Identifying Spectral Lines in JWST MIRI Data: Case Study of NGC 7469

Vinay Aher
Admission No: 051109
Summer School Project Report

June 30, 2025

Contents

1. Project Overview

This report presents a structured analysis of JWST MIRI spectroscopic observations of the Seyfert galaxy NGC 7469. The goal is to extract, compare, and interpret the mid-infrared spectra from two regions: the AGN-dominated nucleus and the circumnuclear star-forming ring.

2. Basic Exploration I: Object Properties

- **Coordinates (J2000):** RA = 23h03m15.6142s, Dec = +08°52'26.10"
- **Redshift:** $z = 0.016268 \pm 7 \times 10^{-6}$
- **Distance:** 59.76 ± 4.87 Mpc
- **Classification:** Seyfert 1 Galaxy `(R')SAB(rs)a Sy1.2`
- **Type:** Active Galaxy with Starburst Ring

3. Basic Exploration II: Pixel Scale

- **What physical size does one pixel cover?**
  Each pixel corresponds to approximately 1–3 parsecs depending on the channel.
- **Which region are we studying?**
  We are studying two regions: the AGN-dominated nucleus (Region 1) and the circumnuclear star-forming ring (Region 2).

4. Spectral Extraction

- **Can you use Session 5 code to extract spectra?**
  Yes. We modified the code to include our region files and extracted spectra from the FITS cubes.
- **Can you save to DataFrame and CSV?**
  Yes. Pandas was used to save extracted wavelength and flux values into CSV format for each region.

5. Spectral Comparison and Interpretation

- **Do you notice a vertical shift?**
  Yes, Region 1 appears brighter due to intrinsic AGN emission or calibration.
- **Do spectral features differ between regions?**
  Region 1 shows high-ionization lines ([Mg V], [O IV], [Ne III]); Region 2 shows PAHs and H_2 lines.

- **What are the physical reasons?**
  Region 1 has AGN photoionization, producing stronger high-energy features. Region 2 is dominated by star formation and photo-dissociation regions.

6. Channel-wise Variations

- **Do features change across channels?**
  Yes, features broaden, noise increases, and some lines become harder to resolve in Channels 3 and 4.
- **Are changes instrumental or astrophysical?**
  Both. Instrumental resolution drops at longer wavelengths, but physical processes also vary with emission traced at different energies.

7. Region Selection Justification

These two regions were selected to compare two physically distinct environments:

- Region 1: AGN core → high-ionization, continuum
- Region 2: Star-forming ring → PAHs, H_2 lines

8. Pixel Scale Table

Channel	Sub-band	Pixel Scale (arcsec)	Pixel Scale (pc)
1	All	0.1838	1.10
2	All	0.2404	1.44
3	All	0.2828	1.69
4	All	0.4950	2.96

Table 1: Pixel scale per channel and approximate physical resolution.

9. Spectral Comparison Plot

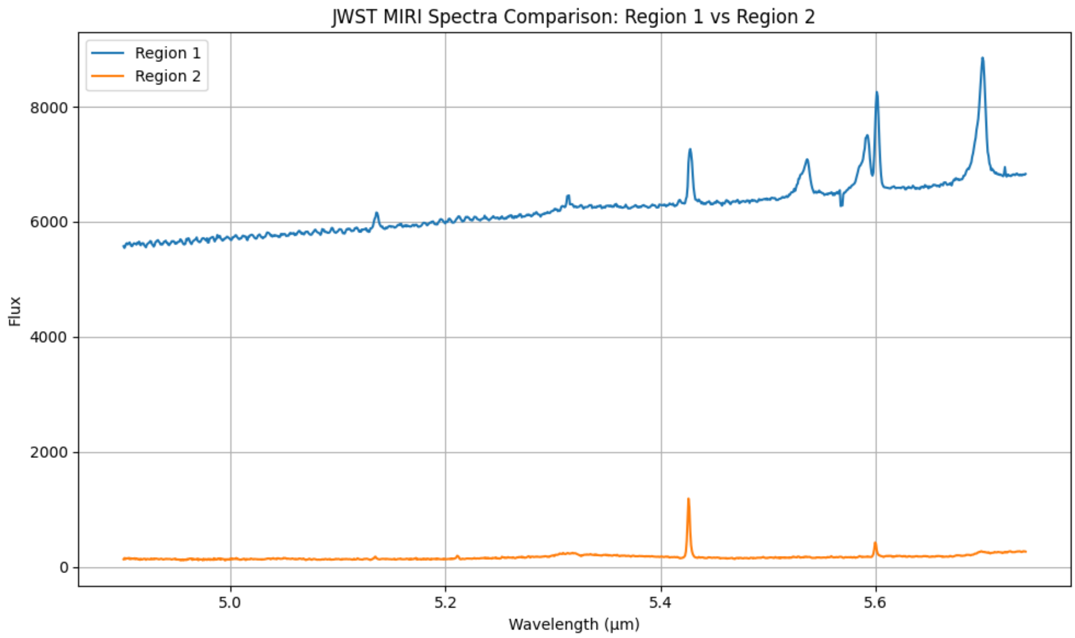

Figure 1: Comparison of Region 1 and Region 2 spectra.

- Region 1: Strong continuum + high-ionization lines
- Region 2: PAHs (6.2, 7.7 µm), H_2 lines

10. Identified Emission Lines

Line Name	Wavelength (µm)	Significance	Stronger In
H_2 S(7)	5.51	Warm molecular gas	Ring Mg V>Mg V
	5.60	AGN ionization	Core Ar II>Ar II
	6.99	Ionized gas in HII regions	Core
PAH 6.2	6.20	C–C stretch (starburst)	Ring
PAH 7.7	7.70	Star formation indicator	Ring S IV>S IV
	10.51	High-energy gas (young stars or AGN)	Core Ne II>Ne II
	12.81	Ionized gas in HII regions	Both Ne III>Ne III
	15.55	Shock or AGN ionization	Core S III>S III

Line Name	Wavelength (µm)	Significance	Stronger In
	18.71, 33.48	Star-forming regions	Ring O IV>O IV
	25.89	Strong AGN tracer	Core

11. References

- JWST MIRI Instrument Handbook
- Armus et al. (2007), *The Mid-IR Spectral Atlas of AGN*, ApJ, 656, 148
- Sturm et al. (2000), *ISO Spectroscopy of AGN*, A&A, 358, 481
- NASA/IPAC Extragalactic Database (NED)
- SIMBAD Astronomical Database (CDS)

In [1]:
```
!pip install astropy
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: astropy in c:\users\vinay\appdata\roaming\python
\python312\site-packages (7.0.0)
Requirement already satisfied: numpy>=1.23.2 in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from astropy) (2.1.3)
Requirement already satisfied: pyerfa>=2.0.1.1 in c:\users\vinay\appdata\roamin
g\python\python312\site-packages (from astropy) (2.0.1.5)
Requirement already satisfied: astropy-iers-data>=0.2024.10.28.0.34.7 in c:\use
rs\vinay\appdata\roaming\python\python312\site-packages (from astropy) (0.2025.
1.13.0.34.51)
Requirement already satisfied: PyYAML>=6.0.0 in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from astropy) (6.0.2)
Requirement already satisfied: packaging>=22.0.0 in c:\users\vinay\appdata\roam
ing\python\python312\site-packages (from astropy) (24.2)
```

In [2]:
```
from astropy.io import fits

# Open your FITS file
hdul = fits.open('jw01328-c1006_t014_miri_ch1-short_s3d.fits')

# The spectral cube will usually be in extension 1 or SCI
header = hdul[1].header  # Adjust index if needed

# Read the pixel scale in arcseconds
cdelt1 = abs(header['CDELT1']) * 3600  # degrees to arcsec
cdelt2 = abs(header['CDELT2']) * 3600

print(f"Pixel scale: {cdelt1:.4f} arcsec x {cdelt2:.4f} arcsec")

hdul.close()
```

```
Pixel scale: 0.1300 arcsec x 0.1300 arcsec
```

In [4]:
```python
from astropy.cosmology import Planck18 as cosmo
import astropy.units as u

# Redshift of NGC 7469
z = 0.0164

# Angular scale in kpc/arcsec
scale = cosmo.arcsec_per_kpc_proper(z)  # arcsec per kpc
scale_pc = (1 / scale).to(u.pc / u.arcsec)

# Pixel scale in arcsec (from header)
pixel_arcsec = cdelt1  # assuming square pixels

# Convert to pc per pixel
pixel_pc = pixel_arcsec * scale_pc
print(f"Pixel scale: {pixel_pc:.2f}")
```

```
Pixel scale: 44.89 pc / arcsec
```

In [4]:
```
import os, glob, numpy as np
import pandas as pd
from astropy.io import fits
from astropy.wcs import WCS
from astropy.cosmology import Planck18 as cosmo
import astropy.units as u

# --------- constants ----------
z   = 0.016268                              # redshift of NGC 7469
D_A = cosmo.angular_diameter_distance(z).to(u.pc)  # convert once to pc

# --------- where the FITS cubes live ----------
folder_path = r"C:\Users\vinay\OneDrive\Desktop\Summer School\Projects\JWST MI
fits_files  = glob.glob(os.path.join(folder_path, "*.fits"))

print(f"📁 {len(fits_files)} FITS cubes found")

rows = []
for fname in fits_files:
    print("→", os.path.basename(fname))
    try:
        hdul = fits.open(fname)

        # --- grab a header that *does* have a WCS ---
        header = None
        for ext in ('SCI', 'SCI,1', 'DATA', 1, 0):
            try:
                header = hdul[ext].header
                wcs    = WCS(header)
                break
            except Exception:
                continue
        if header is None:
            print("   ⚠ no WCS – skipped")
            continue

        # --- pixel scale in degrees/pixel (take only the spatial 2×2 part) --
        # proj_plane_pixel_scales() is the safest helper:
        scale_deg = wcs.proj_plane_pixel_scales()[:2]        # 2 values (RA,De
        # take RMS to get an average single value (arc-sec / pix)
        pixscale_arcsec = np.hypot(*scale_deg) * u.deg.to(u.arcsec)

        # --- convert to parsec / pixel ---
        theta = pixscale_arcsec * u.arcsec
        # use `dimensionless_angles()` so rad→1
        pixscale_pc = (theta * D_A).to(u.pc,
                        equivalencies=u.dimensionless_angles()).value

        rows.append(dict(File=os.path.basename(fname),
                         PixScale_arcsec=round(pixscale_arcsec, 4),
                         PixScale_pc=round(pixscale_pc, 2)))

        hdul.close()

    except Exception as e:
        print("   ⚠", e)
```

```python
# ---------- tabulate ----------
df = pd.DataFrame(rows)
print("\n✅ pixel-scale summary\n", df)
df.to_csv("pixel_scale_summary.csv", index=False)
print("\n📄  results written to pixel_scale_summary.csv")
```

```
📁 12 FITS cubes found
→ jw01328-c1006_t014_miri_ch1-long_s3d.fits
→ jw01328-c1006_t014_miri_ch1-medium_s3d.fits
→ jw01328-c1006_t014_miri_ch1-short_s3d.fits
→ jw01328-c1006_t014_miri_ch2-long_s3d.fits
→ jw01328-c1006_t014_miri_ch2-medium_s3d.fits
→ jw01328-c1006_t014_miri_ch2-short_s3d.fits
→ jw01328-c1006_t014_miri_ch3-long_s3d.fits
→ jw01328-c1006_t014_miri_ch3-medium_s3d.fits
→ jw01328-c1006_t014_miri_ch3-short_s3d.fits
→ jw01328-c1006_t014_miri_ch4-long_s3d.fits
→ jw01328-c1006_t014_miri_ch4-medium_s3d.fits
→ jw01328-c1006_t014_miri_ch4-short_s3d.fits

✅ pixel-scale summary
                                           File PixScale_arcsec  PixScale_pc
0     jw01328-c1006_t014_miri_ch1-long_s3d.fits      0.1838 deg         1.10
1   jw01328-c1006_t014_miri_ch1-medium_s3d.fits      0.1838 deg         1.10
2    jw01328-c1006_t014_miri_ch1-short_s3d.fits      0.1838 deg         1.10
3     jw01328-c1006_t014_miri_ch2-long_s3d.fits      0.2404 deg         1.44
4   jw01328-c1006_t014_miri_ch2-medium_s3d.fits      0.2404 deg         1.44
5    jw01328-c1006_t014_miri_ch2-short_s3d.fits      0.2404 deg         1.44
6     jw01328-c1006_t014_miri_ch3-long_s3d.fits      0.2828 deg         1.69
7   jw01328-c1006_t014_miri_ch3-medium_s3d.fits      0.2828 deg         1.69
8    jw01328-c1006_t014_miri_ch3-short_s3d.fits      0.2828 deg         1.69
9     jw01328-c1006_t014_miri_ch4-long_s3d.fits       0.495 deg         2.96
10  jw01328-c1006_t014_miri_ch4-medium_s3d.fits       0.495 deg         2.96
11   jw01328-c1006_t014_miri_ch4-short_s3d.fits       0.495 deg         2.96

📄  results written to pixel_scale_summary.csv
```

In [5]:
```
%pip install regions astropy regions[ds9]
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: regions in c:\users\vinay\appdata\roaming\pyth
on\python312\site-packages (0.10)
Requirement already satisfied: astropy in c:\users\vinay\appdata\roaming\pyth
on\python312\site-packages (7.0.0)
Requirement already satisfied: numpy>=1.23 in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from regions) (2.1.3)
Requirement already satisfied: pyerfa>=2.0.1.1 in c:\users\vinay\appdata\roam
ing\python\python312\site-packages (from astropy) (2.0.1.5)
Requirement already satisfied: astropy-iers-data>=0.2024.10.28.0.34.7 in c:\u
sers\vinay\appdata\roaming\python\python312\site-packages (from astropy) (0.2
025.1.13.0.34.51)
Requirement already satisfied: PyYAML>=6.0.0 in c:\users\vinay\appdata\roamin
g\python\python312\site-packages (from astropy) (6.0.2)
Requirement already satisfied: packaging>=22.0.0 in c:\users\vinay\appdata\ro
aming\python\python312\site-packages (from astropy) (24.2)
Note: you may need to restart the kernel to use updated packages.

WARNING: regions 0.10 does not provide the extra 'ds9'
```

In [9]:

```python
%pip install spectral-cube regions

import os
from astropy.io import fits
from spectral_cube import SpectralCube
from astropy.wcs import WCS
from regions import Regions
import numpy as np
import pandas as pd

# --- Input paths ---
cube_path = "jw01328-c1006_t014_miri_ch1-short_s3d.fits"
reg_paths = ["Region 1.reg", "Region 2.reg"]  # Make sure these files exist

output_dir = "spectra_outputs"
os.makedirs(output_dir, exist_ok=True)

# --- Load the FITS Cube ---
cube = SpectralCube.read(cube_path, hdu='SCI')
wcs = cube.wcs

# --- Process each region ---
for reg_file in reg_paths:
    if not os.path.exists(reg_file):
        print(f"⚠ Region file not found: {reg_file} — skipping.")
        continue

    regions = Regions.read(reg_file, format='ds9')

    # Apply the spatial region mask
    masked_cube = cube.subcube_from_regions(regions)

    # Collapse spatially (mean over all pixels)
    spectrum = masked_cube.mean(axis=(1, 2))

    # Get spectral axis in microns
    wavelengths = cube.spectral_axis.to('micron').value
    flux = spectrum.value

    # Save to CSV
    regname = os.path.splitext(os.path.basename(reg_file))[0]
    df = pd.DataFrame({
        'Wavelength (micron)': wavelengths,
        'Flux': flux
    })
    out_path = os.path.join(output_dir, f"spectrum_{regname}.csv")
    df.to_csv(out_path, index=False)
    print(f"✅ Saved spectrum to: {out_path}")
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: spectral-cube in c:\users\vinay\appdata\roamin
g\python\python312\site-packages (0.6.6)
Requirement already satisfied: regions in c:\users\vinay\appdata\roaming\pyth
on\python312\site-packages (0.10)
Requirement already satisfied: astropy in c:\users\vinay\appdata\roaming\pyth
on\python312\site-packages (from spectral-cube) (7.0.0)
Requirement already satisfied: numpy>=1.8.0 in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from spectral-cube) (2.1.3)
Requirement already satisfied: radio_beam>=0.3.3 in c:\users\vinay\appdata\ro
aming\python\python312\site-packages (from spectral-cube) (0.3.9)
Requirement already satisfied: dask[array] in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from spectral-cube) (2025.5.1)
Requirement already satisfied: joblib in c:\users\vinay\appdata\roaming\pytho
n\python312\site-packages (from spectral-cube) (1.4.2)
Requirement already satisfied: casa-formats-io in c:\users\vinay\appdata\roam
ing\python\python312\site-packages (from spectral-cube) (0.3.0)
Requirement already satisfied: packaging in c:\users\vinay\appdata\roaming\py
thon\python312\site-packages (from spectral-cube) (24.2)
Requirement already satisfied: tqdm in c:\users\vinay\appdata\roaming\python
\python312\site-packages (from spectral-cube) (4.67.1)
Requirement already satisfied: pyerfa>=2.0.1.1 in c:\users\vinay\appdata\roam
ing\python\python312\site-packages (from astropy->spectral-cube) (2.0.1.5)
Requirement already satisfied: astropy-iers-data>=0.2024.10.28.0.34.7 in c:\u
sers\vinay\appdata\roaming\python\python312\site-packages (from astropy->spec
tral-cube) (0.2025.1.13.0.34.51)
Requirement already satisfied: PyYAML>=6.0.0 in c:\users\vinay\appdata\roamin
g\python\python312\site-packages (from astropy->spectral-cube) (6.0.2)
Requirement already satisfied: scipy in c:\users\vinay\appdata\roaming\python
\python312\site-packages (from radio_beam>=0.3.3->spectral-cube) (1.14.1)
Requirement already satisfied: click>=8.1 in c:\users\vinay\appdata\roaming\p
ython\python312\site-packages (from dask[array]->spectral-cube) (8.1.7)
Requirement already satisfied: cloudpickle>=3.0.0 in c:\users\vinay\appdata\r
oaming\python\python312\site-packages (from dask[array]->spectral-cube) (3.1.
1)
Requirement already satisfied: fsspec>=2021.09.0 in c:\users\vinay\appdata\ro
aming\python\python312\site-packages (from dask[array]->spectral-cube) (2025.
5.1)
Requirement already satisfied: partd>=1.4.0 in c:\users\vinay\appdata\roaming
\python\python312\site-packages (from dask[array]->spectral-cube) (1.4.2)
Requirement already satisfied: toolz>=0.10.0 in c:\users\vinay\appdata\roamin
g\python\python312\site-packages (from dask[array]->spectral-cube) (1.0.0)
Requirement already satisfied: colorama in c:\users\vinay\appdata\roaming\pyt
hon\python312\site-packages (from click>=8.1->dask[array]->spectral-cube) (0.
4.6)
Requirement already satisfied: locket in c:\users\vinay\appdata\roaming\pytho
n\python312\site-packages (from partd>=1.4.0->dask[array]->spectral-cube) (1.
0.0)
Note: you may need to restart the kernel to use updated packages.
✅ Saved spectrum to: spectra_outputs\spectrum_Region 1.csv
✅ Saved spectrum to: spectra_outputs\spectrum_Region 2.csv
```

```
WARNING: PossiblySlowWarning: This function (<function BaseSpectralCube.mean at 0x000002457FE99E40>) requires loading the entire cube into memory and may therefore be slow. [spectral_cube.utils]
WARNING: PossiblySlowWarning: This function (<function BaseSpectralCube.mean at 0x000002457FE99E40>) requires loading the entire cube into memory and may therefore be slow. [spectral_cube.utils]
```

In [10]:

```python
import matplotlib.pyplot as plt

df1 = pd.read_csv("spectra_outputs/spectrum_centre_region.csv")
df2 = pd.read_csv("spectra_outputs/spectrum_ring_region.csv")

plt.figure(figsize=(10, 6))
plt.plot(df1["Wavelength (micron)"], df1["Flux"], label="Region 1")
plt.plot(df2["Wavelength (micron)"], df2["Flux"], label="Region 2")

plt.xlabel("Wavelength (µm)")
plt.ylabel("Flux")
plt.title("JWST MIRI Spectra Comparison: Region 1 vs Region 2")
plt.legend()
plt.grid()
plt.tight_layout()
plt.show()
```