# Summer School 2025

# Astronomy & Astrophysics

# Project Report

## Prepared By

**Student Name: Narjish Khan**
**Institution Name: Tezpur University**
**Institution Roll No: PHI23009**
**ISA Admission No: 821700**

## Project Name

**Estimating the Dynamical Mass of the Galaxy Cluster**

## Submitted To

**Name: Mr. Sahil Sakkarwal**

**Designation: Program Supervisor**

**Institution: India Space Academy**

## Project 1: Estimating the Dynamical Mass of the Galaxy Cluster

# Introduction

Galaxy clusters are the largest gravitationally bound systems in the universe, comprising hundreds to thousands of galaxies, vast quantities of hot intra cluster gas, and a dominant component of dark matter. As such, they play a crucial role in our understanding of cosmic structure formation and evolution. Studying the properties of galaxy clusters provides insight into the distribution of matter on large scales, the dynamics of galaxy interactions, and the underlying cosmological parameters that govern the universe.

In this project, we analyze a galaxy cluster using spectroscopic redshift data to investigate its physical and dynamical characteristics. Redshift measurements allow us to identify galaxies that are gravitationally bound within the cluster and distinguish them from foreground and background objects. By analyzing the distribution of galaxy velocities and spatial positions, we can infer the cluster's systemic redshift, velocity dispersion, spatial extent, and total gravitational mass.

Through a combination of statistical analysis and cosmological calculations, this study aims to derive key parameters that describe the cluster's structure and mass distribution. The resulting measurements, particularly the velocity dispersion and dynamical mass, offer important clues about the cluster's formation history and the role of dark matter in its assembly.

# Methodology

### Step 1: Identify Galaxies that are members of a cluster

To identify cluster member galaxies using redshift data, we look for galaxies with similar redshifts, accounting for small variations due to peculiar velocities within the cluster. These variations cause a redshift spread around the cluster's mean redshift, known as velocity (or redshift) dispersion, typically around 500–1000 km/s for rich clusters—equivalent to a redshift range of about ±0.0017–0.0033. Galaxies within this range are considered likely cluster members.

Below is a point wise explanation of what I did to identify galaxies that are member of cluster

- Combine Multiple Observations Per Object: For each unique galaxy (identified by its object ID), we averaged the spectroscopic redshift (specz) values to get a single, representative redshift per object. For other properties like right ascension (RA), declination (Dec), and projected separation (proj_sep), we assumed they remain constant and took the first recorded value.
- Calculate Redshift Statistics: We calculated the mean and standard deviation of the averaged redshift values across all objects to understand the typical redshift and the spread of the data.
- Apply the 3-Sigma Rule: To identify galaxies likely to be part of the same cluster, we applied a common statistical rule: galaxies with redshifts within three standard deviations ($\pm 3\sigma$) of the mean redshift are considered part of the cluster. This helps exclude outliers and foreground/background galaxies.
- Filter for Cluster Members: Using the defined lower and upper redshift limits, we selected only the galaxies whose redshifts fall within this 3-sigma range, identifying them as probable cluster members.
- Count Cluster Galaxies: We counted the number of galaxies that fall within this filtered redshift range. In this case, **91 galaxies** were identified as cluster members.

- Visualize Redshift Distributions: Finally, we created two histograms: One showing the redshift distribution of all galaxies. Another showing only the redshift distribution of the filtered (cluster) galaxies. This helps visualize how the filtering isolated a central group from the broader distribution.

## Step 2: Determine the cluster redshift and characteristic velocity dispersion of galaxies that belong to the cluster in units of km/s

- Select Cluster Members: We Begin with a set of galaxies identified as belonging to a galaxy cluster.
- Calculate Mean Cluster Redshift: We Compute the average of the spectroscopic redshift (specz) values for all the cluster member galaxies. This average represents the systemic redshift of the cluster
- Compute Relative Velocities: For each galaxy, we calculate its line-of-sight velocity relative to the cluster's redshift using the relativistic Doppler velocity formula, which accounts for the relativistic effects at higher redshifts
- Inspect Velocity Distribution: we generate descriptive statistics (like mean, standard deviation, min, max) of the computed velocities to understand the distribution.
- Calculate Velocity Dispersion: we determine the velocity dispersion by calculating the standard deviation of these relative velocities.
- Final Output: The mean cluster redshift is reported =**0.08007**

The velocity dispersion is given by **1.212e+03 km/s** which is a key indicator of the cluster's mass.

## Step 3: Estimate the cluster Size in Megaparsecs (Mpc)

- Compute Cluster Center: We take the average RA and Dec of all cluster member galaxies.
- Calculate Angular Separation: For each galaxy, we compute its angular distance from the cluster center (in arcminutes).
- Visualize Distribution: we plot a histogram showing how many galaxies fall at each angular separation — helps reveal the cluster's spatial extent
- Estimate Co-moving Distance (r): We use a low-redshift cosmological approximation involving redshift, speed of light, H0 and deceleration parameter q0.
- Calculate Angular Diameter Distance DA: We divide co-moving distance by (1+z).
- Convert Angular to Physical Size: We find the maximum angular separation theta in arcminutes, convert it to radians.
- Compute Cluster Diameter in Mpc: We multiply DA by theta in radians to get physical diameter. We Halve it to get the cluster radius.

## Step 4: Estimate the dynamical mass of the Cluster

- Apply the Virial Theorem: We use the formula:

$$3\sigma$$

- Define the Physical Quantities

σ: Velocity dispersion (converted from km/s to m/s).

R: Radius of the cluster (half of the physical diameter, converted to meters).

G: Gravitational constant in appropriate units (SI or astrophysical units).

- Calculate Dynamical Mass: We plug the values into the virial formula to compute the cluster's mass in kilograms or solar masses.
- Convert to Solar Masses: We divide the result by the mass of the Sun () to express the result in solar masses.
- Interpret the Result

The final value **4.79e+14 M⊙** gives an estimate of the clusters total gravitational mass, including dark matter.

### Step 5: Consistency between Dynamical mass and luminous mass

We calculated the luminous mass but the estimate of the dynamical mass is not consistent with what would be expected based solely on the luminous mass of the galaxy cluster. While the total visible mass from stars in the 91 galaxies is considerable, it accounts for only about 1.9% of the total mass that is inferred from the gravitational forces required to keep the cluster bound together. The vast majority—approximately 98.1%—of the cluster's mass appears to be invisible, as it does not emit, absorb, or reflect detectable light. This striking discrepancy indicates that the observable matter, such as stars and gas, is insufficient to explain the strong gravitational effects measured, such as the motions of galaxies within the cluster and the gravitational lensing of background light. Therefore, the difference must be attributed to non-luminous components, most likely a combination of hot intra cluster gas and, more significantly, dark matter. Such a high mass ratio between the dynamical mass and the luminous mass serves as compelling evidence that the cluster is dominated by matter we cannot directly see. This supports the widely accepted view in modern astrophysics that dark matter is the dominant form of matter in the universe, playing a crucial role in the formation, stability, and evolution of large-scale cosmic structures like galaxy clusters.

# Result

The Resultant Dynamical mass of the galaxy cluster is 4.79e+14 M⊙

# Conclusion

In this project, we estimated the dynamical mass of a galaxy cluster using spectroscopic redshift data. The process involved identifying cluster member galaxies, calculating their velocity dispersion, determining the cluster's size, and applying the virial theorem.

We began by averaging redshift values for each galaxy and used the 3-sigma rule to identify 91 likely cluster members based on their redshift proximity to the mean. Histograms of the redshift distribution confirmed a clear clustering around a mean redshift of 0.08007.

Next, we calculated each galaxy's velocity relative to the cluster using the relativistic Doppler formula. The velocity dispersion was found to be 1212 km/s, indicating the cluster's gravitational binding strength.

To estimate the physical size, we calculated the cluster center from RA and Dec values, measured maximum angular separations, and converted these into a physical radius using cosmological parameters.

Finally, we applied the virial theorem to estimate the dynamical mass, arriving at ~$4.79 \times 10^{14}$ solar masses.

This estimate of the dynamical mass is not consistent with what would be expected based solely on the luminous mass of the cluster. The total visible mass from stars in the 91 galaxies accounts for only about 1.9% of the total mass inferred from gravitational dynamics. The remaining ~98.1% appears to be invisible, pointing strongly to the presence of dark matter and hot intra cluster gas. This significant discrepancy underscores the dominant role of dark matter in binding the cluster and supports the modern astrophysical understanding that most of the universe's mass exists in a non-luminous form.

# 1_dynamical_mass (1)

June 30, 2025

#### 0.0.1 Step 1: Importing Necessary Libraries

We begin by importing Python libraries commonly used in data analysis and visualization: - `numpy` for numerical operations - `matplotlib.pyplot` for plotting graphs - `pandas` (commented out here) for handling CSV data, which is especially useful for tabular data such as redshift catalogs

> Tip: If you haven't used `pandas` before, it's worth learning as it offers powerful tools to manipulate and analyze structured datasets.

For reading big csv files, one can use numpy as well as something called "pandas". We suggest to read pandas for CSV file reading and use that

[23]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from astropy.constants import G, c
from astropy.cosmology import Planck18 as cosmo
import astropy.units as u
```

Before we begin calculations, we define key physical constants used throughout:

- $ H_0 $: Hubble constant, describes the expansion rate of the Universe.
- $c$ : Speed of light.
- $G$: Gravitational constant.
- $q_0$ : Deceleration parameter, used for approximate co-moving distance calculations.

We will use **`astropy.constants`** to ensure unit consistency and precision.

[2]:
```
# Constants:

H_0 =2.2685e-18  # Hubble constant in SI
c =2.99792458e8  # Speed of light in m/s
G =2.16345e-27  # Gravitational constant in pc kg^-1 (m/s)^2
q0=-0.534  # Deceleration parameter (assumed from Planck fit KEEP it as it is)
```

Read the csv data into the python using the method below

[3]:
```
df = pd.read_csv("D:\jupyter project\Skyserver_SQL6_15_2025 6_23_29 PM.csv") #␣
↪Download the data as instructed in the pdf
df
```

```
<>:1: SyntaxWarning: invalid escape sequence '\j'
<>:1: SyntaxWarning: invalid escape sequence '\j'
C:\Users\Acer\AppData\Local\Temp\ipykernel_12980\4057187350.py:1: SyntaxWarning:
invalid escape sequence '\j'
  df = pd.read_csv("D:\jupyter project\Skyserver_SQL6_15_2025 6_23_29 PM.csv") #
Download the data as instructed in the pdf
```

[3]:

```
    #Table1
objid               ra        dec       photoz    photozerr specz      speczerr
proj_sep          umag      umagerr     gmag      gmagerr     rmag      rmagerr
obj_type
1237671768542478711 257.82458 64.133257 0.079193 0.022867  0.08244731
1.657283E-05 8.34773316331893 18.96488 0.04337703 17.49815 0.005671791 16.75003
0.004707855         3
                                                           0.08246633
1.43452E-05  8.34773316331893 18.96488 0.04337703 17.49815 0.005671791 16.75003
0.004707855         3
1237671768542478713 257.83332 64.126043 0.091507 0.014511  0.08121841
2.131049E-05 8.01125941762232 20.22848 0.07201946 18.38334 0.007762768 17.46793
0.005827651         3
1237671768542544090 257.85137 64.173247 0.081102 0.009898  0.07956107
2.217769E-05 8.73927598457626 19.21829 0.05013492 17.1897  0.004936492 16.22043
0.003769081         3
…
…
1237671939804627464 258.44994 64.025909 0.081894 0.013624  0.0718016
2.505877E-05 9.3161404176649  20.05525 0.06565119 18.35306 0.008029834 17.51662
0.005922728         3
1237671939804627483 258.43205 64.123685 0.077184 0.010523  0.08079002
2.528532E-05 8.14615367880558 20.08116 0.08206341 18.15476 0.007831404 17.27835
0.005706132         3
1237671939804627518 258.46676 64.119499 0.088762 0.016934  0.08287635
5.564596E-06 8.98602887839081 18.92277 0.0362908  17.83406 0.00681372  17.36895
0.006322416         3
1237671939804627535 258.48871 64.111343 0.079721 0.009565  0.08089872
2.86031E-05  9.48337428131356 19.85553 0.06715845 17.81776 0.006377555 16.88524
0.004626879         3
                                                           0.08089609
2.330272E-05 9.48337428131356 19.85553 0.06715845 17.81776 0.006377555 16.88524
0.004626879         3

[140 rows x 1 columns]
```

[4]:

```
print(df.columns)
```

```
Index(['#Table1'], dtype='object')
```

[5]:
```
df = pd.read_csv("D:\jupyter project\Skyserver_SQL6_15_2025 6_23_29 PM.csv",␣
  ↪header=1)
print(df.columns)
```

```
Index(['objid', 'ra', 'dec', 'photoz', 'photozerr', 'specz', 'speczerr',
       'proj_sep', 'umag', 'umagerr', 'gmag', 'gmagerr', 'rmag', 'rmagerr',
       'obj_type'],
      dtype='object')
```

```
<>:1: SyntaxWarning: invalid escape sequence '\j'
<>:1: SyntaxWarning: invalid escape sequence '\j'
C:\Users\Acer\AppData\Local\Temp\ipykernel_12980\3713588628.py:1: SyntaxWarning:
invalid escape sequence '\j'
  df = pd.read_csv("D:\jupyter project\Skyserver_SQL6_15_2025 6_23_29 PM.csv",
header=1)
```

### 0.0.2 Calculating the Average Spectroscopic Redshift (specz) for Each Object

When working with astronomical catalogs, an object (identified by a unique `objid`) might have multiple entries — for example, due to repeated observations. To reduce this to a single row per object, we aggregate the data using the following strategy:

"'python averaged_df = df.groupby('objid').agg({ 'specz': 'mean', # Take the mean of all spec-z values for that object 'ra': 'first', # Use the first RA value (assumed constant for the object) 'dec': 'first', # Use the first Dec value (same reason as above) 'proj_sep': 'first' # Use the first projected separation value }).reset_index()

[6]:
```
# Calculating the average specz for each id:
averaged_df = df.groupby("objid").agg({
    "specz": "mean",        #Average redshift for each galaxy
    "ra": "first",          #Keep the first instance of Right Ascension(Ra)
    "dec": "first",         #keep the first instance of Declination(Dec)
    "proj_sep": "first"     #keep the first projected separation value
}).reset_index()
```

To create a cut in the redshift so that a cluster can be identified. We must use some logic. Most astronomers prefer anything beyond 3*sigma away from the mean to be not part of the same group.

Find the mean, standard deviation and limits of the redshift from the data

[7]:
```
mean_z = averaged_df["specz"].mean()
std_z = averaged_df["specz"].std()

#Define the redshift limits using the 3 sigma rule
low_limit = mean_z - 3*std_z
up_limit = mean_z + 3*std_z
```

[8]:
```
df_cluster_members = averaged_df[(averaged_df["specz"] >= low_limit) &␣
  ↪(averaged_df["specz"] <= up_limit)]
```

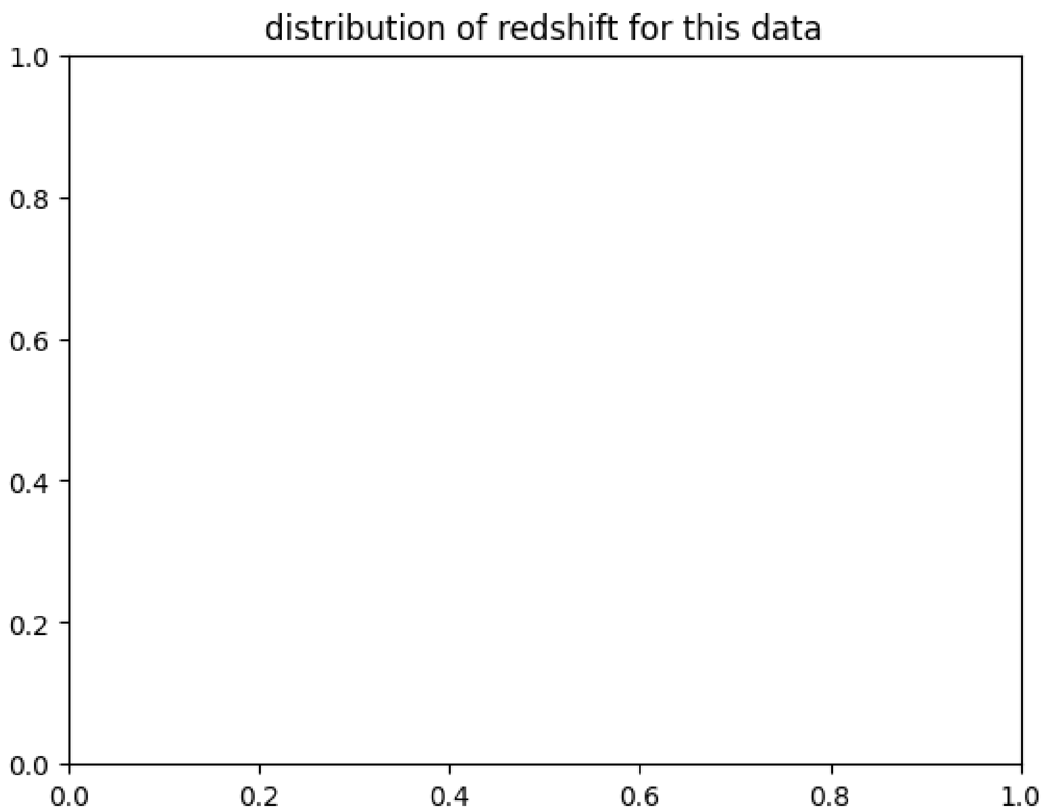
[9]:
```
num_members = len(df_cluster_members)
print(f"Number of galaxies identified in the cluster: {num_members}")
```

```
Number of galaxies identified in the cluster: 91
```

You can also use boxplot to visualize the overall values of redshift

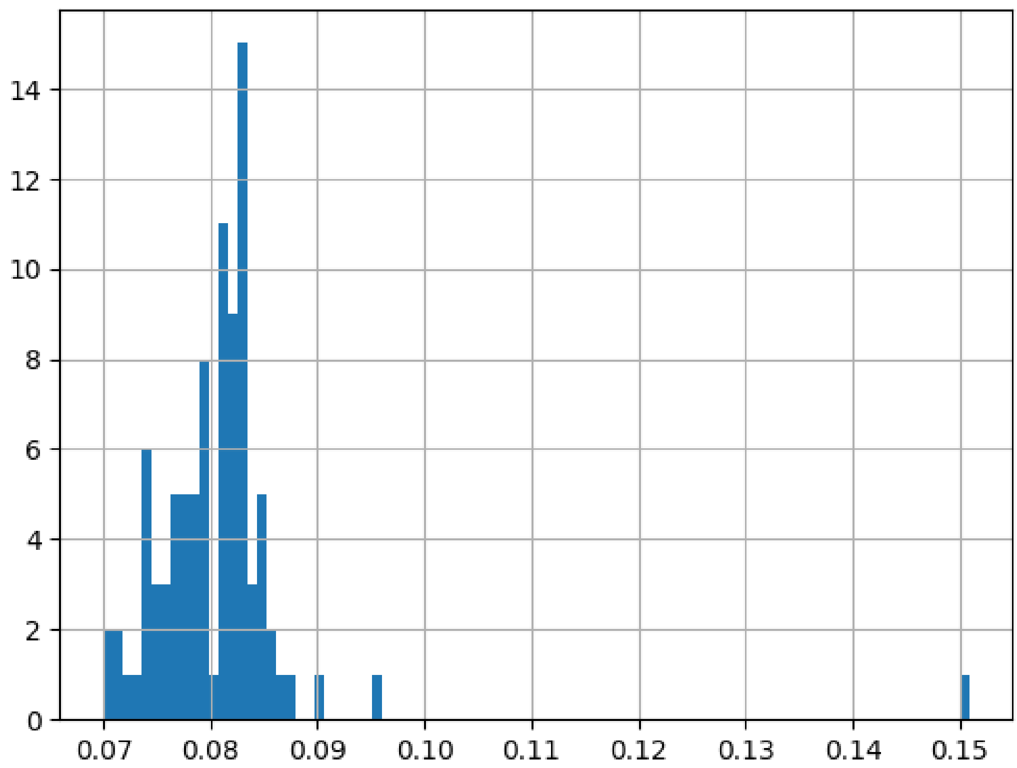
[10]:
```
# Plot the dsitribution of redshift as histogram and a boxplot
plt.title("distribution of redshift for this data")
```

[10]:
```
Text(0.5, 1.0, 'distribution of redshift for this data')
```




But the best plot would be a histogram to see where most of the objects downloaded lie in terms of redshift value

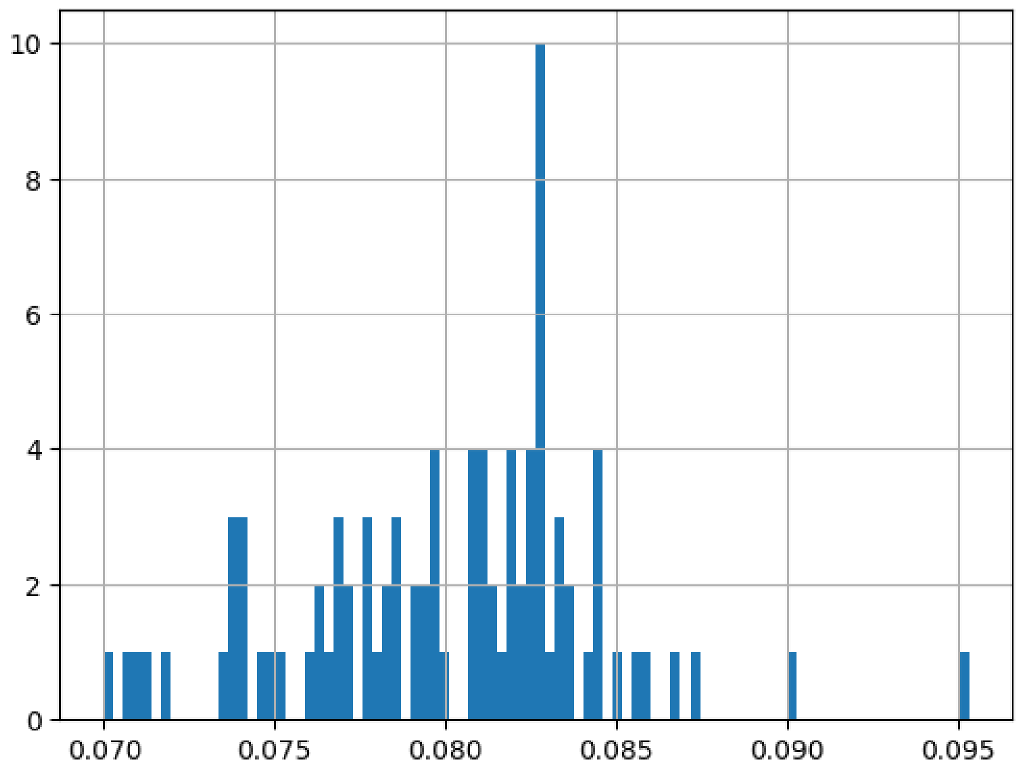
[11]:
```
plt.hist(averaged_df['specz'],bins=90)
plt.grid()
plt.show()
```




Filter your data based on the 3-sigma limit of redshift. You should remove all data points which are 3-sigma away from mean of redshift

[12]:
```
# Filtering the data based on specz values, used 3 sigma deviation from mean as␣
↪upper limit.
filtered_df = averaged_df[(averaged_df['specz'] >= low_limit) &␣
↪(averaged_df['specz'] <= up_limit)]
```

[13]:
```
plt.hist(filtered_df['specz'], bins = 90)
plt.grid()
plt.show()
```




Use the relation between redshift and velocity to add a column named velocity in the data. This would tell the expansion velocity at that redshift

[14]:
```
averaged_df['velocity']=averaged_df['specz']*c
averaged_df
```

[14]:
```
                   objid     specz         ra        dec   proj_sep  \
0    1237671768542478711  0.082457  257.82458  64.133257  8.347733
1    1237671768542478713  0.081218  257.83332  64.126043  8.011259
2    1237671768542544090  0.079564  257.85137  64.173247  8.739276
3    1237671768542544107  0.080842  257.89303  64.141138  6.839642
4    1237671768542544127  0.084575  257.91585  64.107290  5.666108
..                   ...       ...        ...        ...       ...
87   1237671939804627462  0.082060  258.45078  64.020363  9.483937
88   1237671939804627464  0.071804  258.44994  64.025909  9.316140
89   1237671939804627483  0.080790  258.43205  64.123685  8.146154
90   1237671939804627518  0.082876  258.46676  64.119499  8.986029
91   1237671939804627535  0.080897  258.48871  64.111343  9.483374

         velocity
0    2.471993e+07
1    2.434867e+07
```

```
2    2.385281e+07
3    2.423576e+07
4    2.535507e+07
..            ...
87   2.460108e+07
88   2.152644e+07
89   2.422024e+07
90   2.484570e+07
91   2.425243e+07

[92 rows x 6 columns]
```

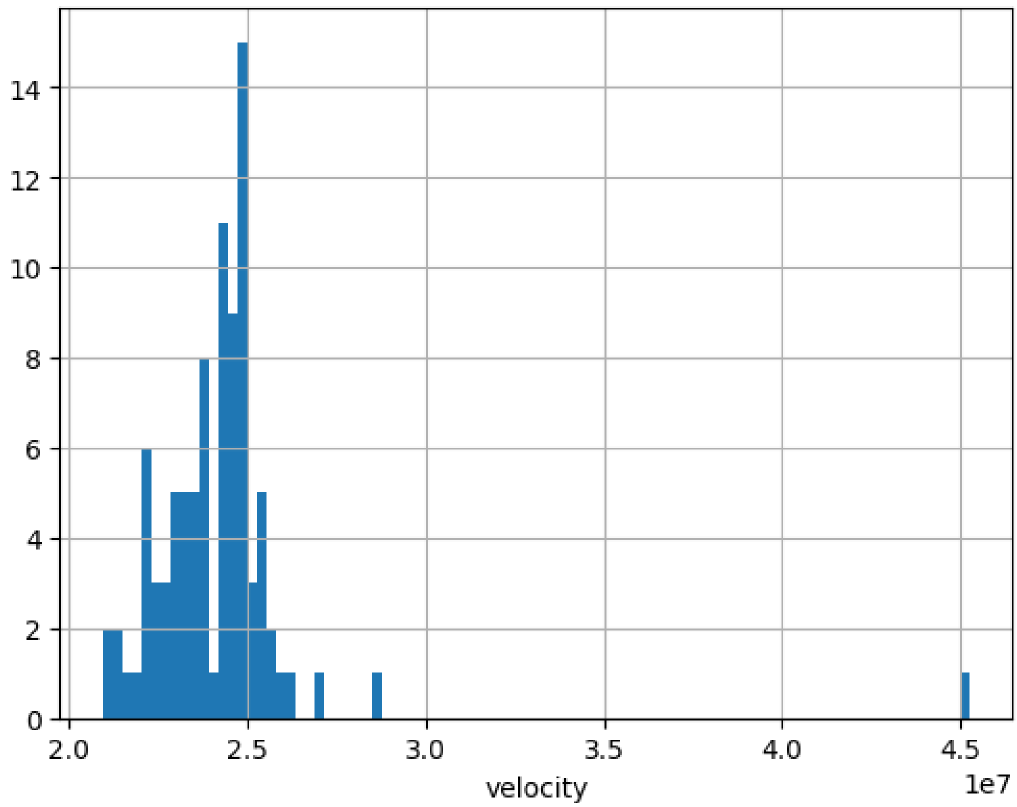
[15]:

```python
print(averaged_df.head())
```

```
                objid     specz         ra        dec   proj_sep      velocity
0  1237671768542478711  0.082457  257.82458  64.133257  8.347733  2.471993e+07
1  1237671768542478713  0.081218  257.83332  64.126043  8.011259  2.434867e+07
2  1237671768542544090  0.079564  257.85137  64.173247  8.739276  2.385281e+07
3  1237671768542544107  0.080842  257.89303  64.141138  6.839642  2.423576e+07
4  1237671768542544127  0.084575  257.91585  64.107290  5.666108  2.535507e+07
```

[16]:

```python
#plot the velocity column created as hist
plt.hist(averaged_df['velocity'], bins=90)
plt.xlabel('velocity')
plt.grid()
plt.show()
```




use the dispersion equation to find something called velocity dispersion. You can even refer to wikipedia to know about the term wiki link here

It is the velocity dispersion value which tells us, some galaxies might be part of even larger groups!!

[17]:
```
mean_velocity = averaged_df['velocity'].mean()
```

[18]:
```
velocity_dispersion=np.sqrt(np.sum((averaged_df['velocity']-mean_velocity)**2)/
 ↪len(df_cluster_members))
```

[19]:
```
velocity_dispersion
```

[19]:
```
np.float64(2571503.959743073)
```

### 0.0.3 Step 2: Calculate Mean Redshift of the Cluster

We calculate the average redshift (`specz`) of galaxies that belong to a cluster. This gives us an estimate of the cluster's systemic redshift.

`cluster_redshift = filtered_df['specz'].mean()`

[20]:
```
cluster_redshift = filtered_df['specz'].mean()
```

The velocity dispersion ( v ) of galaxies relative to the cluster mean redshift is computed using the relativistic Doppler formula:

$$v = c \cdot \frac{(1+z)^2 - (1+z_{\text{cluster}})^2}{(1+z)^2 + (1+z_{\text{cluster}})^2}$$

where: - ( v ) is the relative velocity (dispersion), - ( z ) is the redshift of the individual galaxy, - ( $z_{\text{cluster}}$ ) is the mean cluster redshift, - ( c ) is the speed of light.

[21]:
```
z_cluster= df_cluster_members['specz'].mean()
z=df_cluster_members['specz']
a = (1+z)**2-(1+z_cluster)**2
b = (1+z)**2+(1+z_cluster)**2
v_relv=c*(a/b)
v_relv
```

[21]:
```
0     6.623653e+05
1     3.191853e+05
2    -1.397790e+05
3     2.147463e+05
4     1.248541e+06
          …
87    5.525494e+05
88   -2.302424e+06
89    2.003810e+05
90    7.785333e+05
91    2.301663e+05
Name: specz, Length: 91, dtype: float64
```

[24]:
```
v_relv=c.to('km/s').value*(a/b)
v_relv
```

[24]:
```
0      662.365302
1      319.185348
2     -139.779039
3      214.746305
4     1248.541035
         …
87     552.549373
88   -2302.424337
89     200.381006
90     778.533295
91     230.166253
Name: specz, Length: 91, dtype: float64
```

Pro tip: Check what the describe function of pandas does. Does it help to get quick look stats for your column of dispersion??

[25]:
```
v_relv.describe()
```

[25]:
```
count      91.000000
mean       -2.449331
std      1218.492945
min     -2814.230840
25%      -806.606785
50%       237.179091
75%       754.977576
max      4206.136789
Name: specz, dtype: float64
```

[26]:
```
mean_vrelv = v_relv.mean()
N = len(v_relv)
```

[27]:
```
disp=float(np.sqrt(np.sum((v_relv-mean_vrelv)**2)/N))
disp
```

[27]:
```
1211.7794338417104
```

[28]:
```
print(f"The value of the cluster redshift = {z_cluster:.4}")
print(f"The characteristic value of velocity dispersion of the cluster along␣
↪the line of sight = {disp:.4} km/s.")
```

```
The value of the cluster redshift = 0.08007
The characteristic value of velocity dispersion of the cluster along the line of
sight = 1.212e+03 km/s.
```

[29]:
```
ra_center = df_cluster_members['ra'].mean()
dec_center = df_cluster_members['dec'].mean()
```

[30]:
```
from astropy.coordinates import SkyCoord
```

[31]:
```
coords = SkyCoord(ra=df_cluster_members['ra'].values*u.deg,
                  dec=df_cluster_members['dec'].values*u.deg)
center = SkyCoord(ra=ra_center*u.deg,dec=dec_center*u.deg)

center
```

[31]:
```
<SkyCoord (ICRS): (ra, dec) in deg
    (258.16071527, 64.07383142)>
```

[32]:
```
separations = coords.separation(center).arcminute
```

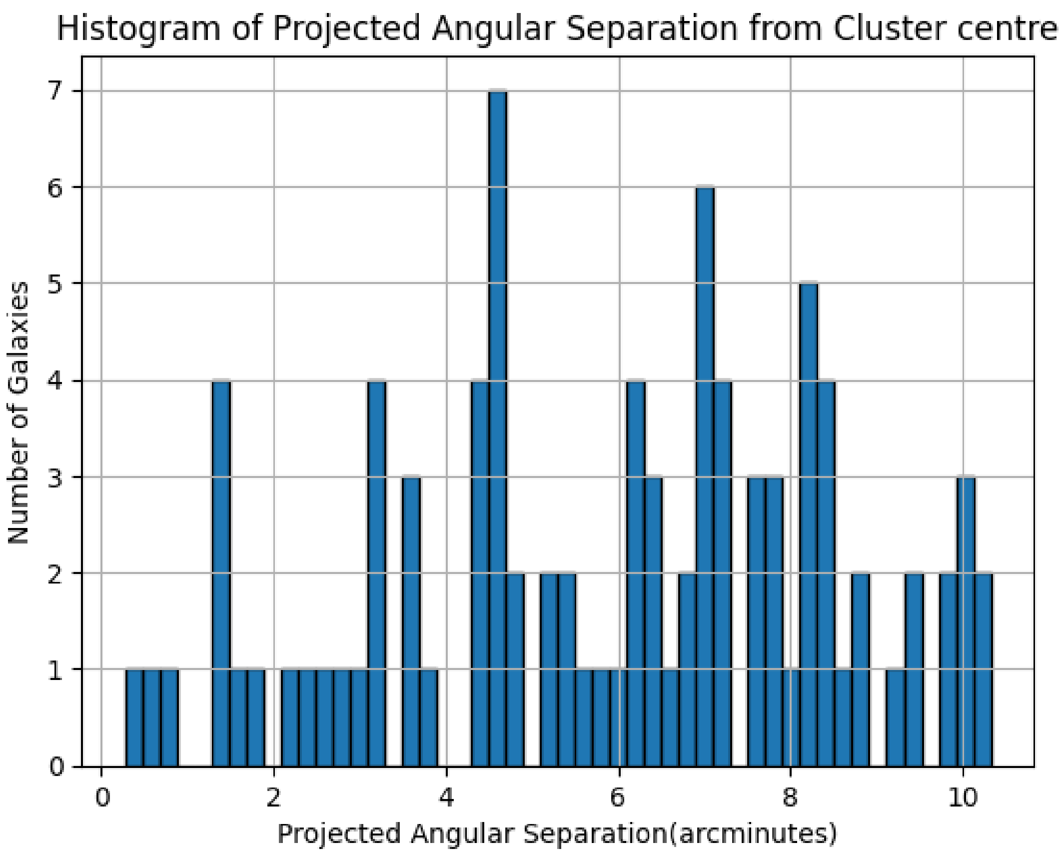
### 0.0.4 Step 4: Visualizing Angular Separation of Galaxies

We plot a histogram of the projected (angular) separation of galaxies from the cluster center. This helps us understand the spatial distribution of galaxies within the cluster field.

- The x-axis represents the angular separation (in arcminutes or degrees, depending on units).
- The y-axis shows the number of galaxies at each separation bin.

[33]:
```
plt.hist(separations, bins = 50, edgecolor = 'black')
plt.xlabel('Projected Angular Separation(arcminutes)')
plt.ylabel('Number of Galaxies')
plt.title('Histogram of Projected Angular Separation from Cluster centre')
plt.grid()
plt.show
```

[33]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```




#### 0.0.5 Determining size and mass of the cluster:

#### 0.0.6 Step 5: Estimating Physical Diameter of the Cluster

We now estimate the **physical diameter** of the galaxy cluster using cosmological parameters.

- `r` is the **co-moving distance**, approximated using a Taylor expansion for low redshift:

$$r = \frac{cz}{H_0}\left(1 - \frac{z}{2}(1 + q_0)\right)$$

where $q_0$ is the deceleration parameter

- `ra` is the **angular diameter distance**, given by:

$$D_A = \frac{r}{1+z}$$

- Finally, we convert the observed angular diameter (in arcminutes) into physical size using:

$$\text{diameter (in Mpc)} = D_A \cdot \theta$$

where \$ \$ is the angular size in radians, converted from arcminutes.

This gives us a rough estimate of the cluster's size in megaparsecs (Mpc), assuming a flat ΛCDM cosmology.

[34]:
```
c = (c.to('km/s').value)
```

[35]:
```
z_=z_cluster
q0 = -0.534
H_0 = 70
r =float((c*z_/H_0)*(1-(z_/2)*(1+q0)))
r
```

[35]: 336.5133133467062

[36]:
```
D_A=float(r/(1+z_))
D_A
```

[36]: 311.5668238308137

[37]:
```
theta_arcmin = float(separations.max())
theta_arcmin
```

[37]: 10.319471061123961

[38]:
```
theta_rad = np.deg2rad(theta_arcmin/60)
diameter = float(D_A*theta_rad)
diameter
```

[38]: 0.9352651712023171

[39]:
```
R = float(diameter/2)
R
```

[39]: 0.46763258560115856

[40]:
```
r=336.5133133467062
ra=311.5668238308137
diameter=0.9352651712023171
diameter #in Mpc
```

[40]: 0.9352651712023171

### 0.0.7 Step 6: Calculating the Dynamical Mass of the Cluster

We now estimate the **dynamical mass** of the galaxy cluster using the virial theorem:

$$M_{\text{dyn}} = \frac{3\sigma^2 R}{G}$$

Where: - \$  \$ is the **velocity dispersion** in m/s (`disp * 1000`), - \$ R \$ is the **cluster radius** in meters (half the physical diameter converted to meters), - \$ G \$ is the **gravitational constant** in SI units, - The factor of 3 assumes an isotropic velocity distribution (common in virial estimates).

We convert the final result into **solar masses** by dividing by \$ 2 ×10^{30} , kg \$.

> This mass estimate assumes the cluster is in dynamical equilibrium and bound by gravity.

[41]:
```
G = 4.302e-9
```

[44]:
```
### Calculating the dynamical mass in solar masses:
M_dyn = (3*(disp**2)*R)/G
print(f"Dynamical mass: {M_dyn:.2e} M ")
```

```
Dynamical mass: 4.79e+14 M
```

[45]:
```
num_galaxies = 91

#Assumed average stellar mass per galaxy {in solar mass}
avg_stellar_mass = 1e11       #M

#Total luminous mass
M_lum = num_galaxies * avg_stellar_mass
print(f"Luminous Mass: {M_lum:.2e}M ")
```

```
Luminous Mass: 9.10e+12M
```

[46]:
```
mass_ratio = M_dyn / M_lum
print(f"Mass Ratio (M_dyn / M_lum): {mass_ratio:.2f}")
```

```
Mass Ratio (M_dyn / M_lum): 52.62
```

Only about 1.9% of the cluster's mass is luminous, the rest, approximately 98.1% can be dark matter and hot intracluster gas, a signature of the hidden universe. The total gravitional pull keeping the cluster together can't be explained by stuff that we can see. Even more intriguing, nearly all of cluster's mass is invisible, making an important evidence about dark matter.