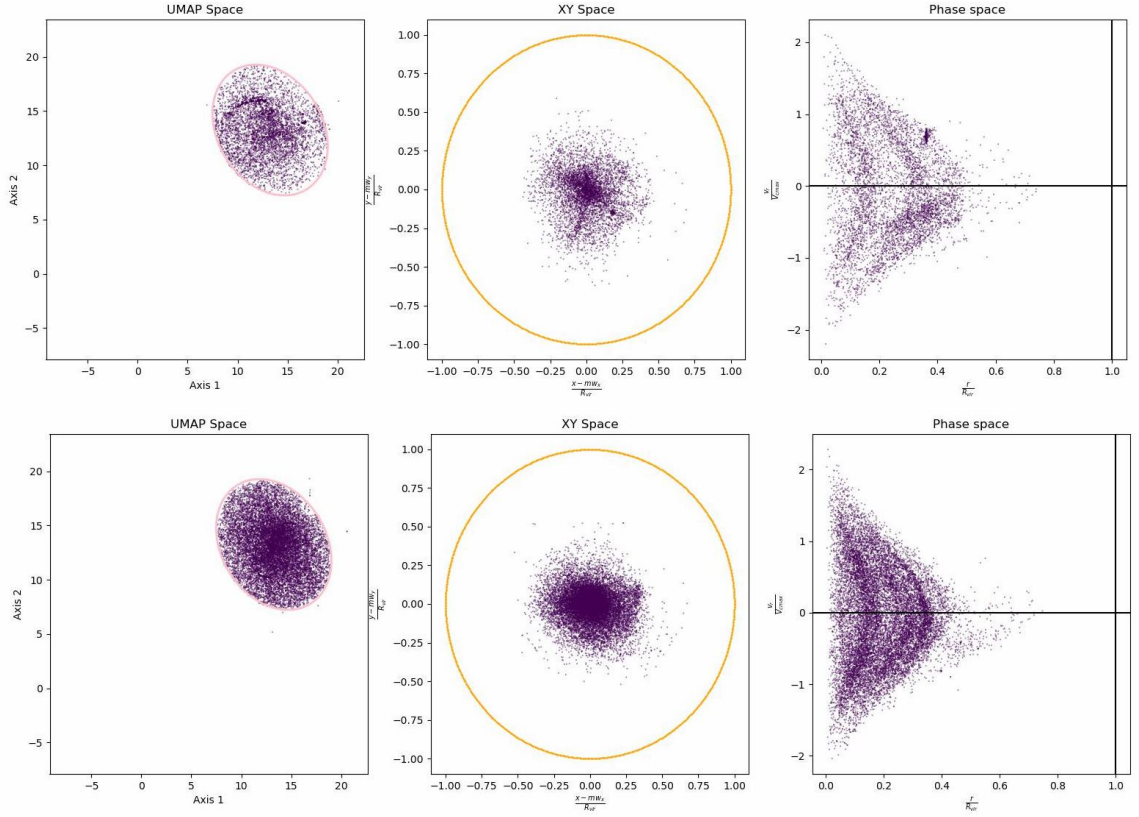


Figure 3.6: Two subhalos that have almost completely phase mixed. The left column shows the UMAP space distribution. The pink ellipse is a reference boundary put in place by hand. The orange curve in the centre column shows the virial boundary of the MW. The black lines in the right column show the $V_r = 0$ and $r = 0$ reference lines. *Top* - a subhalo of mass $1.6 \times 10^9 M_\odot$. *Bottom* - a subhalo of mass $4.3 \times 10^9 M_\odot$.

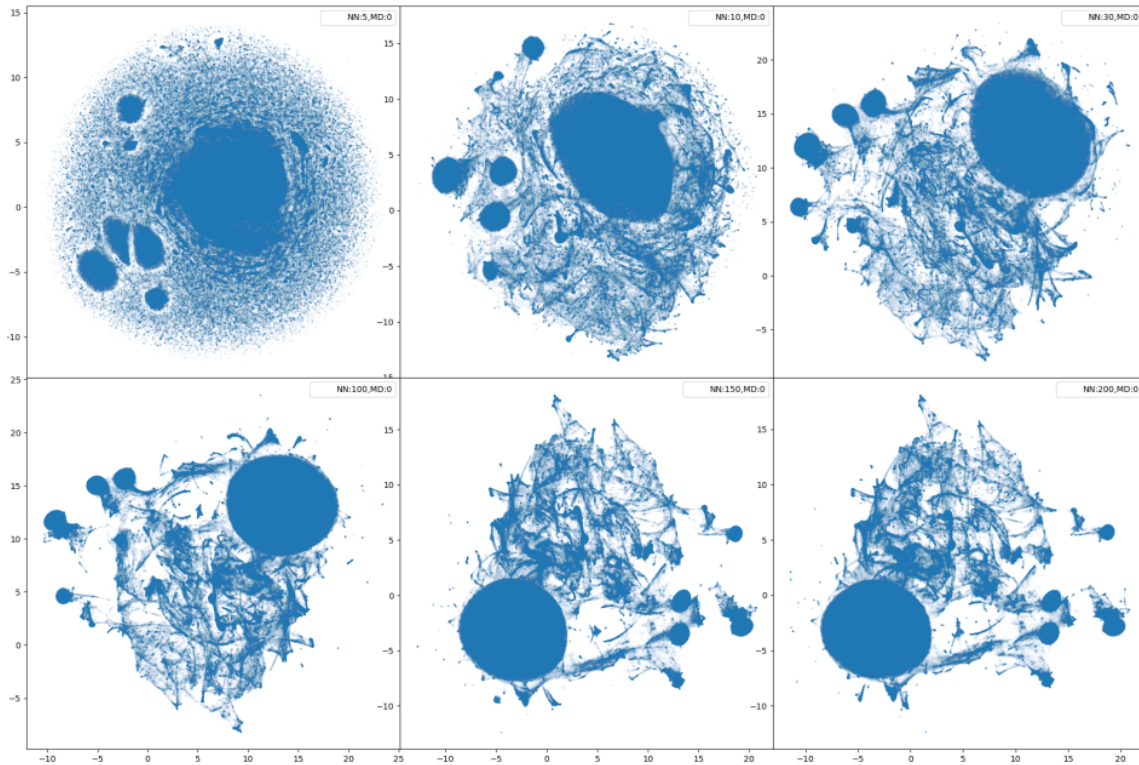
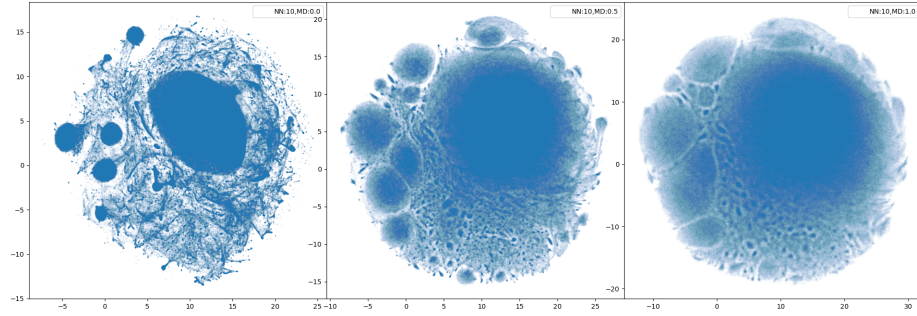
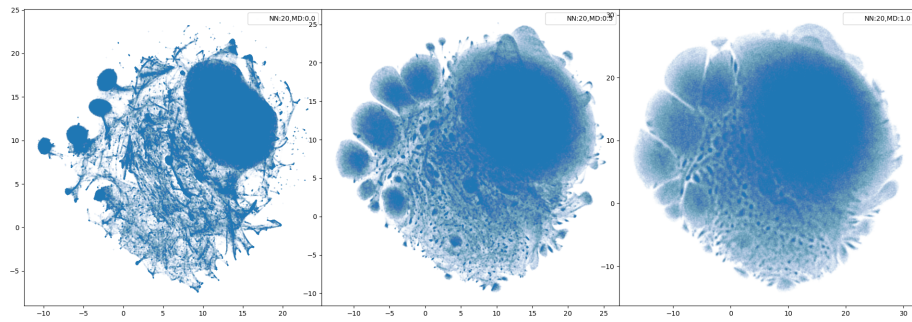


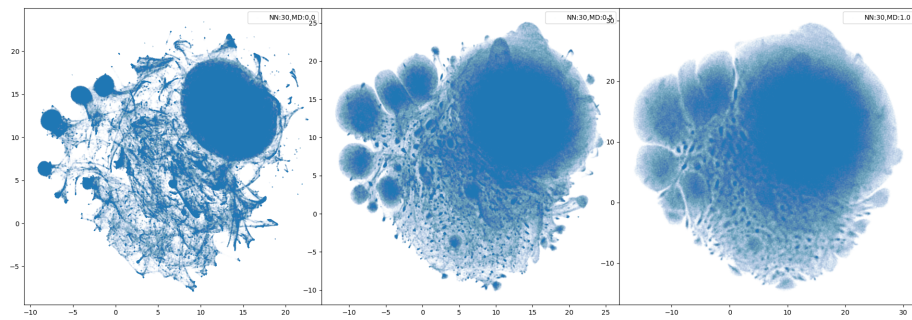
Figure 3.7: 2D UMAP-reduced maps generated from 6D phase space information. All maps were generated using $min_dist = 0$ and $metric = \text{Euclidean}$. $n_neighbors$ values of 5, 10, 30, 100, 150, 200 are used. The orientation of the maps may differ due to the stochastic nature of the algorithm.



(a)



(b)



(c)

Figure 3.8: Figure 3.8a shows three 2D UMAP-reduced maps generated using $n_neighbors = 10$ and $min_dist = 0, 0.5, 1$ respectively. Figure 3.8b shows maps for the same min_dist values for a $n_neighbors$ value of 20. Figure 3.8c shows the same for a $n_neighbors$ value of 30. All maps use the Euclidean metric.

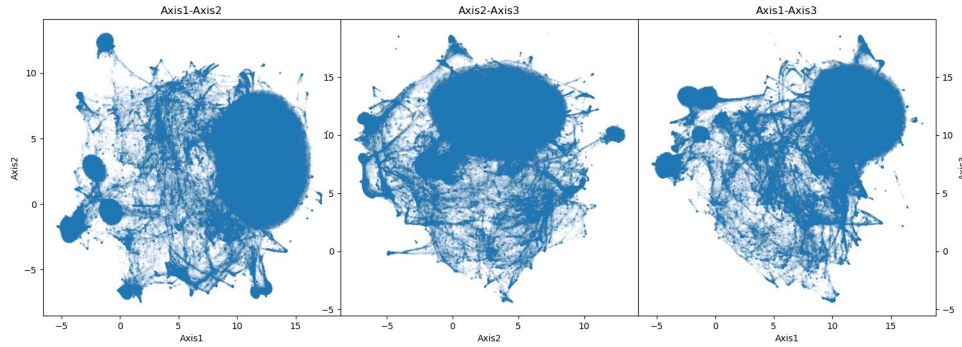


Figure 3.9: UMAP generated with $n_neighbors = 30, min_dist = 0, n_components = 3$. *Left* shows the first two axes, *centre* shows the second and third axes and *right* shows the first and third axes.

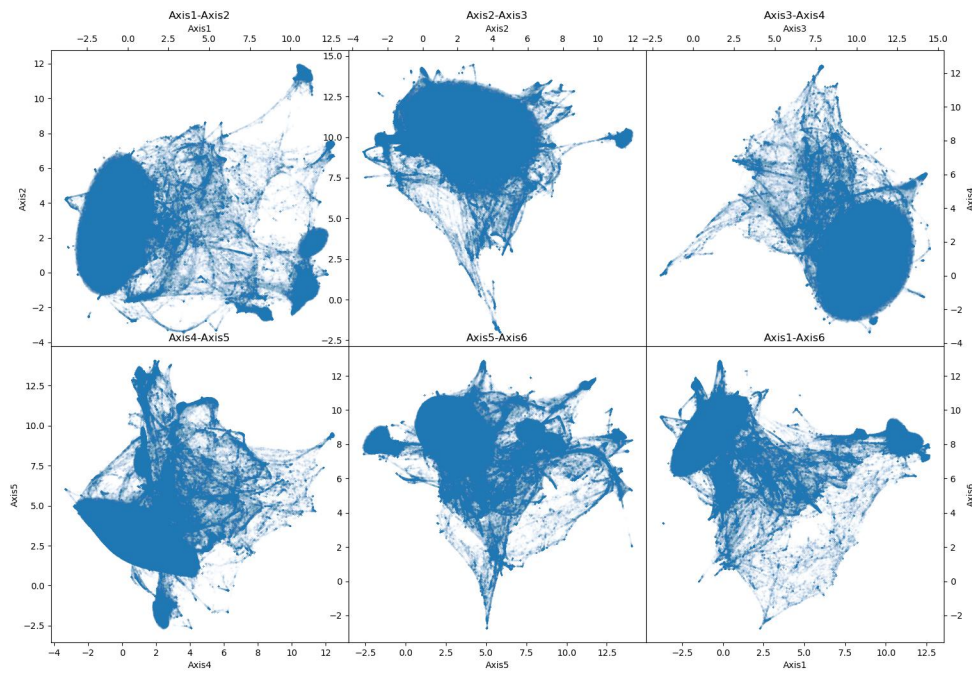


Figure 3.10: UMAP generated with $n_neighbors = 30, min_dist = 0, n_components = 6$. *Top row* shows the first 4 axes and *Bottom row* shows the rest of the axes.

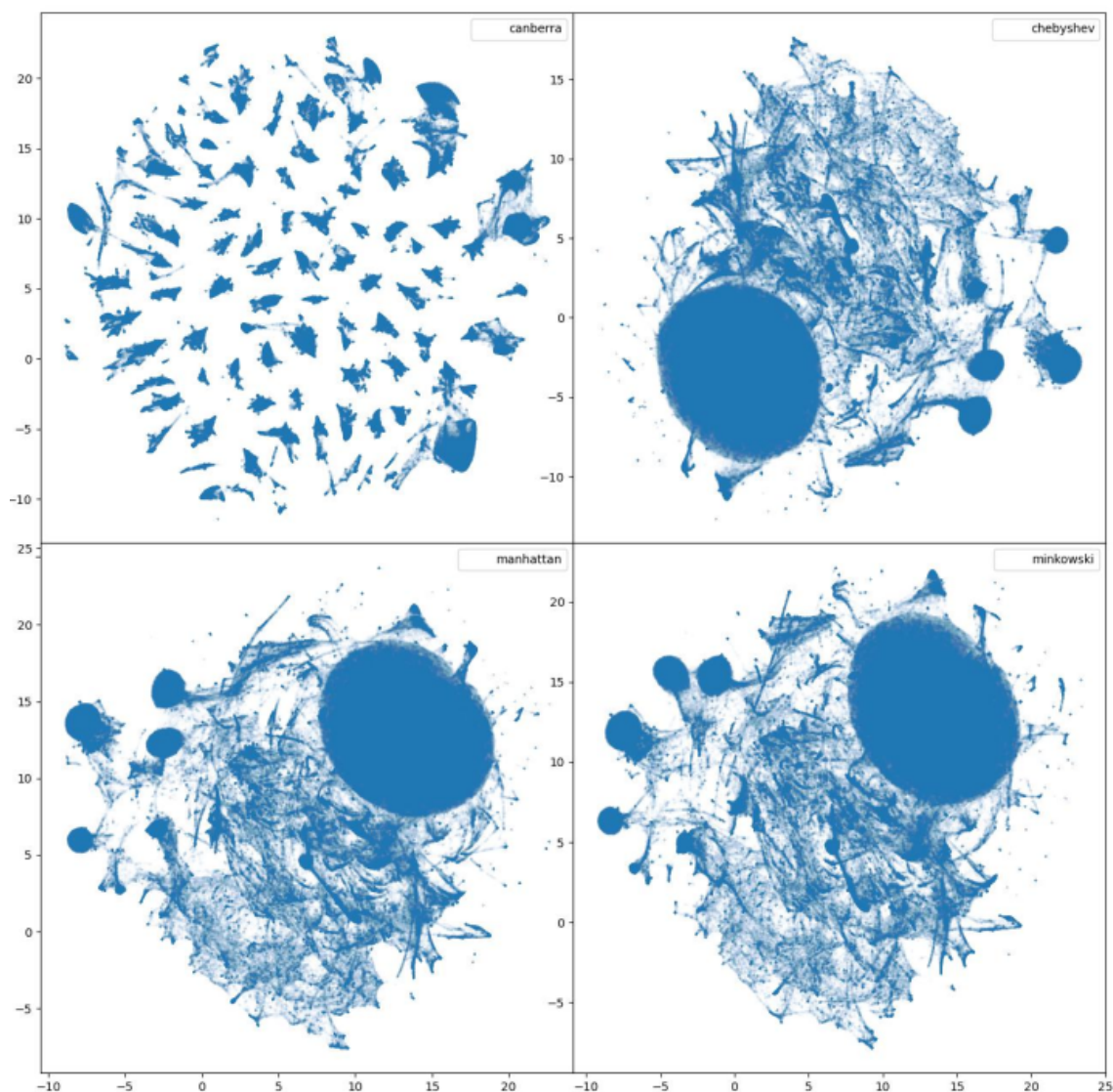


Figure 3.11: Maps for $n_neighoburs$ value of 30 and min_dist value of 0 for various distance metrics. Canberra metric $d(\vec{p}, \vec{q}) = \sum_{i=1}^n \frac{|p_i - q_i|}{|p_i| + |q_i|}$. Chebyshev metric $d(\vec{p}, \vec{q}) = \max_i (|p_i - q_i|)$. Manhattan metric $d(\vec{p}, \vec{q}) = \sum_{i=1}^n |p_i - q_i|$. Minkowski metric $d(\vec{p}, \vec{q}) = (\sum_{i=1}^n (-1)^{i+1} |p_i - q_i|^k)^{1/k}$ for $k = 2$. The orientation of the maps differs due to the stochastic nature of the algorithm.

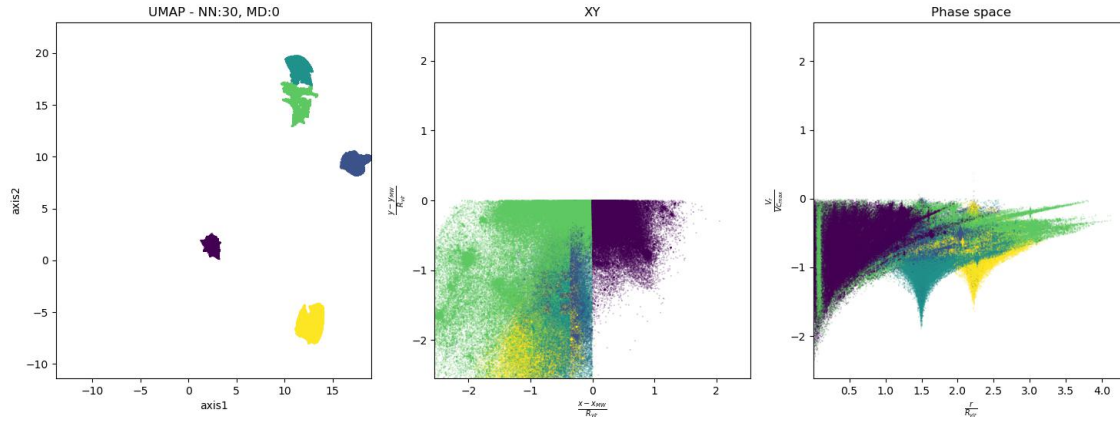


Figure 3.12: The 5 most massive clusters identified by DBSCAN. The left plot shows the location in the UMAP-reduced map. The centre plot shows the same clusters (identified by colours) in the XY plane (taken with respect to the MW centre and normalised with $R_{\text{vir,MW}}$). The solid division between the negative and positive y values shows the unphysical nature of the clusters. The right plot is the phase space plot of the same clusters (identified by colours). We see the strict cut on the V_r values, which points to the unphysical nature of the clusters.

UMAP run with 6D input (phase space) with a `n_neighoburs` value of 30, `min_dist` value of 0 and `n_componenets` value of 2. DBSCAN - run with `eps` of 0.08 and `min_samples` of 100. Particles classified as noise - 0.68%

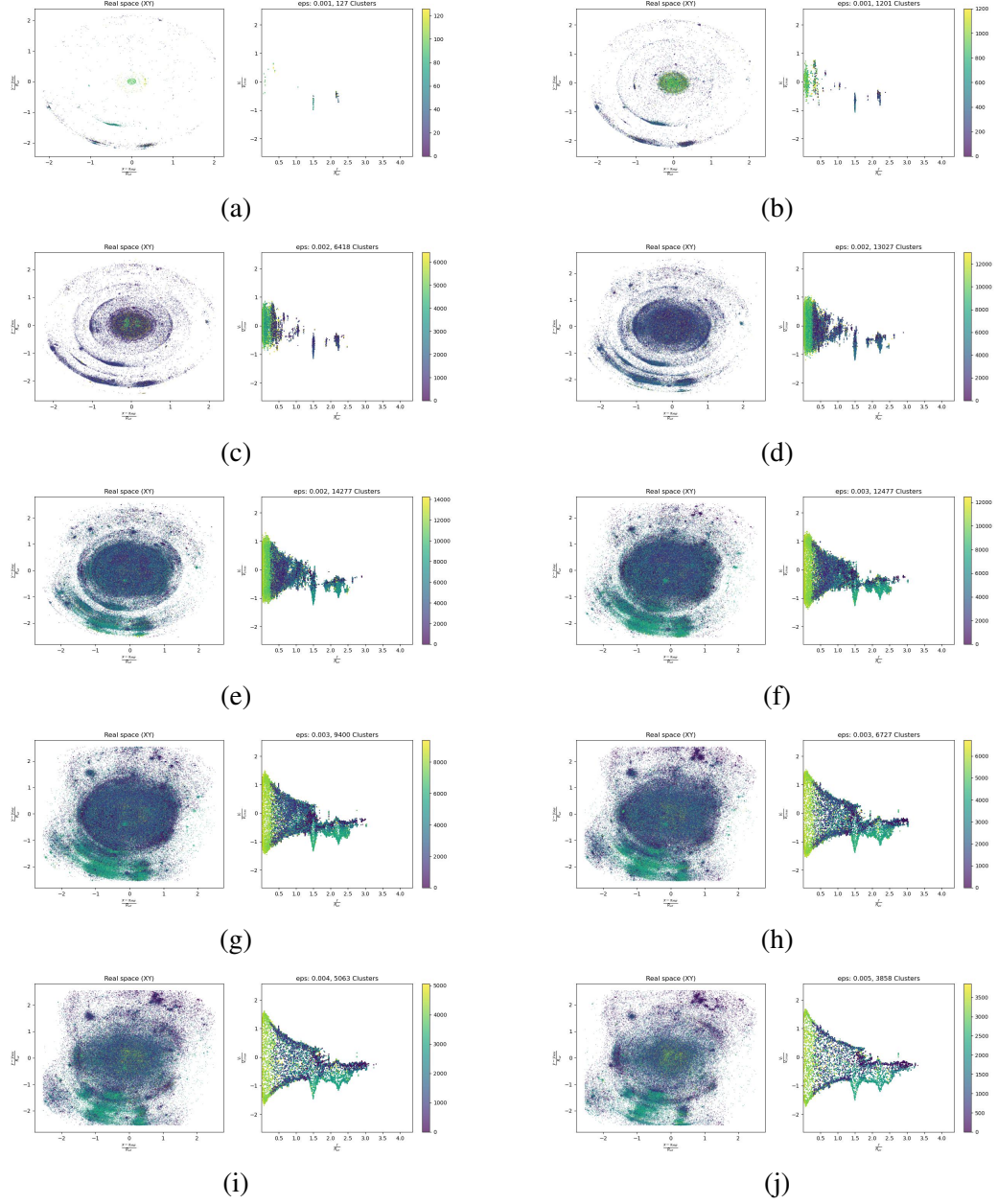


Figure 3.13: 2D phase space ($V_r - r$) iterative DBSCAN outputs for the first 10 steps. The first plot in each pair represents the clusters in XY space, and the second in each pair represents the 2D phase space $V_r - r$. The title of each pair of plots shows the eps parameter value and the number of clusters identified in that particular step. After each step, the identified clusters are removed from the input data for the next step. For step 2, all the clusters from step 1 will be removed. For step 3, all from step 2 will be removed and so on. This is elaborated in section 2.3.2.

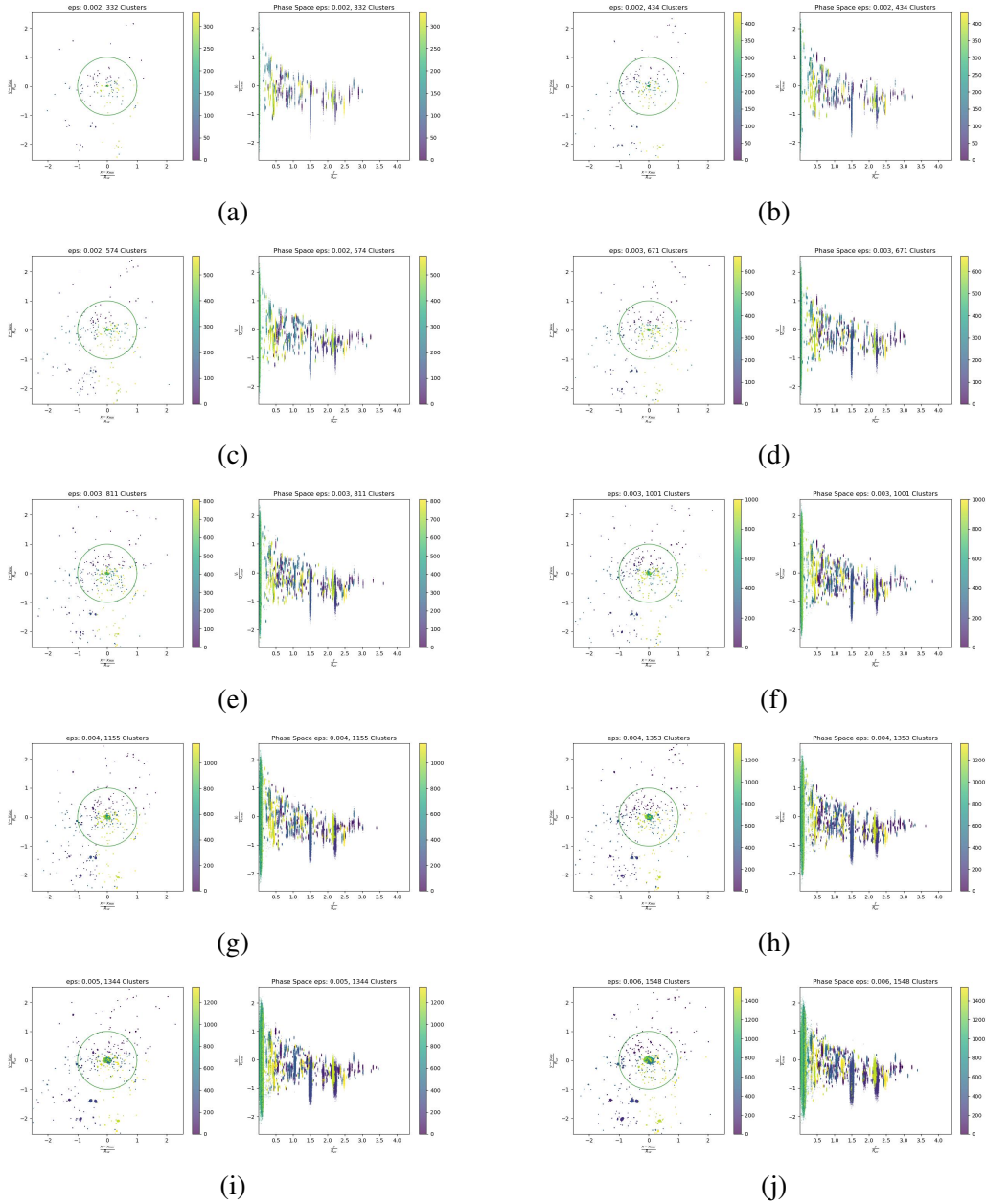


Figure 3.14: 3D position space iterative DBSCAN outputs for iterations 7 through 16. The first 6 iterations clustered a total of 26,373 (0.52% of the total particles). The first plot in each pair represents the clusters in XY space, and the second in each pair represents the 2D phase space $V_r - r$. The title of each pair of plots shows the eps parameter value and the number of clusters identified in that particular step. After each step, the identified clusters are removed from the input data for the next step. For step 2, all the clusters from step 1 will be removed. For step 3, all from step 2 will be removed and so on. This is elaborated in section 2.3.2.

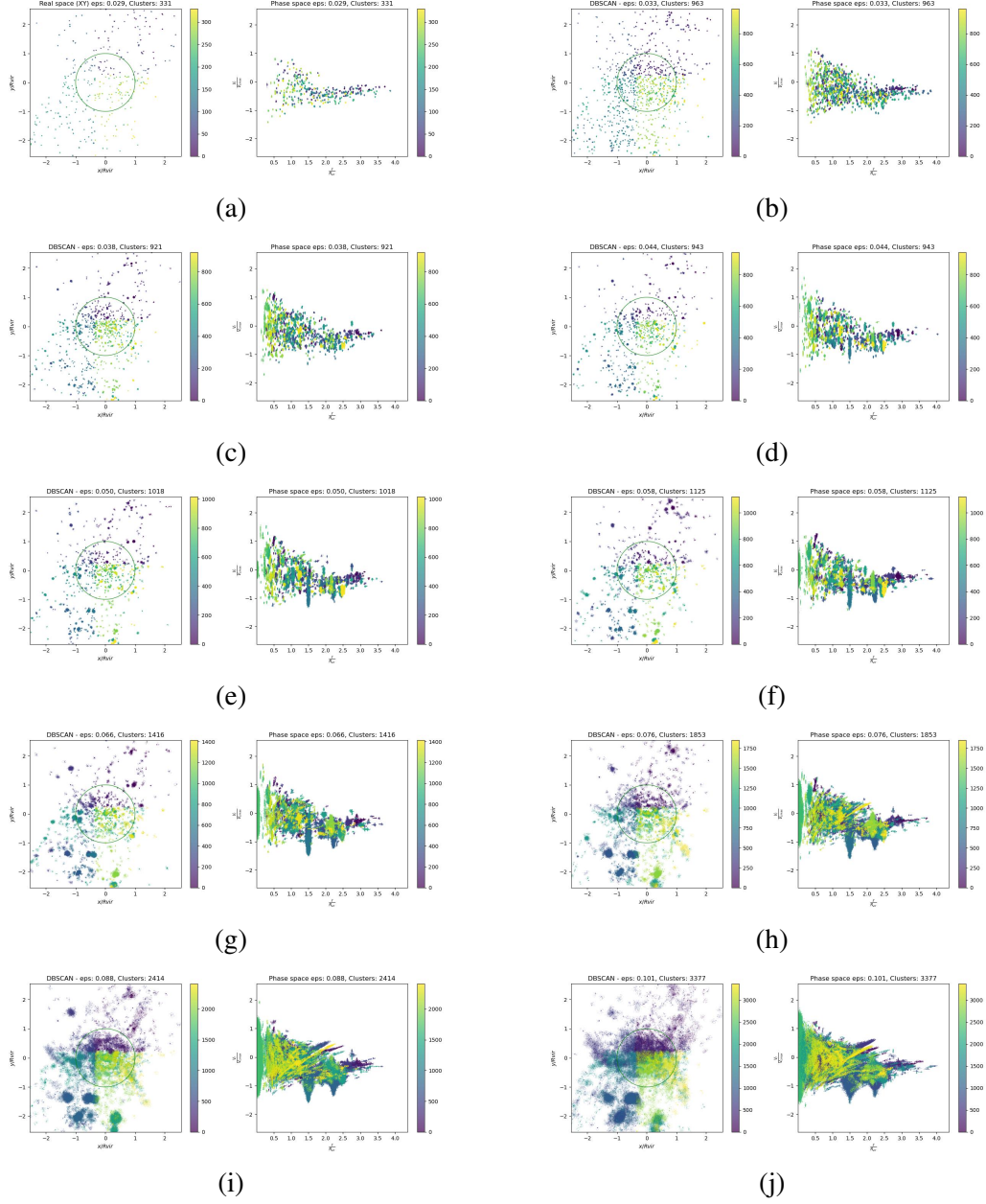


Figure 3.15: 6D phase space (x, y, z, v_x, v_y, v_z) iterative DBSCAN outputs from the first 10 steps. The first plot in each pair represents the clusters in XY space, and the second in each pair represents the 2D phase space $V_r - r$. The title of each pair of plots shows the eps parameter value and the number of clusters identified in that particular step. After each step, the identified clusters are removed from the input data for the next step. For step 2, all the clusters from step 1 will be removed. For step 3, all from step 2 will be removed and so on.

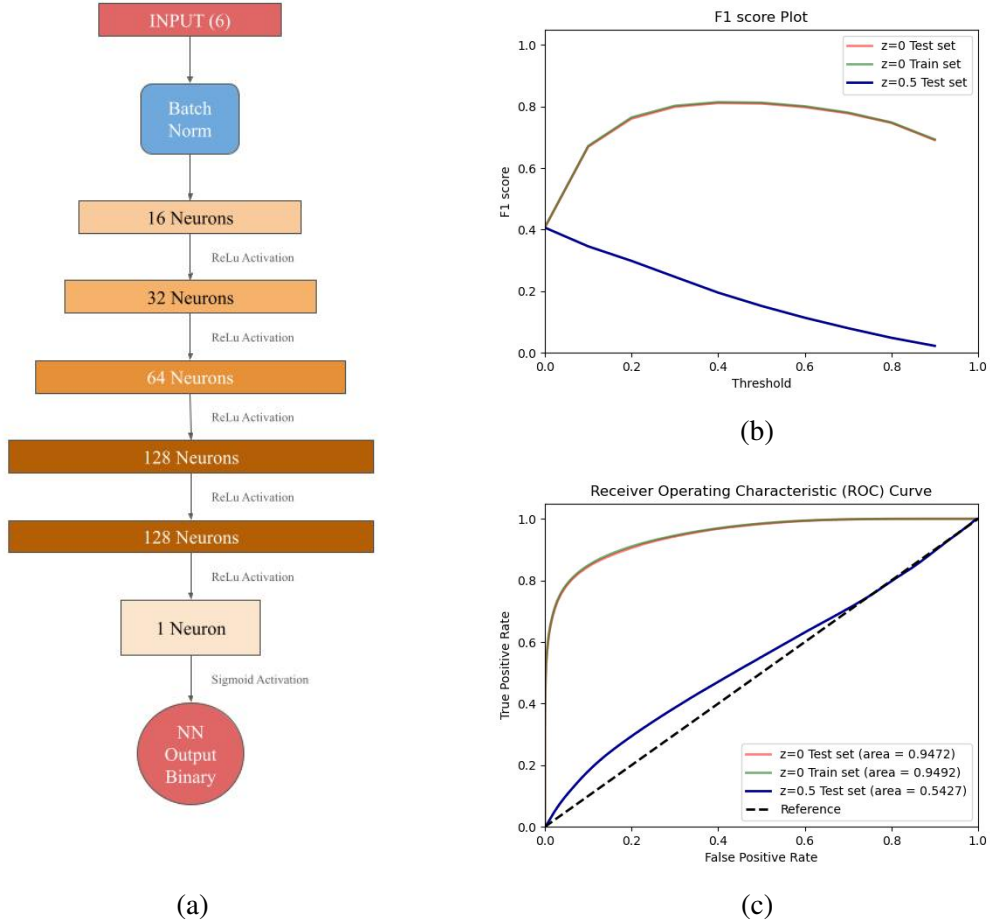


Figure 3.16: Figure 3.16a shows the architecture of the network. Figure 3.16b shows the F1 Score for varying thresholds on different datasets. Figure 3.16c shows the ROC curves for training and testing sets. The ROC curves for both datasets from $z = 0$ overlap. The dotted line is for reference. F1 score and ROCs are explained in the body and footnotes of Section 3.5

Chapter 4

Conclusion

This was an exploratory work to devise an alternate method to find structures in cosmological simulations. Since the current state of the art is capable of finding spherical structures like halos and subhalos but fails when it comes to elongated structures like streams, we wanted to explore various possible machine learning algorithms to find spherical as well as elongated structures. Finding such structures can be useful in constraining the micro-physical properties of dark matter, like annihilation rates and reaction rates, which depend on the velocity distribution and spatial density of particles. The current constraints on the DM annihilation cross sections, for example, is $\langle \sigma v \rangle \leq [2, 3] \times 10^{-25} \text{cm}^3 \text{s}^{-1}$ [59]. These numbers are assuming a WIMP-like scenario. Since these cross-sections are quite low, highly dense regions of dark matter are promising places to look for signatures of dark matter interactions. Dark matter halos and streams are some of the densest regions of dark matter. Due to the high densities, we can use these regions to constrain certain known channels of annihilation.

In our quest to find *the one* ML technique that would identify streams and subhalos, we stumbled upon UMAP. UMAP is a non-linear dimension reduction algorithm which we use to reduce 6D phase space to a 2D representation. In doing so, we find that, on the largest scale, UMAP is capable of separating out the biggest halos in the data. Amongst the MW particles (particles within the virial boundary of the MW), there is a segregation of particles based on how *relaxed* [57] the particles are. UMAP also separates out different parts of an infalling/tidally disrupted subhalo. Therefore, in a very broad sense, one can say that UMAP separates particles based on their dynamics and is capable of finding ex-

tended structures like subhalos and streams. The efficacy of this technique still has to be rigorously tested before we can claim we have achieved our goal. The current halo finders do not perform so well when it comes to identifying streams. So, UMAP takes us a step further in identifying streams and other elongated substructures that could help constrain the microphysics of dark matter and perform various studies of the dynamics of particles under various dark matter paradigms. If the viability of UMAP as a structure-finder is established, maybe we can explore other topological techniques and GNNs to find various structures in simulations.

Bibliography

- [1] F. Zwicky. Die rotverschiebung von extragalaktischen nebeln. *Helvetica Physica Acta* 6, 110-127, 1933.
- [2] Vera C. Rubin and Jr. Ford, W. Kent. Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions. , 159:379, February 1970. doi: 10.1086/150317.
- [3] V. C. Rubin, N. Thonnard, and Jr. Ford, W. K. Extended rotation curves of high-luminosity spiral galaxies. I. The angle between the rotation axis of the nucleus and the outer disk of NGC 3672. , 217:L1–L4, October 1977. doi: 10.1086/182526.
- [4] V. C. Rubin, Jr. Ford, W. K., K. M. Strom, S. E. Strom, and W. Romanishin. Extended rotation curves of high-luminosity spiral galaxies. II. The anemic Sa galaxy NGC 4378. , 224:782–795, September 1978. doi: 10.1086/156426.
- [5] V. C. Rubin, Jr. Ford, W. K., and N. Thonnard. Extended rotation curves of high-luminosity spiral galaxies. IV. Systematic dynamical properties, Sa -> Sc. , 225: L107–L111, November 1978. doi: 10.1086/182804.
- [6] C. J. Peterson, V. C. Rubin, Jr. Ford, W. K., and M. S. Roberts. Extended rotation curves of high-luminosity spiral galaxies. III. The spiral galaxy NGC 7217. , 226: 770–776, December 1978. doi: 10.1086/156658.
- [7] V. C. Rubin, W. K. Jr. Ford, and M. S. Roberts. Extended rotation curves of high-luminosity spiral galaxies. V. NGC 1961, the most massive spiral known. , 230: 35–39, May 1979. doi: 10.1086/157059.
- [8] V. C. Rubin, Jr. Ford, W. K., and N. Thonnard. Rotational properties of 21 SC galaxies

- with a large range of luminosities and radii, from NGC 4605 ($R=4\text{kpc}$) to UGC 2885 ($R=122\text{kpc}$). , 238:471–487, June 1980. doi: 10.1086/158003.
- [9] V. C. Rubin, Jr. Ford, W. K., N. Thonnard, and D. Burstein. Rotational properties of 23Sb galaxies. , 261:439–456, October 1982. doi: 10.1086/160355.
- [10] V. C. Rubin, D. Burstein, Jr. Ford, W. K., and N. Thonnard. Rotation velocities of 16 SA galaxies and a comparison of Sa, SB and SC rotation properties. , 289:81–104, February 1985. doi: 10.1086/162866.
- [11] Claude Carignan, Laurent Chemin, Walter K. Huchtmeier, and Felix J. Lockman. The extended h α rotation curve and mass distribution of m31. *The Astrophysical Journal*, 641(2):L109–L112, March 2006. ISSN 1538-4357. doi: 10.1086/503869. URL <http://dx.doi.org/10.1086/503869>.
- [12] Marc Seigar and J. Berrier. *Galaxy Rotation Curves in the Context of LambdaCDM Cosmology*. 08 2011. ISBN 978-953-307-423-8. doi: 10.5772/22992.
- [13] Matthias Bartelmann. Gravitational lensing. *Classical and Quantum Gravity*, 27(23):233001, November 2010. ISSN 1361-6382. doi: 10.1088/0264-9381/27/23/233001. URL <http://dx.doi.org/10.1088/0264-9381/27/23/233001>.
- [14] Ruth Durrer. The cosmic microwave background: the history of its experimental investigation and its significance for cosmology. *Classical and Quantum Gravity*, 32(12):124007, June 2015. ISSN 1361-6382. doi: 10.1088/0264-9381/32/12/124007. URL <http://dx.doi.org/10.1088/0264-9381/32/12/124007>.
- [15] Douglas Clowe, Maruša Bradač, Anthony H. Gonzalez, Maxim Markevitch, Scott W. Randall, Christine Jones, and Dennis Zaritsky. A Direct Empirical Proof of the Existence of Dark Matter. , 648(2):L109–L113, September 2006. doi: 10.1086/508162.
- [16] Jean-Philippe Uzan. *The big-bang theory: construction, evolution and status*, 2016.
- [17] Derek F. Jackson Kimball, Leanne D. Duffy, and David J. E. Marsh. *Ultralight Bosonic Dark Matter Theory*, pages 31–72. Springer International Publishing, Cham, 2023. ISBN 978-3-030-95852-7. doi: 10.1007/978-3-030-95852-7_2. URL https://doi.org/10.1007/978-3-030-95852-7_2.
- [18] Daniel Baumann. *Cosmology*. Cambridge University Press, 2022.

- [19] ANTONINO DEL POPOLO. Nonbaryonic dark matter in cosmology. *International Journal of Modern Physics D*, 23(03):1430005, February 2014. ISSN 1793-6594. doi: 10.1142/s0218271814300055. URL <http://dx.doi.org/10.1142/S0218271814300055>.
- [20] Joseph Silk. Cosmic Black-Body Radiation and Galaxy Formation. , 151:459, February 1968. doi: 10.1086/149449.
- [21] Antonino Del Popolo and Morgan Le Delliou. Small scale problems of the Λ CDM model: A short review. *Galaxies*, 5(1):17, February 2017. ISSN 2075-4434. doi: 10.3390/galaxies5010017. URL <http://dx.doi.org/10.3390/galaxies5010017>.
- [22] Ricardo A. Flores and Joel R. Primack. Observational and Theoretical Constraints on Singular Dark Matter Halos. , 427:L1, May 1994. doi: 10.1086/187350.
- [23] B. Moore. Evidence against dissipationless dark matter from observations of galaxy haloes. *Nature*, 370:629, 1994. doi: 10.1038/370629a0.
- [24] Michael Boylan-Kolchin, James S. Bullock, and Manoj Kaplinghat. Too big to fail? the puzzling darkness of massive milky way subhaloes. *Monthly Notices of the Royal Astronomical Society: Letters*, 415(1):L40–L44, July 2011. ISSN 1745-3925. doi: 10.1111/j.1745-3933.2011.01074.x. URL <http://dx.doi.org/10.1111/j.1745-3933.2011.01074.x>.
- [25] The aquarius project: the subhaloes of galactic haloes. 391(4). ISSN 1365-2966. doi: 10.1111/j.1365-2966.2008.14066.x. URL <http://dx.doi.org/10.1111/j.1365-2966.2008.14066.x>.
- [26] Mark R. Lovell, Violeta Gonzalez-Perez, Sownak Bose, Alexey Boyarsky, Shaun Cole, Carlos S. Frenk, and Oleg Ruchayskiy. Addressing the too big to fail problem with baryon physics and sterile neutrino dark matter. *Monthly Notices of the Royal Astronomical Society*, 468(3):2836–2849, March 2017. ISSN 1365-2966. doi: 10.1093/mnras/stx621. URL <http://dx.doi.org/10.1093/mnras/stx621>.
- [27] Stacy Y. Kim, Annika H. G. Peter, and Jonathan R. Hargis. Missing Satellites Problem: Completeness Corrections to the Number of Satellite Galaxies in the Milky Way are Consistent with Cold Dark Matter Predictions. , 121(21):211302, November 2018. doi: 10.1103/PhysRevLett.121.211302.

- [28] Alex Drlica-Wagner, Yao-Yuan Mao, Susmita Adhikari, Robert Armstrong, Arka Banerjee, Nilanjan Banik, Keith Bechtol, Simeon Bird, Kimberly K. Boddy, Ana Bonaca, Jo Bovy, Matthew R. Buckley, Esra Bulbul, Chihway Chang, George Chapline, Johann Cohen-Tanugi, Alessandro Cuoco, Francis-Yan Cyr-Racine, William A. Dawson, Ana Díaz Rivero, Cora Dvorkin, Denis Erkal, Christopher D. Fasnacht, Juan García-Bellido, Maurizio Giannotti, Vera Gluscevic, Nathan Golovich, David Hendel, Yashar D. Hezaveh, Shunsaku Horiuchi, M. James Jee, Manoj Kaplinghat, Charles R. Keeton, Sergey E. Koposov, Casey Y. Lam, Ting S. Li, Jessica R. Lu, Rachel Mandelbaum, Samuel D. McDermott, Mitch McNanna, Michael Medford, Manuel Meyer, Moniez Marc, Simona Murgia, Ethan O. Nadler, Lina Necib, Eric Nuss, Andrew B. Pace, Annika H. G. Peter, Daniel A. Polin, Chanda Prescod-Weinstein, Justin I. Read, Rogerio Rosenfeld, Nora Shipp, Joshua D. Simon, Tracy R. Slatyer, Oscar Straniero, Louis E. Strigari, Erik Tollerud, J. Anthony Tyson, Mei-Yu Wang, Risa H. Wechsler, David Wittman, Hai-Bo Yu, Gabrijela Zaharijas, Yacine Ali-Haïmoud, James Annis, Simon Birrer, Rahul Biswas, Jonathan Blazek, Alyson M. Brooks, Elizabeth Buckley-Geer, Regina Caputo, Eric Charles, Seth Digel, Scott Dodelson, Brenna Flaugher, Joshua Frieman, Eric Gawiser, Andrew P. Hearin, Renee Hložek, Bhuvnesh Jain, Tesla E. Jeltema, Savvas M. Koushiappas, Mariangela Lisanti, Marilena LoVerde, Siddharth Mishra-Sharma, Jeffrey A. Newman, Brian Nord, Erfan Nourbakhsh, Steven Ritz, Brant E. Robertson, Miguel A. Sánchez-Conde, Anže Slosar, Tim M. P. Tait, Aprajita Verma, Ricardo Vilalta, Christopher W. Walter, Brian Yanny, and Andrew R. Zentner. Probing the fundamental nature of dark matter with the large synoptic survey telescope, 2019.
- [29] Cedric Lacey and Shanu Cole. Merger rates in hierarchical models of galaxy formation – II. Comparison with N-body simulations. *Monthly Notices of the Royal Astronomical Society*, 271(3):676–692, 12 1994. ISSN 0035-8711. doi: 10.1093/mnras/271.3.676. URL <https://doi.org/10.1093/mnras/271.3.676>.
- [30] Rafael García, Edgar Salazar, Eduardo Rozo, Susmita Adhikari, Han Aung, Benedikt Diemer, Daisuke Nagai, and Brandon Wolfe. A better way to define dark matter haloes. *Monthly Notices of the Royal Astronomical Society*, 521(2):2464–2476, mar 2023. doi: 10.1093/mnras/stad660. URL <https://doi.org/10.1093/mnras/stad660>.
- [31] Volker Springel, Simon D. M. White, Adrian Jenkins, Carlos S. Frenk, Naoki Yoshida,

Liang Gao, Julio Navarro, Robert Thacker, Darren Croton, John Helly, John A. Peacock, Shaun Cole, Peter Thomas, Hugh Couchman, August Evrard, Jörg Colberg, and Frazer Pearce. Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature*, 435(7042):629–636, jun 2005. doi: 10.1038/nature03597. URL <https://doi.org/10.1038%2Fnature03597>.

- [32] Donald G. York, J. Adelman, Jr. John E. Anderson, Scott F. Anderson, James Annis, Neta A. Bahcall, J. A. Bakken, Robert Barkhouser, Steven Bastian, Eileen Berman, William N. Boroski, Steve Bracker, Charlie Briegel, John W. Briggs, J. Brinkmann, Robert Brunner, Scott Burles, Larry Carey, Michael A. Carr, Francisco J. Castander, Bing Chen, Patrick L. Colestock, A. J. Connolly, J. H. Crocker, István Csabai, Paul C. Czarapata, John Eric Davis, Mamoru Doi, Tom Dombeck, Daniel Eisenstein, Nancy Ellman, Brian R. Elms, Michael L. Evans, Xiaohui Fan, Glenn R. Federwitz, Larry Fiscelli, Scott Friedman, Joshua A. Frieman, Masataka Fukugita, Bruce Gillespie, James E. Gunn, Vijay K. Gurbani, Ernst de Haas, Merle Haldeman, Frederick H. Harris, J. Hayes, Timothy M. Heckman, G. S. Hennessy, Robert B. Hindsley, Scott Holm, Donald J. Holmgren, Chi hao Huang, Charles Hull, Don Husby, Shin-Ichi Ichikawa, Takashi Ichikawa, Željko Ivezić, Stephen Kent, Rita S. J. Kim, E. Kinney, Mark Klaene, A. N. Kleinman, S. Kleinman, G. R. Knapp, John Korienek, Richard G. Kron, Peter Z. Kunszt, D. Q. Lamb, B. Lee, R. French Leger, Siriluk Limmongkol, Carl Lindenmeyer, Daniel C. Long, Craig Loomis, Jon Loveday, Rich Lucinio, Robert H. Lupton, Bryan MacKinnon, Edward J. Mannery, P. M. Mantsch, Bruce Margon, Peregrine McGehee, Timothy A. McKay, Avery Meiksin, Aronne Merelli, David G. Monet, Jeffrey A. Munn, Vijay K. Narayanan, Thomas Nash, Eric Neilsen, Rich Neswold, Heidi Jo Newberg, R. C. Nichol, Tom Nicinski, Mario Nonino, Norio Okada, Sadanori Okamura, Jeremiah P. Ostriker, Russell Owen, A. George Pauls, John Peoples, R. L. Peterson, Donald Petravick, Jeffrey R. Pier, Adrian Pope, Ruth Pordes, Angela Prosapio, Ron Rechenmacher, Thomas R. Quinn, Gordon T. Richards, Michael W. Richmond, Claudio H. Rivetta, Constance M. Rockosi, Kurt Ruthmansdorfer, Dale Sandford, David J. Schlegel, Donald P. Schneider, Maki Sekiguchi, Gary Sergey, Kazuhiro Shimasaku, Walter A. Siegmund, Stephen Smee, J. Allyn Smith, S. Snedden, R. Stone, Chris Stoughton, Michael A. Strauss, Christopher Stubbs, Mark SubbaRao, Alexander S. Szalay, Istvan Szapudi, Gyula P. Szokoly, Anirudda R. Thakar, Christy Tremonti, Douglas L. Tucker, Alan Uomoto, Dan Vanden Berk, Michael S. Vogeley, Patrick Waddell, Shu i Wang, Masaru Watan-

- abe, David H. Weinberg, Brian Yanny, and Naoki Yasuda. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579–1587, sep 2000. doi: 10.1086/301513. URL <https://doi.org/10.1086%2F301513>.
- [33] Margaret J. Geller and John P. Huchra. Mapping the universe. *Science*, 246(4932): 897–903, 1989. doi: 10.1126/science.246.4932.897. URL <https://www.science.org/doi/abs/10.1126/science.246.4932.897>.
- [34] Matthew Colless, Gavin Dalton, Steve Maddox, Will Sutherland, Peder Norberg, Shaun Cole, Joss Bland-Hawthorn, Terry Bridges, Russell Cannon, Chris Collins, Warrick Couch, Nicholas Cross, Kathryn Deeley, Roberto de Propriis, Simon P. Driver, George Efstathiou, Richard S. Ellis, Carlos S. Frenk, Karl Glazebrook, Carole Jackson, Ofer Lahav, Ian Lewis, Stuart Lumsden, Darren Madgwick, John A. Peacock, Bruce A. Peterson, Ian Price, Mark Seaborne, and Keith Taylor. The 2dF Galaxy Redshift Survey: spectra and redshifts. *Monthly Notices of the Royal Astronomical Society*, 328(4):1039–1063, 12 2001. ISSN 0035-8711. doi: 10.1046/j.1365-8711.2001.04902.x. URL <https://doi.org/10.1046/j.1365-8711.2001.04902.x>.
- [35] Giuseppina Battaglia, Amina Helmi, Heather Morrison, Paul Harding, Edward W. Olszewski, Mario Mateo, Kenneth C. Freeman, John Norris, and Stephen A. Shectman. The radial velocity dispersion profile of the galactic halo: constraining the density profile of the dark halo of the milky way. *Monthly Notices of the Royal Astronomical Society*, 364(2):433–442, December 2005. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2005.09367.x. URL <http://dx.doi.org/10.1111/j.1365-2966.2005.09367.x>.
- [36] Prajwal Raj Kafle, Sanjib Sharma, Geraint F. Lewis, and Joss Bland-Hawthorn. On the shoulders of giants: Properties of the stellar halo and the milky way mass distribution. *The Astrophysical Journal*, 794(1):59, September 2014. ISSN 1538-4357. doi: 10.1088/0004-637x/794/1/59. URL <http://dx.doi.org/10.1088/0004-637X/794/1/59>.
- [37] Steffen R. Knollmann and Alexander Knebe. Ahf: Amiga’s halo finder. *The Astrophysical Journal Supplement Series*, 182(2):608–624, May 2009. ISSN 1538-4365. doi: 10.1088/0067-0049/182/2/608. URL <http://dx.doi.org/10.1088/0067-0049/182/2/608>.

- [38] P. S. Behroozi, R. H. Wechsler, and H.-Y. Wu. The ROCKSTAR Phase-space Temporal Halo Finder and the Velocity Offsets of Cluster Cores. , 762:109, January 2013. doi: 10.1088/0004-637X/762/2/109.
- [39] Matthieu Schaller. Friends-of-friends algorithm, 2019. URL https://swift.dur.ac.uk/docs/_sources/FriendsOfFriends/algorithm_description.rst.txt.
- [40] Greg L. Bryan and Michael L. Norman. Statistical Properties of X-Ray Clusters: Analytic and Numerical Comparisons. , 495(1):80–99, March 1998. doi: 10.1086/305262.
- [41] Josh Barnes and Piet Hut. A hierarchical $O(N \log N)$ force-calculation algorithm. , 324(6096):446–449, December 1986. doi: 10.1038/324446a0.
- [42] Peter S. Behroozi, Risa H. Wechsler, Hao-Yi Wu, Michael T. Busha, Anatoly A. Klypin, and Joel R. Primack. Gravitationally consistent halo catalogs and merger trees for precision cosmology. *The Astrophysical Journal*, 763(1):18, December 2012. ISSN 1538-4357. doi: 10.1088/0004-637x/763/1/18. URL <http://dx.doi.org/10.1088/0004-637X/763/1/18>.
- [43] Yao-Yuan Mao, Marc Williamson, and Risa H. Wechsler. The dependence of sub-halo abundance on halo concentration. *The Astrophysical Journal*, 810(1):21, aug 2015. doi: 10.1088/0004-637X/810/1/21. URL <https://dx.doi.org/10.1088/0004-637X/810/1/21>.
- [44] Volker Springel, Naoki Yoshida, and Simon D. M. White. GADGET: a code for collisionless and gasdynamical cosmological simulations. , 6(2):79–117, April 2001. doi: 10.1016/S1384-1076(01)00042-2.
- [45] Volker Springel. The cosmological simulation code gadget-2. *Monthly Notices of the Royal Astronomical Society*, 364(4):1105–1134, 12 2005. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2005.09655.x. URL <https://doi.org/10.1111/j.1365-2966.2005.09655.x>.
- [46] Martín Crocce, Sebastián Pueblas, and Román Scoccimarro. Transients from initial conditions in cosmological simulations. *Monthly Notices of the Royal Astronomical Society*, 373(1):369–381, 10 2006. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2006.11040.x. URL <https://doi.org/10.1111/j.1365-2966.2006.11040.x>.

- [47] Oliver Hahn and Tom Abel. Multi-scale initial conditions for cosmological simulations. *Monthly Notices of the Royal Astronomical Society*, 415(3):2101–2121, 08 2011. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2011.18820.x. URL <https://doi.org/10.1111/j.1365-2966.2011.18820.x>.
- [48] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD’96*, page 226–231. AAAI Press, 1996.
- [49] Ricardo Campello, Davoud Moulavi, and Joerg Sander. Density-based clustering based on hierarchical density estimates. volume 7819, pages 160–172, 04 2013. ISBN 978-3-642-37455-5. doi: 10.1007/978-3-642-37456-2_14.
- [50] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.
- [51] Wikipedia contributors. Nerve complex — Wikipedia, the free encyclopedia, 2023. URL https://en.wikipedia.org/w/index.php?title=Nerve_complex&oldid=1183950593. [Online; accessed 15-March-2024].
- [52] Leland McInnes. Umap documentation - how umap works, 2018. URL https://umap-learn.readthedocs.io/en/latest/how_umap_works.html.
- [53] Benedikt Diemer and Andrey V. Kravtsov. Dependence of the outer density profiles of halos on their mass accretion rate. *Astrophys. J.*, 789:1, 2014. doi: 10.1088/0004-637X/789/1/1.
- [54] S. Adhikari, N. Dalal, and R. T. Chamberlain. Splashback in accreting dark matter halos. , 11:019, November 2014. doi: 10.1088/1475-7516/2014/11/019.
- [55] Surhud More, Benedikt Diemer, and Andrey Kravtsov. The splashback radius as a physical halo boundary and the growth of halo mass. *Astrophys. J.*, 810(1):36, 2015. doi: 10.1088/0004-637X/810/1/36.
- [56] Jeremy Tinker, Andrey V. Kravtsov, Anatoly Klypin, Kevork Abazajian, Michael Warren, Gustavo Yepes, Stefan Gottlöber, and Daniel E. Holz. Toward a Halo Mass Function for Precision Cosmology: The Limits of Universality. , 688(2):709–728, December 2008. doi: 10.1086/591439.

- [57] Angelo F. Neto, Liang Gao, Philip Bett, Shaun Cole, Julio F. Navarro, Carlos S. Frenk, Simon D. M. White, Volker Springel, and Adrian Jenkins. The statistics of Λ CDM halo concentrations. , 381(4):1450–1462, November 2007. doi: 10.1111/j.1365-2966.2007.12381.x.
- [58] Jason Brownlee. How to calculate precision, recall, and f-measure for imbalanced classification, 2020. URL <https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/>.
- [59] Shin’ichiro Ando and Daisuke Nagai. Fermi-lat constraints on dark matter annihilation cross section from observations of the fornax cluster. *Journal of Cosmology and Astroparticle Physics*, 2012(07):017, jul 2012. doi: 10.1088/1475-7516/2012/07/017. URL <https://dx.doi.org/10.1088/1475-7516/2012/07/017>.