import numpy as np

import matplotlib.pyplot as plt

from scipy.optimize import curve\_fit

# MBT velocity law

def mbt\_velocity(r, a, b):

return a \* (1 - np.exp(-b \* r))

# NGC 925 — Approximate rotation curve (kpc, km/s)

r = np.array([1, 2.5, 4, 5.5, 7, 8.5, 10, 11.5])

v\_obs = np.array([35, 65, 85, 95, 102, 106, 108, 109]) # Based on Pisano+ 2000 and Hague & Wilkinson 2014

# Fit MBT model

popt, \_ = curve\_fit(mbt\_velocity, r, v\_obs, p0=[110, 0.4])

v\_fit = mbt\_velocity(r, \*popt)

residuals = v\_obs - v\_fit

rmse = np.sqrt(np.mean(residuals\*\*2))

r2 = 1 - np.var(residuals) / np.var(v\_obs)

# Plot

plt.figure(figsize=(8, 6))

plt.plot(r, v\_obs, 'o', label='Observed', color='black')

plt.plot(r, v\_fit, '-', label=f'MBT Fit: a={popt[0]:.1f}, b={popt[1]:.2f}', color='royalblue')

plt.xlabel("Radius (kpc)")

plt.ylabel("Rotation Speed (km/s)")

plt.title("MBT Fit — NGC 925")

plt.legend()

plt.grid(True)

plt.tight\_layout()

plt.show()

# Print stats

print(f"Best-fit a: {popt[0]:.2f} km/s")

print(f"Best-fit b: {popt[1]:.3f} kpc⁻¹")

print(f"RMSE: {rmse:.2f} km/s")

print(f"R²: {r2:.3f}")