Master's thesis
Astronomy

Your Title Here

Anni Järvenpää

August 6, 2018

Tutor: Dr. Till Sawala
Professor Peter Johansson

Censors: prof. Smith
doc. Smythe

UNIVERSITY OF HELSINKI
Department of Physics

PL 64 (Gustaf Hällströmin katu 2a)
00014 University of Helsinki

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty Faculty of Science	Laitos — Institution — Department Department of Physics

Tekijä — Författare — Author

Anni Järvenpää

Työn nimi — Arbetets titel — Title

Your Title Here

Oppiaine — Läroämne — Subject

Astronomy

Työn laji — Arbetets art — Level Master’s thesis	Aika — Datum — Month and year August 6, 2018	Sivumäärä — Sidoantal — Number of pages 0 pages

Tiivistelmä — Referat — Abstract

Abstract goes here.

Avainsanat — Nyckelord — Keywords

Your keywords here

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — övriga uppgifter — Additional information

Contents

1. Introduction

1.1 TL;DR version of prerequisite information

1. galaxies form
   - Why?
   - When?
   - How?
   - Where?
2. galaxies form in groups
3. our local group is one of these
4. something about large scale distribution of galaxies

1.2 History of Local Group Research

LG objects visible with naked eye -> realization they are something outside our galaxy -> realization they are something very much like our galaxy

First determining distance was difficult, now mass is more interesting question

1.3 Aim of This Thesis

Whatever the main results end up being, presented in somewhat coherent manner and hopefully sugar-coated enough to sound Important and Exciting.

2. Theoretical Background

Cosmology determines the properties of the Universe, including its origin, the rules by which it evolves and the structures that arise within it (?). Thus many fields of astronomy and astrophysics, including the study of galaxies and galaxy groups such as the Local Group and its members, are tightly connected to the study of cosmology. This section gives a brief explanation of the current cosmological understanding, its relevant implications and how the Local Group is currently viewed in ta cosmological context.

2.1 Basics of Cosmology

Understanding processes that take place on very large scales or take long times requires understanding some basic cosmology. The following sections will cover the most basic concepts of cosmology, the evolution of the Universe on both large and small scales, and the ΛCDM model which is currently the cosmological model that best matches the observations on multiple scales (?). Sections 2.1.1 and 2.1.2 cover large scales at which the cosmological principle applies and section 2.1.3 covers smaller scales of individual dark matter haloes.

2.1.1 Evolution of the Universe

The current cosmological understanding is based on a subset of the general theory of relativity together with simple hypotheses such as the cosmological principle, which states that on large scales the Universe is spatially homogeneous and isotropic (?). At a given location, an observer might not see this, as is easy to understand when considering two observers located at the same point but moving relative to each other. In this situation, it is clear that at least one of the observers will see a dipole in the surrounding velocities and thus will not observe the universe to be isotropic. Nonetheless, in an isotropic universe, for every point in space we can define a so-called fundamental observer as the observer who sees the universe as isotropic (?). These fundamental observers correspond to a cosmological rest frame, which can be determined at a given location e.g. by observing the cosmic microwave background and subtracting the velocity corresponding to the observed dipole component (?).

The existence of such fundamental observers has interesting consequences. The existence of any large-scale flows is clearly prohibited, as these would violate the isotropy. In a three-dimensional universe any curl of the velocity field around any fundamental observer is also forbidden. This can be easily seen by considering the hairy ball theorem, which states that a tangential vector field on a sphere must be zero in at least one point (?). Thus if there was a curl in the surrounding velocity field, there would always still be at least one direction in which the tangential velocity is zero and thus the field would not be isotropic. This means that there cannot be any tangential motion and thus the only allowed motion happens in the radial direction: the Universe can either expand or contract.

The expansion of the Universe can be parametrized using the dimensionless scale factor a, whose time evolution is governed by the Friedmann equations

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}\left(\rho + 3\frac{P}{c^2}\right) + \frac{\Lambda c^2}{3} \tag{2.1}$$

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho - \frac{Kc^2}{a^2} + \frac{\Lambda c^2}{3}, \tag{2.2}$$

where G is the gravitational constant, P is pressure, ρ is energy density and K and Λ are cosmology-related parameters (?). K specifies the curvature of the Universe, determined by the overall density of the Universe (?). The allowed values for K

are -1 corresponding to a hyperbolic universe, 0 to a flat one and 1 to a spherical universe (**?**). These are also often called open, critical and closed universes respectively. Modern measurements suggest that the Universe is flat within the measurement error, but small deviations from the $K = 0$ cannot be ruled out (**?**). Λ is the cosmological constant driving the expansion of the Universe, often described as dark energy or vacuum energy (**?**).

The concept of scale factor a is crucial for this thesis as it affects measured values of both distances and velocities. It increases or decreases as the Universe expands or contracts, and thus it can be used to relate distances at different times (**?**). For a universe that is monotonically expanding, the scale factor can also be used as an alternative time coordinate, as it has a one-to-one correspondence to time. It is often convenient to measure not the proper distance l between a pair of objects, but instead their so-called comoving distance r: (**?**)

$$r = \frac{l}{a} \tag{2.3}$$

For a pair of objects with no relative motion, the comoving distance will remain constant as the size of the Universe changes. A comoving coordinate system is often used in cosmological simulations, as the expansion of the universe is included in the scale factor instead of all of the coordinates having to be recalculated as the size of the universe varies (**?**).

Observations have confirmed that the Universe is indeed expanding, first observed by **?**. The rate of the expansion is denoted with the Hubble parameter $H(t)$, which is defined using the proper distance and the rate of change of the proper distance between a pair of fundamental observers:

$$H(t)\, l = \frac{\mathrm{d}l}{\mathrm{d}t} \tag{2.4}$$

The Hubble parameter is also closely related to the scale factor, as (**?**)

$$H(t) = \frac{\dot{a}(t)}{a(t)}. \tag{2.5}$$

Often in calculations the Hubble parameter is replaced with the reduced Hubble parameter, often denoted by h and defined as (?)

$$h(t) = \frac{H(t)}{100 \text{ km/s/Mpc}}. \tag{2.6}$$

The early measurements trying to determine the current value of the Hubble parameter, often called the Hubble constant, were prone to error as the distance estimates to extragalactic objects were inaccurate. For example the first measurement by ? yielded a value of over 500 km/s/Mpc due to systematically erroneous distance measurements derived using Cepheid variables. Later estimates offer reasonably accurate results such as the ? value of 67.77±0.77 km/s/Mpc for the current expansion speed, though the results of different experiments have considerable scatter.

Another factor affecting measurements based on extragalactic objects is that the observed proper velocities contain not only the expansion of the Universe but also peculiar motions of the objects, which is why measurements based on other probes such as the cosmic microwave background in case of ? are valuable. Information contained in the peculiar motions can still be interesting. In this thesis, the mass of the Local Group is estimated using radial velocity measurements within a few megaparsecs of a number of simulated Local Universe analogues. At scales this small, the expansion of the Universe is greatly affected by local gravity fields, and thus expansion measurements can be used to infer the mass enclosed within the Local Group.

While the scale factor is one possible way of expressing time, it is not the only one. As the universe expands and we observe objects receding, the light emitted from them is shifted to longer wavelengths. The further away the emitter is, the more the space between the observer and emitter will expand making the effect stronger. The relative change in the wavelength is called redshift z, defined as

$$z \equiv \frac{\lambda_o - \lambda_e}{\lambda_e} \tag{2.7}$$

where λ_o is the observed and λ_e the emitted wavelength (?). If the effect of peculiar motions is ignored, redshift is directly related not only to the distance to the emitter but also to the scale factor at the time of the emission. For arbitrary scale factors at the time of the emission and observation, the relation between z and a is (?)

$$1 + z = \frac{a(t_o)}{a(t_e)}. \tag{2.8}$$

In most situations observations are done at $a = 1$, in which case the scale factor at the time of the emission can be written

$$a = \frac{1}{1 + z}. \tag{2.9}$$

2.1.2 Composition of the Universe

Using the Friedmann equations 2.1 and 2.2 to model the evolution of the Universe requires not only the equations and cosmological parameters introduced in section 2.1.1 but also the composition of the Universe to be known in order to determine the density, pressure, cosmological constant and ultimately even the curvature of the Universe. According to the current understanding, the Universe is made of three components: non-relativistic matter, relativistic matter and dark energy (?). At present time, about 69 % of the energy density ρ in the Universe is dark energy and 31 % is non-relativistic matter, including both baryonic and dark components (?). Most of the non-relativistic matter, around 84 %, is dark matter, with baryonic matter only contributing around 16 % of total matter energy density (?). The energy density of relativistic matter, i.e. photons and standard-model neutrinos, in the present-day universe is negligible (?).

This has not always been true, as these ratios change over time as the Universe evolves. It is easy to see that, as the Universe expands, the energy density of matter behaves as a^{-3}, as the volume of the Universe increases as a^3 and in an adiabatic system no matter is created or disappears. Radiation is diluted similarly to matter, but in addition to the effect of increasing volume, the growing universe also causes the wavelength of the radiation to increase, resulting in the energy density decreasing as a^{-4}. The dark energy, as the name suggests, can be thought as arising from the space itself, and thus a change in a does not affect the energy density of the component (?). The change of energy density of a component may also correspond to a change in the pressure or temperature of the component (?).

The different time evolutions of the components also mean that the dominant component of the Universe changes as the scale factor grows. At very early times the Universe was radiation dominated, but as the radiation energy density decreases faster than the energy densities of the two other components, a matter dominated era followed. As dark energy is the only component with constant energy density, it will be the final dominant energy component. This transition from matter to dark energy dominated era has happened in the recent past (?).

The geometry of the Universe is also determined by its contents. A density threshold known as critical density and defined as

$$\rho_{crit,0} \equiv \frac{3H_0^2}{8\pi G} \tag{2.10}$$

acts as a threshold value that separates the different geometries (?). Subscript zero comes from $z = 0$ and denotes present-day values, but all introduced quantities can also be determined at any other time. The overall density of the Universe can be parametrized using this critical density:

$$\Omega_0 \equiv \frac{\bar{\rho}_0}{\rho_{crit,0}} \tag{2.11}$$

where $\bar{\rho}_0$ is the mean density of the Universe and Ω_0 is known as the density parameter (?). Different values of Ω_0 correspond to different geometries: for values of $\Omega_0 < 1$ the Universe is hyperbolic, for $\Omega_0 = 1$ flat and for $\Omega_0 > 1$ spherical (?).

As the contents of the Universe can be divided into the three categories of dark energy and relativistic and non-relativistic matter, the total density parameter is also a sum of three density parameters:

$$\Omega_0 = \Omega_{m,0} + \Omega_{r,0} + \Omega_{\Lambda,0} \tag{2.12}$$

where m, r and Λ stand for non-relativistic matter ("matter"), relativistic matter ("radiation") and dark energy (?). Each of these can be calculated similarly to the equation 2.11 but replacing the overall density with density of the corresponding component (?). As in the Universe the value of Ω_0 is very close to unity, the different density parameters conveniently correspond to the relative densities of the components.

2.1.3 Structure Formation in the Linear Regime

All structures in the Universe arise from the small perturbations in the matter density in the early universe and the physics that govern their evolution. These primordial density fluctuations that later develop into the structures such as galaxy clusters and voids separating them can still be seen in the cosmic microwave background (CMB) (?). Starting at the end of inflation, the density contrast of such perturbations in the dark matter begins to grow. While dark matter is collisionless, after inflationary epoch the baryons are still coupled to radiation via Thomson scattering and experience pressure which slows their structural evolution down. Consequently, their evolution is delayed relative to the dark matter, and baryons sink into potential wells already formed from the dark matter.

At $z \approx 1100$, the time of recombination and origin of the CMB photons, these fluctuations are still very small, having $\Delta\rho/\rho \approx 10^{-5}$, but sufficiently overdense regions have already collapsed (?). The gas inside these structures seen as hot spots in the CMB has just reached a density maximum and is shock-heated, while gas inside smaller fluctuations has already begun expanding again, and larger scales have not yet reached their maximum density at the time of the CMB.

At first, the evolution of overdense regions differs from the evolution of the surroundings only by the speed of the expansion of the Universe in that region: expansion is slower in denser regions. Later some of these volumes will start to collapse. This requires the volume to be sufficiently dense to allow gravity to overcome the expansion of the universe as described by the Friedmann equations.

To understand the evolution of density perturbations at early times when the perturbations are still small and evolve linearly, let us consider an ideal fluid of density ρ, moving at proper velocity $\mathbf{v}$ and experiencing the gravitational field with potential ϕ. Growth of a perturbation in this medium is governed by three equations: the equation of continuity describing the conservation of mass, the Euler equation governing the motions in the fluid and the Poisson equation describing the gravitational field, or

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho \nabla_{\mathbf{x}} \cdot \mathbf{v} = 0\,, \tag{2.13}$$

$$\frac{\mathrm{D}\mathbf{v}}{\mathrm{D}t} = -\frac{\nabla_{\mathbf{x}} P}{\rho} - \nabla_{\mathbf{x}} \phi \tag{2.14}$$

and

$$\nabla^2_{\mathbf{x}} \phi = 4\pi G \rho \tag{2.15}$$

respectively (**?**). Here $\mathbf{x}$ denotes proper coordinates and $\frac{\mathrm{D}}{\mathrm{D}t}$ is the convective time derivative, defined as

$$\frac{\mathrm{D}}{\mathrm{D}t} = \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} \tag{2.16}$$

and describing the time derivative when moving with the fluid (**?**).

Next let us follow **?** and introduce a small perturbation by replacing $\mathbf{v}$, ρ, P and ϕ with $\mathbf{v_0} + \delta\mathbf{v}$, $\rho_0 + \delta\rho$, $P_0 + \delta P$ and $\phi_0 + \delta\phi$ respectively, having subscript zero represent the properties of the unperturbed medium. Now equations 2.13–2.15 can be written as

$$\frac{\mathrm{D}\rho_0}{\mathrm{D}t} + \frac{\mathrm{D}\delta\rho}{\mathrm{D}t} + \rho_0 \nabla_{\mathbf{x}} \cdot \mathbf{v}_0 + \rho_0 \nabla_{\mathbf{x}} \cdot \delta\mathbf{v} + \delta\rho \nabla_{\mathbf{x}} \cdot \mathbf{v}_0 + \delta\rho \nabla_{\mathbf{x}} \cdot \delta\mathbf{v} = 0\,, \tag{2.17}$$

$$\frac{\mathrm{D}\mathbf{v}_0}{\mathrm{D}t} + \frac{\mathrm{D}\delta\mathbf{v}}{\mathrm{D}t} = -\frac{\nabla_{\mathbf{x}}(P_0 + \delta P)}{\rho_0 + \delta\rho} - \nabla_{\mathbf{x}}\phi_0 - \nabla_{\mathbf{x}}\delta\phi \tag{2.18}$$

and

$$\nabla^2_{\mathbf{x}}\phi_0 + \nabla^2_{\mathbf{x}}\delta\phi = 4\pi G\rho_0 + 4\pi G\delta\rho. \tag{2.19}$$

These equations can be greatly simplified by subtracting the unperturbed versions from each and assuming that the initial state is homogeneous and isotropic i.e. $\nabla P_0 = 0$ and $\nabla\rho_0 = 0$. Using the knowledge that in the linear regime the perturbations are small and thus discarding second order terms, the equations take the following forms:

$$\frac{\mathrm{D}}{\mathrm{D}t}\frac{\delta\rho}{\rho_0} = -\nabla_{\mathbf{x}} \cdot \delta\mathbf{v} \tag{2.20}$$

$$\frac{\partial\delta\mathbf{v}}{\partial t} + (\delta\mathbf{v} \cdot \nabla_{\mathbf{x}})\mathbf{v}_0 = -\frac{\nabla_{\mathbf{x}}\delta P}{\rho_0} - \nabla_{\mathbf{x}}\delta\phi \tag{2.21}$$

$$\nabla^2_{\mathbf{x}}\delta\phi = 4\pi G\delta\rho \tag{2.22}$$

As the Universe is expanding, it is natural to transit to comoving coordinates, defined in equation 2.3 and denoted by $\mathbf{r}$. This also affects the velocities, and using the dot to denote time derivative we can write

$$\mathbf{v} = \dot{\mathbf{x}} = \dot{a}(t)\mathbf{r} + a(t)\dot{\mathbf{r}} = \dot{a}(t)\mathbf{r} + \mathbf{u}, \tag{2.23}$$

where $\mathbf{u}$ denotes the perturbed comoving velocity. From this it follows that $\delta\mathbf{v} = a\mathbf{u}$. In comoving coordinates the operator $\nabla_{\mathbf{x}}$ is also replaced with $\frac{1}{a}\nabla_{\mathbf{r}}$. Using these and $(a\mathbf{u} \cdot \nabla_{\mathbf{x}})\dot{a}\mathbf{r} = \mathbf{u}\dot{a}$, the equations can be written as

$$\frac{\mathrm{D}}{\mathrm{D}t}\frac{\delta\rho}{\rho_0} = -\nabla_{\mathbf{r}} \cdot \mathbf{u} \tag{2.24}$$

$$\frac{\partial\mathbf{u}}{\partial t} + 2\frac{\dot{a}}{a}\mathbf{u} = -\frac{\nabla_{\mathbf{r}}\delta P}{\rho_0 a^2} - \frac{1}{a^2}\nabla_{\mathbf{r}}\delta\phi. \tag{2.25}$$

$$\nabla_{\mathbf{r}}^2 \delta\phi = 4\pi G \delta a^2 \rho \tag{2.26}$$

Taking the comoving divergence, equation 2.25 gives

$$\nabla_{\mathbf{r}} \cdot \dot{\mathbf{u}} + 2\frac{\dot{a}}{a}\nabla_{\mathbf{r}} \cdot \mathbf{u} = -\frac{\nabla_{\mathbf{r}}^2 \delta P}{\rho_0 a^2} - \frac{1}{a^2}\nabla_{\mathbf{r}}^2 \delta\phi. \tag{2.27}$$

There the last term on the right side contains the left side of equation 2.26, the first term on the left can be found in equation 2.24 after taking the convective time derivative and the second term on the left already contains the right side of 2.24. Inserting these yields

$$\frac{\mathrm{D}^2}{\mathrm{D}^2 t}\frac{\delta\rho}{\rho_0} + 2\frac{\dot{a}}{a}\frac{\mathrm{D}}{\mathrm{D}t}\frac{\delta\rho}{\rho_0} = \frac{\nabla_{\mathbf{r}}^2 \delta P}{\rho_0 a^2} + 4\pi G \delta\rho. \tag{2.28}$$

The overdensities in the Universe are often denoted using the density contrast $\Delta = \delta\rho/\bar{\rho}$, where the bar denotes the universal mean (**?**). In this case the unperturbed density ρ is the mean density, so the left side of equation 2.29 can be expressed using Δ. If the density perturbations are assumed to be adiabatic, the adiabatic sound speed c_s relates perturbations in density and pressure as $\delta P/\delta\rho = c_s^2$. Inserting these yields

$$\frac{\mathrm{D}^2 \Delta}{\mathrm{D}^2 t} + 2\frac{\dot{a}}{a}\frac{\mathrm{D}\Delta}{\mathrm{D}t} = \frac{c_s^2 \nabla_{\mathbf{r}}^2 \Delta\rho_0}{a^2 \rho_0} + 4\pi G \Delta\rho_0. \tag{2.29}$$

Now a trial solution of $\Delta \propto e^{i(\mathbf{k}_r \cdot \mathbf{r} - \omega t)}$ corresponding to a wave with comoving wavevector $\mathbf{k}_r$ yields a wave equation

$$\frac{\mathrm{D}^2 \Delta}{\mathrm{D}^2 t} + 2\frac{\dot{a}}{a}\frac{\mathrm{D}\Delta}{\mathrm{D}t} = \Delta(4\pi G \rho_0 - k^2 c_s^2). \tag{2.30}$$

Here $\mathbf{k}_r$ has been replaced by $a\mathbf{k}$ to transform it to proper coordinates. The wave equation is a linear second-order partial differential equation that describes the evolution of perturbations.

Whether the waves described by equation 2.30 are oscillatory or unstable depends on the sign of the right side of the equation. If $c_s^2 k^2 > 4\pi G \rho_0$ the perturbations are oscillating sound waves supported by the internal pressure of the denser regions, but if $c_s^2 k^2 < 4\pi G\rho_0$ the wave is unstable and the modes grow (?). In a static universe these density perturbations grow exponentially but in an expanding universe the growth is slower: e.g. in simple Einstein-de Sitter universe with $\Omega_0 = 1$ and $\Omega_\Lambda = 0$ the growth is only algebraic (?).

2.1.4 Non-Linear Collapse

The equation 2.30 represents the perturbations well if all matter behaves as non-relativistic fluid so that Newtonian physics suffice to describe the perturbations, the perturbations are assumed to be spatially small compared to the observable universe and the density contrast is smaller than unity (?). In the present-day Universe it is clear that many structures with $\Delta \gg 1$ exist so the linear model alone is not sufficient to explain the evolution of structures. In the general case the evolution of these non-linear perturbations cannot be predicted analytically and simulations are often used instead to gain insight into their dynamics (?).

One special case in which an analytical solution can be presented is spherical top-hat collapse in which there is a uniform spherical density perturbation with no angular momentum inside a uniform density field (?). The evolution of such a perturbation resembles the evolution of an individual universe with $K = 1$: initially the perturbation expands with the surrounding Universe, but its expansion slows down relative to its surroundings and eventually a sufficiently dense perturbation reaches a point after which it starts to contract (?). As the size of the perturbation as a function of time is a cycloidic function, it is easiest to express in parametric form

$$a_p = \frac{\Omega_0}{2(\Omega_0 - 1)}(1 - \cos\theta) \tag{2.31}$$

$$t = \frac{\Omega_0}{2H_0(\Omega_0 - 1)^{3/2}}(\theta - \sin\theta) \tag{2.32}$$

where a_p is the relative size of the perturbation and t is time (?). Parameter θ evolves from 0 to 2π. From this equation one can see that a_p reaches its maximum at $\theta = \pi$. This maximum size is known as the turnaround radius of the perturbation and the corresponding time as the turnaround time (?).

At $\theta = 0$ and $\theta = 2\pi$ the model predicts a radius of zero and therefore an infinite density for the perturbation. Thus it is clear that in addition to the often unrealistic assumption of a perfectly uniform and spherical density perturbation the model has other limitations as well. In reality, smaller perturbations are present within the perturbation to cause the larger perturbation to fragment and the perturbation feels tidal forces from other perturbations that surround it, which apply torques (?). Different structures can also collide and merge.

The evolution of these fragments depends on their composition. Collisionless matter such as cold dark matter will simply experience relaxation and end up as virialized structures, but structures containing baryons have more variation (?). In gas, the non-linear evolution of the perturbation is more complicated as the gas feels pressure and shocks can occur when the gas compresses (?). Unless the gas is able to radiate away energy by effective cooling, the relaxation ends when the structure is in hydrostatic equilibrium (?).

In this work, only dark matter is studied. These collapsed objects made of dark matter are called dark matter haloes (?). The exact definition for when an object is dense enough to be considered a halo and where the edge of a halo lies vary somewhat depending on the source, but for halo catalogues analysed in this thesis a halo is defined to extend to the radius at which the mean density of a spherical volume drops below 200 times the critical density of the Universe. The mass and radius of such a halo are denoted M_{200} and r_{200}.

As the cosmological model and its parameters affect the structure formation, analysing the observations of structures at different stages of their evolution provides a way to compare cosmological models (?). Currently the standard model is the so-called ΛCDM model, according to which the Universe consists mostly of dark energy and cold dark matter in ratios given in section 2.1.2, with baryonic matter making up only a small fraction of total mass (?). Observations at the scale of the Local Group and its surroundings are not well suited for constraining the nature of dark energy, but the existence of dark matter can already be seen and its properties studied at scales of an individual galaxy (?).

For example the mass of possible warm dark matter particle can be greatly restricted (?) and hot dark matter made of standard model neutrinos can be excluded as a sole dark matter component based on the fact that non-linear structure has been able to form in the distribution of galaxies by the current age of the Universe (?). This is due to the properties of hot dark matter particles, i.e. low-mass dark matter particles that would have decoupled from the radiation while still relativistic and their thermal motions would allow them to escape from gravity wells and smooth out structures smaller than some tens of Mpc (?). The same connection between the particle mass and speed of thermal motions and thus size of smoothed-out structures also sets the lower limit for a warm dark matter particle mass.

2.2 The Local Group and Its Mass

Galaxy groups are systems of galaxies defined by the number of galaxies within a volume. Exact definitions vary, but typically they are required to have at least three galaxies and a volume with numerical overdensity of the order of 20 (?). The upper limit of the size of a galaxy group is set by the least massive galaxy clusters: again, the different definitions exist but typically if a group would have over 50 members with apparent magnitudes $m < m_3 + 2$, m_3 denoting the magnitude of the third brightest member, it is classified as a galaxy cluster instead (?).

The Local Group, the galaxy group containing the Milky Way, is the best-known of these galaxy groups as it offers great opportunity for precise observations: much fainter dwarf galaxies can be observed in it than in any other galaxy group and objects subtend large areas on the sky allowing smaller details to be observed than in more distant galaxy groups (??). This makes it an appealing target for testing astrophysical and cosmological models (?). The Local Group has two main members: the Milky Way and the Andromeda Galaxy (M31). In total more than hundred galaxies are known to exist within 3 Mpc of the Sun, more than half of these being satellites for either of the primaries (?). As the distance range is quite wide, some of the more distant galaxies are likely not bound to the Local Group, but on the other hand there are likely numerous faint dwarf galaxies that remain unseen (?). The number of these unknown galaxies within the Local Group could be as high as several hundred (?).

The number of galaxies is not the only uncertainty regarding the Local Group: also the total mass of the system and the individual masses of the Milky Way and M31 galaxies are still highly uncertain with different estimates easily differing by a factor of 2–3 (??). This is problematic as in addition to constraining dark matter models as mentioned in section 2.1.4, the Local Group has a key role in testing the currently dominant ΛCDM model. The model explains the large-scale structure of the Universe where it is able to make accurate predictions, but at distance scales of a single galaxy group the quality of the predictions suffers (?).

Two of the possible discrepancies between observations and ΛCDM predictions are the missing satellites problem and the too-big-to-fail problem. The first arises from the fact that both analytical calculations and cosmological simulations produce more low-mass satellites in systems similar to the Milky Way or M31 than can be observed (?). To some extent the number of luminous galaxies and dark matter haloes can be expected to differ as stars do not form in all haloes due to reionization and feedback processes such as supernova feedback (??). The problem is also somewhat alleviated by the high estimated number of satellite galaxies that are luminous but faint and yet unseen. It is still uncertain whether the missing satellites problem actually exists as many modern simulations such as (?) agree well with the observed number of faint satellite galaxies.

Regardless of the existence of the missing satellites problem, the most massive dark matter haloes in dark matter only simulations also seem to be more numerous than observed galaxies in the the Local Group are. This is known as the too-big-to-fail problem. For example the Milky Way has only three satellite galaxies with central densities high enough to allow maximum circular velocities of more than 30 km/s, but dark matter only simulations tend to produce double or even triple this number of similar mass dark matter haloes (?). In this case the galaxies should be massive enough to form stars regardless of reionization, but central masses of the satellites can still be reduced as a result of satellites interacting with the primary and e.g. tidal stripping and ram pressure stripping lowering the central densities (?). As with the missing satellites problem, the difference between the simulations and observations decreases as baryons and baryonic effects are included in the simulations (?).

Both of these possible problems are sensitive to the mass of the Local Group: a massive halo is expected to have more subhaloes than a less massive one has. Thus the mass of the Local Group and its members is an intriquing question not only for building up knowledge about our surroundings but also for testing cosmological models. Numerous studies have been conducted to find out its mass, and the following sections outline some of the methods that have been used.

2.2.1 Timing Argument

One possible way of estimating the lower end of the range of possible Local Group masses is to use the timing argument, first introduced by ?. It is based on the mutual kinematics of the Milky Way and M31 galaxies and the assumption that they have formed close to each other and are now on orbiting their mutual mass centre, approaching each other for the first time. This corresponds to the structure formation model presented in section 2.1.3: initially the expansion of the Universe drives the pair further away from each other, but eventually the mass of the system is able to overcome the expansion and the galaxies start to approach each other.

For a zero angular momentum system, **?** obtain an estimate of minimum total mass of the system using Kepler's third law

$$P^2 = \frac{4\pi}{GM}a \tag{2.33}$$

and the fact that energy is conserved:

$$\frac{GM}{2a} = \frac{GM}{D} - E_k. \tag{2.34}$$

Many of the quantities that appear in the equations are either known or can be approximated: the current distance D between the Milky Way and the center of mass of the system can be estimated using the distance to the M31 galaxy, an upper limit for period P can be obtained using the current age of the Universe and kinetic energy, E_k, only depends on the velocity of the galaxy, easily obtained using radial velocity measurements as long as the tangential velocity is assumed to be negligible. Thus what remains to be solved are semimajor axis, a, and the effective mass at the center of gravity, M. The lower limit for the mass of the system derived by **?** was 1.8×10^{12} M$_\odot$. This is clearly less than current estimates that tend to favour masses around 5×10^{12} M$_\odot$ (**??**), but still significantly larger than the observed baryonic mass content of the two galaxies (**?**).

The estimate can be improved by using the Kepler's laws in parametric form:

$$r = a(1 - \cos(E)) \tag{2.35}$$

and

$$t = \left(\frac{a^3}{\mu}\right)^{1/2} (E - \sin(E)) \tag{2.36}$$

where E is the eccentric anomaly and μ is gravitational parameter (?). These two equations can be combined to determine the value of $\frac{\mathrm{d}r}{\mathrm{d}t}$ as

$$\frac{\mathrm{d}r}{\mathrm{d}t} = \frac{\mathrm{d}r}{\mathrm{d}E}\frac{\mathrm{d}E}{\mathrm{d}t} = \frac{\mathrm{d}r}{\mathrm{d}E}\left(\frac{\mathrm{d}t}{\mathrm{d}E}\right)^{-1}. \tag{2.37}$$

Calculating the derivatives using 2.35 and 2.36 and inserting them yields the following equation:

$$\frac{\mathrm{d}r}{\mathrm{d}t} = \left(\frac{\mu}{a}\right)^{1/2} \frac{\sin(E)}{1 - \cos(E)}. \tag{2.38}$$

Solving for a and $\sqrt{\mu}$ from equations 2.35 and 2.36 yields

$$a = \frac{r}{1 - \cos(E)} \tag{2.39}$$

and

$$\sqrt{\mu} = \frac{a^{\frac{3}{2}}}{t}(E - \sin(E)) \tag{2.40}$$

respectively and inserting them into equation 2.38 gives

$$\frac{vt}{r} = \frac{\sin(E)\,(E - \sin(E))}{(1 - \cos(E))^2}, \tag{2.41}$$

where velocity v and distance r can be deduced from observations and if the Milky Way and M31 are on their first orbit then t is the age of the Universe. Now E can be solved numerically. Inserting equations 2.39 and the solved value of E into 2.40, a value for μ and the combined mass of the pair is acquired.

Getting an estimate is simple, but the results are burdened with the assumptions that are made about the system. Modern values for the velocity and distance yield a virial mass of $(4.23 \pm 0.45) \times 10^{12}$ $\mathrm{M}_{\odot}$ for the Local Group (?). This is a considerably higher mass than calculations of the Milky Way and M31 masses using kinematic tracers of the gravitational field suggest (?), but at least some of the effect might be explained by the fact that the timing argument is sensitive to a larger volume of mass compared to satellite galaxies and other kinematic tracers (?).

The assumption of the galaxies being on their first approach is likely to be valid as a higher number of completed orbits would result in a mass so high that it does not seem physically realistic. Applying Kepler's laws also requires the masses to be approximated as point masses. If the dark matter haloes of the galaxies are assumed to be spherically symmetrical, this is not a problem if the haloes are sufficiently far from each other, but at times when the center of one halo is encompassed in the other halo, the accuracy of the model is compromised. Indeed the mass estimates acquired using timing argument seem to be best consistent with the mass enclosed in two spheres surrounding the two galaxies with radii of half the distance between the galaxies (?).

The assumption of radial orbit is likely fairly good as the estimated tangential velocities are small. For example ? give a tangential velocity of 17 km/s for M31 with $v_t < 34$ km/s at 1 σ confidence, which is small compared to their radial velocity of -109.3 ± 4.4 km/s. It is also possible to carry out the calculations above for elliptical orbits as is done by ?. A non-zero tangential velocity increases the resulting mass: for example ? give a mass of $4.27 \pm 0.53 \times 10^{12}$ $\mathrm{M}_{\odot}$ when the tangential velocity is included. Even if external forces felt by the galaxies do not result in significant tangential velocities for the galaxies, they can affect the system in other ways. Large-scale structure surrounding the Local Group can apply forces and torques and smaller members of the Local Group interact with the primary members as was noted by ?.

2.2.2 Other Mass Estimation Methods

Other structures can also be used to infer the masses of Milky Way and M31 galaxies. For example ? use Leo I, a Milky Way satellite with high galactocentric velocity, to estimate the mass of the Milky Way by a calculation similar to the timing argument. This is done by assuming that the satellite is bound to the Milky Way which is the only body gravitationally influencing it and that the satellite is on a radial orbit, having passed its periapsis once and now moving away from the Milky Way and towards the apoapsis. This yielded a lower limit of 1.3×10^{12} $M_\odot$ for the mass of the Milky Way (?).

Another option is to use the Large and Small Magellanic Clouds, a pair of well-studied Milky Way satellites. For example ? use circular velocities within the Magellanic clouds and their distances and velocities relative to the Milky Way to estimate its mass. This is done by constructing the probability density functions (PDFs, see section 4.1.2) of these three measurements from a simulation halo catalogue and then using Bayes' theorem to find the Milky Way mass range that is most probable given the observations.

Naturally this type of analysis can suffer from effects not included in the model. If for example the system happens to be peculiar in some sense, its characteristics might cause effects that are not reflected in the PDFs of the selected variables. This is studied by e.g. ? who measure the effects of local galaxy density and nearby clusters and the fact that the Magellanic Clouds are a fairly close pair and thus rare. They find that subhalo pairs similar to the Magellanic Clouds are rare in haloes resembling the Milky Way and including the subhalo pair criterion in analysis used in ? brings the estimated mass down considerably, reducing the most probable mass by a factor of two (?).

Including more constraints for the sample from which the PDF is determined naturally brings it closer to the one that would represent the Milky Way, but the range of allowed values has to be large enough to cover the uncertainty of measurements from the real system. Tightening the criteria also requires having a larger simulation to find a statistically representative sample of haloes. Naturally the Milky

Way system can also be located in either of the tails of the probability distribution, which is true for all mass estimation methods based on statistics instead of physics.

Larger number of objects can also be used in analysis. For example the properties of satellite galaxies can be utilized in mass estimation as is done by e.g. **?** who use the velocity dispersion of the satellites as an indicator of the virial velocity of the host halo and **?** who use the ellipticity distribution of Milky Way satellites to constrain the range of likely masses. Other useful objects include individual high-velocity stars from which the galactic escape speed can be estimated (**?**) and the orbits of stellar streams (**?**) among others.

The mass can also be determined based on a larger volume than is occupied by the two main galaxies of the Local Group and their satellites: for example **?** and **?** estimate the mass of the Local Group by studying the local Hubble flow. This is possible as the gravitational interaction of the surrounding galaxies with the Milky Way and M31 galaxies cause the surrounding galaxies to recede slower than ones surrounding empty regions of the Universe, and the strength of this effect can be translated to a mass estimate using e.g. Bayesian analysis. The velocity dispersion around the Hubble flow can also be used as is done by for example **?**. The properties of the Hubble flow are also used in this thesis as one estimator of the Local Group mass.

Different papers present varying values for the masses of the Local Group and its galaxies, both due to the strengths and limitations of the methods used and differences in the used observed and simulated data. A collection of mass estimates for Milky Way is shown in figure 2.1 by **?**. The masses are calculated by different authors using various mass estimation methods, some of which have been briefly introduced in this section. The plot clearly illustrates the fact that currently the masses of the Local Group galaxies have errors of over a factor of two.

Especially the timing argument stands out from the other methods. This is in

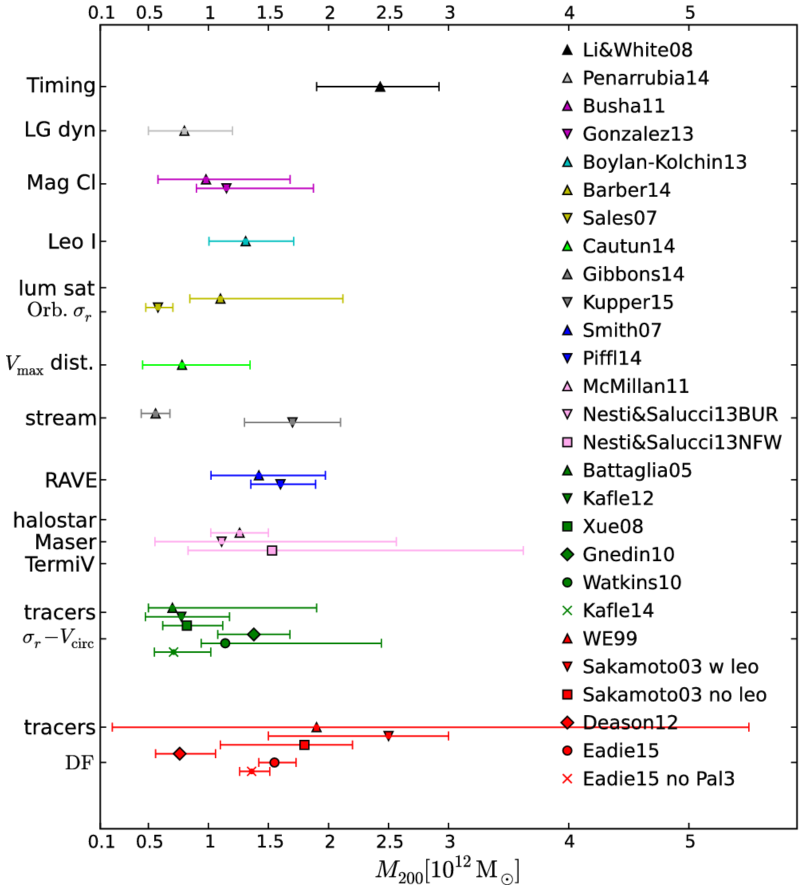

Figure 2.1: M_{200} masses of the Milky Way as obtained by different authors using various mass estimation methods, collected and plotted by ?. Error bars show the 1σ range where assuming Gaussian distribution was appropriate. Masses derived using similar methods are plotted in same colour.

agreement with a study of ? where timing argument masses were found to overestimate the true mass of a Local Group like system with small tangential velocity and low local overdensity by a mean factor of 1.6. This is due to the timing argument not being sensitive to the tangential velocity or environment of the system.

The other mass estimates also have considerable scatter, sometimes even within the same mass estimation method. At least some of this scatter can be a result of some estimates being more than ten years old and thus their methods and data possibly outdated, but the scatter also reflects the fact that the exact mass of the

Local Group is currently a question without a definitive answer. Thus exploring new ways of estimating the mass and refining the old results is important.

3. Simulations and Simulation Codes

Numerical simulations are a valuable tool in astrophysical research as they bypass many major restrictions characteristic to observational astronomy. For example the dark matter content of the Universe cannot be directly observed and following the evolution of a single object is impossible, with the exception of the most rapid events such as supernovae. Much can also be done using analytical models, but they are often valid only for a simplified problem or suffer from other limitations. For example the Zel'dovich approximation can be used to calculate the non-linear evolution of density perturbations in the Universe up to shell crossing, but after the structures have collapsed to sheets along one of their axes, the model ceases to be valid (**?**).

The data used in this master's thesis also originates from a cosmological N-body simulation. This section shortly introduces first some fundamental operational principles of N-body simulations and then discusses specifics of the simulations from which the data used here originates from. Lastly, the extraction and properties of the resulting data set are discussed.

3.1 N-Body Simulations

N-body simulations are a type of computer simulations that follow a number of particles interacting with each other (?). They are often used in computational astrophysics as well as other fields of physics. For some applications, including for example simulating motions of planets in a planetary system, it is possible to have each body represented by a single simulation particle. This kind of simulations are run using collisional N-body codes (?). In cosmological simulations the number of simulated objects such as stars let alone dark matter particles is too great for them to be simulated as separate particles. In this case a collisionless N-body simulation is used, meaning that the simulated volume is approximated to contain a smooth density field, evolution of which is studied by following a number of particles representing the system (?). These particles do not correspond to any real structures but instead they can be thought to sample the probability density distribution (see section 4.1.2 for more on probability distributions) of positions and velocities (?). In this thesis, only collisionless simulations are discussed.

The concept of both collisional and collisionless N-body simulations is simple: the current positions and velocities of the particles are known and a physical model is used to calculate how they should advance over a small period of time known as the time step (?). Simple dark matter only simulations might only handle gravitational interactions of the particles, but many simulation codes such as GADGET-3 (?) (see section 3.2.1 for more about GADGET family simulation codes) and Enzo (?) are also able to simulate hydrodynamics of baryonic matter and can include star formation, feedback and other baryonic processes.

As the particles of a collisionless system are not any real structures, e.g. stars, star clusters or galaxies, the situation in which two particles come near each other and thus feel strong forces resulting in large accelerations is problematic (?). This is because the underlying density distribution is smooth and thus such two-body scattering is unphysical. The problem can be alleviated by softening, i.e. modifying the gravitational force calculations so that for small interparticle distances the force diminishes (**?**). At large separations the force remains unchanged, and the distance within which the force differs from Newtonian gravity is called the softening length (**?**).

There are multiple ways to handle both the force calculations and updating the positions and velocities for the particles (**?**). For the force calculations, the most popular algorithms are based on either hierarchical trees or particle meshes, but for updating the positions, a wide variety of integrators have been developed for a range of different needs (**?**). The simulations discussed in this thesis are run using a modified version of GADGET-3 and thus use the TreePM method for force calculations accompanied by a leapfrog integrator. TreePM is a mix of a hierarchical tree for short-range forces and a particle mesh for long-range forces. The description of the particle mesh is omitted, but the **?** hierarchical tree algorithm is introduced in the following subsection, followed by a short description of the leapfrog integrator.

3.1.1 Hierarchical Tree Algorithm

In many applications of computational astrophysics the desired number of particles in a simulation is too great to allow calculating interparticle forces by direct summation as its time complexity is $\mathcal{O}(n^2)$, where n is the number of particles in the simulation. One of the alternatives is to organize the particles into a tree data structure, which allows distant particles residing close to each other to be approximated as a single more massive particle. This approach was first introduced by **?** and **?**, the latter of which will be followed here.

Constructing the tree starts with setting the full simulation box as the root of the tree. This root cube is then subdivided into eight equally sized sub-cubes called

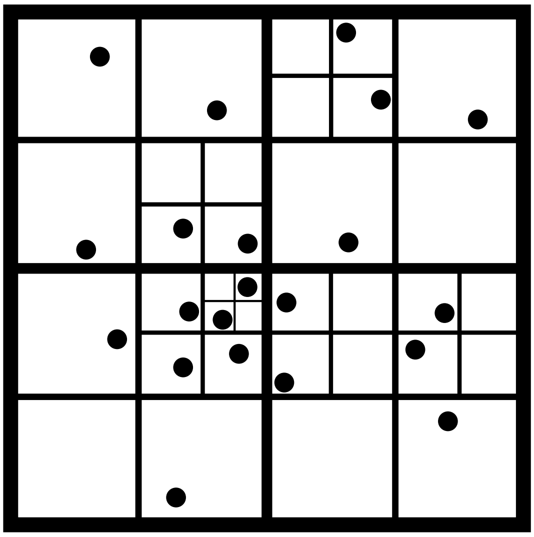

Figure 3.1: A two-dimensional example of particles (black dots) being assigned to cells using the Barnes-Hut algorithm (**?**). The outermost square is the simulation box, i.e. the root of the tree, and the smaller squares with decreasing line thicknesses are its descendants. Note how each cell contains either one particle, no particles or four smaller subcells, each of which may be further divided into subcells of their own.

cells. These eight cells are children of the root node. This starts a recursive process where each new cell is again divided into eight subcells recursively until each of the cells contains either no particles, one particle or eight subcells. When a new cell is created, the total mass of the enclosed particles and the location of the mass centre of the particles within it is stored as a pseudoparticle to allow easy calculation of the approximate gravitational effect the particles within a cell have on a distant particle.

To aid in understanding the process, a two-dimensional simplification of the simulation box divided into cells is shown in Fig. 3.1, together with the corresponding tree in Fig. 3.2. The thick outer line of Fig. 3.1 is the simulation box, corresponding to the topmost node of the tree in Fig. 3.2. This root cell is first divided into four subcells (in contrast to the eight subcells in the three-dimensional case), which is

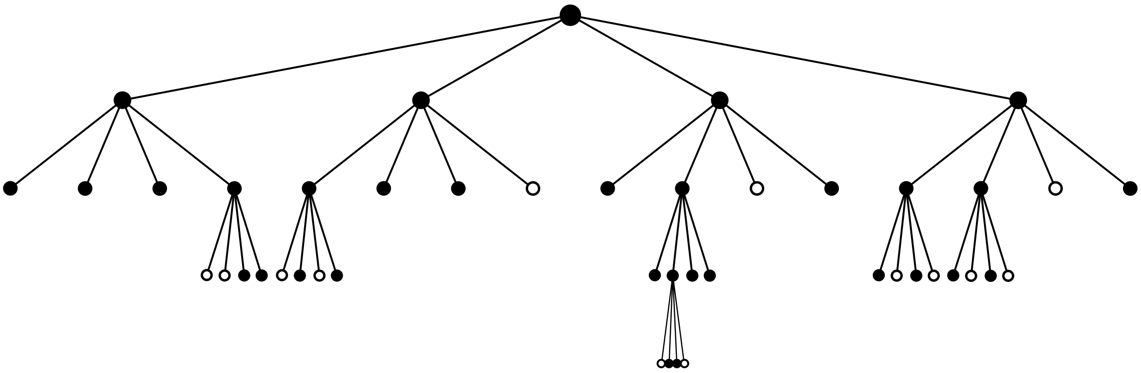

Figure 3.2: Cells of Fig. 3.1 shown as a tree, each level of subcells in Fig. 3.1 traversed from left to right and top to bottom corresponding to each level of nodes from left to right. Cells containing one or more particles are shown as filled circles. Cells with no particles are not used in force calculations and thus they do not need to be stored in computer memory, but they are shown here as non-filled circles to emphasize the structure of the tree.

shown with the second-thickest line dividing the simulation box into quarters in Fig. 3.1. These four cells are the four children of the root node in the tree. Each of the subcells contains more than one particle, so each subcell is again split into four quarters, each of them thus receiving four new child nodes. The newly-created quarters of quarters form the third level of the tree in Fig. 3.2.

Some of these new cells are empty or only have a single particle. For them the recursion halts and the nodes of the tree are leaves. The rest are again split and a new level is added to the tree, but as some cells were complete already the fourth level of the tree is not full. In this particular case only one cell requires division beyond the fourth level, producing the last four leaves of the tree on the fifth level.

Often, as is the case in this example, some of the leaf nodes contain no particles. In the case of a real simulation, these cells of course do not need to be saved as there is no information to store, but in this example case all are drawn to emphasize the regular structure of the tree. In the three-dimensional case, where each internal cell has eight subcells, the formed tree is known as an octree, whose two-dimensional analogue, the quadtree, is constructed in the previous example.

After constructing the tree, it can be utilized to speed up the force calculations. When calculating the gravitational acceleration felt by a single particle, the tree is traversed starting from the root. The ratio l/D of the length of a side of the cell (l) and the distance from the particle to the pseudoparticle representing the centre of mass within the cell (D) is then calculated. If the ratio is smaller than a predefined accuracy parameter θ representing the opening angle of the cell, the cell is treated as if everything within it was replaced with the pseudoparticle having the combined mass of the cell. Otherwise the subcells of the cell are examined recursively in the same way until a small enough subcell is found or a leaf of the tree is reached, at which point the pseudoparticle and the real particle in the cell are equivalent and the force can be calculated with the maximum accuracy possible for the simulation.

Using the Barnes-Hut algorithm, the tree construction and the force calculations both have time complexity of $\mathcal{O}(n \log n)$. This is a significant improvement over the $\mathcal{O}(n^2)$ of the direct summation considering that the accuracy cost is fairly small: **?** report accuracy of about 1 % for a single force evaluation when $\theta = 1$ and the accuracy can be improved by either setting a smaller θ or including multipole moments in the pseudoparticles (**?**).

The algorithm is also straightforward to parallelize as different branches of the tree can be assigned to their own threads, though memory management has to be done carefully when forces outside the current branch of a thread are calculated (**?**). One way to circumvent the memory management issue is to use a particle mesh based integrator for long range forces as GADGET-3 does (**?**). The algorithm can also handle problems where the range of densities in the simulation box is large, which is important for applications such as galaxy mergers and cosmological simulations. This, together with their competitive time complexity, makes tree based codes an appealing tool for many astrophysical simulations (?).

3.1.2 Leapfrog Integrator

A number of different integrators suitable for astronomical and astrophysical applications have been developed for solving different problems (?). No integrator is optimal for every task and thus factors such as the integration time, amount of memory available per particle, smoothness of the potential and the cost of a single gravitational field evaluation should be considered (?). One of the integrators that are well suited for cosmological simulations is the leapfrog integrator, which is also used by GADGET-3 simulation code (?).

When a fixed time step is used, the leapfrog integrator conserves the energy of the system and is time-reversible (?). While a variable time step is possible and often used, it requires some modifications to the algorithm, presented in e.g. ? and implemented in the GADGET-3 simulation code among others. Other benefits of the integrator are its second order accuracy and the fact that it does not require excessive amounts of memory per particle as only the current state of the system is needed in calculating the next step (?). It is also second-order accurate and symplectic (?). Due to symplecticity of the integrator, numerical dissipation of energy does not happen and thus the integrator rivals some integrators with higher-order accuracy such as the fourth order Runge-Kutta when the number of simulated time steps is large (?).

Timestepping with the leapfrog integrator consists of two phases, the drift and the kick steps, which are alternated with a half-step offset as shown in Fig. 3.3 (?). During the kick step, the momenta of the particles are updated and during the drift step the positions of the particles are changed according to the momenta calculated during the kick step (?). When synchronized output of both positions and velocities is desired, a half-timestep advance or backtrack to one of the variables is needed to

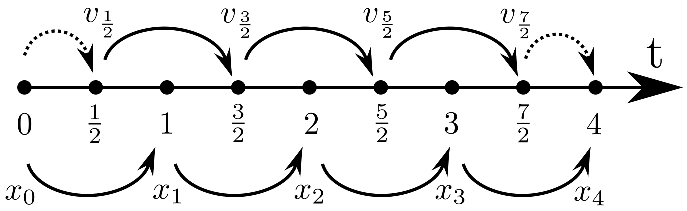

Figure 3.3: Timesteps taken by the leapfrog algorithm, with positions (x) updated as indicated with the lower arrows and velocities (v) as indicated by the upper arrows. It is not important for the algorithm which of the two is chosen to step through the integer times, the choice of it being x in this figure is arbitrary. The dashed short arrows depict the half-steps that are needed when a synchronized output is desired.

determine their values at the same point in time. This kind of synchronization steps at start and end of the integration are indicated in Fig. 3.3 with dashed arrows.

3.1.3 Halo Finding

The data output by a cosmological simulation consists of individual particles tracing the underlying density field. To make comparisons with the real Universe, a way of matching structures in the simulation to observable objects is needed. In a dark matter only simulation no structure would obviously be directly observable but luckily many properties of dark matter haloes from simulations can be compared with the estimated dark matter haloes of observed galaxies. Making such comparisons naturally requires structures to be identified first, which makes structure finding a key step in the data analysis for many astrophysical and cosmological applications (?).

A number of different halo finders designed to read N-body data and extract locally overdense gravitationally bound systems have been developed to suit different needs, most based either on locating density peaks or collecting and linking together particles located close to each other based on some metric (?). Two meth-

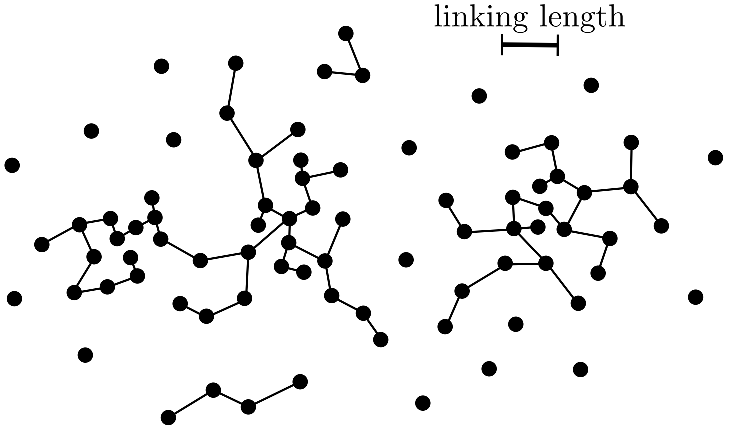

Figure 3.4: Friends-of-friends groups found from a mock data set using the length of the indicator in the upper right area of the illustration as the linking length. Particles are depicted as black dots and the links connecting particles within groups are shown with black lines.

ods, the friends-of-friends (FOF) and SUBFIND algorithms, are discussed here. Both algorithms were developed separately, FOF by **?** and SUBFIND by **?**, but they work well when used together by inputting FOF results to SUBFIND as a starting point for the subhalo finding (**?**).

The FOF algorithm is simple and based purely on the spatial separation of the particles: pairs of particles residing closer to each other than a chosen threshold distance called the linking length are marked to reside within the same group by linking them together (**?**). When all particles have been processed, each distinct subset of particles linked to each other is defined to be a group (**?**). Figure 3.4 presents an example of a set of particles grouped using the FOF algorithm. Depending on the specifics of the data and its intended usage, one might want to discard the smallest groups with only a few particles. This is because they are more likely than bigger groups to be just realizations of the random noise instead of actual physical structures, but also as they might not be massive enough to represent the structures that are being studied.

The algorithm is made appealing by its simplicity, small number of free parameters and the ability to find structures of arbitrary shape (?). However, it is prone to link unrelated structures via thin bridges that might consist of only a single chain of particles. This behaviour can be seen in Fig. 3.4: the leftmost group has two dense areas that could be cores of two separate groups connected by a single-particle bridge. Removal or even suitable movement of this particle would result in the group separating into two distinct groups. The two existing big groups in the figure also stretch quite close to each other in their lower parts: moving or adding one particle between the chains of particles protruding towards each other could result in the two groups merging. In addition, the algorithm as is cannot be used to detect substructure within larger objects (?). If desired, this can be done by some modified versions of the algorithm such as hierarchical friends-of-friends algorithm by ?.

In contrast to groups found by FOF, the SUBFIND algorithm has been developed to extract physically well-defined subhaloes that are self-bound and locally overdense from a given parent group. SUBFIND can work with an arbitrary parent group, but FOF groups are well-suited for parent groups. The algorithm is simple and with appropriately long linking length the FOF groups are unlikely to split a physical structure between FOF groups (?). It is still possible that independent structures end up in the same FOF group, but SUBFIND is able to distinguish them as two separate objects.

Unlike FOF, SUBFIND uses a local density estimate instead of individual particle pairs. It labels all locally overdense regions enclosed by an isodensity contour traversing a saddle point as substructure candidates (?). This is done by lowering an imaginary density threshold through the range of the density field: particles surrounding a common local density maximum are assigned to a common substructure candidate until two separate substructure regions join in a saddle point of the potential (?). When a saddle point is reached, the two substructure candidates it connects are both stored individually to be processed further and the saddle point particle is added to a new substructure candidate containing the particles from both of the smaller candidates (?). Thus the algorithm is able to identify a hierarchy of substructures within each other (?).

A two-dimensional example of this substructure candidate identification is shown in Fig. 3.5. The algorithm starts from the particle in the most dense area of the simulation. At that point, the particle has no neighbours that would be at a higher density than the particle itself, and thus it becomes the first particle of a substructure candidate. All of the particles belonging to this substructure candidate are marked with striping in the figure. The algorithm iterates through the particles in order of decreasing density, always finding that the next particle has one neighbouring particle in higher density area than the particle itself and thus adding it to the same dashed group, until the second local density maximum is reached. At that point, a new substructure candidate, marked with a checkerboard pattern, is created. The following particles are assigned to their respective substructure candidates based on their single higher potential neighbour until a saddle point between the two is reached, at which point the state of the substructure candidates is shown in Fig. 3.5. Then the current substructure candidates are complete and will be saved. Next the particles belonging to the two structures can be joined by the saddle point particle to form a new bigger substructure candidate.

Unfortunately, now some particles are assigned to multiple substructure candidates and it is not clear that all particles within one substructure candidate are part of an actual physical structure (?). Is is very much possible that some particles are just passing by and if the same particles were re-examined at a later time, they would no longer be anywhere near the structure they were supposed to belong to. Hence the next step in the analysis is to eliminate unbound particles from each group

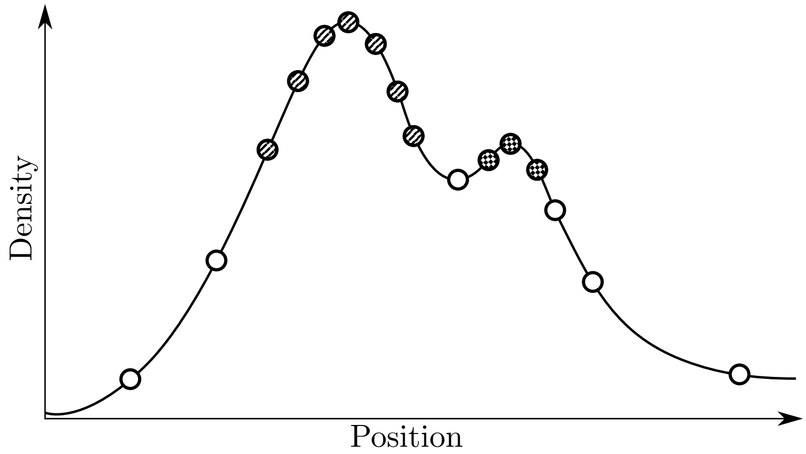

Figure 3.5: Intermediate stage of the SUBFIND algorithm, shown just before it reaches the first saddle point. Circles depict simulation particles and the line the underlying density field. Striped and checkered circles both belong to their own subhalo candidates whereas the white ones are not yet labelled.

by iteratively removing particles with positive total energy until all of the remaining particles are bound to each other by their mutual gravitational attraction (**?**). At this stage, each particle is labelled only based on the smallest structure it resides in, which solves the problem of a single particle belonging to multiple structures (**?**).

After the iterative pruning stage some substructure candidates can vanish completely or be left with very few members. These substructure candidates with less than some minimum number of bound particles can be discarded (**?**). The structure candidates surviving the pruning can then be considered to represent physical structures and are labelled as subhaloes (**?**). In this thesis, all of the analysis is based on catalogues containing such subhaloes.

3.2 Simulation Runs

[insert some form of attribution for the simulations, e.g. citation if available]. The simulations used in this thesis are cosmological zoom-in simulations, meaning that a higher resolution was used in regions of interest than in the rest of the simulation box. For these simulations, this was done by identifying regions relevant for the study from an output of a low-resolution simulation at $z = 0$, after which the resolution of these volumes was increased and the new simulations were run. In this case, the interesting regions were defined as the surroundings of a pair of dark matter haloes resembling ones of the Milky Way and Andromeda galaxies, the two main galaxies of the Local Group.

The simulations were dark matter only simulations, meaning that gravity is the only modeled interaction and all matter behaves as dark matter. For my analysis only the dark matter halo information was needed, so instead of using the full simulation output, the analysis was conducted on subhalo catalogues. The subhalo information was extracted as described in section 3.1.3. The resulting data set consists of subhalo catalogues from 448 zoom simulations, each centered on one region resembling the Local Group in the low-resolution simulation. The following sections shortly introduce the code used to run the simulation and the parameters of the simulations used for both stages of the zoom-in simulations.

3.2.1 Modified gadget-3

This master's thesis is based on data obtained from simulations run using a simulation code called GADGET-3, an update to the GADGET-2 which is described by **?**. The code is the same as used in the EAGLE project, so detailed descriptions of the changes between the versions can be found in **?**. However, the changes mostly affect the handling of baryonic matter so understanding the basic GADGET-2 gives a good basis for understanding the simulations (**?**).

GADGET uses a TreePM algorithm to compute forces, meaning that short-range forces are calculated using a tree method as described in section 3.1.1 and a particle mesh is employed for long-range forces (**?**). As the code is parallel, the normal tree-construction algorithm is problematic in regard to splitting the nodes of the octree between tasks (?). To ensure a balanced workload amongst all processors, the particles are split between tasks by constructing a space-filling fractal curve known as Peano-Hilbert curve, and splitting it into segments with approximately equal number of particles on each segment (?). The properties of the curve ensure that particles close to each other in space are usually also near each other along the curve, which means that close-range forces can frequently be calculated without the need to access memory belonging to other processors (?). In regions where an octree constructed from particles belonging to a processor should contain particles assigned to other processors, a pseudoparticle resembling in principle the pseudoparticles used when calculating forces for cells where $l/D < \theta$ is inserted, instead of the full particle information (?).

Updating the positions of the particles is done using the leapfrog integrator described in section 3.1.2, but the integrator is modified to allow using variable time step lengths (?). These modifications are important, as in cosmological simulations it is not sensible to use the same time step in all parts of the simulation as there are both high-density regions and sparse void areas, first of which requiring a time step so small that a lot of computational time is wasted while integrating the latter with more detail than needed.

3.2.2 Parent simulation

The first step in this kind of a zoom-in simulation is to construct initial conditions and run a low-resolution box. A large box with a comoving side length of 542.16 Mpc/h, h being the reduced Hubble parameter defined in equation 2.5, was used to ensure that the box is large enough to contain a sample of more than a hundred Local Group analogues and that the structure of the Universe is represented on all relevant scales in the simulation. At $z = 0$ the simulation volume is 800^3 Mpc^3

Box size	542.16^3 Mpc^3/h^3
Number of particles	$5040^3 \approx 1.3 \times 10^{11}$
Particle mass	1.6×10^8 M$_\odot$
Comoving softening length	2.3 kpc/h

Table 3.1: Properties of the parent simulation.

in physical units.

The volume contained about 1.3×10^{11} dark matter particles, each with a mass of about 1.6×10^8 M$_\odot$. This determines the mass resolution of the simulation: objects smaller than the mass of a single particle are not seen at all and reliable analysis cannot be made of objects that are made of fewer than 20–40 particles, the minimum number depending on the used halo finder (**?**). This means that for example distribution of spiral and elliptical galaxies can be studied but all dwarf galaxies are not resolved. The spatial resolution of the simulation is determined by the softening length: the distance within which the gravitational interaction calculations are modified in order to remove singularities and avoid extreme accelerations. The properties of the parent simulation are shown in Table 3.1.

In the parent simulation a comoving softening length of 2.3 kpc/h was used, meaning that the softening length grows with the simulation box. For regions expanding with the simulated universe this is advantageous, but for e.g. virialized structures with no spatial growth this is an unwanted behaviour. For simulations aimed at studying objects with specific mass and thus specific redshift when they collapse, it is possible to specify a redshift after which the softening length is kept constant in physical units, but for simulations exploring a range of different objects this is not possible.

The cosmology of the simulation is a standard ΛCDM cosmology with cosmological parameters from the Planck space telescope measurements (**?**). The most

H_0	67.77 km/s/Mpc
Ω_m	0.307
Ω_b	0.0455
Ω_Λ	0.693
σ_8	0.8288

Table 3.2: Most important cosmological parameters of the simulations.

important parameters are shown in Table 3.2. For a dark matter only simulation the Ω_b, corresponding to the baryonic component of the Universe, is treated as dark matter. The σ_8 parameter shown in the table is defined as the amplitude of the primordial power spectrum at scales of 8 Mpc/h.

The low-resolution simulation was then run and when it reached redshift $z = 0$, haloes were identified using the FOF and SUBFIND algorithms as described in section 3.1.3. From the resulting subhalo catalogue, halo pairs resembling the Local Group were identified. In the parent simulation, the following properties were required from a pair of haloes:

- Mutual distance of the two haloes lies between 700 and 840 kpc.
- The haloes are moving towards each other with a radial velocity between 100 and 170 km/s.
- Tangential velocity of the haloes is smaller than 50 km/s.
- Haloes have masses between 5×10^{11} and 5×10^{12} $M_\odot$.
- The two haloes are the most massive ones within 2 Mpc.

This criterion produced 448 hits in the parent box and a zoom simulation was created for each of these Local Group analogues.

3.2.3 Zoom simulations

The zoom simulations simulated the same box as the parent simulation, but new initial conditions with enhanced resolution around Local Group analogues and coarser resolution elsewhere were created for each pair of objects. The large scale structure and evolution in these zooms is similar to the parent simulation, but due to increased resolution near the Local Group analogues they can be studied in more detail. The change of resolution was implemented by grouping the dark matter particles to three categories ranging from the least massive type 1 particles to the most massive type 3 particles.

Whereas type 1 particles have a fixed mass in each simulation, type 2 and 3 particles have masses ranging from type 2 particles with only slightly larger mass than the type 1 particles to type 3 particles with masses comparable to a million type 1 particles. Masses of type 2 and 3 particles vary, but each particle type has its own softening length with type 1 particles having the smallest and type 3 particles the largest softening lengths. While the number and mass of type 1 particles varied slightly from simulation to simulation, ranging from 3.7×10^7 to 4.7×10^7 $\mathrm{M}_{\odot}$ per particle, every simulation used the same softening length of 1.0 kpc/h for them. The GADGET-3 simulation code is also capable of handling baryonic matter such as gas and stars, but the simulations used in this thesis were dark matter only simulations so no other particles than the three types of dark matter particles were present.

The type 1 particles were assigned to regions identified to end up in or pass by the central Local Group analogue, surrounded by a region of type 2 particles. The rest of the box was filled with type 3 particles. Where two different types of particles interact, the larger of the two softening lengths is applied for the interaction in order to reduce unphysical two-body scatterings. The number and mass of type 1 particles in a simulation varies based on the extent and density of the regions forming the Local Group analogue. A histogram showing the numbers of high resolution particles

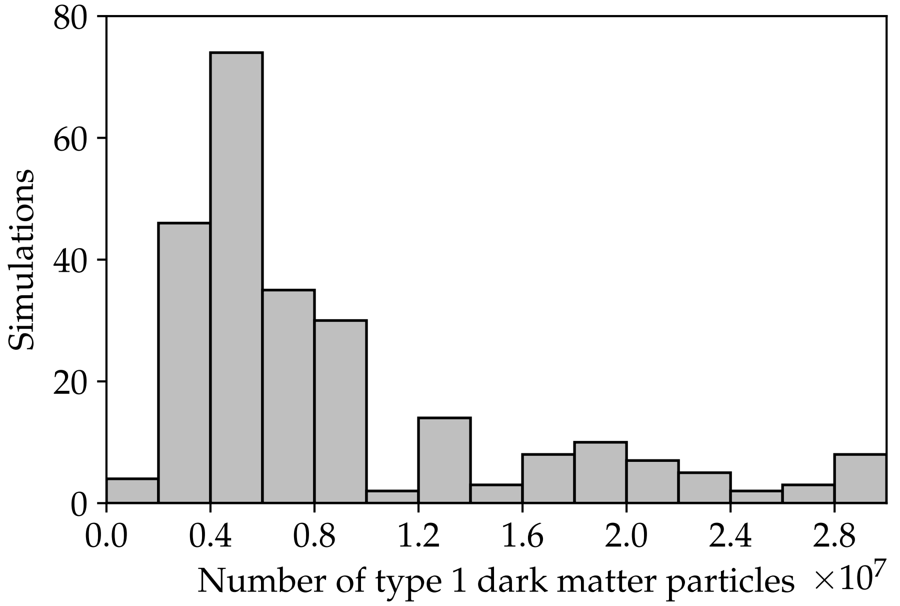

Figure 3.6: The number of simulations with given number of type 1 particles. Values near the lower end of the distribution are most common, but many simulations have more than 2×10^7 particles.

in each simulation is presented in Fig. 3.6. The number of type 1 particles ranges from about 1.6×10^6 to nearly 30×10^6 with lower values being most typical: most of the simulations have less than 7×10^6 high resolution particles.

At $z = 0$ a subhalo catalogue was created for each of the zoom simulations. This was done using a combination of FOF and SUBFIND algorithms as was done with the parent simulation. A lower limit of 20 particles in each subhalo was used and smaller were structures ignored. Instead of the full particle data, all analysis conducted for this thesis was done using these subhalo catalogues. Initially there were 448 of these catalogues in the data set. Simulations were run and halo catalogues generated by Till Sawala but all further analysis presented is done as a part of this thesis.

The 448 simulation runs did not easily translate to 448 Local Group analogues. First of all, picking the Local Group analogues from the parent simulation had resulted in most halo pairs being picked twice. Two zoom simulations were then run for each of these pairs resulting essentially the same simulation being run twice, with the second run providing no new information. Removing one of each of these pairs from the data set resulted in a total of 250 unique zoom simulations.

The analysis started with identifying Local Group analogues in all of the catalogues. This was done by identifying pairs of haloes that are made of type 1 particles and fulfil the following criteria:

- Distances between centres of potential are in range 0.6–1.0 Mpc.
- The haloes are moving towards each other with radial velocity no larger than 170 km/s.
- Tangential velocity of the haloes is smaller than 50 km/s.
- Masses of individual haloes lie between 4×10^{11} and $5 \times 10^{12} \mathrm{M}_{\odot}$.
- The two haloes are the most massive ones within 2 Mpc.

The criteria were chosen so that the systems resemble the Local Group but the allowed ranges are broad enough to accommodate uncertainties in the observed values and to produce a large enough data set for statistical analysis. The ranges are somewhat wider than with the parent simulation, which reduces the probability of a Local Group analogue that was found in the parent simulation no longer fulfilling the criteria after resimulation. This is important for statistical analysis as with the parent simulation criteria only 58 Local Group analogues are found in the zooms, compared to the 199 simulations with Local Group analogues found with these criteria.

Even with the loose criteria, the sample size was reduced from the number of unique simulations as some resimulations did not contain a pair of haloes fulfilling the Local Group criteria. This can happen as the zooms are not identical to the parent

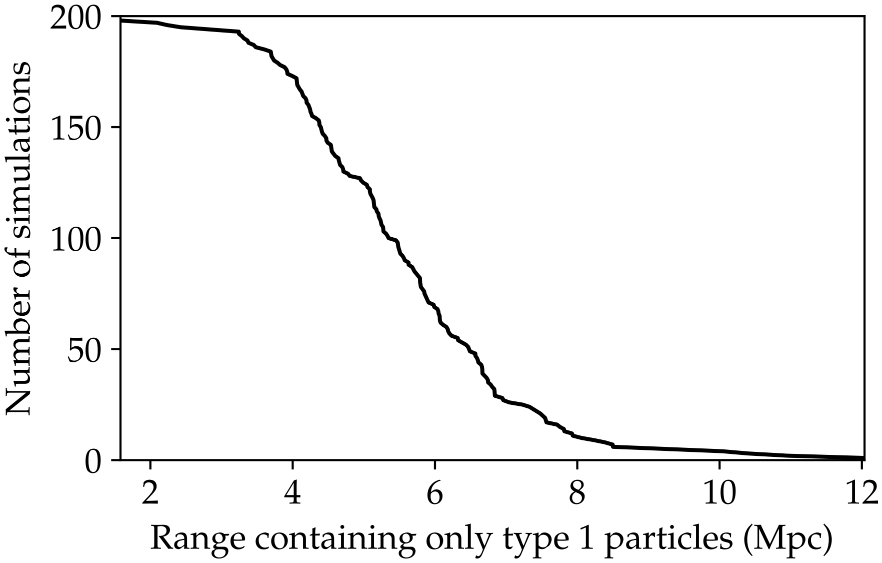

Figure 3.7: Number of simulations that have only high resolution type 1 particles at least up to a given distance as a function of distance.

simulation and especially Local Group analogues with some value near the edge of the allowed range can end up no longer meeting all criteria after resimulation. Some of the simulations also contained more than one Local Group analogue, in which case the pair with largest distance to the nearest type 2 or 3 particle was chosen for analysis.

As the analysis conducted for this thesis uses not only the properties of a Local Group analogue but also its surroundings, the extent of high resolution regions is important. As can be seen in Fig. 3.7, most of the simulations contain only type 1 particles up to 4 Mpc from the Local Group analogue. For larger distances, the number of simulations uncontaminated by type 2 or 3 particles decreases until after 8 Mpc only few simulations have only type 1 particles. TODO: sum up the fact that some analysis required excluding some simulations based on contamination

4. Mathematical and statistical methods

täällä tarvittavat esitiedot ja önnönnöö, listaa mm. mitä aiot kertoa kunhan tiedät itsekään

4.1 Statistical Background

Precision of the used equipment limits accuracy of all data gathered from physical experiments, simulations or observations. Thus assessing whether for example measurements support a model requires using statistical methods. The main methods relevant for this thesis are covered here. The methods are shortly introduced in the following sections together with basic statistical concepts that are necessary to understand the methods.

4.1.1 Hypothesis testing and p-values

A common situation in scientific research is that one has to compare a sample of data points to either a model or another sample in order to derive a conclusion from the dataset. In statistics, this is known as hypothesis testing (**?**). For example, this can mean testing hypotheses such as “these two variables are not correlated” or “this sample is from a population with a mean of 1.0”. Next paragraphs shortly introduce the basic concept of hypothesis testing and methods that can used to test the hypothesis “these two samples are drawn from the same distribution” following the approach of **?** and **?**.

The process of hypothesis testing as described by **?** begins with forming a null hypothesis H_0 that is formatted such that the aim for the next steps is to either reject it or deduce that it cannot be rejected with a chosen significance level. The negation of the null hypothesis is often called research hypothesis or alternative hypothesis and denoted as H_1. For example, this can lead to H_0 “this dataset is sampled from a normal distribution” and H_1 “this dataset is not sampled from a normal distribution”. Choosing the hypotheses in this manner is done because often the research hypothesis is difficult to define otherwise.

After setting the hypothesis one must choose an appropriate test statistic. Ideally this is chosen such that the difference between cases H_0 and H_1 is as large as possible. Then one must choose the significance level α which corresponds to the probability of rejecting H_0 in the case where H_0 actually is true. This fixes the critical region i.e. the values of test statistic that lead to the rejection of the H_0. This kind of probability based decision making is always prone to error. Rejecting H_0 despite it being true is known as error of the first kind. However, this is not the only kind of error possible. It might also occur that H_0 is false but it does not get rejected, which is known as error of the second kind.

There is no one optimal way of choosing α, but instead one should try to find a balance between false rejections of the null hypothesis and not being able to reject the null hypothesis based on the dataset even if it is false. When sample size (often denoted N) is large, smaller values of α can often be used as decisions get more accurate when N grows. For example tässä työssä α oli jokin ja N jotain muuta. TODO!

It is crucial not to look at the test results before choosing α in order to avoid intentional or unintentional fiddling with the data or changing the criterion of acceptance or rejectance to give desired results. Only after these steps should the test statistic be calculated. If the test statistic falls within the critical region, H_0 should be rejected and otherwise stated that H_0 cannot be rejected at this significance level. The critical values for different test statistics are widely found in statistical textbooks and collections of statistical tables or they can be calculated using statistical or scientific libraries available for many programming languages.

Despite statistical tests having a binary outcome "H_0 rejected" or "H_0 not rejected", a continuous output is often desired. This is what p-values are used for. The name p-value hints towards probability, but despite its name p-value is not equal to the probability that the null hypothesis is true. These p-values are functions of a test statistic and the p-value for a certain value t_{obs} of a test statistic gives the probability that under the condition that H_0 is true, the value of a test statistic for a randomly drawn sample is at least as extreme as t_{obs}. Therefore, if the p-value is smaller than α, H_0 is to be rejected.

4.1.2 Distribution functions

Some statistical tests such as the Kolmogorov-Smirnov test and the Anderson-Darling test make use of distribution functions such as cumulative density function (CDF) and empirical distribution function (EDF) in determining the distribution from which a sample is drawn.

To understand CDF and EDF, one must first be familiar with probability density function (PDF). As the name suggests, a PDF is a function the value of which at some point x represents the likelihood that the value of the random variable would equal x. This is often denoted as $f(x)$. Naturally for continuous functions the probability of drawing any single value from the distribution is zero, so these values should be interpreted as depicting relative likelihoods of different values. For example if $f(a) = 0.3$ and $f(b) = 0.6$ we can say that drawing value b is twice as likely as drawing value a. (**?**)

Another way to use the PDF is to integrate it over a semi-closed interval from negative infinity to some value a to obtain the CDF, often denoted with $F(x)$:

$$F(x) = \int_{-\infty}^{x} f(x')\, dx'. \tag{4.1}$$

This gives the probability of a random value drawn from the distribution having value that is smaller than x. The relation between the PDF and the CDF is illustrated in Fig. 4.1, where PDFs and CDFs are shown for three different distributions. It is easy to see the integral relation between PDF and CDF and how wider distributions have wider CDFs. (**?**)

Both the PDF and the CDF apply to whole populations or to the sets of all possible outcomes of a measurement. In reality the sample is almost always smaller than this. Therefore one cannot measure the actual CDF. Nevertheless, it is possible to calculate a similar measure of how big a fraction of measurements falls under a given value. This empirical counterpart of the CDF is known as empirical distribution function (EDF), often denoted $\hat{F}(x)$, and for a dataset $X_1, X_2, ..., X_n$ containing n samples it is defined to be

$$\hat{F}(x) = \frac{1}{n}\sum_{i=1}^{n} I[X_i \leq x] \tag{4.2}$$

where I is the indicator function, value of which is 1 if the condition in brackets is true, otherwise 0. (**?**)

Due to the EDF being a result of random sampling, it may deviate from the underlying CDF considerably as can be seen by comparing CDFs in Fig. 4.1 and corresponding EDFs in Fig. 4.2. This example is somewhat exaggerated with its N=35 as the actual dataset used in this thesis has N>100, but reducing the sample size makes seeing the effects of random sampling easier. The latter figure also has EDFs corresponding to two random samples drawn from the distribution of the

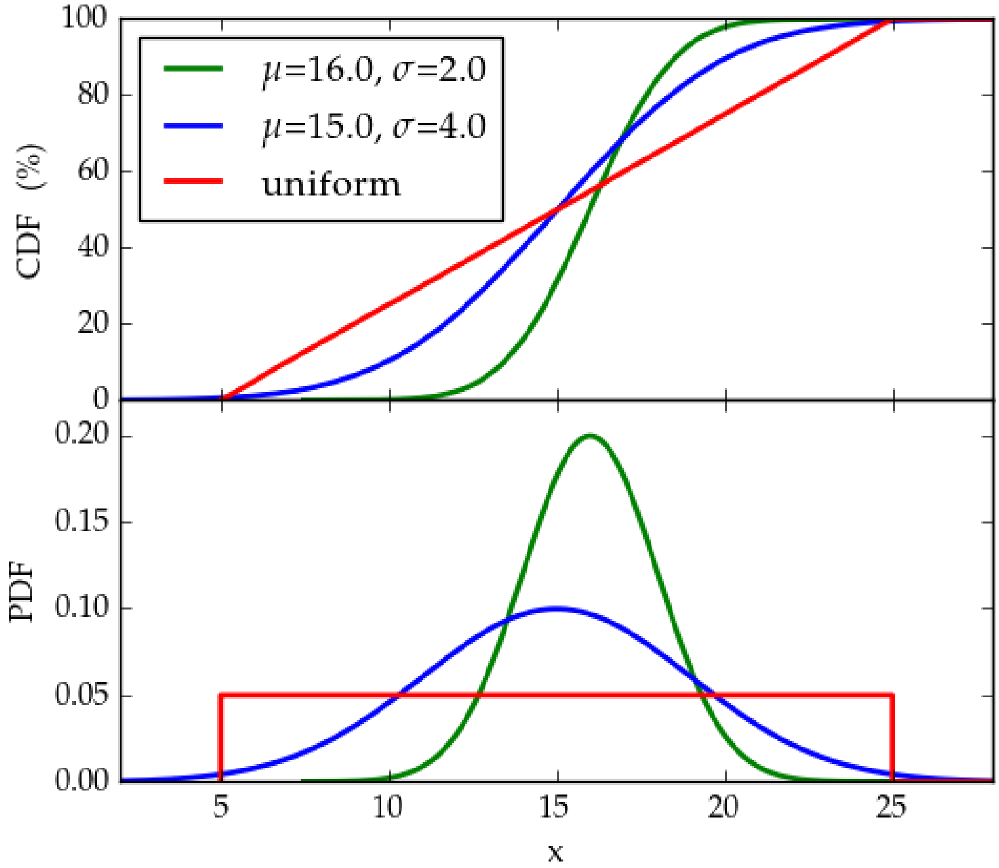

Figure 4.1: Cumulative distribution functions (top panel) and probability distribution functions (bottom panel) for three random samples drawn from different distributions, two of which are normal and one is uniform. Parameters μ and σ of the normal distribution describe the mean and the spread of the distribution respectively, large values of σ corresponding to a wide distribution.

green curve in the first figure to further illustrate the differences that can arise from random sampling. This randomness also makes determining whether two samples are drawn from the same distribution difficult.

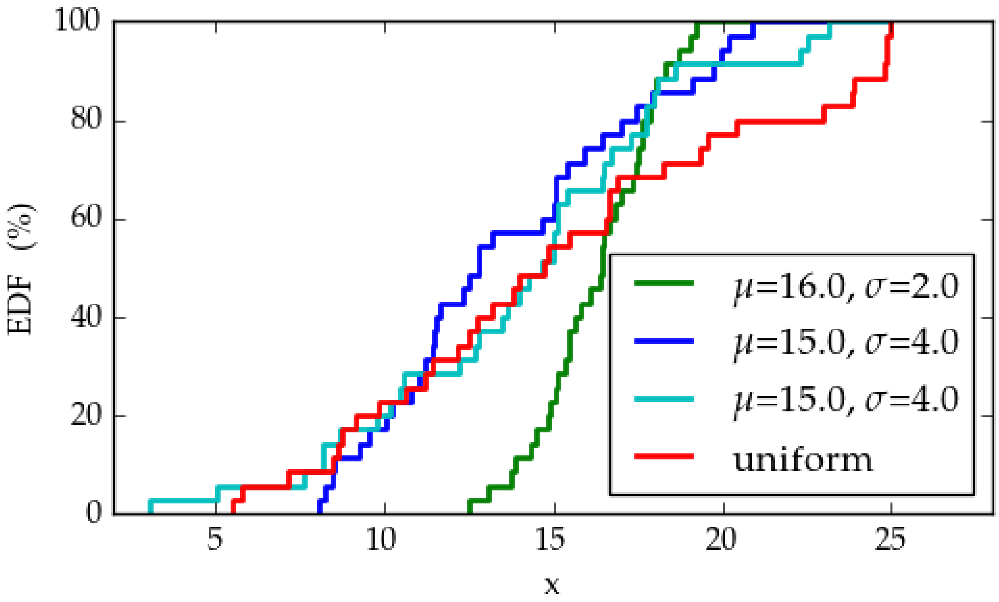

Figure 4.2: Empirical distribution function for four random samples (N=35) drawn from the same distributions as in Fig. 4.1. Note that both the blue and the cyan data are drawn from the same distribution.

4.2 Linear Regression

Regression analysis is a set of statistical analysis processes that are used to estimate functional relationships between a response variable (denoted with y) and one or more predictor variables (denoted with x in case of single predictor or $x_1 \dots x_i$ if there are multiple predictor variables) (?). In this section, we will cover both simple regression, where there is only one response variable, and multiple linear regression, where there are more than one predictor variables. The models also contain an ε term that represents the scatter of measured points around the fit. One of the models used is the linear regression model, which can be used to fit any relationship where the response variable is a linear function of the model parameters (?). In addition to the widely known and used models where the relationship is a straight line, such as $y = \beta_0 x + \varepsilon$, all models where relationship is linear in unknown parameters β_i

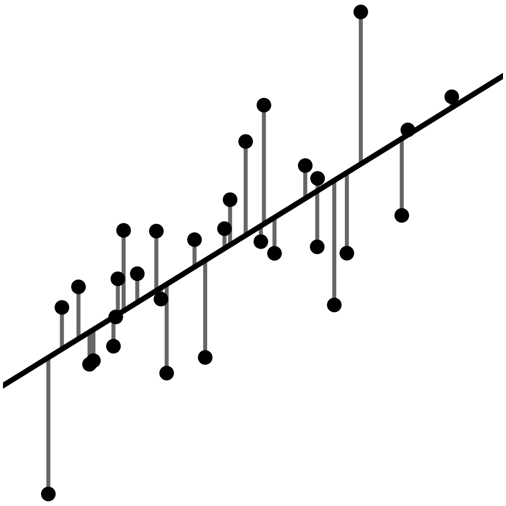

Figure 4.3: Ordinary least squares fit (black line) to a data set (black circles) with the response variable on the vertical axis. The fitted line minimizes the sum of squares of vertical distances between the data points and the fitted line, shown in grey.

are linear (?). Thus for example $y = \beta_0 x^2 + \varepsilon$ and $y = \beta_0 e^x + \beta_1 \tan x + \varepsilon$ are linear models. On the other hand, all models where the relationship is not linear, for example $y = x^{\beta_0} + \varepsilon$ and $y = \beta_0 x + \cos(\beta_1 x) + \varepsilon$, are nonlinear.

4.2.1 Simple linear regression

Simple linear regression is a model with a single predictor variable and a single response variable with a straight line relationship, i.e.

$$y = \beta_0 + \beta_1 x + \varepsilon \tag{4.3}$$

where parameter β_0 represents the y axis intercept of the line and β_1 is the slope of the line (?). The parameters can be estimated using method of least squares, i.e. choosing parameter values that minimize the sum of squared differences between the data points and the fitted line (?).

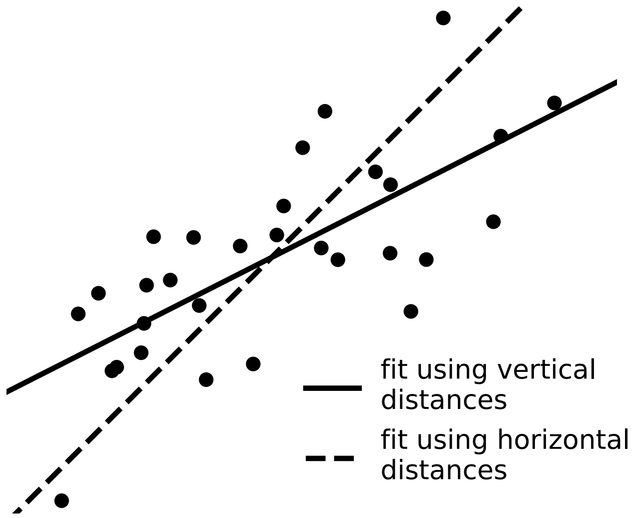

Figure 4.4: Ordinary least squares fits using either vertical or horizontal distances, both resulting in different fit. In order to avoid such ambiquity, ordinary least squares should only be used when the predictor variable is measured exactly.

The best-known method of minimizing the sum of squared error is the ordinary least squares (OLS) estimator. The OLS method uses differences in the response variable as shown in Fig. 4.3 and thus the minimized sum is

$$\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i) \tag{4.4}$$

where x_1 and y_i are single values of the measured quantities (**?**). This approach requires that the values of the predictor variable are known exactly without error and all uncertainty is in the values of the response variable (**?**). In those situations where this assumption is not valid, results acquired using OLS may be counterintuitive. This can be seen for example in Fig. 4.4 where OLS is used to calculate two linear fits: one where x is used and predictor variable and y as response variable and another where y is the predictor and x the response. As the minimized distance is different, the fitted lines also differ.

When dividing the variables to the independent variable with no error and a response variable with possible measurement error is not a justifiable choice, OLS

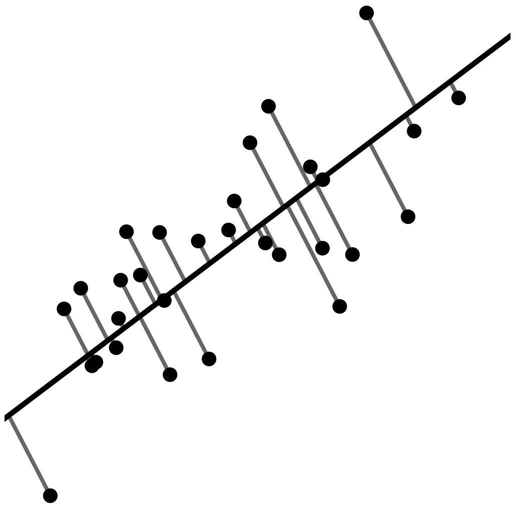

Figure 4.5: Total least squares minimizes the sum of squares of orthogonal distances of the data points from the fitted line. Distances between the data points (black circles) and the fitted line (black line) are shown in grey.

should not be used. For example when fitting the Hubble flow to observations, both observed distances and radial velocities are likely to contain some uncertainty and thus OLS is not an optimal choice. One alternative for OLS is total least squares (TLS, also known as orthogonal least squares in some sources such as **?**) regression can be used instead of OLS (**?**). The major difference between OLS and TLS is that instead of vertical distance, the minimized squared distance is measured between a point and its projection to the fitted line, thus providing minimum of the sum of the squared orthogonal distances from the line (**?**). These minimized distances are shown in Fig. 4.5.

4.2.2 Multiple linear regression

4.3 Principal Component Analysis

Principal component analysis (PCA) is a statistical procedure first introdueced by **?** to aid physical, statistical and biological investigations where fitting a line or a plane to n-dimensional dataset is desired. When performing PCA, one transforms a data set to a new set of uncorrelated variables i.e. ones represented by orthogonal basis vectors. These variables are called principal components (PCs) (**?**). This approach also solves the problem of sometimes arbitrary choice of division of the data into dependent and independent variables introduced in section 4.2.1 (**?**).

PCA can be used to both reduce and interpret data (**?**). Often PCA alone does not produce the desired result, but instead PCs are used as a starting point for other analysis methods such as factor analysis or multiple regression (**?**). These applications are introduced in the following subsections together with a short description of performing PCA and interpreting its results. In addition to these applications, PCA is also used in image compression, face recognition and other fields (**?**).

4.3.1 Extracting Principal Components

In order to understand the process of obtaining principal components of a data set let us follow the procedure on a two-dimensional data set shown in the top panel of Fig. 4.6 with black dots. The first step of finding the PCs is to locate the centroid of the dataset i.e. the mean of the data along every axis (**?**). This is marked with a red x in the top panel of Fig. 4.6.

The best-fit line and therefore the PCs always pass through the centroid of the system (**?**), so subtracting the location of the centroid from the data is a natural next step, as this ensures that in the next step only the slope has to be determined.

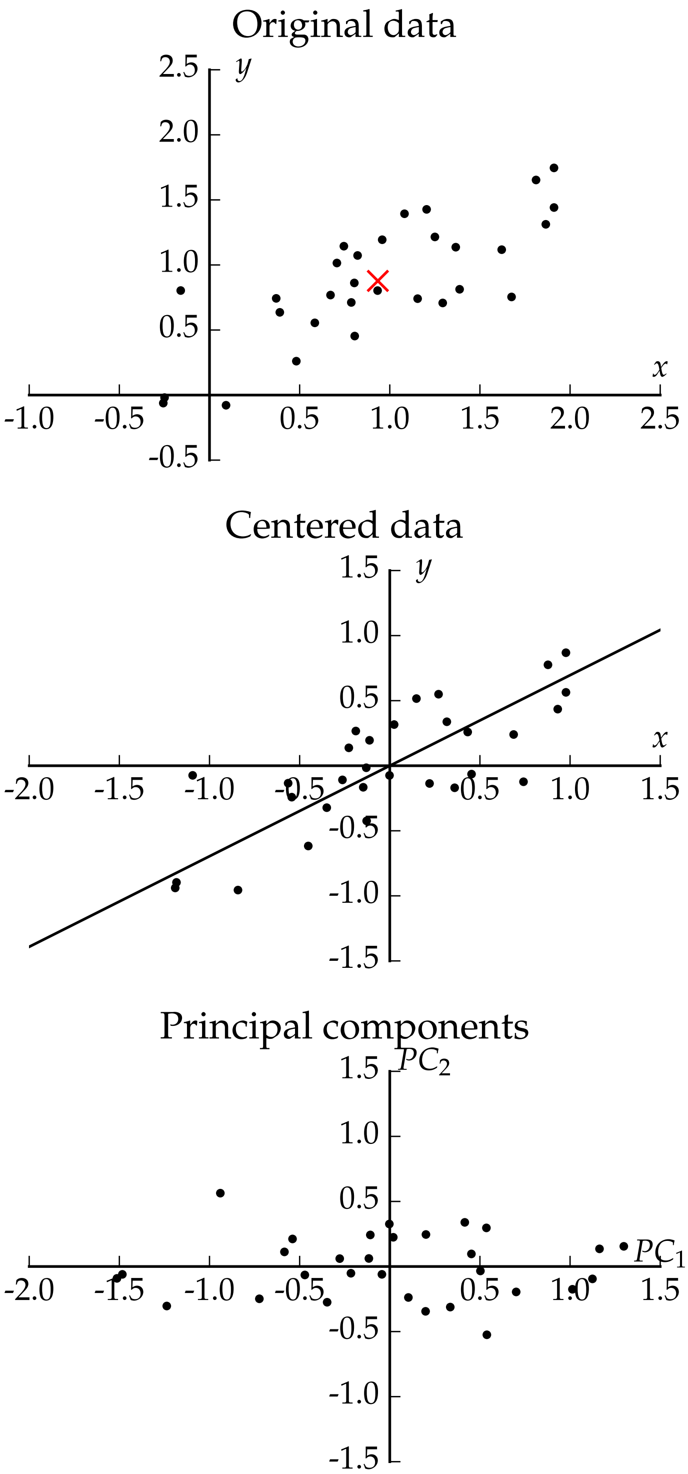

Figure 4.6: Extracting the PCs of a two-dimensional data set. First the origin is moved to the centroid of the data, original location of which is shown with a red x in the top panel. Next the line along which the variance of the data points is largest is determined, shown with the black line in the middle panel. This is the first PC, to which the second PC is orthogonal and thus fully determined. In the bottom panel the data is plotted along these principal components.

This is done in the middle panel of the Fig. 4.6. If the variables have different units, each variable should be scaled to have equal standard deviations (**?**) unless the linear algebra based approach with correlation matrices, as explained in e.g. **?**, is used.

If this scaling is not performed, the choice of units can arbitrarily skew the principal components. This is easy to see when considering for example a case where one has distances to galaxies in megaparsecs and their masses in units of 10^{12} $M_{\odot}$, both of which might result in standard deviations being of the order of unity and PCA might thus yield principal components that are not dominated by neither variable alone. Now, say another astronomer has a similar data set, but distances are given in meters. In this case, most of the variation is in the distances, so distances will also dominate the PCs. If all variables are measured in the same units, scaling can be omitted in some cases (**?**).

Now the first PC can be located by finding the line that passes through the origin and has the maximum variance of the projected data points (**?**), shown with a black line in the middle panel of Fig. 4.6 for our data set. PCs are always orthogonal and intersect at the origin, so in the two-dimensional example case the second and final PC is fully determined. The data set can now be represented using the PCs as is shown in the bottom panel of the Fig. 4.6.

For a data set with more than two dimensions, the second PC is chosen such that it and the first PC are orthogonal and that variance along the new PC is again maximised (**?**). This can be repeated for each dimension of the data set or, if dimensionality reduction is desired, only for a smaller number of dimensions.

This level of understanding is often enough to successfully apply PCA to a problem, because PCA has ready-made implementations for many programming languages such as `prcomp` in R (**?**) and `sklearn.decomposition.PCA` in the scikit-learn library for Python (**?**). If a more mathematical approach is desired, **?** explains

PCA together with covariance matrices, eigenvectors and eigenvalues required to understand the process very clearly. **?** also includes a very thorough description of PCA.

4.3.2 Excluding Less Interesting Principal Components

Even though a data set has as many principal components as there are measured variables, one is often not interested in all of them as the last principal components might explain only a tiny fraction of the total variation in the data (**?**). Reducing the dimensionality of the problem also greatly eases visualizing and interpreting the data. Thus one might want to retain only the first few PCs when PCA is used to, for example, compress, visualize or just interpret a data set (**??**). Unfortunately, many of the rules and methods used to determine the number of PCs to retain are largely without a formal basis or require assuming a certain distribution which is often not justifiable with the data (**?**). With careful consideration these methods can nevertheless aid a researcher in making informed decisions and reasoned conclusions, so some rules are introduced in this section.

If the PCA is performed to aid visualizing the data set, retaining only the two first PCs can be a justified choice as two is the maximum number of dimensions that are easy to visualize on two-dimensional media such as paper and the two first PCs determine the best-fit plane for the data (**?**). Of course the question whether the two PCs are sufficient to describe the data reasonably well still remains unanswered in this case.

One widely used technique was introduced by **?** to be used in factor analysis, but is also very much applicable to PCA (**?**). This so called Cattell scree test involves plotting the variance of the data points along each PC versus the index of the PC. These plots tend to look similar to what is shown in Fig. 4.7, resembling a steep cliff with eroded material accumulated at the base, which is why these plots are known

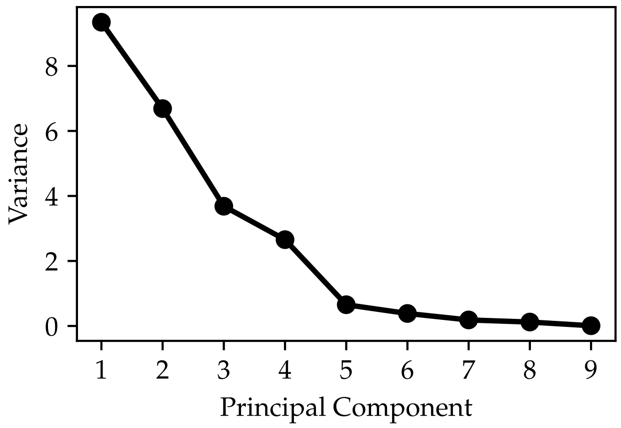

Figure 4.7: Example of a scree plot of randomly generated normally distributed data. In this case the plot has a clear elbow at the fifth PC with the PCs 5-9 appearing roughly on a line. Thus the last four PCs could be omitted if dimensionality reduction is desired.

as scree plots and the nearly linear section of the plot is called the scree.

When the scree plot has two clearly different areas, the steep slope corresponding to the first PCs and a more gently sloping scree for the latter PCs, locating this elbow in the plot connecting the two areas will give the number of PCs that should be included (**?**), which in case of Fig. 4.7 would yield five PCs. Some sources such as (**?**) suggest that in some cases the PC corresponding to the elbow should be discarded, which will result in one less PC.

Unfortunately, as Cattell also acknowledges in his paper, all cases are not as easy to analyse as the one in Fig. 4.7 and may prove difficult to discern for an inexperienced researcher. This problem might arise from for example noise in the linear part of the plot or the scree line consisting of two separate linear segments with different slopes. The first case has no easy solution, but in the latter case Cattell suggests using the smaller number of PCs.

Another straightforward method for choosing how many PCs to retain is to examine how much of the total variation in data is explained by first PCs and including components only up to a point where pre-defined percentage of the total variance is explained (**?**). Whereas the previous method posed a challenge in determining which PC best matches the exclusion criteria, when using this approach the problem arises from choosing the threshold for including PCs. **?** suggests that a value between 70 % and 90 % of the total variation is often a reasonable choice, but admits that the properties of the data set may justify values outside this range. Unfortunately, the suggested range is quite wide, so it may contain multiple PCs and therefore it is up to the researcher to determine the best number of PCs, while the criterion again acts only as an aid in the process.

4.3.3 Principal Component Regression

4.4 Error analysis

4.5 Comparing two samples drawn from unknown distributions

A common question in multiple fields of science is whether two or more samples are drawn from the same distribution. The most relevant methods that can be used to address this problem are introduced here following **?** and **?** apart from introducing the χ^2 test which is mostly based on the approach of **?**.

Questions related to comparing samples can emerge for example when comparing effectiveness of two procedures, determining if the instrument has changed over time or whether observed data is compatible with simulations. There are multiple two-sample tests that can address this kind of questions, e.g. χ^2, Kolmogorov-Smirnov, Cramér-von Mises and Anderson-Darling tests.

In addition to comparing two samples, these tests can be used as one-sample tests to determine whether it is expected that the sample is from a particular distribution. However, some restrictions apply when using the one-sample variants. Some of these tests use categorical data, i.e. data where variables fall in pre-defined categories, and compares numbers of samples in different categories, whereas the others are applied to numerical data and compare empirical distribution functions (EDF) of the datasets.. Examples of such categories might be for example “galaxies that are active” or “data points between values 1.5 and 1.6”.

4.5.1 χ^2 test

Astronomical data often involves classifying objects into categories such as “stars with exoplanets” and “stars without exoplanets” or the spectral classes of stars (**?**). One tool for analysing such categorical data is χ^2 test, which can be used both to determine whether a sample can be drawn from a certain distribution and to test whether two samples can originate from a single distribution.

The method described here is sometimes referred to as Pearson’s χ^2 test due to existence of other tests where χ^2 distribution is used. In some cases, such as with small 2×2 contingency tables and when expected cell counts are small, other variants of χ^2 test should be used. For example the Yates’s χ^2 test or the Fisher exact test work better in these cases than the χ^2 test.

For one-sample test, the χ^2 test uses the number of measurements in each bin together with a theoretical estimate calculated from the null hypothesis. For example one might have observed exoplanets and tabulated the number of planet-hosting stars of different spectral class as is shown in Table 4.1 and now wants to test the observations against null hypothesis “Distribution of stellar classes for observed exoplanet-hosting stars is equal to that of main sequence stars in solar neighbourhood as given by **?**” using significance level $\alpha = 0.01$. The data is categorical, so now χ^2 test is a justified choice.

Stellar class	Number of observed planetary systems
A	6
F	38
G	39
K	134

Table 4.1: Example of categorical data.

Stellar class	Observations (f_o)	Theory (f_e)
A	6	6
F	38	28
G	39	71
K	134	112
total	217	217

Table 4.2: Data of Table 4.1 together with expected values if null hypothesis was true.

In this case the first step would be to calculate the expected observation counts for each bin according to the null hypothesis. Table 4.2 contains these expected counts (f_e) together with the observations (f_o). These observed and expected values are then used to calculate the χ^2 test statistic, defined as

$$\chi^2 = \sum_i \frac{(f_o - f_e)^2}{f_e}. \tag{4.5}$$

With the data given above this results in $\chi^2 \approx 23.6$. The data has four bins, so the degree of freedom is $4 - 1 = 3$. Next one can compare the calculated χ^2 value to a tabulated critical value for our significance level $\alpha = 0.01$. These tabulated values can be widely found in statistics textbooks and books specifically dedicated to statistical tables.

In this case according to **?** the critical value is 11.34, which means that as $23.6 > 11.34$ one can reject the null hypothesis and conclude that at 1% significance level the distribution of stellar classes for observed exoplanet-hosting stars is not equal to that of main sequence stars in solar neighbourhood. This of course can either be due to exoplanets being more numerous around some stellar classes than others or arise from some observational effect such as the observer observing more of the later type stars and thus arbitrarily skewing the distribution of the exoplanet finds.

The χ^2 test can also be used to test for independence of two or more samples. The data is again tabulated and now the χ^2 test statistic is calculated as

$$\chi^2 = \sum_i \sum_j \frac{(f_{oij} - f_{eij})^2}{f_{eij}} \tag{4.6}$$

where f_{oij} denotes the observed frequency in cell (i, j) and f_{eij} is the expected frequency for that cell. The expected frequency can be calculated using the following formula

$$f_{eij} = \frac{R_i C_i}{N} \tag{4.7}$$

where R_i is the number of samples in row i, C_j is the number of samples in column j and N is the total sample size.

According to **?**, the degrees of freedom is $(R-1)(C-1)$ where R is the number of rows and C is the number of columns in tabulated data. This is true in many if not most cases, but the way of collecting data can affect the degrees of freedom in both one-sample and multi-sample cases, as **?** explains. For example, if the one-sample model is not renormalized to fit the total number of observed events or, in two-sample case, the sample sizes differ, the degrees of freedom equal to number of bins N_b instead of $N_b - 1$.

Before performing the χ^2 test on a dataset, it is important to confirm that the data meets the assumptions for χ^2 test, given for example in (**?**) and (**?**). First of all, the data has to consist of counts i.e. not for example percentages or fractions.

These counts should be independent of each other and there has to be enough of them, generally > 50 is sufficient. Bins should also be chosen such that all bins have at least five counts according to the null hypothesis. If the last condition is not met, one can can consider combining bins.

4.5.2 Kolmogorov-Smirnov test

For astronomers, one of the most well-known statistical test is the Kolmogorov-Smirnov test, also known as the KS test. It is computationally inexpensive to calculate, easy to understand and does not require binning of data. It is also a nonparametric test i.e. the data does not have to be drawn from a particular distribution.

In the astrophysical context this is often important because astrophysical models usually do not fix a specific statistical distribution for observables and it is common to carry out calculations with logarithms of observables, after which the originally possibly normally distributed residuals will no longer follow a normal distribution. When using the KS test, the values on the x-axis can be freely reparametrized: for example using $2x$ or $\log x$ on x-axis will result in same value of the test statistic as using just x (**?**).

The test can be used as either one-sample or two-sample test, both of which are very similar. For two-sample variate the test statistic for the KS test is calculated based on empirical distribution functions $\hat{F}_1$ and $\hat{F}_2$ derived from two samples and the test statistic

$$D = \sup_x |\hat{F}_1(x) - \hat{F}_2(x)| \tag{4.8}$$

uses the maximum vertical distance of the EDFs. This test statistic is then used to determine the p-value and thus decide whether the null hypothesis can be rejected. For one-sample variate the procedure is similar, but EDF $\hat{F}_2$ is substituted with the CDF that corresponds to the null hypothesis.

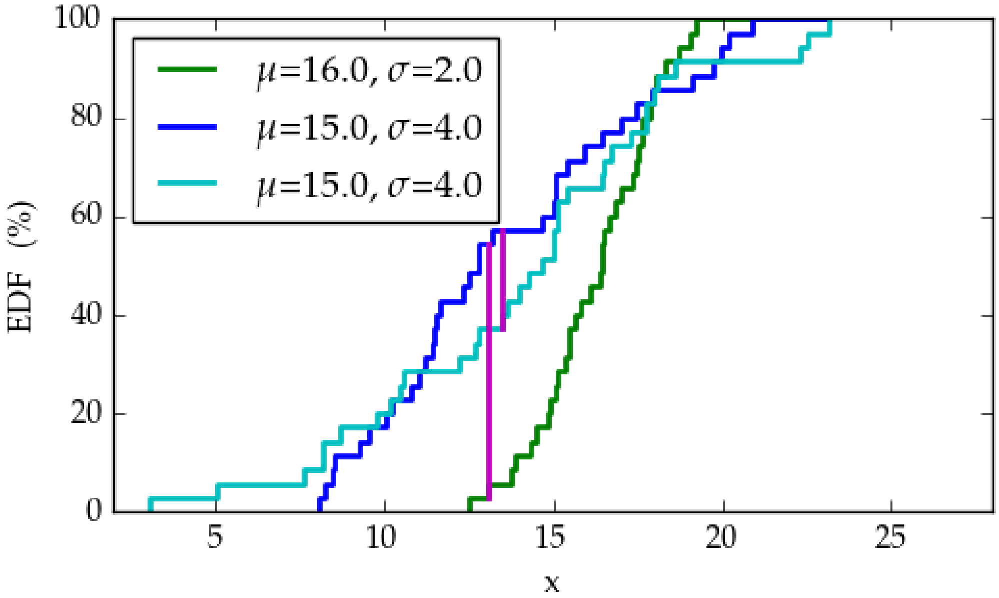

Figure 4.8: KS test parameter values (magenta vertical lines) shown graphically for three samples from Fig. 4.2.

As an example, let us consider two pairs of samples from Fig. 4.2: green and blue (two samples drawn from different normal distributions) and blue and cyan (two samples drawn from the same normal distribution). We can formulate the test and null hypotheses for both pairs as H_0="the two samples are drawn from the same distribution" and H_1="the two samples are not drawn from the same distribution" and choose a significance level of for example $\alpha = 0.05$ or $\alpha = 0.01$.

The test statistic is then calculated and for these samples we get $D = 0.51$ for the green-blue pair and $D = 0.20$ for the blue-cyan pair. Test statistics are illustrated in Fig. 4.8 where the test statistics D are shown as vertical magenta lines. According to Python function `scipy.stats.ks_2samp`, these values of D correspond to p-values 9.9×10^{-5} and 0.44 respectively, which means that the null hypothesis "green and blue samples are drawn from the same distribution" is rejected at both 0.05 and 0.01 significance levels but the null hypothesis "blue and cyan samples are drawn from the same distribution" cannot be rejected.

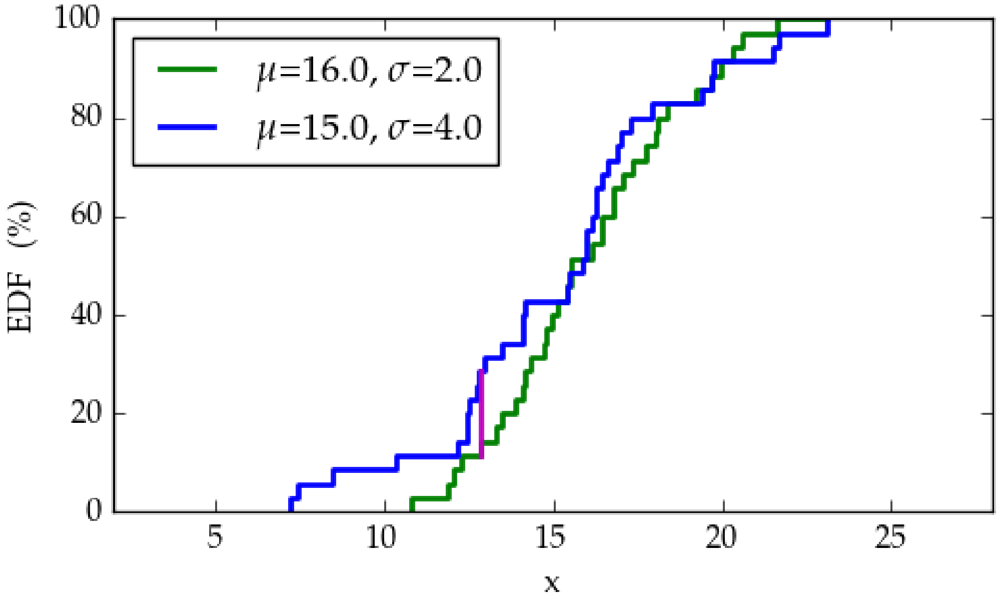

Figure 4.9: KS test ran on another pair of samples drawn from blue and green distributions in Fig. 4.1.

In this case the KS test produced result that matches the actual distributions from which the samples were drawn. Using a different random realization might have resulted in a different conclusion, for example the one shown in Fig. 4.9 results in $D = 0.17$ that corresponds to a p-value of 0.64 i.e. null hypothesis could not have been rejected using the α specified earlier. In a similar manner there can be cases where two samples from one distribution are erroneously determined not to come from the same distribution if the samples differ from each other enough due to random effects.

The latter example case also illustrates one major shortcoming of the KS test: it is not very sensitive to small-scale differences near the tails of the distribution. For example in Fig. 4.9 the blue sample goes much further left, but because EDF is always zero at the lowest allowed value and one at the highest one the vertical distances near the tails are small and the test is most sensitive to differences near the median value of the distribution. On the other hand, the test performs quite well when the samples differ globally or have different means. (**?**)

The KS test is also subject to some limitations and it is important to be aware of them in order to avoid misusing it. First of all, the KS test is not distribution free if the model parameters, e.g. mean and standard deviation for normal distribution, are estimated from the dataset that is tested. Thus the tabulated critical values can be used only if model parameters are determined from some other source such as a simulation, theoretical model or another dataset.

Another severe limitation of KS test is that it is only applicable to one-dimensional data. If the dataset has two or more dimensions, there is no unique way of ordering the points to plot EDF and therefore if KS test is used, it is no longer distribution free. Some variants that can handle two or more dimensions have been invented, such as ones by **?** and **?**, but the authors do not provide formal proof of validity of these tests. Despite this, the authors claim that Monte Carlo simulations suggest that the methods work adequately well for most applications.

4.5.3 Other tests based on EDFs

Unsatisfactory sensitivity of the KS test motivates the use of other more complex tests. Such tests are for example the Cramér-von Mises test (CvM) and Anderson-Darling (AD) test, both of which have their strengths. Similar to KS test, both of these can be used as one-sample or two-sample variants.

First of these tests integrates over the squared difference between the EDF of the sample and CDF from the model or two EDFs in case of two-sample test. The test statistic W^2 for one-sample case can be expressed formally as

$$W^2 = \int_{-\infty}^{\infty} [\hat{F}_1(x) - F_0(x)]^2 \, dF_0(x) \tag{4.9}$$

For two-sample version, the theoretical CDF F_0 has to be replaced with another empirical distribution function $\hat{F}_2$.

Due to integration, the CvM test is able to differentiate distributions based on both local and global differences, which causes it to often perform better than the KS test. Similar to the KS test, the CvM test also suffers from EDFs or an EDF and a CDF being equal at the ends of the data range, which again makes the test less sensitive to differences near the tails of the distribution.

In order to achieve constant sensitivity over the entire range of values, the statistic has to be weighted according to the proximity of the ends of the distribution. The AD test does this with its test statistic defined as

$$A^2 = N \int_{-\infty}^{\infty} \frac{[\hat{F}_1(x) - F_0(x)]^2}{F_0(x)[1 - F_0(x)]} \, dF_0(x) \tag{4.10}$$

where N is the number of data points in sample. This weighing makes the test more powerful than the KS and CvM tests in many cases. (**??**)

Also other more specific tests exist, such as the Kuiper test which is well suited for cyclic measurements. The test should always be chosen to match the dataset such that it best differentiates between the null and research hypotheses.

4.6 Cluster Analysis

The aim of cluster analysis is to find groups of similar data points within a data set, these groups being called clusters (**?**). Ideally data points within each cluster are as similar to each other as possible and dissimilar to data points outside the cluster (**?**). It has many applications in multiple fields ranging from machine learning to biology and it can also be applied in astronomy and astrophysics to e.g. classify objects and possibly even discover new ways to classify them (**??**). For example **?** were able to divide gamma ray bursts to three distinct categories separated by a combination of their durations, brightnesses and spectra.

The definition of a cluster is intentionally loose, allowing the exact definition to vary between problems to which it is applied (**?**). Differences can arise for example from whether one data point can be part of multiple clusters and whether all data points should be part of some cluster or if the dataset contains noise not belonging to any cluster. Many different algorithms have been developed for performing cluster analysis, each best suited for some specific type of data (**?**). The algorithm used should always be chosen based on the desired properties of the clusters and the properties of the data set.

One clustering algorithm that is well suited for finding clusters based on a density threshold in a spatial data set is DBSCAN, originally presented by **?**. It produces a density-based clustering, meaning that the identified clusters consist of data points residing in high-density regions, each cluster separated from others by a low-density region (**?**). This definition allows both finding clusters of arbitrary shape and working with noisy data with data points that do not belong to any cluster (**?**).

The algorithm uses two parameters defining the properties of the clusters: ε (sometimes denoted Eps) and MinPts. The ε parameter resembles the linking length used by the FOF algorithm introduced in section 3.1.3 as it sets the size of the neighbourhood used when determining whether a data point is part of a cluster. The distance function can be chosen freely to best fit the problem, possible metrics being e.g. Euclidean or Manhattan distance. The points within the distance ε from a given data point are said to belong to its ε-neighbourhood. This definition implies that a data point also belongs to its own ε-neighbourhood.

If more than MinPts data points belong to the ε-neighbourhood of a data point, it belongs to a cluster and is defined to be a core point of that cluster. If a core point has another core point in its ε-neighbourhood, the two core points belong to the same cluster. For example in Fig. 4.10 the red points are all core points of one cluster, but as no blue points are within ε-neighbourhood of any red point, the blue points form their own cluster. When MinPts $= 2$, the resulting clustering is identical to one produced by the FOF algorithm.

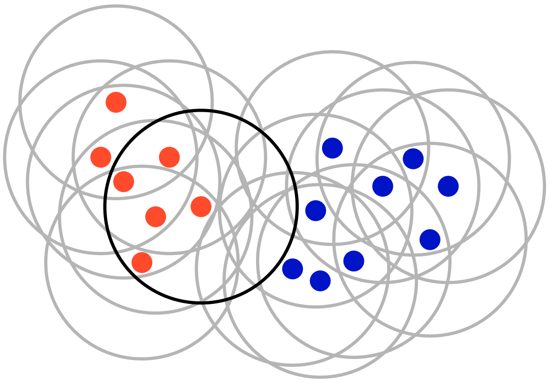

Figure 4.10: An example data set shown as dots, surrounded by circles showing the extent of ε-neighbourhoods of each data point. To reduce clutter, all but one ε-neighbourhood border are shown in grey. When MinPts $= 4$, the data points are separated to two clusters, shown in red and blue here. As every point has at least four data points within its ε-neighbourhood, all data points are core points.

Some points might belong to an ε-neighbourhood of a core point but have ε-neighbourhoods with less than MinPts data points in them. These points are border points and they are also members of the cluster containing the core point in the ε-neighbourhood of the border point. For example in Fig. 4.11 the light blue border point belongs to the cluster defined by the dark blue core points. Cluster membership does not propagate further from border points, meaning that e.g. the grey data point next to the light blue point in Fig. 4.11 is not part of any cluster despite belonging to the ε-neighbourhood of one of the cluster members as this member is a border point.

In a situation where a border point is in ε-neighbourhoods of core points belonging to different clusters, as happens for example to the grey data point in Fig. 4.12, it is not clear to which cluster the border point should be assigned. The density-based definition of a cluster presented by **?** would classify these border points to all clusters whose core points reside in the ε-neighbourhoods of these border point. Often a clustering with each data point belonging to a maximum of one cluster is

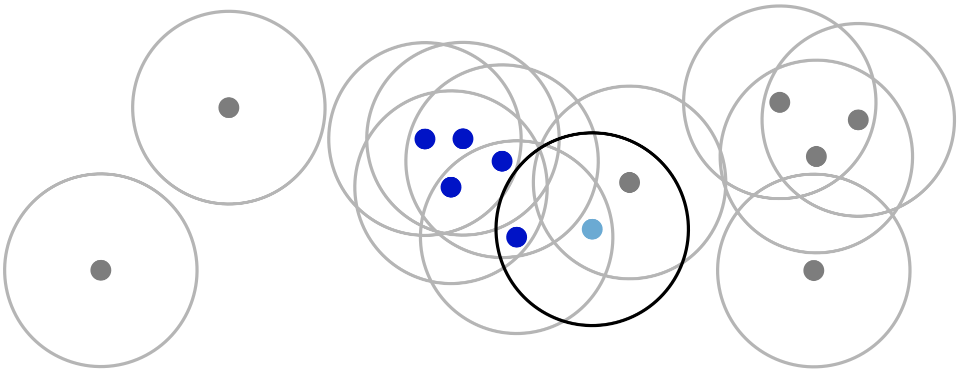

Figure 4.11: A single cluster with MinPts = 4 and ε shown with circles. The dark blue points all have at least four points in their ε-neighbourhood so they are core points, but the light blue point has an ε-neighbourhood of only three points. Thus despite it being part of the ε-neighbourhood of a core point, it is not a core point. Instead, it is a border point of the cluster. The grey point in the ε-neighbourhood of the light blue point does not have four points in its ε-neighbourhood nor does it belong to an ε-neighbourhood of any core point, so it is not part of the cluster. Similarly none of the other grey points have enough particles in their neighbourhoods so they do not belong to any cluster.

desired instead, so the DBSCAN algorithm classifies this kind of border points only to the first cluster they are discovered in.

This introduces some possibly unexpected behaviours. First of all, while the algorithm is deterministic when run multiple times on the same data set, the clustering is dependent on the order of the data points (**?**). If the order of data points is permuted, the cluster to which these border points between two clusters are assigned might change. Fortunately this is a fairly rare occasion and the effects of assigning a single border point to one cluster or another are often insignificant (**?**).

Another edge case is even more extreme. Consider the configuration of data points is Fig. 4.13 where the ε-neighbourhood of the grey point contains one border point from each of the three other clusters. As these are only border points, the grey

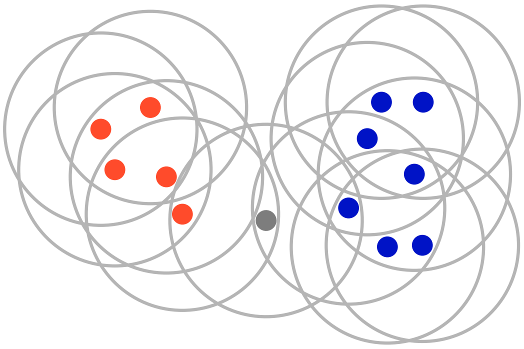

Figure 4.12: In this situation the grey data point could belong to either red or blue cluster. The DBSCAN algorithm will be added to the cluster that is discovered first, so depending on the order of the data points it might be part of either red or blue cluster.

point is not member of any of the three coloured clusters, but its ε-neighbourhood has four points, so it is a core point of its own. Now assuming that the red, blue and yellow clusters are all found before the grey point is processed, the grey data point becomes the core point of a cluster containing only a single data point. These clusters with fewer than MinPts members are of course even rarer than the occasions when the cluster membership of a border point is ambiguous, as they require border points of multiple clusters stretching close to the core area of a small cluster without merging with it. Even with these fairly rarely occurring undesirable features, the DBSCAN algorithm is in many cases more robust than the FOF algorithm, as choosing an approppriately large value of MinPts for example reduces the probability of a thin bridge of data points connecting otherwise separate clusters.

The properties of the identified clusters naturally depend on the used parameters ε and MinPts. Different heuristics exist for determining suitable values for them. For MinPts, **?** provide a value of twice the dimensionality of the dataset, e.g. for two dimensional data MinPts $= 4$ should be chosen. For some datasets a higher value can be better if the dataset for example contains duplicate data points or is very large or high-dimensional (**?**).

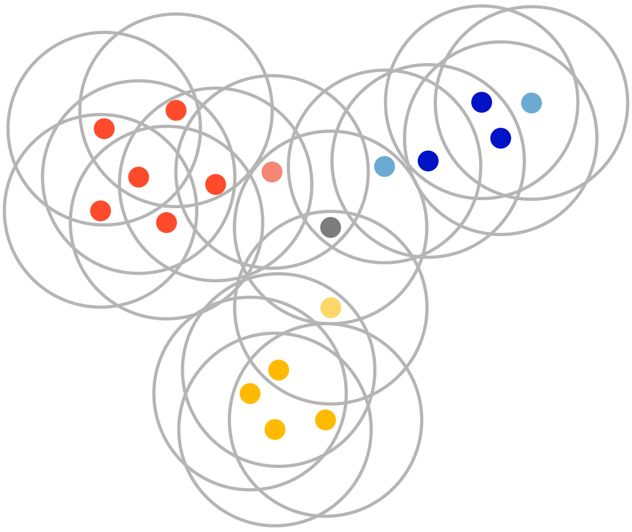

Figure 4.13: Another clustering with MinPts = 4. One border point from red, blue and yellow clusters each resides within the ε-neighbourhood of the grey point, but as these are border points (shown with lighter colours), the grey point belongs to none of the other clusters. The grey point has four points in its ε-neighbourhood, making it a core point. If the three other clusters are discovered before the cluster membership of the grey data point is determined, it becomes the only data point of its own cluster.

According to **?**, ε can be set after choosing the value of MinPts by inspecting the distances of data points to their k^{th} nearest neighbour. Here k can be set to the same value as MinPts, though the results do not seem to vary greatly if k is increased (**?**). When the k-distances are sorted in descending order and plotted, the resulting plot often shows an elbow similar to the one in the scree plot in Fig. 4.7. Examples of such plots can be seen in Fig. 4.14 that shows k-distance plots for six different values of k ranging from 2 to 7. **?** suggest that the distance corresponding to the elbow should be used as a value of ε, with the points to the left of the elbow corresponding to noise. The location of the elbow is usually not very sensitive to small changes in k, as can be seen in Fig. 4.14 where the location of the elbow changes less than 0.1 radians over the six different values of k. If domain knowledge

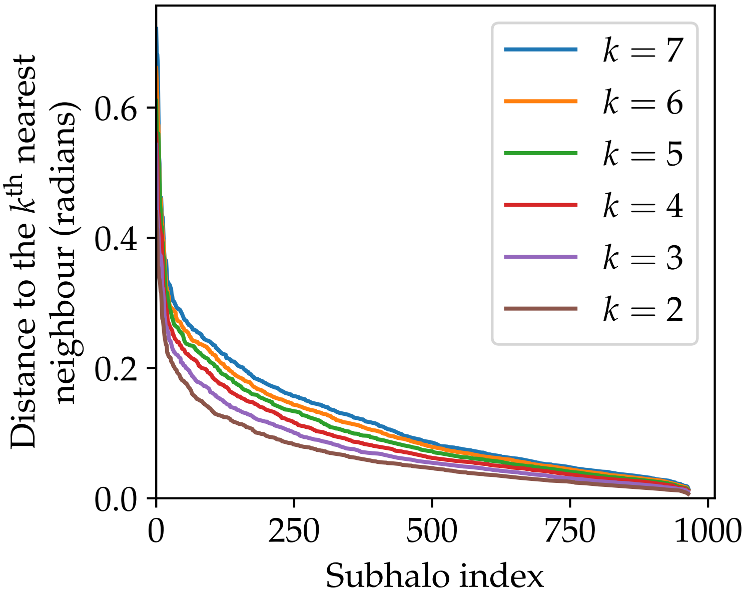

Figure 4.14: Six $k^{\rm th}$ nearest neighbour distance plots with k ranging from 2 to 7. Variations in the location of the elbow are reasonably small when k is varied.

is available, ε and MinPts can also be chosen differently to best suit the data set (?).

5. Findings from Halo Catalogue Analysis

TODO intro

5.1 Properties of the Local Group analogues

As the Local Group analogue identification criteria described in section 3.2.3 allow for a range of different halo pairs, the properties of the found Local Group analogues vary. This can be seen in Fig. 5.1 where both distance between the main haloes and the velocity components span nearly the whole allowed range of values. The plot also contains a histogram of overdensity within 2 Mpc of the Milky Way analogue.

The peak of the radial velocity histogram is located near 100 km/s, meaning that most of the identified Local Group analogues have radial velocities near the measured value for the Andromeda galaxy (**?**). Pairs with low radial velocity are rare, as according to the timing argument in order to have a radial velocity close to zero the mass of the Local Group would typically have to be small, which is excluded by the mass criterion used. For tangential velocities, the distribution is much more uniform, though values near the ends of the allowed interval are still fairly rare. The **?** tangential velocity measurement result of 17.0 km/s would also fall in one of the densely populated bins, though the tangential velocity measurements have much larger uncertainties than radial velocities: the 1σ confidence region extends to

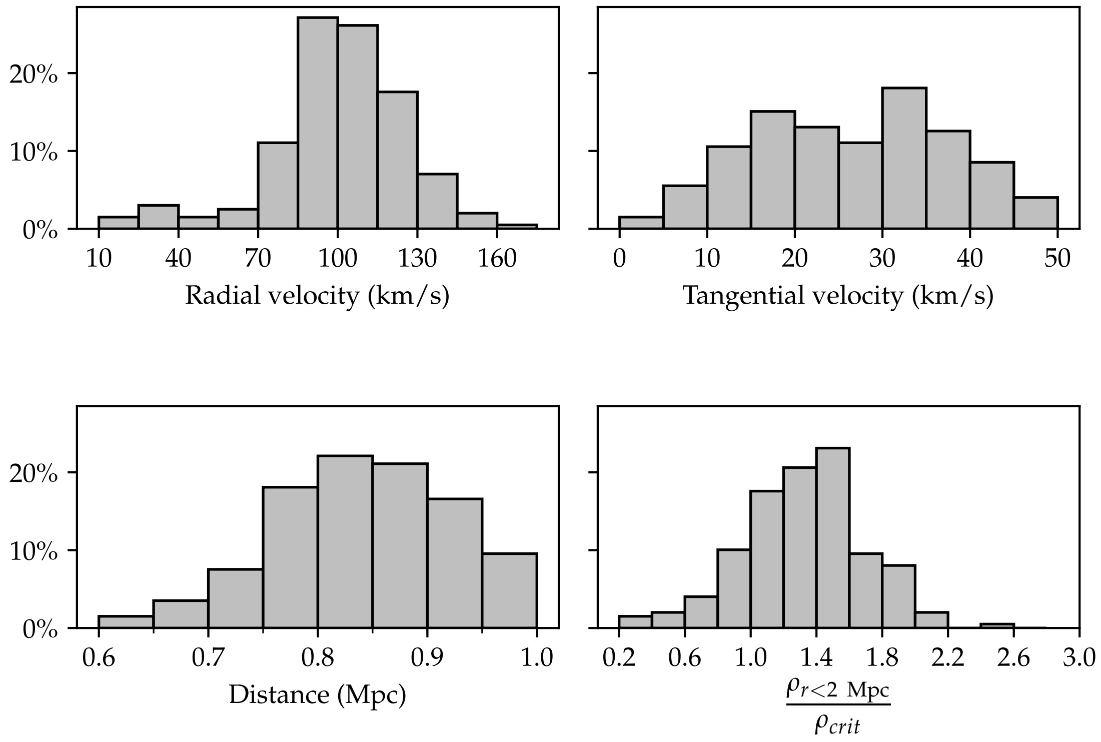

Figure 5.1: Histograms of the mutual radial and tangential velocities (top row) and distances (bottom left panel) of the main haloes in each Local Group analogue found in the simulations. The bottom right panel shows the overdensity within 2 Mpc of the Milky Way analogue in each simulation. The densities are calculated based on the combined mass of subhaloes, so the values of overdensity are somewhat smaller than they would be if all mass within 2 Mpc was included.

34.3 km/s, covering more than half of the allowed range in the Local Group analogue identification criterion (**?**).

Modern distance estimates for M31 range from 770 to 800 kpc with errors of few tens of kpc, meaning that the distance is slightly shorter than in most of the simulated Local Group analogues but still near the peak of the distribution (**?**). Analogues with shorter distances between the primaries are somewhat rare whereas distances at the longer end of the allowed range are more common. This is likely an effect of the mass and radial velocity criteria together with the fact that a spherical shell of constant thickness has larger volume on larger distances and thus if a constant probability density function is assumed also more subhaloes.

As can be seen in the overdensity histogram, most of the Local Group analogues are located in an overdense region. Densities in the histogram only include the mass within subhaloes, so the total densities within the examined 2 Mpc regions is somewhat larger. There is also a small number of Local Group analogues in very underdense regions. This is possible only if the primary haloes are on the lower end of the allowed mass range, they have only low mass satellites and there are no other comparable structures within the 2 Mpc distance. Very high density regions are also rare as they require either a large number of smaller haloes or a halo with a mass comparable to the primaries of the Local Group to fall within the 2 Mpc region.

The combined mass of the primaries in the Local Group analogues and its distribution between the haloes also varies from simulation to simulation. This is illustrated in Fig. 5.2 that shows the mass of the more massive of the primaries in each simulation against the combined mass of the pair. The allowed range of the masses is outlined with the grey lines, the lower one showing the threshold where the more massive halo holds exactly half of the total mass and the upper one illustrating the maximum possible mass in the more massive halo when the Local Group analogue criteria require the smaller one to have a mass of at least 4×10^{11} $\mathrm{M}_{\odot}$.

There are Local Group analogues both with almost equal masses and with the smaller halo barely exceeding the minimum mass, though as can be seen in the figure, the former case is more common. Especially for total masses larger than 2×10^{12} $\mathrm{M}_{\odot}$ the area near the upper limit for the mass of the more massive primary is nearly vacant of data points. This is likely caused by environments with density high enough for them to be able to host the more massive primary usually having other massive subhaloes in them too. Thus in dense environments it is common that

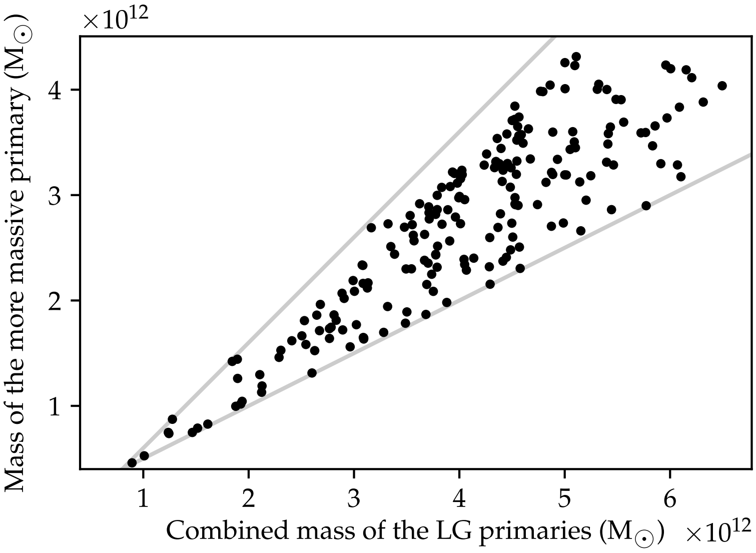

Figure 5.2: The masses of the more massive primary plotted against the combined mass of the primaries, each black dot representing one Local Group analogue. Grey lines show the allowed range: the more massive one has to contain at least half of the combined mass and the smaller one has to have a mass of at least 4×10^{11} $M_\odot$.

more than one halo with a mass larger than that of the smaller primary is present within the 2 Mpc distance used to check that the two primaries are the dominant haloes in their surroundings.

The Local Group identification criteria allow the combined mass of the primaries between 8×10^{11} $M_\odot$ and 1×10^{13} $M_\odot$, but whereas the lower end of the range is populated, there are no Local Group analogues with combined mass of 7×10^{12} $M_\odot$ or more. This is likely because high mass haloes are relatively rare in the sample: only about one fifth of all haloes in a primary pair of a Local Group analogue have a mass of more than 3×10^{12} $M_\odot$. This is illustrated in Fig. 5.3 that shows histograms of masses both separately for the smaller and larger primary in each pair and all primaries together. It can also be seen that the masses do not span the full allowed range, as the most massive subhalo in a Local Group analogue has a mass of 4.3×10^{12} $M_\odot$, clearly below the maximum allowed mass of 5×10^{12} $M_\odot$.

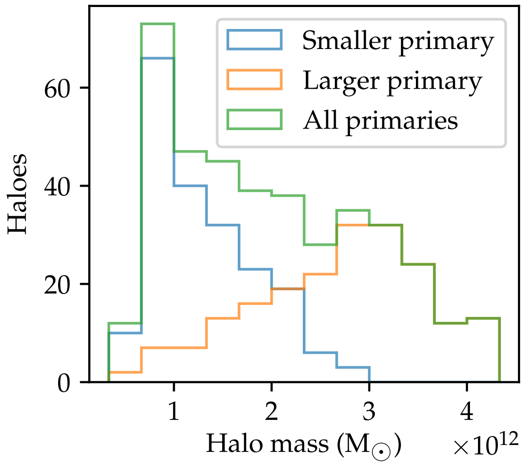

Figure 5.3: Histograms showing the masses of the identified Local Group analogue primaries. The blue and yellow histograms contain only the less or more massive subhalo of each primary respectively whereas the green histogram shows the masses of all subhaloes in a primary pair of a Local Group analogue.

It is notable how the distribution of the masses of the larger haloes in the primary pairs extends all the way to the lowest masses whereas there are no primary pairs with the mass of the smaller halo exceeding 3×10^{12} M$_\odot$.

Overall, the 199 Local Group analogues chosen from the zoom simulations span a wide range of different two-halo systems. Only the values near the very edges of the allowed ranges in each variable are rare. This allows an analysis of correlations of the mass of the system with its other properties.

5.2 Hubble Flow Measurements

The Local Group analogues analysed in this project do not only differ in the properties of the primaries but also in the environments in which they reside. The primaries can for example have few or many satellites around them and the velocity dispersion within the system can vary. Similarly, the volume further out from the system can be densely populated with haloes possibly arranged in filaments or walls, or the Local Group analogue can be located in a very low density environment. These environmental factors are explored in this section and a way to quantify a selection of them for further analysis is presented.

5.2.1 Different Hubble Flows

A selection of these different environments is illustrated in Fig. 5.4 showing radial velocities as seen from the lower mass primary plotted against the distance of the subhalo. This corresponds to what a Hubble diagram would look like as seen from the Milky Way analogue of that simulation. Each of the four panels contains all subhaloes in that simulation up to the closest type 2 particle. The simulations shown in this plot are chosen such that they show a variety of different Hubble Flows, but naturally four simulations can never show the full range of all 199 analysed simulations nor all structures that can occur.

The simulations on the top row in Fig. 5.4 both show fairly small dispersion of radial velocities at each distance, corresponding to a cold Hubble flow. In the upper left plot, there is a region with a much larger velocity dispersion between approximately 4 and 5 Mpc, likely corresponding to a massive subhalo surrounded by satellite galaxies, some of which are receding and some approaching the observer as they orbit the primary or primaries of the system. The upper right plot on the other hand shows no such variation in the velocity dispersion, apart from the dispersion within the nearest 1 Mpc or so being higher than in the rest of the plot due to the velocities of the satellite galaxies within the Local Group analogue, whereas the upper left plot has fairly low velocity dispersion within the Local Group.

The simulations presented in the bottom row of the figure show considerably higher velocity dispersions. Especially the bottom right plot, with radial velocities in it easily spanning a range of 400 km/s at all distances except within 1 Mpc of the

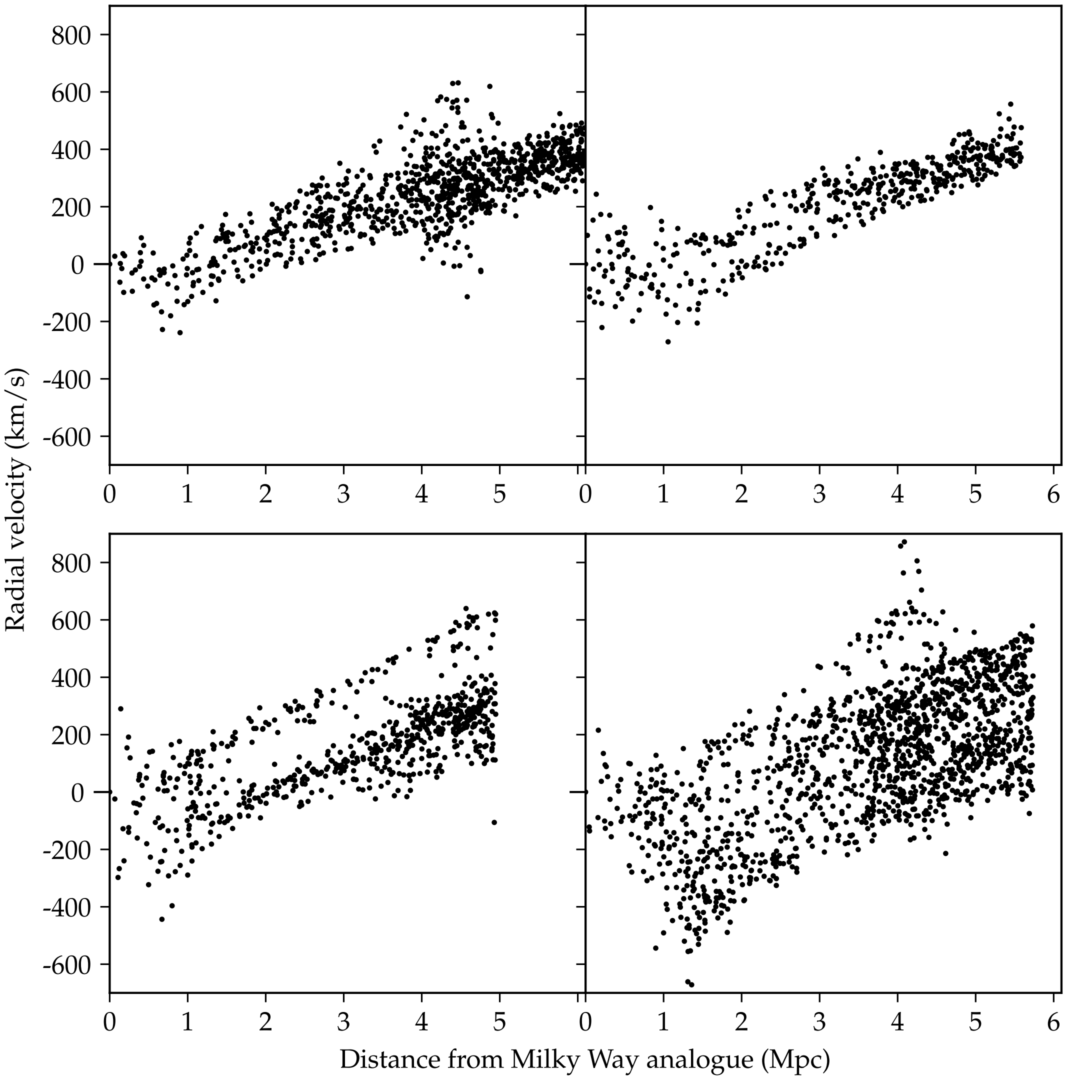

Figure 5.4: Hubble flows around Milky Way analogues in four simulations differing in for example the total number of subhaloes, scatter of the flow, scatter within the Local Group and the number of bound structures present within the flow. Each black dot represents one subhalo in a simulation and subhaloes are drawn up to the distance at which the first type 2 particle resides.

Milky Way analogue, differs from the top row plots. The bottom right plot also has a much greater number of subhaloes than the other plots. The bottom left plot, on the other hand, seems to show a Hubble flow that is divided into two parts with a vertical offset of around 100 km/s. This kind of situation might occur, for example, if the Local Group analogue is moving relative to the surrounding structures.

These kinds of differences in the surroundings of the Local Group analogues might be correlated with the mass of the primaries and could thus provide a way to gauge the mass of the real Local Group by observing its surroundings. These correlations are explored in section 5.4, but in order to conduct statistical analysis, the properties of the Hubble flow have to be quantified first.

5.2.2 Measuring the Hubble Flow

The first step in quantifying the properties of the Hubble flows around the Local Group analogues is to determine the basic shapes of each flow. In this work, the radial velocity field is modelled as consisting of two parts: at small distances when going outward from the Milky Way analogue, the radial velocity decreases or perhaps stays nearly constant with a possibly large scatter, and at larger radii a positive linear relationship between distance and radial velocity is expected. Naturally this is not an exact description of the situation, as the transition from a gravitationally bound Local Group analogue to a linear Hubble flow is not instantaneous and the gravitational pull of the Local Group also affects objects outside it.

Fitting a linear relation to the linear Hubble flow observed around the Milky Way analogues is, in principle, simple, as for example the ordinary least squares regression (see section 4.2.1) can be used. Determining the distance from which the linear model can be applied, on the other hand, is less straightforward and multiple possible schemes exist. Possibly the simplest option would be to set an arbitrary distance threshold and exclude all haloes within this distance from the

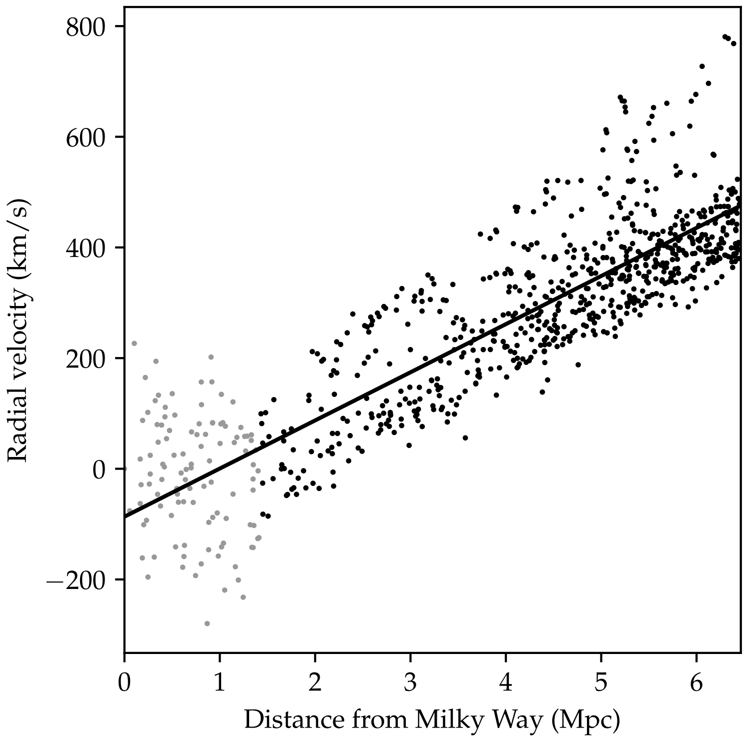

Figure 5.5: Hubble flow fit (black line) calculated by excluding a number of innermost subhaloes resulting in the smallest mean squared residual. For this simulation, this was achieved when excluding the first 1.4 Mpc of haloes, producing a measured Hubble flow slope of 87.0 km/s/Mpc. The excluded subhaloes are shown as grey points and subhaloes used to fit the Hubble flow as black dots.

fitting procedure. For some simulations, this will likely exclude more data points than is necessary and for others it is possible that not enough points will be excluded. Even when the exclusion threshold is chosen carefully, in a large sample a constant distance will always be suboptimal for some simulations, so a solution that is able to adapt to the properties of each simulation is desired.

When not otherwise stated, in this work the number of nearest subhaloes to be ignored is chosen based on the sum of squares of the residuals. This is done by finding the number of innermost subhaloes up to 2 Mpc away from the Milky Way analogue to be ignored in order to get the smallest mean squared residual for the ordinary least squares fit. The final result of this kind of fitting procedure is illustrated in Fig. 5.5, where all subhaloes are shown as dots: the excluded ones in grey and ones used in fitting in black. The best fit line is also shown. In this simulation the excluded range corresponds well to the distinction a human would likely make when asked to determine the distance at which the motions inside Local Group are no longer the dominant effect on the radial velocity.

This approach is naturally not without weaknesses. For example if the innermost subhaloes within Local Group have a very small radial velocity dispersion, no subhaloes might be excluded as a large number of data points with small dispersion can reduce the mean squared residual even if they skew the result. Similarly groups of data points with large scatter can get excluded even if they otherwise follow the Hubble Flow neatly. Luckily these problems are usually not severe as even when the maximum of 2 Mpc of the nearest region is excluded from fitting, most simulations still have more than 3 Mpc worth of data to use for fitting and thus small changes in including or excluding subhaloes in the innermost regions do not produce large effects. This is also confirmed in the following subsection where the dependence of Hubble flow parameters on the distance are explored.

When the results of the fitting are expressed in this work, the Hubble flow fitting result is not expressed in the most familiar slope-intercept form of $y = kx + b$ giving the slope and the y intercept. Instead the result is expressed in terms of the Hubble flow slope H_0 and the distance at which the fitted radial velocity is zero, corresponding to giving the x intercept instead of y. This convention is chosen as the zero point indicates the distance at which subhaloes are no longer bound to the Local Group, which in turn can be used to probe the mass of the Local Group. Sometimes the velocity dispersion of the Hubble flow is also interesting. In order to calculate it, the Hubble flow fit is first subtracted from the radial velocities, after which the velocity dispersion is obtained by taking the standard deviation of the differences.

5.2.3 Dependence of the Hubble Flow Parameters on Distance

The distance from the Milky Way analogue to the nearest type 2 particle varies from simulation to simulation as was seen in Fig. 3.7 in section 3.2.3. The range of closest to furthest distance to the nearest type 2 particle is also quite substantial: the closest particle can be closer than 2 Mpc away from the Milky Way analogue but some simulations contain only type 1 particles in a spherical volume with radius of more than 10 Mpc. Thus if the Hubble flow measurements were very sensitive to the distance up to which data is available, simulations might appear to have different Hubble flows just because they are measured up to different distances.

The possibility of this affecting the results of Hubble flow measurements is explored in Fig. 5.6, the blue curve of which shows the median value of Hubble flow slope H_0 when all haloes within a given radius are used to construct an ordinary least squares fit. As the closest subhaloes are also a part of the data used in fitting, the fits done only based on the nearest haloes produce negative values of H_0. The smallest values are seen at the radius where the M31 analogues reside in the simulations, produced by the M31 and its satellite system moving towards the Milky Way. Moving the exclusion threshold further out, the measured values of H_0 grow as more and more of the actual Hubble flow is included until at approximately 3 Mpc the median value reaches a maximum and no longer changes.

The same plot also includes a curve showing the median overdensity within a radius, mass being measured as the combined mass of all subhaloes with a centre of potential within the given distance from the Milky Way. Near the Milky Way the overdensity is large but it decreases when the examined volume grows until M31 is reached, at which point the median overdensity rises again, showing a bump in

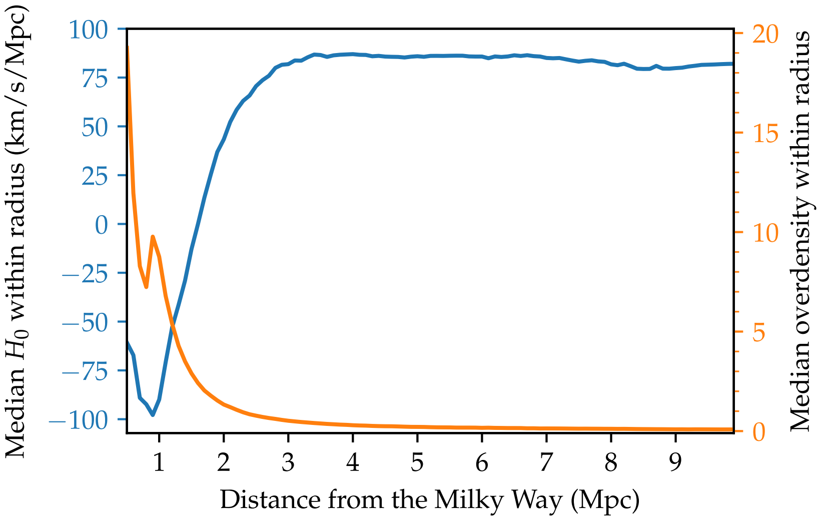

Figure 5.6: Median H_0 results (blue curve) and overdensities (orange curve) in the simulation sample up to a given radius. The plot shows clearly how both of the values evolve within the Local Group, including the effect of the M31 and its satellites, and how the values stabilize at larger radii. The overdensity is measured using only mass contained in subhaloes and thus at large distances it tends to a value smaller than unity. At each radius the plot only contains simulations that are free of type 2 particles up to that distance, so the last measured values are medians of values from very few simulations.

the curve. After this the decrease in density continues until after 3 Mpc there is very little mass belonging to subhaloes compared to the volume. As the mass only contains mass enclosed in subhaloes, at large radii the overdensity does not tend to one but to a smaller value.

The two curves in Fig. 5.6 correspond to each other neatly: at regions with high overdensity structures are gravitationally bound and H_0 measurements yield negative values and, at larger distances, as the overdensity decreases the values of H_0 rise closer to the global value in the simulation. At very large distances of more than 7 Mpc the H_0 curve shows slight variations, but as there are only few simulations that are free of type 2 particles up to those distances, these are likely not actual features but instead produced by local variations in the Hubble flow of possibly as few as a single simulation.

It is also notable that the measured median value of H_0 rises to a value of more than 80 km/s/Mpc, i.e. clearly higher than the global value of 67.77 km/s/Mpc for the simulations. This can happen as the Local Group analogue selection criteria also places indirect restrictions on the surroundings of the system. This effect can also possibly be used when determining the mass of the Local Group.

As the point after which the measured median value for H_0 no longer changes is reached fairly soon, it seems that e.g. 5 Mpc region containing only type 1 particles is sufficient for obtaining a reasonably reliable estimate of the shape of the Hubble flow surrounding the Local Group analogue. This is important for further statistical analysis as only a small fraction of the simulations is free of type 2 particles up to for example 7 Mpc whereas more than half of the simulations still contain only type 1 particles within 5 Mpc.

As the H_0 values shown in Fig. 5.6 include all subhaloes up to a given distance, they differ from values obtained as described in section 5.2.2. The plot also does not give any insight into whether the observed relation between distance and radial velocity in simulations is linear or not. In order to examine these effects, Fig. 5.7 shows how the median H_0 changes as a function of distance when measured using all haloes in a moving 2 Mpc bin. The plot also shows the extents of the ranges containing the middle 90%, 75% and 50% of the H_0 values in each bin. The merging of different ranges at large radii is caused by the plot including in each bin only those simulations that do not have type 2 particles in that bin, so there are very few data points in the last bins.

Like Fig. 5.6, this plot also shows how initially the values of H_0 grow when the bin moves away from the Milky Way analogue, but the curve reaches a significantly

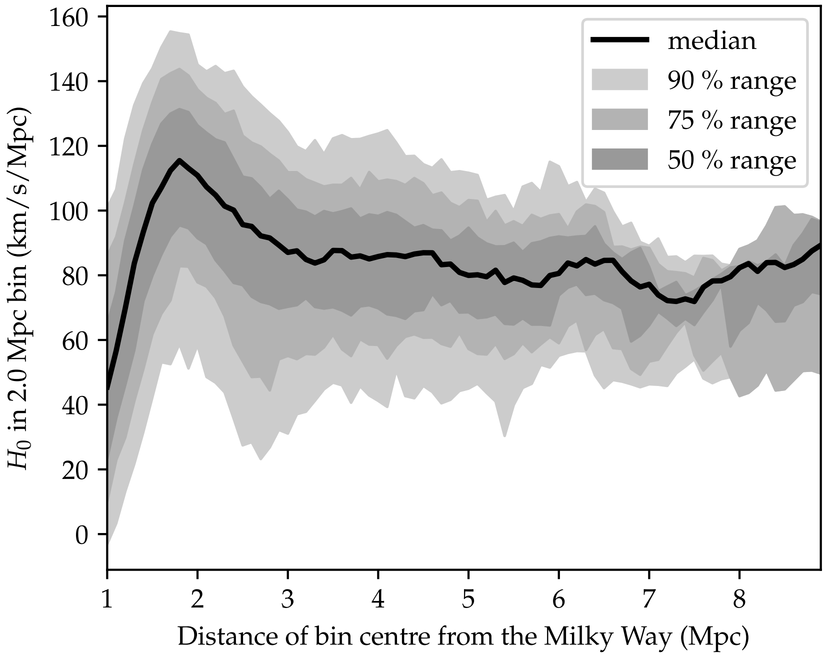

Figure 5.7: Median H_0 when fitted in all subhaloes with distances from the Milky Way analogue in a 2 Mpc bin. The grey areas show ranges containing the middle 90%, 75% and 50% of the measurements. The value of H_0 is calculated only for bins that do not contain any type 2 particles, which results in the total number of data ponints decreasing outwards. For approximately the last megaparsec, the small number of available data points causes the 90% and 75% limits merging.

higher value of H_0 than the curve in Fig. 5.6. This is caused by the bin including both M31 and its satellites moving towards the Milky Way analogue and haloes outside the Local Group moving away, which results in the measured H_0 being exaggerated. At larger distances the measurement no longer has haloes bound to the Local Group analogue in it and it stabilizes to a value near the one in Fig. 5.6.

As the unbinned figure, Fig. 5.7 also starts to show some variation in the value of H_0 near a distance of 7 Mpc for the bin centre. This is also likely at least partially related to the number of simulations from which the median is calculated getting smaller, but actual features in the data set can also affect the result. For example

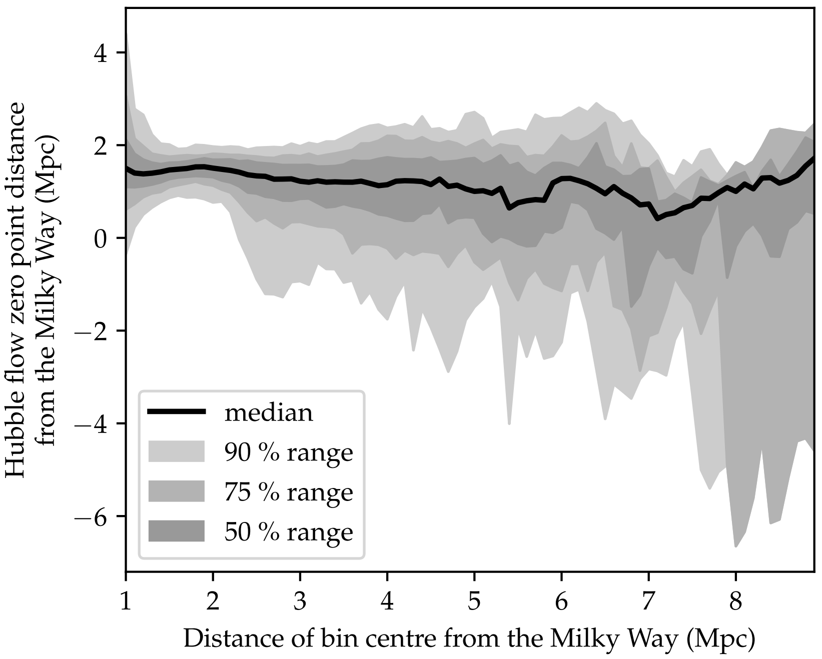

Figure 5.8: Median Hubble flow zero point distance from the Milky Way analogue, measured from a Hubble flow fitted to all subhaloes in a 2 Mpc bin in each simulation, accompanied with the ranges containing the middle 90%, 75% and 50% of the measurements. As in Fig. 5.7, the zero point is calculated only in bins that do not contain type 2 particles and thus for large distances there are less measurements, resulting in the 90% and 75% ranges merging.

massive haloes moving relative to the Milky Way and hosting their own satellite systems can also result in anomalous values of H_0 when the distance range used in the measurement is small.

In addition to the measured value of H_0, the zero point of the fitted Hubble flow can also change if the distance range within which the fit is calculated changes. This is explored in Fig. 5.8 showing how the Hubble flow zero point changes if it is measured from a fit to all haloes with radial distances in a 2 Mpc bin and the bin centre is moved. Unlike H_0, the zero point does not change drastically based on whether M31 analogue and its satellites are inside the bin limits. This is likely caused by the steepening of the fitted line getting cancelled out by the data to which the line is fitted being further from the origin.

This plot also suggests that a volume with a relatively small radius of e.g. 5 Mpc or even less is sufficient to gain good fitting results, as there does not seem to be any systematic change in the zero point at large distances. As this result is compatible with the conclusions drawn from Figs. 5.6 and 5.7, a maximum distance of 5 Mpc is adopted for the Hubble flow fitting using the procedure described in the previous subsection. This allows using a constant upper limit for the range used in fitting while still being able to use more than half of all Local Group analogues in the analysis and without losing the accuracy of the fitting.

As with the H_0 plots, the zero point measurement results also vary more on larger distances. When looking at the plotted ranges, it is clear that in the outermost bins the scatter is very large. Even “negative distances” to the zero point are present, meaning that the fit would suggest that even the Local Group is expanding. These kinds of results are possible when the data set used in fitting no longer contains subhaloes that are bound to the Local Group analogue and the slope of the fitted line is more gradual and closer to the global value of 67.77 km/s/Mpc, at which point the zero point can move close to the Milky Way analogue or if the H_0 is very small even to “negative” distances. This means that the fit is no longer sensitive to the overdensity within the Local Group. Especially when only distant haloes are used, the negative radial velocities within the Local Group might not affect the fit significantly.

5.3 Anisotropy of Hubble flow

Overall, the plots in the previous subsection suggest that the Hubble flows around the analysed Local Group analogues are dependent on the distance at which the value is measured up to about 3 Mpc distance, after which the measured parameters

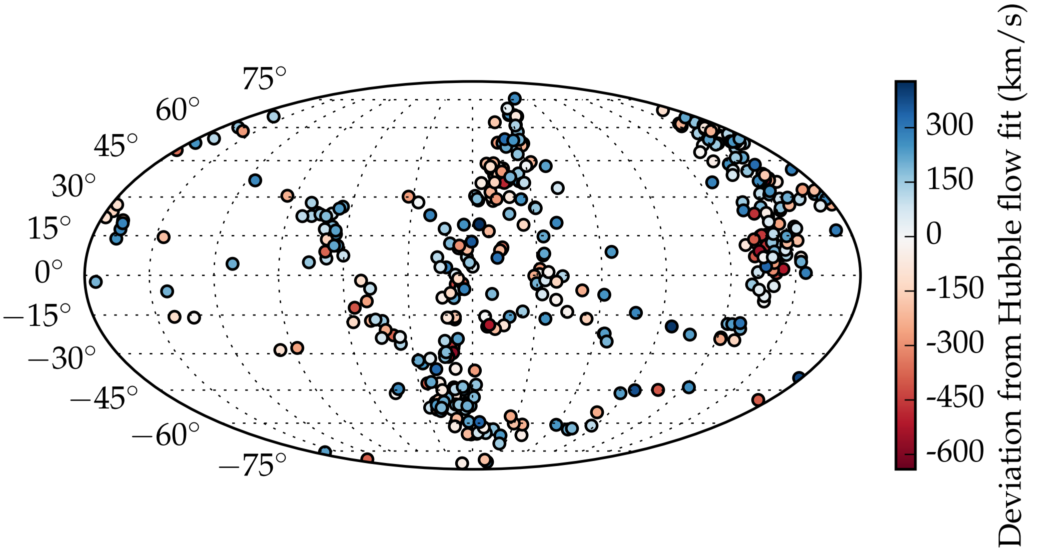

Figure 5.9: Projections of subhaloes, marked with dots, around the less massive LG primary with distances ranging from 1.5 Mpc to 5.0 Mpc. The radial velocity of each subhalo relative to the Hubble flow fit is shown with colour: red subhaloes are moving away from the observer at smaller speeds than predicted by the Hubble flow fit and blue subhaloes at larger velocities.

no longer change. The possibility of directional dependencies, on the other hand, is completely ignored up to now. In order to explore possible anisotropies in the Hubble flows too, this subsection explores the distribution of subhaloes and their radial velocities on the sky as seen from the Milky Way analogue in each simulation.

The distribution of subhaloes at distances up to 5 Mpc are clearly anisotropic, as, like all matter in the Universe, the subhaloes tend to be arranged to filaments and walls instead of being evenly distributed. The velocities of these structures and the location of the Local Group relative to them affect the observed radial velocities. Being able to identify structures such as other galaxy groups and clusters can help improve the Hubble flow measurements and the distribution of the radial velocities can reveal correlations with the mass of the Local Group.

The locations and radial velocities in one simulation are illustrated in Fig. 5.9, which clearly shows how the subhaloes form dense structures separated by voids with few to no subhaloes in them. In order to avoid obstructing the structures with the satellites of the Local Group primaries, the subhaloes are plotted starting at the distance of 1.5 Mpc. The orientation of the coordinate frame in the plot is arbitrary.

The colours of the dots in the plot indicate how the subhaloes move relative to the Hubble flow fitted to the data: red points recede faster than the fitted Hubble flow would suggest and blue points recede slower. Visual inspection of the plot would suggest that the radial velocities might also be anisotropically distributed. For example approximately between -45° and -60° near the centerline of the plot there is a collection of subhaloes coloured almost uniformly blue and most of the reddest subhaloes with radial velocities exceeding that of the Hubble flow by more than 450 km/s are located near each other at the right edge of the plot.

This kind of visual inspection, especially when done only to individual simulation outputs, can not be the sole means of determining whether the radial velocities are anisotropically distributed. In the following subsections two approaches to quantifying the possible anisotropy are introduced. First, in section 5.3.1, the dependency between how the radial velocities depend on the angle between the observed subhalo and the M31 analogue in the simulation is applied. Later in section 5.3.2 another approach using clustering algorithm introduced in section 4.6 is used to study the structures surrounding the Local Group analogues.

5.3.1 Angular Dependencies

Many established ways to orient the coordinate system around the observer are not usable in dark matter only simulations. For example the plane of the galaxy, let alone the orbit of anything analogous to the Sun, is not resolved. Thus in this work, a complete uniform coordinate system is not used across simulations.

Instead, when a directional binning is desired, subhaloes are classified based on how they relate to the vector from the less massive to more massive primary in each simulation. This is chosen as, due to the asymmetrical shape of the Local Group that is dominated by the two primary galaxies, it is expected that near the Local Group analogues the Hubble flow is not isotropic. The Milky Way analogues moving relative to the centres of corresponding Local Group analogues also affects the observed velocities.

As another similarly justified coordinate axis that could be used with the line connecting the Milky Way and M31 analogues does not exist, when subhaloes are binned based on their observed location on the sky, the angle (marked with ϕ) between the M31 and the subhalo is used as the only coordinate. Thus a subhalo located very near the M31 analogue on the sky would have a small value of ϕ and a subhalo that is located on the line connecting the Milky way and M31 analogues in the direction away from M31 as seen from the Milky Way would have $\phi = 180°$.

This scheme will naturally result in bins with constant angular width being smaller in terms of the volume when located near the line connecting the primaries and larger with values of ϕ near 90°, and thus it is not suitable for e.g. comparing numbers of subhaloes in each bin without scaling the counts according to the volume within each bin. For calculating quantities like Hubble flow parameters this kind of binning works well though, as long as also the smallest bins are large enough to have a sufficient number of data points in them.

This kind of angular binning is used to produce Fig. 5.10 that shows how H_0 and the zero point of the Hubble flow change as a function of ϕ. For this plot, 20° bins were used and the Hubble flows were fitted using subhaloes up to 5 Mpc away from the Milky Way analogue. Simulations that had their first type 2 particle at a distance of less than 5 Mpc were not used. Also if any bin in a simulation contained less than 15 subhaloes, this bin was excluded from the analysis. A separate Hubble flow fit was calculated for each bin in each simulation using the fitting procedure

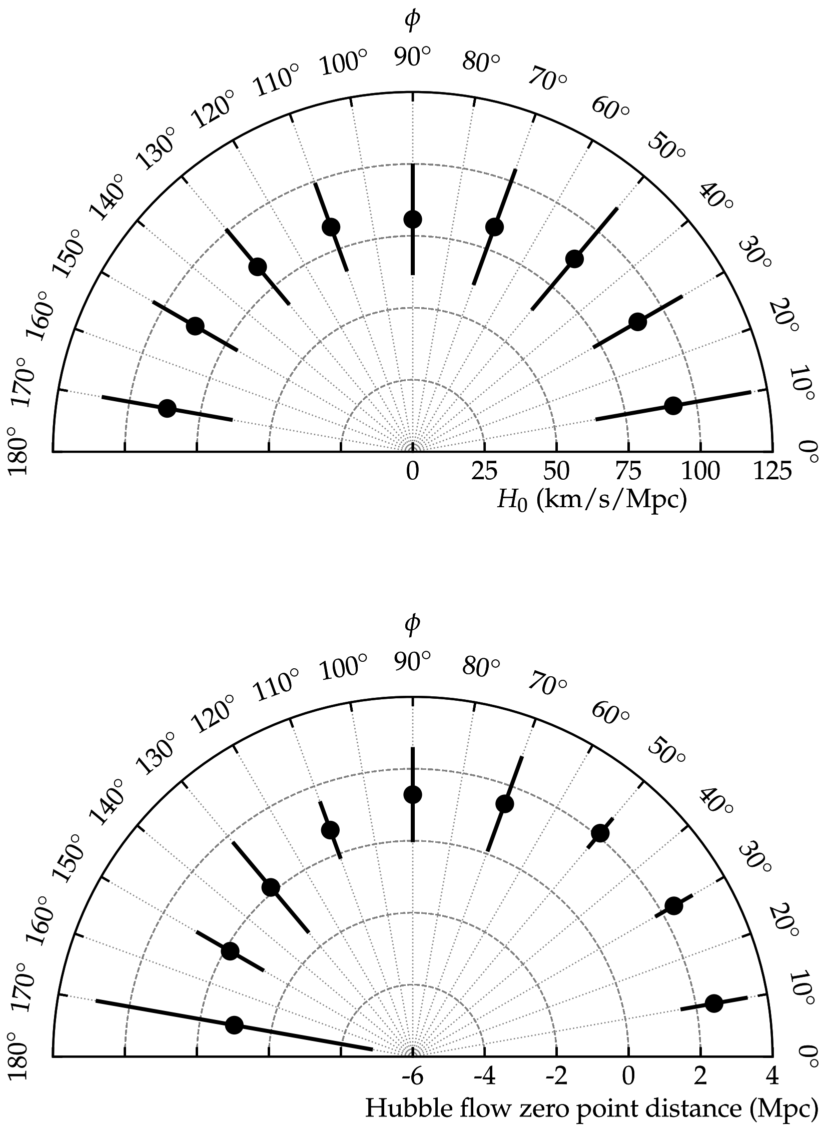

Figure 5.10: Hubble constants and Hubble flow zero points when measured in 20° bins in each simulation, the angle being measured as the angle between M31 and each subhalo as seen from the Milky Way. The dots mark the mean result and the whiskers show the standard deviation in each bin. Note that in the zero point distance plot the distance axis does not start from zero but from -6 Mpc instead.

introduced in section 5.2.2. The figure shows the means and standard deviations of the results in each bin.

From the plots, it can be seen that at distances of less than 5 Mpc the Hubble flow around the simulated Milky Way analogues is not isotropic. This is clear when looking at the zero point distance plot, where the three bins closest to the direction towards the M31 analogue have the mean zero point at more than 2 Mpc away from the Milky way analogue whereas when looking away from the M31 analogues, it is common that the Hubble flow has not yet reached a radial velocity of 0 km/s when the Milky Way analogue is reached, resulting in negative values of the zero point distance.

This is likely to a great extent caused by a combination of the observer being located at the Milky Way analogue instead of the centre of the Local Group and the Local Group selection criteria also requiring the observer to be moving toward the M31 analogue. This means that the effect the Local Group has on the surrounding Hubble flow will not be isotropic, as when looking in the direction of a large ϕ, the observer is looking almost directly away from the Local Group whereas for small values of ϕ most of the Local Group is within the observed region. The movement of the Milky Way analogue also affects the results: if the mass center of the Local Group is assumed to be in rest relative to its surroundings, the movement of the Milky Way analogue towards it will add either a positive or negative radial velocity to all haloes within each bin. These effects cause the plot to appear like the measured values of zero point are offset to the right.

The differences in the H_0 plot are more subtle, but these measurements are not independent of the direction either. Perhaps the most prominent effect in it is that when H_0 is measured near $\phi = 90°$, the obtained values tend to be smaller than when measured either towards or away from the M31 analogue. Differences between measurements done near the ends of the allowed range are difficult to discern

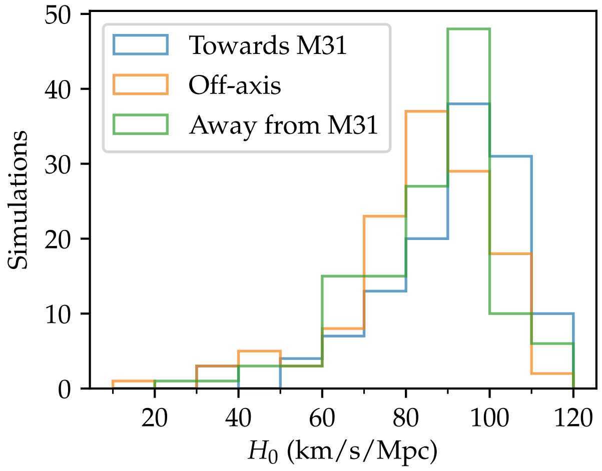

Figure 5.11: The distribution of H_0 measurement results when subhaloes with angle between them and M31 is less than 45°(blue histogram), more than 135°(green histogram) or in between (orange histogram).

from this plot, though it appears that the values of H_0 measured towards the M31 analogue might be slightly higher than when measured away from it.

In order to compare the values of H_0 when measured towards or away from the M31 analogue or in the perpendicular direction, Fig. 5.11 shows the histograms of H_0 fitting results in three bins with unequal sizes. This plot shows that the measurements done in either direction along the axis connecting the Milky Way and M31 analogues have the peak of the distribution at a higher value than the measurements in the off-axis direction. The highest measured values of H_0 are also more common in the two on-axis bins. The difference is especially clear between the measurements done in the $\phi < 45°$ and the off-axis bins, with the difference between the off-axis and $\phi > 135°$ bins being smaller.

The same observation can be made by comparing medians of the H_0 measurements in each bin. The smallest median H_0 of 86.9 km/s/Mpc is measured in the off-axis bin with the $\phi > 135°$ having slightly higher median value of 89.9 km/s/Mpc. The highest value was measured in the bin with $\phi < 45°$ where the median H_0 was

94.2 km/s/Mpc. This agrees well with what is seen in Figs. 5.10 and 5.11, hinting that the Hubble Flows around the Milky Way analogues are indeed anisotropic at the examined distances.

5.3.2 Clustering

One way of studying the anisotropies of the Hubble flow in more detail is to apply a clustering algorithm to the subhaloes in each simulation and then fit the Hubble flow separately to subsets of all haloes depending on their cluster memberships. In this work, clustering is used to find groups of subhaloes located near each other on the sky as seen from the Milky Way analogue. This is done with the DBSCAN algorithm introduced in section 4.6.

The algorithm has two free parameters, MinPts and ε, which have to be chosen before the clustering is performed. Effect of changing the parameters on the mean number and diameter of the found clusters is explored in Figs. 5.12 and 5.13. As can be seen from Fig. 5.12, for small values of MinPts the clusters found per simulation tends to be very high as e.g. a single subhalo or a pair of subhaloes can form a cluster. Only when ε is set to a very high value the number of found clusters gets smaller when the clusters merge, up to a point when nearly all subhaloes form one cluster.

When MinPts is increased, the number of found clusters decreases as clusters with less dense central regions no longer qualify as clusters. The only exception to this is that at high values of ε, increasing the required MinPts can break large clusters with multiple dense regions into two or more smaller clusters that are no longer connected with core particles. These changes are also reflected in Fig. 5.13, where merging and separating of clusters affects the mean diameter. Only for small values of ε increasing MinPts has only a small effect on the diameters, which is explained by clusters with small ε being either small or non-existent with all values

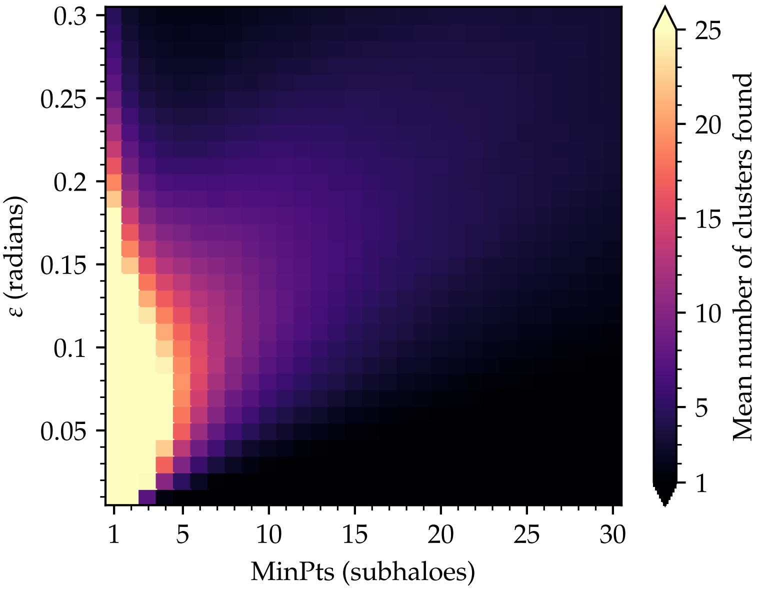

Figure 5.12: Mean number of clusters found in a simulation for all simulations in dataset with different DBSCAN parameters.

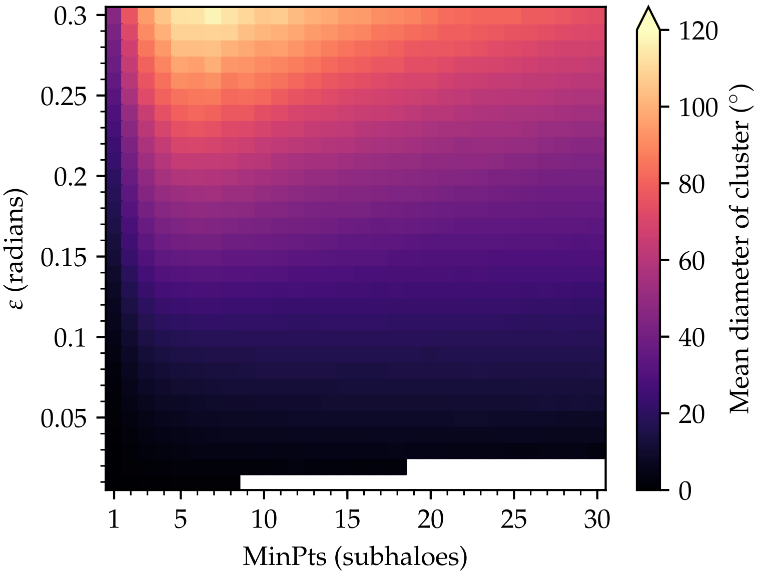

Figure 5.13: Mean diameter of clusters found for all simulations in dataset with different DBSCAN parameters, the diameter of a cluster being defined as the largest distance between a pair of points that are members of the cluster. The white area in the lower right corresponds to the area where no clusters were found.

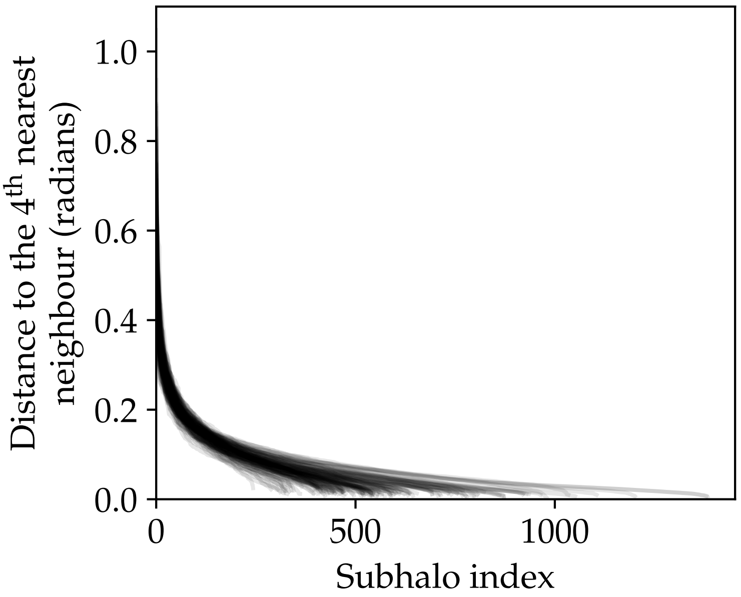

Figure 5.14: Sorted 4-dist graphs for subhaloes between 1.5 and 5.0 Mpc from the Local Group in each simulation that is uncontaminated at least up to 5 Mpc from the Local Group. Each line shows the sorted distances from each particle in one simulation to its 4$^{\text{th}}$ nearest neighbour. The plotted distances are measured as the angular distance between the subhaloes as seen from the Milky Way analogue.

of MinPts.

Figs. 5.12 and 5.13 aid in choosing the values of the parameters by revealing clearly bad combinations that lead to nearly all subhaloes belonging to one cluster or to a large number of small clusters with only a few members in each. Still determining the best values for ε and MinPts requires using tools introduced in section 4.6. As the clustering is done based on the position of the subhaloes as seen on the sky, the data set is two-dimensional and thus the recommended value of MinPts is 4, which is used in the analysis.

The accompanying value of ε is selected using a 4-distance graph seen in Fig. 5.14. The lines representing different simulations are fairly tightly packed near the elbow of the plot, suggesting that using the same value of ε for all simulations produces acceptable results. Due to the scatter of the lines, selecting the optimal value is challenging, though the elbow seems to be located near the value $\varepsilon = 0.16$.

Figure 5.12 shows the effect of varying minsamples and ε on number of clusters found in each simulation (1.5 Mpc $<$ r $<$ 5.0 Mpc again). Regions where there are ridiculously many clusters and ones where there are one or zero, relevant region in between, some areas have similar number of clusters but do the clusters look the same, see plots that don't exist yet

Figures 5.15 and 5.16 show how the clustering results vary when clustering parameters are varied.

Figure 5.18 shows how derived values of slope and zero-point for the Hubble flow change when the Hubble flow fitting is carried out on partial data chosen based on the cluster membership of the haloes.

5.4 Statistical Estimate of the Local Group Mass

Analysis similar to Fattahi et al 2016 paper

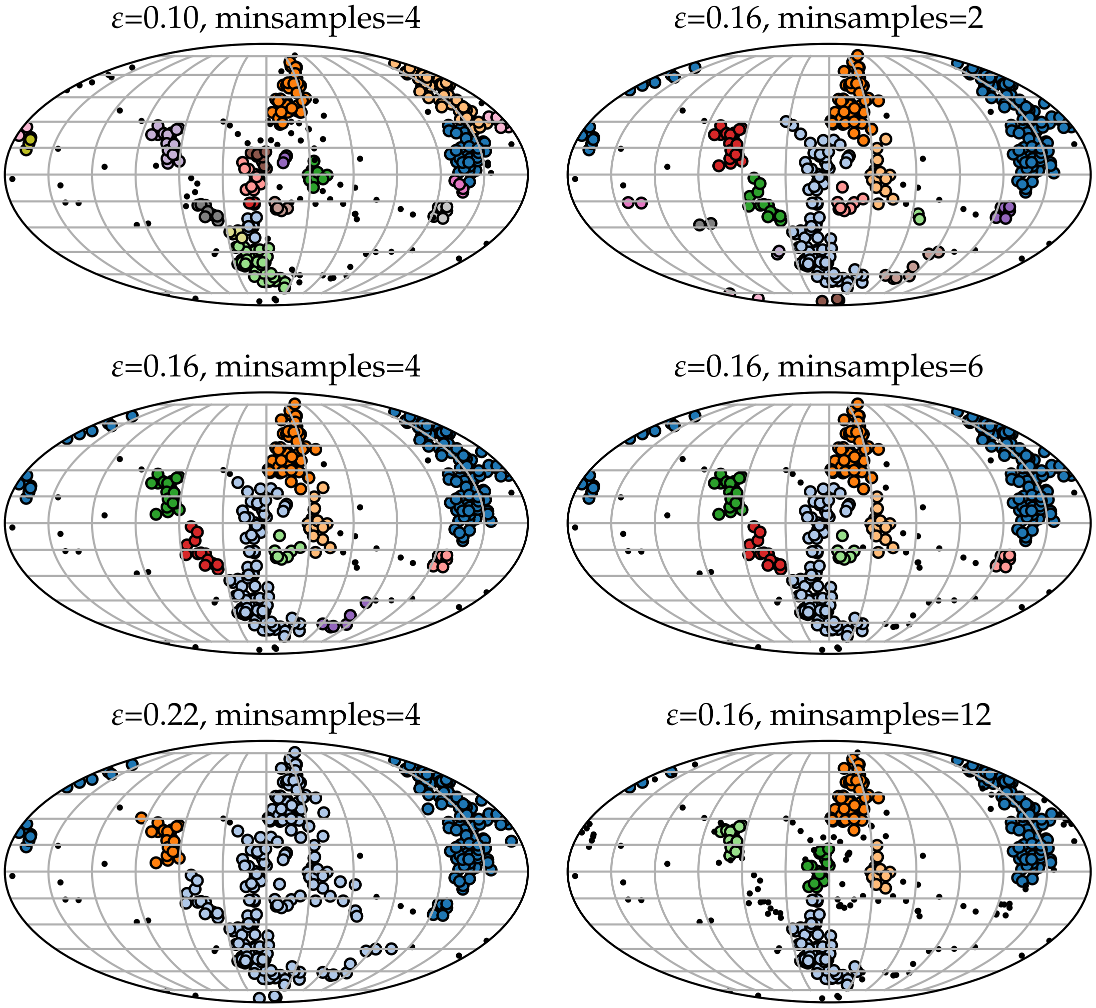

Figure 5.15: Results of DBSCAN clustering on same simulation output with different clustering parameters. TODO: erittele parametrien muuttamisen vaikutukset

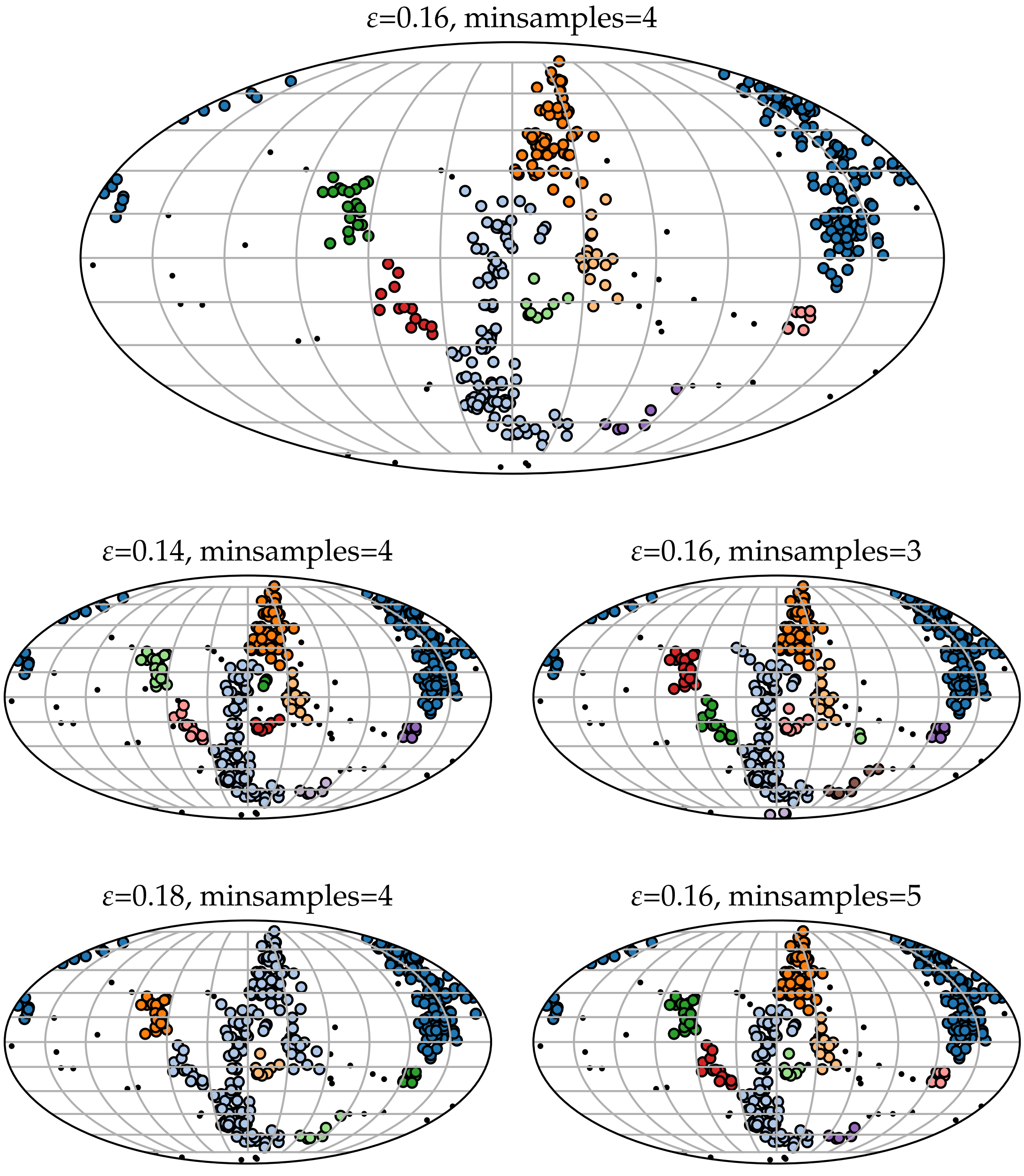

Figure 5.16: The effect of slightly variying the clustering parameters around the values ε=0.16 and minsamples=4 used when analysing clustered data. TODO: more sensitive to ε

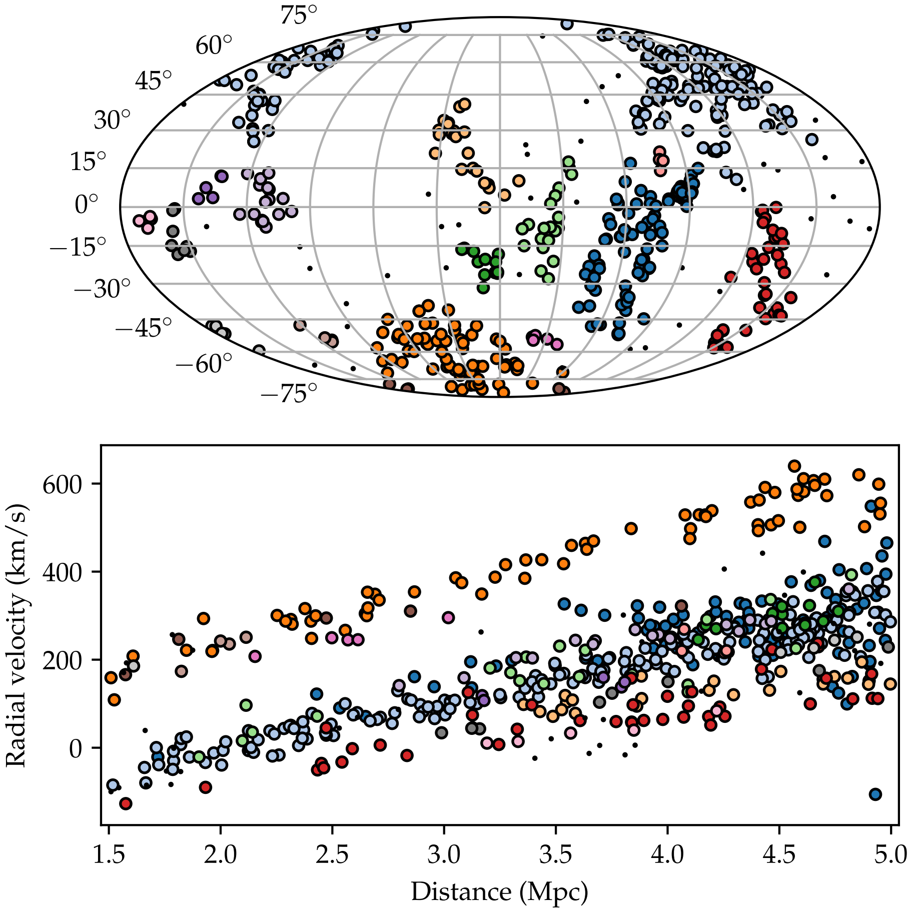

Figure 5.17: TODO

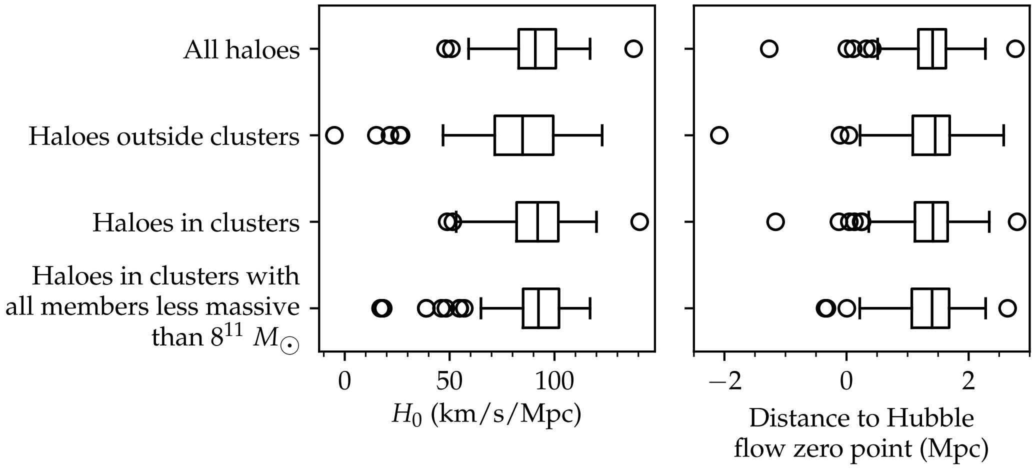

Figure 5.18: Hubble constant and distance to the point at which velocity due to the fitted Hubble flow is zero calculated from different samples. HUOM OBS NB sim 299 excluded,

6. Conclusions

A. Principal Components

PC	H_0	HF zero	HF zero (clustered)	HF zero (not clustered)	σ_{radvel}	σ_{radvel} (clustered)	σ_{radvel} (not clustered)	$v_{r,LG}$	$v_{t,LG}$	r_LG
1	-0.386	-0.449	-0.324	-0.379	-0.393	-0.211	-0.384	-0.211	0.097	0.013
2	0.147	0.221	0.248	0.150	-0.064	-0.599	-0.056	-0.599	-0.103	0.332
3	0.287	0.145	0.186	0.223	-0.531	0.258	-0.535	0.258	0.115	0.313
4	0.009	0.020	-0.152	0.102	0.151	-0.047	0.136	-0.047	0.934	0.221
5	-0.024	-0.171	-0.273	-0.105	0.140	0.134	0.227	0.134	-0.270	0.840
6	0.015	-0.092	0.706	-0.663	0.049	0.065	0.072	0.065	0.147	0.127
7	-0.848	0.156	0.323	0.343	-0.074	0.071	0.060	0.071	-0.001	0.128
8	-0.162	0.582	-0.206	-0.315	0.456	0.024	-0.529	0.024	-0.018	0.061
9	0.041	-0.572	0.239	0.326	0.549	-0.008	-0.452	-0.008	-0.016	0.031
10	0.000	-0.000	0.000	0.000	-0.000	0.707	-0.000	-0.707	0.000	-0.000

Table A.1: component H0s zeropoints inClusterZeros outClusterZeros allDispersions clusterDispersions unclusteredDispersions radialVelocities tangentialVelocities LGdistances