



easyspec: An Open-source Python Package for Long-slit Spectroscopy

Raniere de Menezes¹ , Francesco Massaro^{2,3,4,5} , Michela Negro⁶ , Claudia M. Raiteri⁴ , Harold Peña-Herazo⁷ , and Jose A. Acosta-Pulido^{8,9}

¹ Centro Brasileiro de Pesquisas Físicas, 22290-180 Rio de Janeiro, RJ, Brazil; ranieri@cbpf.br

² Dipartimento di Fisica, Università degli Studi di Torino, via Pietro Giuria 1, I-10125 Turin, Italy

³ Istituto Nazionale di Fisica Nucleare, Sezione di Torino, via Pietro Giuria 1, I-10125 Turin, Italy

⁴ INAF—Osservatorio Astrofisico di Torino, via Osservatorio 20, 10025 Pino Torinese, Italy

⁵ Consorzio Interuniversitario per la Fisica Spaziale (CIFS), via Pietro Giuria 1, 10125 Turin, Italy

⁶ Department of Physics & Astronomy, Louisiana State University, Baton Rouge, LA 70803, USA

⁷ East Asian Observatory, 660 N. A’ohoku Place, Hilo, HI 96720, USA

⁸ Instituto de Astrofísica de Canarias, Calle Vía Láctea, s/n, 38205 La Laguna, Tenerife, Spain

⁹ Departamento de Astrofísica, Universidad de La Laguna, 38206 La Laguna, Tenerife, Spain

Received 2025 April 18; revised 2025 July 4; accepted 2025 July 17; published 2025 August 18

Abstract

In modern-day astronomy, near-infrared, optical, and ultraviolet spectroscopy are indispensable for studying a wide range of phenomena, from measuring black hole masses to analyzing chemical abundances in stellar atmospheres. However, spectroscopic data reduction is often performed using instrument-specific pipelines or legacy software well-established and robust within the community that are often challenging to implement and script in modern astrophysical workflows. In this work, we introduce easyspec, a new Python package designed for long-slit spectroscopy, capable of reducing, extracting, and analyzing spectra from a wide range of instruments—provided they deliver raw FITS files, the standard format for most optical telescopes worldwide. This package is built upon the well-established long-slit spectroscopy routines of the Image Reduction and Analysis Facility (IRAF), integrating modern coding techniques and advanced fitting algorithms based on Markov chain Monte Carlo simulations. We present a user-friendly open-source Python package that can be easily incorporated into customized pipelines for more complex analyses. To validate its capabilities, we apply easyspec to the active galactic nucleus G4Jy 1709, observed with the DOLORES spectrograph at the Telescopio Nazionale Galileo, measuring its redshift and estimating its supermassive black hole mass. Finally, we compare our results with a previous IRAF-based study.

Unified Astronomy Thesaurus concepts: Spectroscopy (1558); Astronomy data analysis (1858); Astronomy image processing (2306); Active galactic nuclei (16)

1. Introduction

Long-slit spectroscopy is one of the most powerful diagnostic tools in astronomy. In this technique, light from an astrophysical object—such as a galaxy, nebula, or star—passes through a narrow, elongated slit before being dispersed by a diffraction grating or prism, breaking it into its component wavelengths. A charge-coupled device (CCD) or a complementary metal–oxide–semiconductor detector then records the dispersed light, producing a two-dimensional spectrum where one axis represents wavelength and the other, spatial position along the slit.

Raw data from astronomical cameras are often contaminated by various noise sources, including sensor bias, thermal noise, detector sensitivity variations, and cosmic-ray hits. To ensure accurate spectral analysis, these artifacts must be carefully removed before extracting and calibrating the spectrum in both wavelength and flux. Traditionally, the Image Reduction and Analysis Facility (IRAF; D. Tody 1986; M. D. de La Peña et al. 2001) has been the standard tool for these tasks and

remains widely used in the community (e.g., M. Guerrero et al. 2025; A. J. Schönnell et al. 2025; M. Turchetta et al. 2025).¹⁰

Other legacy software, such as the STARLINK library¹¹ (M. J. Currie et al. 2014; D. Berry et al. 2022), provides valuable tools for astronomical data reduction. However, with the rise of Python as a leading high-level language for astronomical analysis, some legacy routines have been integrated into Astropy (Astropy Collaboration et al. 2013, 2018). Additionally, new software for astronomical data reduction has been actively developed (e.g., J. X. Prochaska et al. 2020; M. C. Lam et al. 2023; T. E. Pickering et al. 2024), offering modern alternatives to traditional tools.

In this work, we introduce easyspec, a user-friendly, open-source Python package for data reduction, spectral extraction, and line fitting in long-slit spectroscopy of astrophysical objects. In the context of modern spectroscopy tools, easyspec distinguishes itself by integrating both data reduction and spectral analysis within a single package. It places particular emphasis on robust line-fitting capabilities and a highly visual, interactive workflow. Diagnostic plots are provided at nearly every stage of the reduction and analysis processes, offering users intuitive insights and enhancing transparency throughout



Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

¹⁰ Support for IRAF was discontinued in 2013 by the NOAO, and it is currently maintained by community effort <https://iraf-community.github.io>.

¹¹ Officially discontinued in 2005 and since then being maintained by the East Asian Observatory: <https://starlink.eao.hawaii.edu/starlink>.

their analysis. To showcase its capabilities, we apply easyspec to the raw spectroscopic data of the active galactic nucleus (AGN) G4Jy 1709 (previously studied by J. Holt et al. 2008; F. Massaro et al. 2023), also known as PKS 2135-20, observed in 2024 September with the 3.58 m Telescopio Nazionale Galileo (TNG) in the Canary Islands, Spain. We estimate the mass of its supermassive black hole and measure its redshift. By streamlining the data analysis process, easyspec enhances user experience and significantly reduces the time required for near-infrared, optical, and near-ultraviolet astronomical spectroscopy.

The tutorials for easyspec and detailed documentation are available on GitHub¹² and the official documentation webpage.¹³ Although here we focus on spectroscopic data observed with TNG, we have successfully tested easyspec in data from other telescopes, such as the 4.1 m Southern Astrophysical Research Telescope (SOAR), at Cerro Pachón, Chile, and the 1.6 m Perkin-Elmer telescope at the Observatório do Pico dos Dias (OPD), Brazil. This paper is structured as follows: Section 2 outlines the software’s dependencies and provides an overview of its functionality. Section 3 describes its application to G4Jy 1709, covering data reduction, spectral extraction, and line fitting, along with a detailed explanation of the adopted methods. Section 4 presents our results on the estimation of the supermassive black hole mass, while Section 5 discusses the validation of easyspec by comparing it with other software and previous studies. Finally, Section 6 summarizes our findings and highlights the potential applications of easyspec.

2. Code Overview and Main Dependencies

The current release of easyspec (v1.0.0.9; R. de Menezes 2025) is available on the Python Package Index¹⁴ and GitHub, along with detailed installation and usage instructions. The package leverages several well-established Python tools for astronomy and data analysis, including Astropy (Astropy Collaboration et al. 2013, 2018) for handling FITS files, ccdproc (M. Craig et al. 2017) for gain correction and cosmic-ray removal (through the L.A. Cosmic algorithm detailed in P. G. Van Dokkum 2001; C. McCully et al. 2018), dust_extinction (K. D. Gordon 2024) for extinction corrections, emcee and corner (D. Foreman-Mackey et al. 2013; D. Foreman-Mackey 2016) for Markov chain Monte Carlo (MCMC) line fitting, and the widely used packages Matplotlib, NumPy, and SciPy (J. D. Hunter 2007; C. R. Harris et al. 2020; P. Virtanen et al. 2020).

The easyspec package is designed to process raw spectroscopic FITS files and is structured into three main modules: cleaning, spectral extraction, and line fitting. In the next section, we apply easyspec to the AGN G4Jy 1709 and provide a detailed explanation of each step in the data reduction and line-fitting process. For a more in-depth analysis, we refer readers to the tutorial notebooks available on GitHub,¹⁵ where we describe step-by-step what each easyspec function is doing and give general advice on how to do long-slit spectroscopy.

3. Application to the Active Galactic Nucleus G4Jy 1709

In 2024 September, we observed the AGN G4Jy 1709, which is classified as a Narrow-line Seyfert 1 (NLS1) object (M.-P. Véron-Cetty & P. Véron 2006), using the low-resolution spectrograph DOLORES at the 3.58 m Telescopio Nazionale Galileo, located at the Roque de los Muchachos Observatory in the Canary Islands, Spain. We performed three 1800 s exposures with a 1.0" slit, no filter, and the LR-R grism (300 lines/mm). For calibration, we acquired 15 bias frames (zero exposure, closed shutter), seven sky flats (0.9 s each), seven lamp¹⁶ exposures (Ar, Ne+Hg, Kr; 0.9 s each), and four exposures (5 s each) for the standard star BD+33d2642, using a slit width of 1.0" for all of them. The major noise sources and instrumental signal offsets affecting these data are:

1. Sensor bias: an electronic offset introduced by the detector’s readout electronics. It is measured by taking a zero-second exposure with the shutter closed and serves as one of the most fundamental calibration steps, as it affects all images. For CCDs, the bias includes a fixed direct current offset in the amplifier, as well as contributions from low-order structure and random noise introduced during the readout process. To correct for this effect, the bias must be subtracted from the data.
2. Dark current: thermal noise that can be mitigated by cooling the detector. This image is obtained in exposures taken with the shutter closed for the same duration as the target observation. To remove this noise, the dark frame must be subtracted from the data. For our observations of G4Jy 1709 this is not an issue.
3. Flat-field variations: pixel-to-pixel sensitivity differences in the light sensor. These variations can be measured by taking a relatively short exposure of a uniformly illuminated white screen inside the telescope dome (or using a sky flat). After subtracting bias and dark frames (if needed), the 2D spectrum must be divided by the flat field to correct for these inhomogeneities. This correction must be applied to any image done with a nonzero exposure and open shutter (i.e., it does not affect the bias and dark frames). It is important to acquire flat-field frames with the highest possible signal-to-noise ratio in order to minimize the propagation of additional noise into the science data during the flat-field correction.
4. Cosmic-ray strikes: longer accumulate more cosmic-ray hits. If several exposures (typically ≥ 5) of the same target are available, cosmic rays can be removed by taking the median of all exposures (avoid using the average since it cannot get rid of the strongest cosmic-ray strikes). If only a few exposures are available, as in our case, specialized algorithms (see Section 3.1) can detect and remove cosmic rays by identifying their sharp edges.

In Figure 1 we show one of the three raw spectra. The x -axis corresponds to the dispersion axis, while the AGN spectrum appears as a faint horizontal line around the y -axis pixel 1000. At this stage, the image includes randomly scattered cosmic-ray hits. To start the data reduction, we import the easyspec `cleaning()` class and load all data with the function `cleaning.data_paths()`. In this function, the user must

¹² <https://github.com/ranieremenezes/easyspec>

¹³ <https://easyspec.readthedocs.io/en/latest/index.html>

¹⁴ <https://pypi.org/project/easyspec/>

¹⁵ <https://github.com/ranieremenezes/easyspec/tree/main/tutorial>

¹⁶ Reference lamp webpage: <https://www.tng.iac.es/instruments/lrs/>

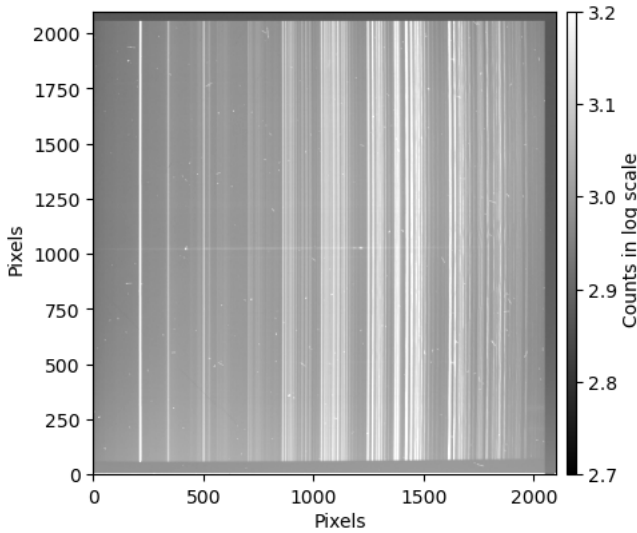


Figure 1. Raw spectrum of G4Jy 1709 for an exposure of 1800 s. The spectrum of our target is the dim horizontal line near the y-axis pixel 1000. In the edges of the image, we can see the dark patches in the light sensor where the light coming from the aperture does not arrive.

specify the paths to the bias, flats, lamps, standard star, and target data, which will then be converted into a dictionary containing the name of each file as the dictionary keywords and the loaded data as the dictionary values. Notably, easyspec never modifies the original files, ensuring they remain intact in their original directories.

3.1. Data Cleaning

The first step in data cleaning is trimming the data. This step is typically used to remove the overscan region or areas of the image that did not capture light from the slit (see the edges of Figure 1). We apply the easyspec function `cleaning.trim()` to crop all data files stored in the dictionary produced by `data_paths()`, selecting the pixel range $x_1 = 30$, $x_2 = 2000$, $y_1 = 400$, $y_2 = 1500$. The output is a new dictionary containing the cropped data, with the same keys as the original. The top panel of Figure 2 displays one of the trimmed spectra.

Next, we compute the average, median, or mode bias using the 15 bias exposures mentioned earlier. This step is easily performed with the easyspec function `cleaning.master("bias")` applied to the subset of bias files in the trimmed data dictionary. In this case, we use the median bias, shown in the middle panel of Figure 2. Since all camera images are affected by bias, we subtract the median bias from all of them at once using the easyspec function `cleaning.debias()`.

Since we are using a refrigerated camera, thermal noise in the CCD is negligible.¹⁷ However, if needed, the user can remove thermal noise using the easyspec function `cleaning.master("dark")`.¹⁸ The next step is flat-field correction, as CCD sensitivity varies across pixels. We compute the median flat-field using the function `cleaning.master("flat")`. However, since CCD sensitivity and grating diffraction efficiency depend on wavelength, the

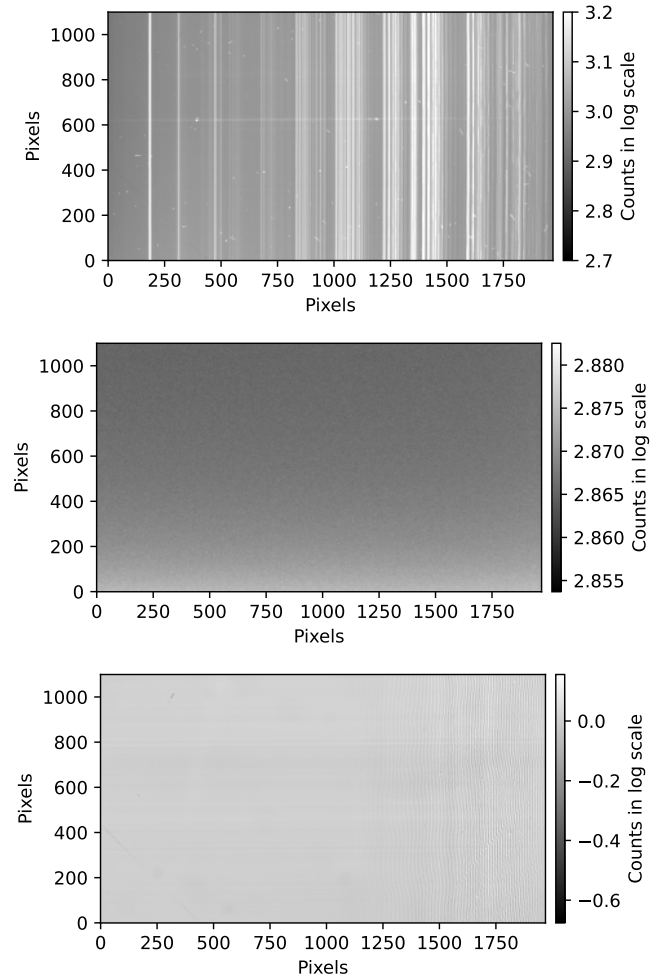


Figure 2. Trimmed raw spectrum (top), median bias (middle), and normalized median flat-field (bottom) for our observations.

median flat-field often exhibits a strong intensity gradient along the dispersion axis.

This is not the correction we seek—we are primarily interested in removing pixel-to-pixel sensitivity variations independent of wavelength. To achieve this, we normalize the median flat by fitting a polynomial to its intensity profile along the dispersion axis (if the flat-field presents wiggles, which are difficult to fit using polynomials, this function allows for the use of a median filter with a customizable smoothing window), using the function `cleaning.norm_master_flat()`. The resulting normalized median flat is shown in the bottom panel of Figure 2. Finally, we apply the normalized flat-field correction to all debiased data files with the function `cleaning.flatten()`.

With the camera’s intrinsic features corrected, the next step is cosmic-ray removal. We use the easyspec function `CR_and_gain_corrections()`, which adopts the LACosmic algorithm, originally developed for IRAF by P. G. Van Dokkum (2001). At this stage, we also correct for CCD gain and read noise, adopting a Laplacian-to-noise limit of 7σ for cosmic-ray detection. Finally, we stack the three cleaned spectra to produce a final 2D spectrum, shown in Figure 3. One of easyspec’s key features is its ability to apply this same process to lamp, standard star, and even other target spectra with minimal additional effort (see the GitHub tutorial for details).

¹⁷ For DOLORES, dark current is negligible even for long exposures: <https://www.tng.iac.es/instruments/lrs/>.

¹⁸ More details here: https://github.com/ranieremenezes/easyspec/blob/main/tutorial/Image_cleaning_easyspec.ipynb.

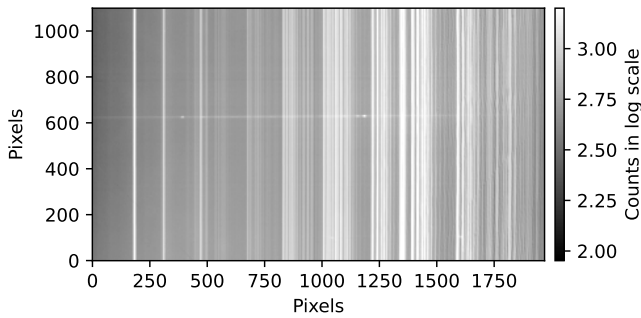


Figure 3. Cosmic-ray corrected and stacked 2D spectrum for G4Jy 1709. We can see the stacked spectrum as the horizontal line just above the y-axis pixel 600. The vertical lines are all sky lines, produced by excited molecules in Earth’s atmosphere.

3.2. Spectral Extraction

With the cleaned 2D spectrum in hand, the next step is to extract its trace (see Figure 4). Although the dispersed target spectrum may appear as a straight horizontal line, this is rarely precisely the case. In most cases, it follows a shallow parabolic curve with minimal curvature, appearing almost straight. We perform this step using the `easyspec` class `extraction()`, specifically the function `extraction.tracing()`, which fits a polynomial to recover the trace of one or multiple spectra in an image. The tracing method can be set to:

1. `argmax` or `moments`: for extracting the strongest spectrum in the image.
2. `multi`: for extracting multiple spectra from the cleaned image.

For our target, we use the `argmax` approach, which selects the pixel with the maximum value in each column and fits a polynomial to these points. This function also allows customization of the polynomial order, minimum average counts per spectrum, and distance between traces (either a constant or a user-defined list). The spectral trace found with `easyspec` is shown in Figure 4 and is automatically assigned the name `spec 0`. If multiple spectra were present in the slit, the `multimethod` would assign them names sequentially from bottom to top: `spec 0`, `spec 1`, `spec 2`, and so on. This method finds all local maxima by direct comparison with neighboring values using the `scipy` function `signal.find_peaks()` applied to take the 1D array of average *x*-axis image values.

Once the trace is identified, we proceed with spectrum extraction using the `extraction.extracting()` function, which applies a Gaussian-weighted model to extract one or more spectra from the image. The initial guess for the Gaussian width is user-defined (with a default value of 3.5 pixels) and is subsequently refined by fitting it to the linearized trace profile. This profile is obtained by averaging the vertical dispersion (i.e., perpendicular to the wavelength axis) over the linearized trace. This function also estimates the systematic uncertainty of the extracted spectrum by performing a Monte Carlo simulation over the uncertainty ranges of the fitted Gaussian parameters. The default number of iterations is 50, though the user can specify any positive integer. The simulation provides an independent error estimate for each spectral bin, which is saved to a text file alongside the calibrated spectrum at the end of the analysis process. After extracting the target spectrum, we use the same trace to extract

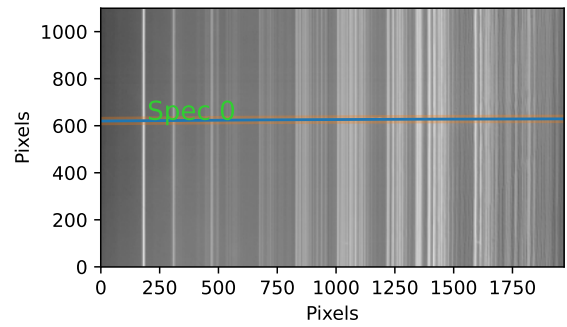


Figure 4. Spectral trace found with `easyspec` (blue line). The orange band around the trace is the region at which the spectrum will be extracted with a Gaussian.

the lamp spectrum from the stacked 2D lamp image. To estimate the sky background, the trace is shifted by ± 30 pixels along the *y*-axis (a user-defined value) to extract two sky spectra—one above and one below the original trace. The average of these sky spectra is then subtracted from the target spectrum. The resulting extracted target and lamp spectra are shown in Figure 5. For those familiar with AGN spectroscopy, the Oxygen and Hydrogen emission lines in the G4Jy 1709 spectrum are immediately recognizable. However, at this stage, the spectrum remains uncalibrated, with the *x*-axis in pixels and the *y*-axis in raw counts. The detailed line identification process will be discussed in Section 3.3.

We now compare our lamp spectrum with the archival lamp spectrum available on the TNG website¹⁹ and use the `easyspec` function `extraction.wavelength_calibration()` to determine the wavelength solution. This function takes an array of selected emission lines from the lamp spectrum (see bottom panel of Figure 5) and fits an *i*th order polynomial (here we choose the third order), mapping wavelengths as a function of dispersion-axis pixels. Out of all the lamp lines identified with `easyspec`, we select those seven that have a corresponding value in the online list. The resulting wavelength solution and fit residuals are presented in Figure 6. The standard deviation of the fit, 1.263 Å, is approximately half the average spectral resolution element, 2.623 Å, indicating a well-constrained calibration. This standard deviation is taken as the systematic error for the wavelength calibration and is saved alongside the calibrated spectrum at the end of the analysis.

With the wavelength solutions established individually for the target and standard star, we use the function `extraction.extinction_correction()` to correct for the wavelength-dependent atmospheric extinction based on their respective air masses, using the extinction curve of Roque de los Muchachos.²⁰ These corrected spectra are displayed in the top and middle panels of Figure 7.

The final step is flux calibration. We derive the flux correction curve using the `std_star_normalization()` function, which compares the measured standard star spectrum with its archival reference spectrum. In this case, we use the BD+33d2642 standard star spectrum from the IRSCAL database. This function begins by extracting the continuum emission from both the archival and observed standard star spectra (already normalized by exposure time) using a median filter, with kernel sizes defined by the user (default is 101 for

¹⁹ <https://www.tng.iac.es/instruments/lrs/>

²⁰ https://www.tng.iac.es/Astronomy/observing/manuals/ps/tech_notes/tn031.pdf

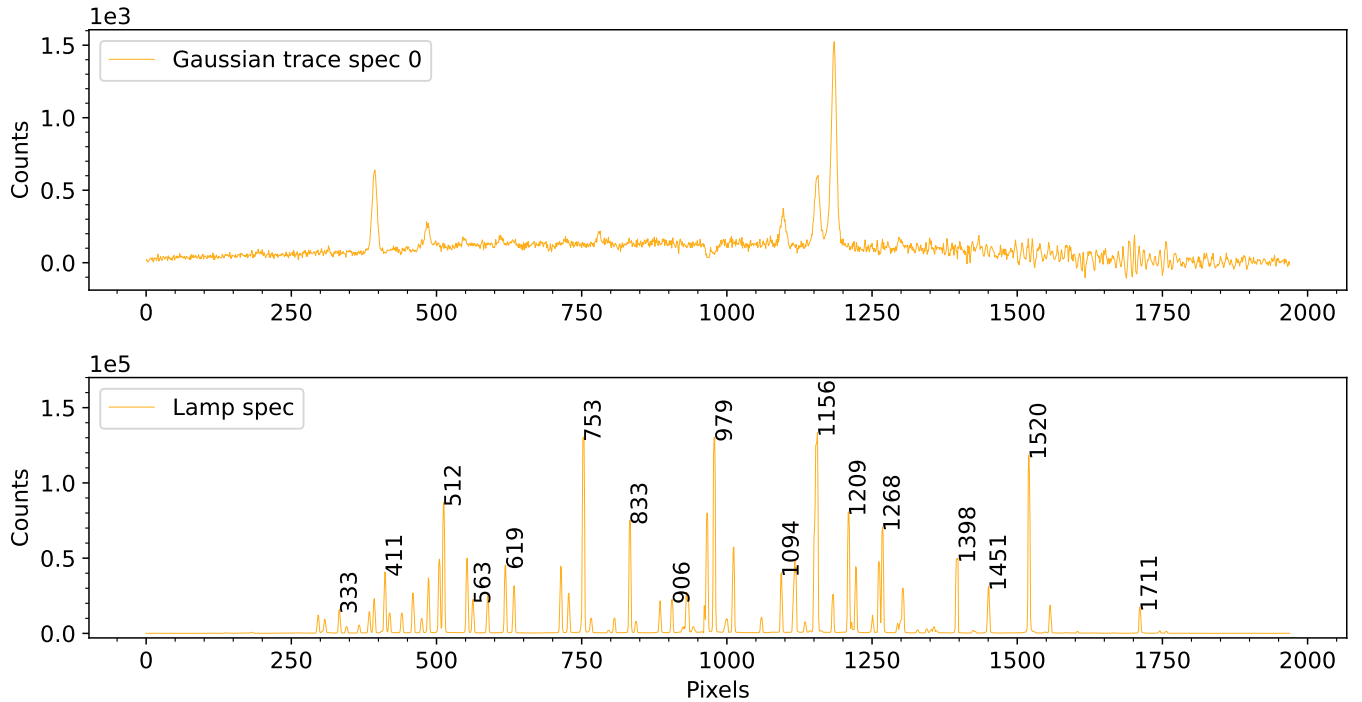


Figure 5. Non-calibrated spectra for G4Jy 1709 (top) and the Ar, Ne+Hg, Kr lamp (bottom). The numbers over the lines represent the pixel in the x -axis where the peak of each line is found. These lines are strategically selected by easyspec, such that they are relatively strong and far apart from each other.

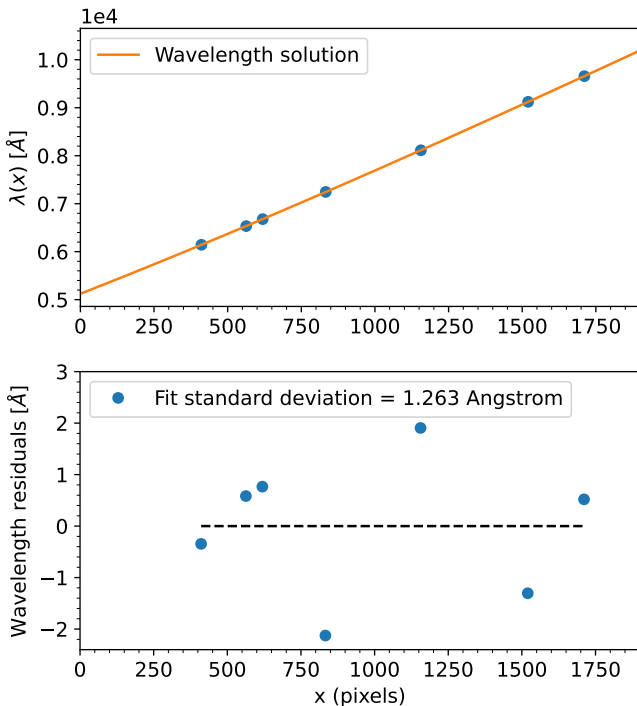


Figure 6. Wavelength solution (top) and residuals (bottom) for the spectrum of G4Jy 1709.

the observed spectrum and 11 for the archival spectrum). If undesired features—such as strong telluric absorption lines—affect the continuum estimation, the user can exclude specific wavelength regions from the median filtering by providing a list of intervals to ignore. The correction factor curve is then computed by dividing the interpolated archival continuum by the interpolated observed continuum. This flux calibration is subsequently applied to the exposure-corrected spectrum of

G4Jy 1709 using the function `target_flux_calibration()`. Finally, we correct for Galactic reddening using the Galactic visual extinction estimate from E. F. Schlafly & D. P. Finkbeiner (2011) and IRSA (2022).²¹ This correction is needed because as light travels through the interstellar medium, dust particles in the Galaxy scatter and absorb shorter-wavelength photons more efficiently than longer-wavelength photons. This process causes the observed light to appear redder than its intrinsic color. The fully calibrated spectrum is presented in the bottom panel of Figure 7 and saved in a text file together with the systematic uncertainties in wavelengths estimated as the standard deviation of the wavelength solution.

3.3. Line Fitting

With the calibrated spectrum ready, we now proceed to the `easyspec` class `analysis()`. The function `analysis.find_lines()` takes user-defined continuum zones as input and then identifies emission and absorption lines with significances above a specified threshold—in this case, 5σ —with respect to the closest user-defined continuum. Here, significance is defined as the line height divided by the standard deviation of the local continuum. In the top panel of Figure 8, we identify six lines that meet the 5σ criterion, with their wavelength positions automatically plotted above them. This function also provides options to set minimum line width and minimum separation, reducing the likelihood of spurious detections (see the GitHub tutorial for details).

At this stage, some expertise in astrophysical spectra interpretation is required. Users must be able to manually identify at least one spectral line before proceeding. In the case of AGN spectroscopy, the group of three emission lines around

²¹ Extinction values for every direction in the sky can be found here: <https://irsa.ipac.caltech.edu/applications/DUST/>.

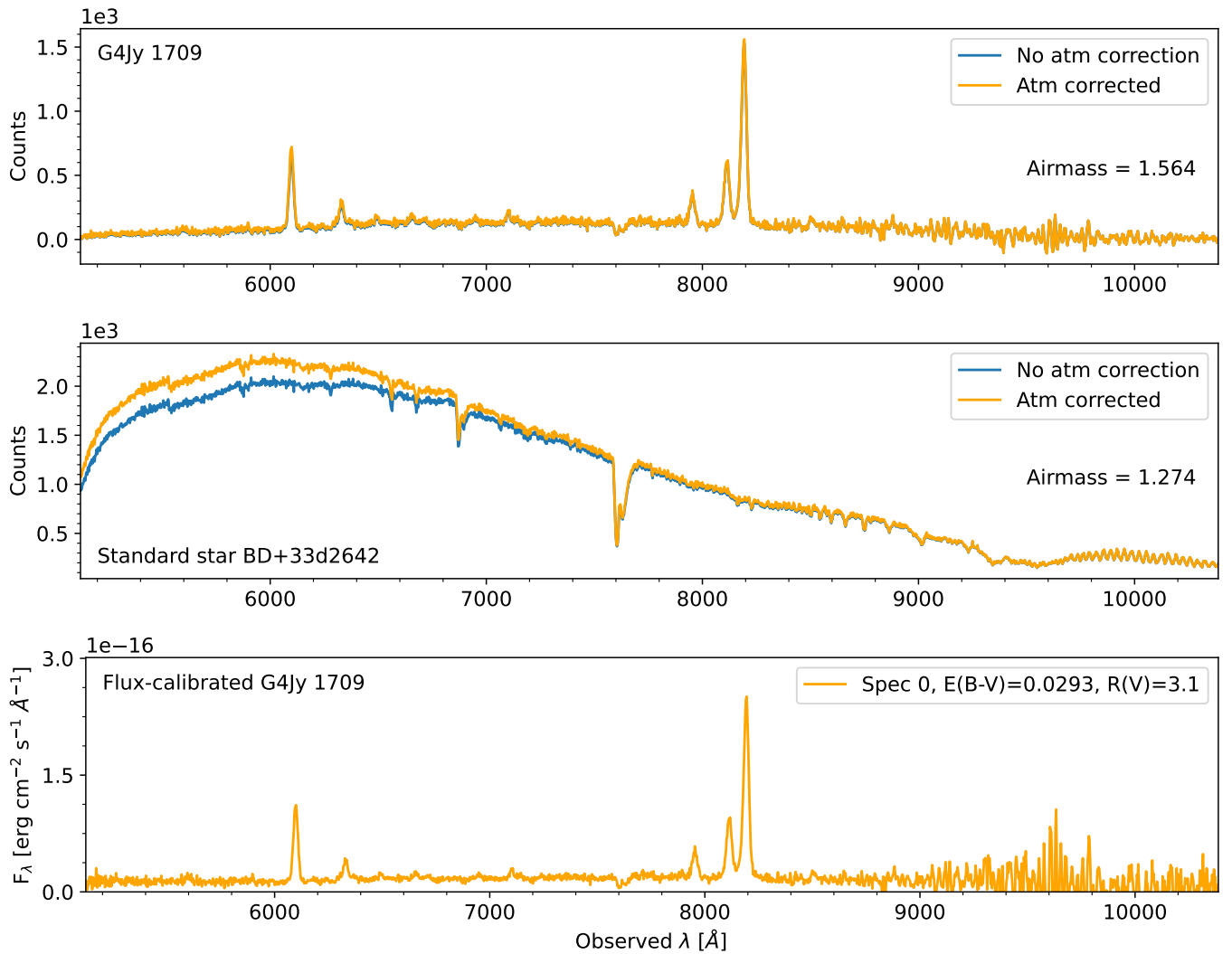


Figure 7. Target and standard star calibrated for atmospheric extinction (top and middle panels, respectively), and flux-calibrated target spectrum (bottom panel). The effect of atmospheric extinction is more relevant toward bluer wavelengths. We also see the presence of a relatively strong telluric line at $\lambda \approx 7600$ Å.

8000 Å is readily recognized as $H\beta$ and two [O III] lines. While identifying more lines improves accuracy, a single confirmed line is sufficient to continue the analysis. Next, we use the previously detected lines as input for the analysis. `fit_lines()` function, which performs an MCMC estimation—based on the `emcee` Python library (D. Foreman-Mackey et al. 2013)—to retrieve the best-fit parameters for all lines, including both isolated and blended cases. This function supports parallelization and allows users to define priors for each line. If no priors are provided, `easyspec` selects them automatically as follows: (i) the line amplitude is initially set as the measured height of the line relative to the continuum—retrieved with the function `analysis.find_lines()`—and allowed to vary between 0.1 and 10 times this value; (ii) the line center (wavelength position) is initially set as the observed wavelength of the line—also retrieved with the function `analysis.find_lines()`—and allowed to vary within ± 100 Å for isolated lines, or within half the distance to the nearest neighboring line if it lies closer than 200 Å; and (iii) the line width is initially set to 10 Å and allowed to vary in the range 0.1–150 Å. Additionally, `easyspec` can be configured to display commonly observed elemental lines in astrophysical spectra. Its line database primarily consists of

entries from the NIST Atomic Spectra Database (Y. Ralchenko 2005).^{22, 23} For this specific case, we request hydrogen, oxygen, and neon. This feature aids in the precise identification of spectral lines, even those below the 5σ threshold. Finally, the MCMC estimation provides a redshift measurement for the detected lines, as shown in the bottom panel of Figure 8.

We then recover the highest likelihood parameters and corresponding asymmetrical errors (68% confidence intervals based on the 0.16 and 0.84 quantiles of the posterior distributions) for all six fitted lines. The user can model all the lines with the same model, which can be Gaussian, Lorentzian, or Voigt, or choose a specific model for each line. Here we are modeling the $H\beta$ line with a Lorentzian, and all other lines with a Gaussian. To go deeper into the MCMC adopted in `easyspec`, including the corner plots with the covariances and temporal evolution of parameters, we refer the reader to the GitHub tutorials.

²² Most of the astrophysical lines used in `easyspec` are available here: <https://astronomy.nmsu.edu/drewski/tableofemissionlines.html>.

²³ The NIST database can be found here: <https://www.nist.gov/pml/atomic-spectra-database>.

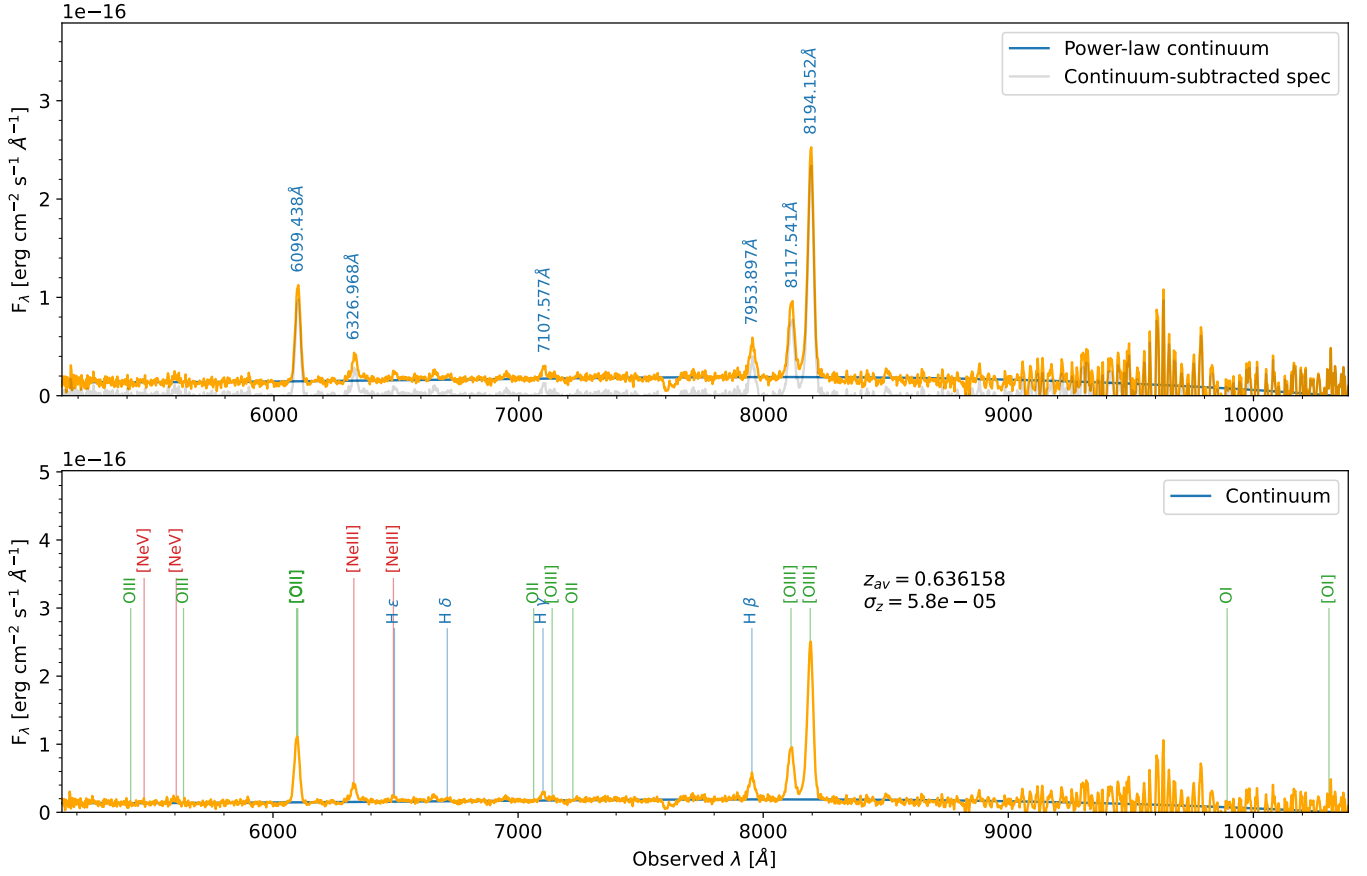


Figure 8. Detection (top) and assignment (bottom) of the lines done by easyspec. In the top panel, all lines with more than 5σ with respect to the local background are detected. In the bottom panel, the line fitting gives us the average redshift z_{av} of the six lines analyzed and its standard deviation. In this panel, we also see the over-plotted Hydrogen, Oxygen, and Neon lines from the easyspec database. It is clear that some weak lines, like [Ne V] and [O I] are lost below the adopted significance threshold.

4. Results

Although we measured the parameters of 6 emission lines in the spectrum of the AGN G4Jy 1709 and found an average redshift of $z_{av} = 0.636158 \pm 0.000058$ (see bottom panel in Figure 8), from now on we focus only on the $H\beta$ line. With this line we can estimate the supermassive black hole mass with the scaling relationships described by M. Vestergaard & B. M. Peterson (2006). These formulas are available in easyspec in the `analysis.BH_mass_Hbeta_VP2006()` function and can be explicitly written as follows:

$$\log M_{BH}(H\beta) = \log \left[\left(\frac{FWHM(H\beta)}{1000 \text{ km s}^{-1}} \right)^2 \left(\frac{\lambda_{L\lambda}(5100 \text{ Å})}{10^{44} \text{ erg s}^{-1}} \right)^{0.5} + (6.91 \pm 0.02) \right], \quad (1)$$

and

$$\log M_{BH}(H\beta) = \log \left[\left(\frac{FWHM(H\beta)}{1000 \text{ km s}^{-1}} \right)^2 \left(\frac{L(H\beta)}{10^{42} \text{ erg s}^{-1}} \right)^{0.63} + (6.67 \pm 0.03) \right], \quad (2)$$

where FWHM stands for the full width at half maximum of the line, $\lambda_{L\lambda}(5100 \text{ Å})$ is the rest-frame continuum luminosity at 5100 Å, and $L(H\beta)$ is the line luminosity. The parameters of

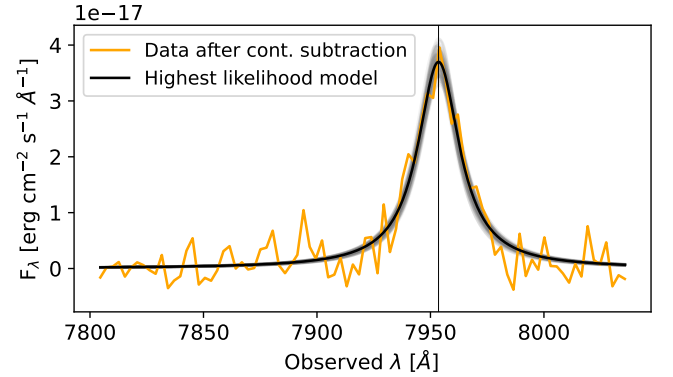


Figure 9. Lorentzian profile fit to the $H\beta$ line with the MCMC method. The fit is done in the observed frame, although the results are available in both the observed and rest frames.

the Lorentzian fit performed in the $H\beta$ line (see Figure 9) are listed in Table 1. By feeding these values to the `analysis.line_physics()` function, we recover the line integrated flux $f_{H\beta} = 1.20^{+0.03}_{-0.03} \times 10^{-15} \text{ erg s}^{-1} \text{ cm}^{-2}$ —computed as the line equivalent width times the continuum value at the line center—and the rest-frame FWHM (already corrected for instrumental broadening for a Lorentzian line profile, i.e., $FWHM = FWHM_{obs} - FWHM_{inst}$, where $FWHM_{inst} \approx \Delta\lambda$ at the position of the $H\beta$ line) in terms of the dispersion velocity $FWHM_v = 881^{+12}_{-10} \text{ km s}^{-1}$, where the conversion from

Table 1
H β Line Parameters Recovered With the MCMC Method

Parameter	Value	Unit
Redshift	$0.63610^{+0.00012}_{-0.00011}$	...
Mean _{obs}	$7953.6^{+0.53}_{-0.58}$	Å
Amplitude _{obs}	$(3.67^{+0.18}_{-0.17}) \times 10^{-17}$	erg cm ⁻² s ⁻¹ Å ⁻¹
FWHM _{obs}	$22.2^{+1.5}_{-1.3}$	Å

Note. The fit is done in the observed frame, although easyspec converts everything to the rest-frame when needed. The amplitude is computed with respect to the local continuum, as shown in Figure 9, while the mean represents the line peak position in the wavelength axis.

Å to km s⁻¹ is done with the formula $\text{FWHM}_v = c \times \text{FWHM}/\lambda_0$, where c is the speed of light and λ_0 is the line rest-frame wavelength. This relatively small FWHM is consistent with the typical values observed in NLS1s (D. E. Osterbrock & R. W. Pogge 1985), and the correlations presented in Equations (1) and (2) can therefore be directly applied to this class of objects, as demonstrated for the three NLS1s analyzed in M. Vestergaard & B. M. Peterson (2006).

We now compute the rest-frame continuum luminosity at 5100 Å and the line luminosity, using the line flux and redshift obtained earlier, that is, $L(H_{\beta}) = f_{H\beta} 4\pi D_L^2$, where D_L is the luminosity distance. These values are then applied to Equations (1) and (2) to estimate the mass of the supermassive black hole, assuming a cosmological model with $H_0 = 70 \text{ km s}^{-1} \text{ Mpc}^{-1}$, $\Omega_\Lambda = 0.7$, and $\Omega_m = 0.3$. The resulting mass estimates, in solar masses, are $\log M_{\text{BH}} = 6.97 \pm 0.43$ and $\log M_{\text{BH}} = 6.71 \pm 0.43$, respectively, showing excellent agreement between the two methods.

This type of mass estimation is most reliable for AGNs with highly inclined accretion disks (i.e., close to 90° relative to the observer’s line of sight). In such cases, the FWHM of emission lines provides an accurate measure of the total Doppler broadening due to the gas dynamics around the black hole, rather than just a projected component of this motion. For smaller inclination angles, a correction factor f is needed to adjust the black hole mass estimate (see e.g., P. Marziani & J. W. Sulentic 2012).

For G4Jy 1709, we have significant indicators suggesting a high-inclination angle:

1. Despite being a powerful radio source (F. Massaro et al. 2023), it lacks detection in γ -rays by the Fermi Large Area Telescope (S. Abdollahi et al. 2020, 2022).
2. The equivalent width of its [O III] $\lambda 5007$ Å line, which is proposed as an orientation indicator (G. Risaliti et al. 2011; S. Bisogni et al. 2017), is relatively large (i.e., $\text{EW} = -221.29^{+1.10}_{-0.94}$ Å), suggesting that our target is viewed from a high-inclination angle.

Given these factors, it is reasonable to assume a correction factor of $f \approx 1$, indicating that no additional correction to the black hole mass estimate is necessary. Additionally, due to the well-established anti-correlation between Fe II and [O III] emission in quasar spectra (T. A. Boroson & R. F. Green 1992), we do not expect a significant iron contribution in this spectrum. Moreover, since G4Jy 1709 is not a blazar, its continuum emission is unlikely to be contaminated by nonthermal emission from AGN jets. Therefore, no subtraction

of nonthermal emission is required (see, e.g., C. M. Raiteri et al. 2019, 2020).

5. Validation

To validate our results, we perform the same spectroscopic analysis with IRAF, following the standard routines as done in R. de Menezes et al. (2020). The comparison between the two data reduction methods is shown in Figure 10. While minor differences can be observed, the overall spectra are nearly identical.

We can furthermore compare the average redshift $z_{\text{av}} = 0.636158 \pm 0.000058$ we measured for G4Jy 1709 with that reported in J. Holt et al. (2008), $z_{\text{Holt}} = 0.63634 \pm 0.00003$. In their study, the authors used IRAF to process a 2D spectrum obtained with the William Herschel Telescope in La Palma, Spain. The excellent agreement between these measurements—consistent within a 1/10000 fraction—further supports the reliability of our reduction process.

Another key comparison with J. Holt et al. (2008) is the FWHM of the [O III] $\lambda 5007$ Å emission line. Our MCMC analysis (Section 3.3) yields $\text{FWHM}_v = 881^{+12}_{-10} \text{ km s}^{-1}$, while in J. Holt et al. (2008) the reported value is $\text{FWHM}_{\text{Holt}} = 919 \pm 7 \text{ km s}^{-1}$. Here we see a $\sim 4\%$ discrepancy between the results, which are still compatible within 2σ error bars. We take this as a decent level of agreement given that their observations were conducted over 15 yr prior using a different instrument. Possible intrinsic spectral variations or systematic differences in data reduction could account for this small offset.

6. Discussion and Conclusions

In this work, we introduced easyspec, a Python package designed for the analysis of near-infrared, optical, and near-ultraviolet long-slit spectra. As a demonstration of its capabilities, we applied it to the AGN G4Jy 1709, measuring its redshift as $z_{\text{av}} = 0.636158 \pm 0.000058$ —a value fully consistent with previous literature.

Moreover, we provided the first estimation of its supermassive black hole mass, obtaining $\log M_{\text{BH}} = 6.97 \pm 0.43$ and $\log M_{\text{BH}} = 6.71 \pm 0.43$ (in terms of solar masses) using two independent scaling relations. These results highlight easyspec as a powerful, flexible, and user-friendly tool for long-slit spectroscopy data reduction and analysis.

The applications of easyspec are extensive, spanning various fields of astronomy, including the study of stars, nebulae, and galaxies. The software is compatible with instruments that provide raw FITS images, making it a versatile tool for spectroscopic data analysis. So far, we have successfully tested easyspec on data from multiple telescopes, such as TNG (as shown in the previous sections), the 4.1 m SOAR telescope, in Chile, and the 1.6 m Perkin-Elmer telescope at the OPD, Brazil. The development of easyspec is part of a broader effort to create modern Python-based tools for astronomers. It follows in the footsteps of its predecessor, easyfermi (R. de Menezes 2022),²⁴ and is the second in a series of open-source packages planned to facilitate astronomical data analysis.

The potential of easyspec extends far beyond the analysis presented here. The software allows users to:

²⁴ <https://github.com/ranieremenezes/easyfermi>

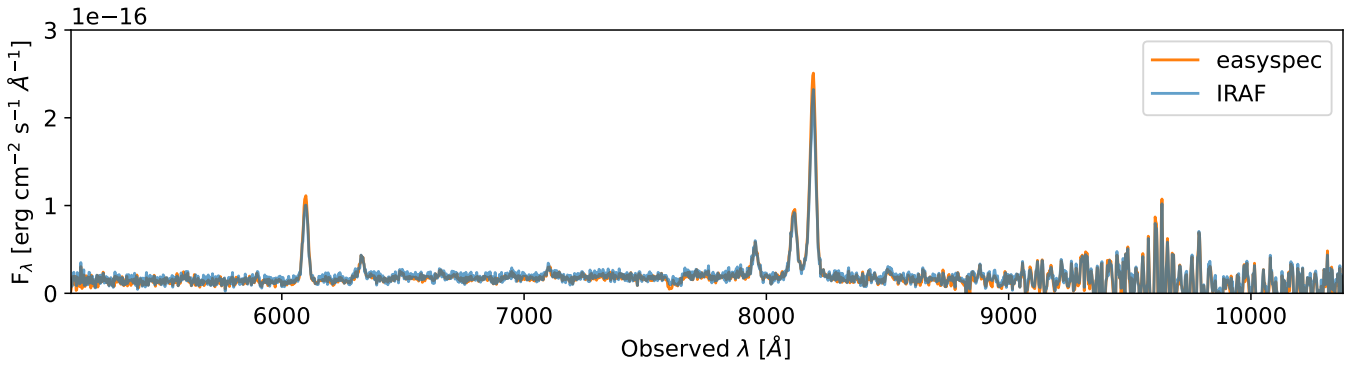


Figure 10. Comparison between the spectral data reduction performed with easyspec and IRAF, showing an excellent agreement between the two data reductions.

1. Fit multiple components (Gaussian, Lorentzian, or Voigt) to a single spectral line.
2. Monitor the temporal evolution and covariances of MCMC parameters.
3. Detect spectral absorption lines.
4. Compute equivalent widths and velocity dispersions of spectral lines.
5. Fit blended lines (up to three overlapping lines).
6. Perform bulk data cleaning for several targets at once (provided they share suitable bias and flats).
7. Extract multiple 1D spectra from a single 2D spectral image.
8. Vertically align exposures in case the telescope loses track between observations.

Additionally, easyspec is highly customizable and can be integrated into complex pipelines, depending on the user's needs. All of these features are described in detail in the GitHub tutorials and documentation. Another noteworthy capability is that the `cleaning()` class can also be applied to photometric data reduction, further expanding the software's versatility.

Acknowledgments

We thank the anonymous referee for the detailed report and for the constructive comments that allowed us to substantially improve the manuscript. C.M.R. acknowledges financial support from the INAF Fundamental Research Funding Call 2023. J.A.-P. acknowledges financial support from the Spanish Ministry of Science and Innovation (MICINN) through the Spanish State Research Agency, under Severo Ochoa Centres of Excellence Programme 2020–2024 (CEX2019-000920-S).

Author Contributions

R.M. was responsible for developing the code and drafting the manuscript. F.M. and M.N. provided the optical spectra featured in the main text and the GitHub tutorials, as well as several conceptual contributions that were incorporated into the code. C.M.R. contributed the theoretical framework for measuring black hole masses and offered numerous suggestions for the spectroscopy reduction process. H.P.H. and J.A.A.P. provided valuable insights regarding the code's features and the data reduction procedures. All authors participated in the critical revision of the manuscript.

The initial idea for this work originated when F.M. approached R.M. with a request to assist in teaching a group of students how to use legacy software for optical spectra

reduction. R.M.'s decision to decline this request ultimately led to the creation of easyspec.

Facilities: TNG (DOLORES), SOAR (Goodman).

ORCID iDs

Raniere de Menezes <https://orcid.org/0000-0001-5489-4925>

Francesco Massaro <https://orcid.org/0000-0002-1704-9850>

Michela Negro <https://orcid.org/0000-0002-6548-5622>

Claudia M. Raiteri <https://orcid.org/0000-0003-1784-2784>

Harold Peña-Herazo <https://orcid.org/0000-0003-0032-9538>

Jose A. Acosta-Pulido <https://orcid.org/0000-0002-0433-9656>

References

- Abdollahi, S., Acero, F., Ackermann, M., et al. 2020, *ApJS*, **247**, 33
- Abdollahi, S., Acero, F., Baldini, L., et al. 2022, *ApJS*, **260**, 53
- Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, *AJ*, **156**, 123
- Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, *A&A*, **558**, A33
- Berry, D., Graves, S., Bell, G. S., et al. 2022, in ASP Conf. Ser. 532, *Astronomical Data Analysis Software and Systems XXX*, ed. J. E. Ruiz, F. Pierfederici, & P. Teuben (San Francisco, CA: ASP), 559
- Bisogni, S., Marconi, A., & Risaliti, G. 2017, *MNRAS*, **464**, 385
- Boroson, T. A., & Green, R. F. 1992, *ApJS*, **80**, 109
- Craig, M., Crawford, S., Seifert, M., et al. 2017, *astropy/ccdproc: v1.3.0.post1*, Zenodo, doi:[10.5281/zenodo.1069648](https://doi.org/10.5281/zenodo.1069648)
- Currie, M. J., Berry, D. S., Jenness, T., et al. 2014, in ASP Conf. Ser. 485, *Astronomical Data Analysis Software and Systems XXIII*, ed. N. Manset & P. Forshay (San Francisco, CA: ASP), 391
- de La Peña, M. D., White, R. L., & Greenfield, P. 2001, in ASP Conf. Ser. 238, *Astronomical Data Analysis Software and Systems X*, ed. F. R. Harnden, Jr., F. A. Primini, & H. E. Payne (San Francisco, CA: ASP), 59
- de Menezes, R. 2022, *A&C*, **40**, 100609
- de Menezes, R. 2025, easyspec: The Easiest Way to do Long-slit Spectroscopy, v1.0.0.9, Zenodo, doi:[10.5281/zenodo.15732789](https://doi.org/10.5281/zenodo.15732789)
- de Menezes, R., Amaya-Almazán, R. A., Marchesini, E. J., et al. 2020, *Ap&SS*, **365**, 12
- Foreman-Mackey, D. 2016, *JOSS*, **1**, 24
- Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, *PASP*, **125**, 306
- Gordon, K. D. 2024, *JOSS*, **9**, 7023
- Guerrero, M., Santamaría, E., Liberato, G., et al. 2025, *A&A*, **694**, A105
- Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, *Natur*, **585**, 357
- Holt, J., Tadhunter, C., & Morganti, R. 2008, *MNRAS*, **387**, 639
- Hunter, J. D. 2007, *CSE*, **9**, 90
- IRSA 2022, Galactic Dust Reddening and Extinction, IPAC, doi:[10.26131/IRSA537](https://doi.org/10.26131/IRSA537)
- Lam, M. C., Smith, R. J., Arcavi, I., et al. 2023, *AJ*, **166**, 13
- Marziani, P., & Sulentic, J. W. 2012, *NewAR*, **56**, 49

- Massaro, F., White, S., García-Pérez, A., et al. 2023, *ApJS*, **265**, 32
- McCully, C., Crawford, S., Kovacs, G., et al. 2018, *astropy/astrocrappy*: v1.0.5, Zenodo, doi:[10.5281/Zenodo.1482019](https://doi.org/10.5281/Zenodo.1482019)
- Osterbrock, D. E., & Pogge, R. W. 1985, *ApJ*, **297**, 166
- Pickering, T. E., Tollerud, E., Shanahan, C., et al. 2024, *Proc. SPIE*, **13101**, 131010S
- Prochaska, J. X., Hennawi, J. F., Westfall, K. B., et al. 2020, *JOSS*, **5**, 2308
- Raiteri, C. M., Acosta Pulido, J., Villata, M., et al. 2020, *MNRAS*, **493**, 2793
- Raiteri, C. M., Villata, M., Carnerero, M., et al. 2019, *MNRAS*, **489**, 1837
- Ralchenko, Y. 2005, *MSAIS*, **8**, 96
- Risaliti, G., Salvati, M., & Marconi, A. 2011, *MNRAS*, **411**, 2223
- Schlafly, E. F., & Finkbeiner, D. P. 2011, *ApJ*, **737**, 103
- Schönell, A. J., Jr, Riffel, R. A., Riffel, R., & Storchi-Bergmann, T. 2025, *MNRAS*, **537**, 3826
- Tody, D. 1986, *Proc. SPIE*, **627**, 733
- Turchetta, M., Sen, B., Simpson, J. A., et al. 2025, *MNRAS*, **538**, 380
- Van Dokkum, P. G. 2001, *PASP*, **113**, 1420
- Véron-Cetty, M.-P., & Véron, P. 2006, *A&A*, **455**, 773
- Vestergaard, M., & Peterson, B. M. 2006, *ApJ*, **641**, 689
- Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, *NatMe*, **17**, 261