Friends of Friends Algorithm for Identifying Star Forming Clouds

Aryan Jain

May 20, 2024

1 Introduction

The Friends of Friends (FoF) algorithm is a widely used method in cosmology for identifying groups of particles, such as dark matter halos, based on their spatial proximity. This algorithm groups particles into clusters by linking those that are within a certain distance of each other. In this project, we apply the FoF algorithm to identify star forming clouds in a simulated galaxy. We begin by generating a random distribution of particles, calculate their pairwise distances, and then use a KD-tree structure to efficiently find clusters of particles within a specified linking length.

2 Methodology

The primary steps in our methodology include generating a particle distribution, calculating pairwise distances, implementing the FoF algorithm, and visualizing the results.

2.1 Generating Particle Distribution

To simulate star forming clouds, we generate particles from multiple Gaussian distributions. Each Gaussian distribution is defined by a mean vector μ and a covariance matrix Σ. The probability density function for a multivariate Gaussian distribution is given by:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right) \tag{1}$$

where $\mathbf{x}$ is a k-dimensional random vector, μ is the mean vector, and Σ is the covariance matrix.

The Cholesky decomposition is used to generate samples from these distributions. Given a covariance matrix Σ, the Cholesky decomposition allows us to express Σ as $\Sigma = LL^T$, where L is a lower triangular matrix. If $\mathbf{z}$ is a standard

normal random vector, the affine transformation to obtain samples from our Gaussian distribution is:

$$\mathbf{x} = \mu + L\mathbf{z} \tag{2}$$

This transformation ensures that the samples $\mathbf{x}$ have the desired mean and covariance.

2.2 Calculating Pairwise Distances

The pairwise Euclidean distances between all particles are calculated to determine the mean distance. This mean distance serves as a reference for setting the linking length in the FoF algorithm. Calculating these distances involves computing the norm of the difference between each pair of particles, which can be expressed as:

$$d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \tag{3}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the positions of particles i and j, respectively.

2.3 Friends of Friends Algorithm

The FoF algorithm identifies clusters by linking particles that are within a specified distance (linking length) of each other. To improve efficiency, a KD-tree is used to quickly query which particles are within this linking length of a given particle. The algorithm then iteratively merges clusters that share common particles until all possible clusters are identified.

2.4 Visualization

The resulting clusters are visualized using 3D scatter plots, with each cluster colored differently. This visualization helps to understand the spatial distribution and structure of the star forming clouds identified by the FoF algorithm.

3 Results

We applied the FoF algorithm to our simulated particle data. The linking length was set as a multiple of the mean pairwise distance between particles. This allowed us to control the granularity of the identified clusters. The results, shown in Figure 1, display the particles in a 3D scatter plot with different colors representing different clusters.

The visualization reveals distinct clusters of particles, indicating the presence of star forming clouds. The KD-tree based implementation of the FoF algorithm efficiently handled the clustering of particles, demonstrating its effectiveness for large-scale simulations.

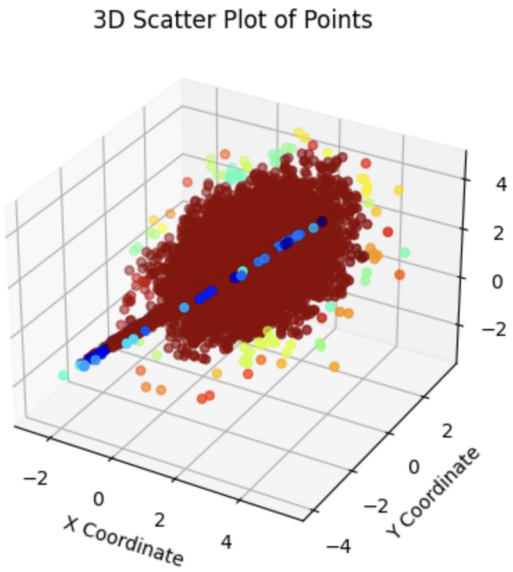

Figure 1: 3D Scatter Plot of Particles Colored by Clusters

4 Conclusion

The Friends of Friends algorithm is a powerful tool for identifying clusters of particles in cosmological simulations. By adjusting the linking length, we can control the granularity of the clusters, making it suitable for studying various structures, from small star forming regions to large galactic clusters. Our implementation, using a KD-tree for efficient neighbor searching, proves effective for handling large datasets typical in cosmological studies.

5 Appendix

5.1 Python Code Overview

Here we provide a brief overview of the Python code used in this project. The main functions include:

- **create_distribution:** Generates particles from Gaussian distributions using specified means and covariance matrices.
- **norm_calculator:** Calculates the pairwise Euclidean distances between particles.
- **mean_distance:** Computes the mean of the pairwise distances.

- **friends_of_friends:** Implements the FoF algorithm using a KD-tree for efficient neighbor searching.
- **plot_clusters:** Visualizes the clusters in a 3D scatter plot.

The complete Python code can be found in the supplementary materials accompanying this document.