Studying the potential of the [OII] emission line as a surrogate for stellar velocity dispersion in Active Galactic Nuclei

Edward Donohue

A Senior Thesis Presented to the Department of Physics, California Polytechnic State University, San Luis Obispo

Advised by:
Dr. Vardha N. Bennert

Contents

1   Abstract

We study the emission profiles of 80 active galaxies to find a surrogate for stellar velocity dispersion. We focus on the width of the emission line of once ionized oxygen, [OII], and compare our results to previous work that used [OIII]. In previous research, [OIII] was found to be a good candidate for a surrogate for stellar velocity dispersion, but analysis of the line was complicated by the presence of wings caused by gas infall and outflow in the region. Emission lines with lower ionization levels, like [OII], are known to have less artificial line-broadening from wings. The study of [OIII] used a double Gaussian profile in an attempt to remove the wings. Here we simply fit the [OII] doublet line with two single Gaussian profiles at fixed separation and width. The main benefits of the [OII] line over the [OIII] line are the simpler fitting process, and the fact that [OII] is in a quiet region on the spectra with little interference from other emission lines. We found that the [OII] line still showed wings significant enough to require a fitting of these wings. [OIII] still appears to be the best candidate for a surrogate for stellar velocity dispersion.

2   Introduction to Active Galactic Nuclei

Active Galactic Nuclei were first observed in the form of Seyfert galaxies and Quasars, short for Quasi-stellar objects. Quasars initially appeared to be a point source that was very luminous and very distant. Seyfert galaxies were classified based on their unusual and broadened emission lines at their nucleus. The current theory classifies both of these phenomena as Active Galactic Nuclei or AGNs. AGNs are galaxies in which a supermassive black hole at the center of the galaxy is actively accreting matter. Note that most or all large galaxies are believed to contain a supermassive black hole at their center, but most are not active. This accretion of matter forms a bright, hot disk which we can observe both directly and indirectly.

The current model of AGNs consists of a central source with an accretion disk, a broad-line region of gas closely surrounding the nucleus, and a much larger narrow-line region. Finally the model predicts a dust torus that is larger than the broad-line region and smaller than the narrow-line region. The orientation of the central features, including the dusty torus, is independent of the orientation of the galaxy. The geometry of the model is shown in Figure 1.
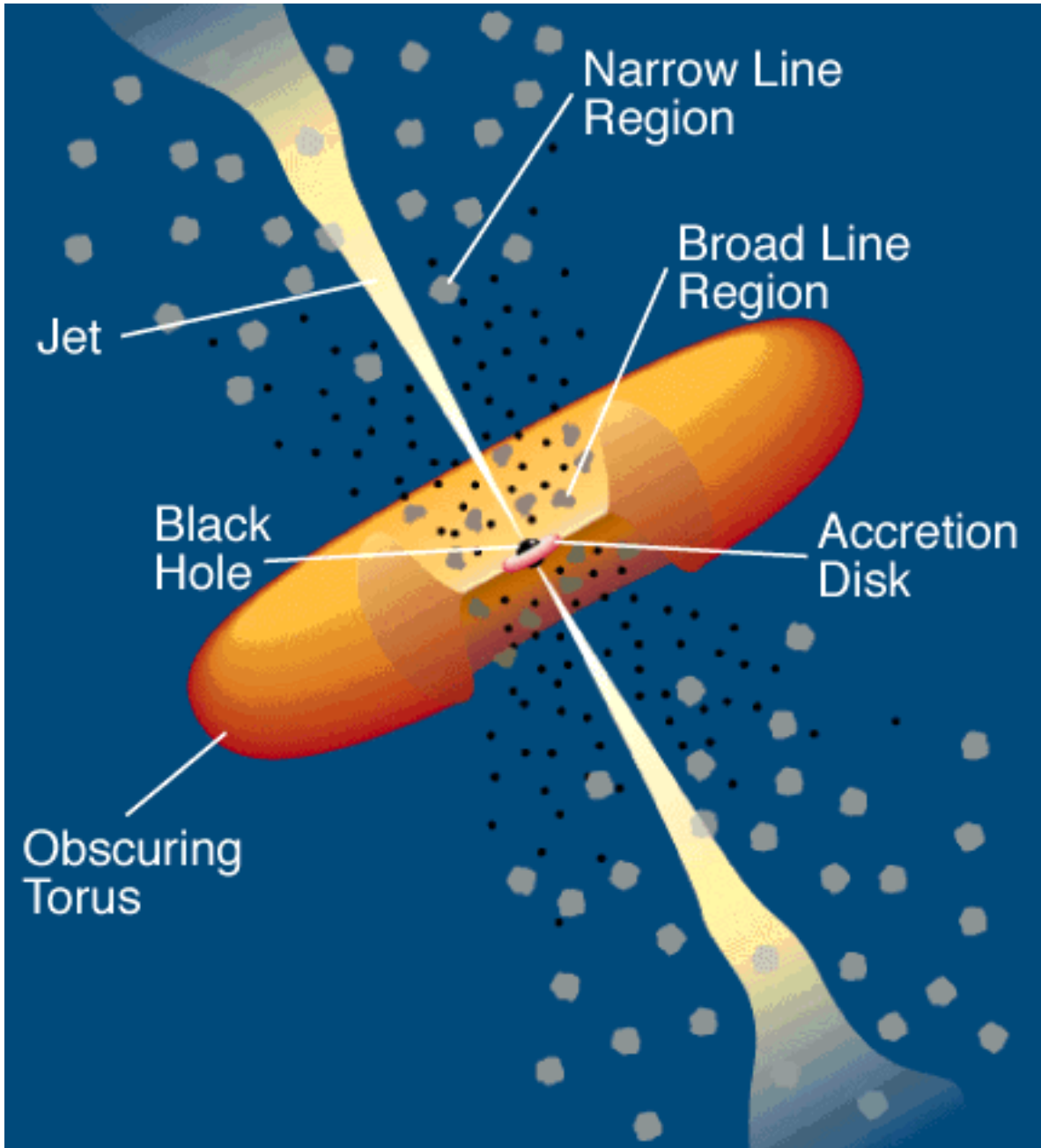
Figure 1: The commonly accepted model of AGNs (Taken from Urry & Padovani 1995).

2.1 Broad-line Region

The broad-line region, or BLR, is characterized by its large amount of Doppler broadening (hence the name) and its high density of gas. The BLR is usually defined as the region with Doppler broadening of at least 1000 km/s (Peterson 1997). The large amount of broadening is due to the region's close proximity to the black hole; the BLR is around 10^{14} m in radius (Peterson 1997). Due to the high density of gas, only permitted lines such as the Balmer series are seen in the BLR. The variability of the BLR directly corresponds to the variability of the accretion disk because the BLR is ionized by the accretion disk. This variability has allowed researchers to estimate the radius of the region with a technique called reverberation mapping. When a significant change in flux is measured in the accretion disk, the light re-emitted by the parts of the BLR closer to Earth will reach us sooner than the light re-emitted by the parts further away. We use this time delay that is caused by the travel time of light to help estimate the radius of the region.

2.2 Narrow-line Region

The narrow-line region, or NLR describes the less dense, slower moving gas that occupies a large area around the BLR and dusty torus. From an astronomer's perspective the NLR is characterized by emission lines with width less than 1000 km/s and a radius of at least 1pc (Peterson 1997). The relatively small amount of Doppler broadening in the NLR as compared to the BLR is due to its larger distance from the central black hole. Because of the large distance from the black hole, the NLR follows the gravitational potential of the entire bulge, just as the stars of the host galaxy do. In the NLR the gravity causes the Doppler broadening observed, in the host galaxy it causes stellar velocity dispersion. Stellar velocity dispersion is a measure of the variation in star velocity relative to the observer, and is a proxy for the gravitational force on the stars. The relationship between Doppler broadening in the NLR and the stellar velocity dispersion in the host galaxy is crucial for the goals of this project. Additionally the NLR is spatially resolved in nearby AGNs. The line profiles of the NLR do not have a simple Gaussian shape but usually have a wider base and are asymmetric due to a (typically) blue wing. The strength of the wing is known to increase with higher ionization levels, for example we would expect [OIII] emission lines to have stronger wings than [OII] in the same AGN. These wings are not caused by the gravitational potential of the bulge, but rather gas outflow (or infall for red wings). We do not observe the opposite effect on the other side of the NLR because the light is obscured. If not accounted for, these wings would give us an overestimation of the strength of the gravitational potential. The strength of the emission lines from the NLR are similar to the BLR even though the NLR is much larger. This is attributed to the low density of the gas in the NLR.

2.3 [OII] and [OIII]

The two emission lines of study in this paper, [OII] and [OIII], both occur in the NLR. These lines were chosen due to their high flux and lack of interference from other emission lines. [OIII] can easily be identified in a spectrum as a prominent, narrow peak with a slightly longer wavelength than Hβ. Its rest wavelength is 5007 angstroms. [OII] can be identified as a prominent, narrow peak in the lower wavelengths of the visual spectrum. It is actually a doublet line with peaks at 3726.1 and 3728.8 angstroms, however the two peaks are often blended past the point of distinguishability. The "II" and "III" in [OII] and [OIII] refer to their respective ionization levels. Because [OII] is once ionized while [OIII] is twice ionized, [OII] should have weaker wings than [OIII] in a galaxy where infall or outflow is present.

2.4 Unification

The model described unifies the types of active galaxies discussed: quasars, type-1 Seyferts and type-2 Seyferts. Quasars are active galaxies in which the central source outshines the host galaxy such that the host galaxy is difficult to detect. Type-1 Seyferts are active galaxies where the broad-line region and the narrow line region are observable. Finally type-2 Seyferts are objects where the broad-line region is obscured by the dust torus. The effort towards unification of Seyfert galaxies was advanced when type-2 Seyferts were observed to have broad lines in polarized light. The light was polarized and at a reduced flux because the light was scattered and not observed directly. These can be explained by objects with a viewing angle where the dust torus only obscures the broad-line region. The unification model also is supported by the presence of ionization cones observed in [OIII]. These cones are part of the narrow line region and extend symmetrically from the nucleus. The conical shape is due to some of the light from the source being obscured by the dusty torus.

2.5 Black hole mass measurements

There are a number of techniques used to estimate the black hole mass of AGNs. The simplest and most intuitive way is to use the rotation speed and distance of the objects orbiting the black hole. With nearby galaxies, measurements of the velocities of the stars orbiting the black hole can be made directly, giving an accurate estimation of the black hole mass. In distant active galaxies the motion of the stars is not directly detectable so we have to find an alternate method. In that case, measurements are taken on the amount of Doppler broadening (usually with the H-beta line) due to rotation in the BLR to determine the rotation speed. Then, reverberation mapping, as discussed earlier is used to determine the radius. Kepler's laws are then applied to estimate the mass. Unfortunately this technique makes assumptions about the geometry and kinematics of the AGN leading to uncertainty in the estimate.

The other main technique is known as the single-epoch method because only one spectral image is required, as opposed to watching for time variability for reverberation mapping. There is a known relationship between the BLR size and the continuum luminosity, so the continuum luminosity can be used as a surrogate. The velocity is determined by the width of the H-beta line, just like in reverberation mapping. The velocity and size are then used to estimate the mass of the black hole.

3 Sample and Analysis Process

3.1 Sample

Our sample consists of about 100 local type-1 AGNs observed with the 10m Keck telescope. This is the same sample used for the [OIII] fitting carried out in research projects by former undergraduate students Kelsi Flatland, Sean Lewis, Nathan Milgram (see their theses published at Kennedy Library) and Donald Loveland (Physics). We use this limited set so that we can feasibly check each object for a successful fit. The galaxies are chosen from the SDSS survey based on their broad $H\beta$ width (i.e. black hole mass) and within a redshift range of $0.02 < z < 0.1$ to allow for the host galaxies to be well resolved. Importantly, the stellar absorption lines from stars in the host galaxy can be detected to measure the stellar velocity dispersion (Harris et al. 2012). This allows us to directly compare the stellar velocity dispersion with the width of the [OII] and [OIII] emission lines to search for correlations.

The long-slit spectra were taken along the major axis of the galaxy so we can address the change of emission line profiles as a function of distance from the center of the galaxy.

3.2 Qualitative analysis

To better understand the differences between the [OII] and [OIII] emission lines we perform an initial visual inspection of the emission lines for each galaxy. We first look for asymmetric line profiles, indicating red or blue wings. Red wings are correlated with gas infall because the gas on the side closest to the observer is moving away and is redshifted. The opposite side of the AGN is obscured so the opposite effect is not observed. Contrastingly, a blue wing means gas outflow is occurring. These wings are caused by non-gravitational gas motion. Thus, they need to be accounted for and subtracted out to determine the width of the central line profile from gas following the gravitational potential of the bulge. As expected, we find stronger evidence of wings in [OIII] than [OII]. We also find examples in [OIII] where the line profile changed as a function of distance from the center of the galaxy. This is likely caused by spatially uneven infall and ouflow. Finally, for some galaxies, off-nuclear spectra exist for which the [OIII] peak is higher than in the center, indicating extra ionization in those areas from star formation, i.e. an HII region. Examples of the different features of emission lines are shown in Figure 2.

In addition to making general observations, a visual statistical analysis of [OIII] line profiles was performed. In this analysis we found 34% of objects with a blue wing, 15% with a red wing, 51% of objects with no clear wings, 17% of objects with evidence of rotation, 7% of objects with a change in profile over the radius of the NLR, and 15% of objects with evidence of a star forming HII region. Note that objects can be in multiple categories.

The initial analysis confirms our expectations of the differences between the [OII] and [OIII] emission lines: The [OIII] lines are much brighter and can be traced far out from the center of the galaxy with a signal-to-noise ratio that justifies a detailed computational analysis. However, amongst the [OIII] lines, we also find many examples with strong blue wings and some with strong red wings, requiring a complex fit. For [OII], only the central line consistently has a high enough signal-to-noise ratio to justify computational analysis. The [OII] emission line profiles are overall very symmetric, despite the fact that it is a blended doublet line in which the peaks of each component in the doublet are not necessarily the same magnitude.
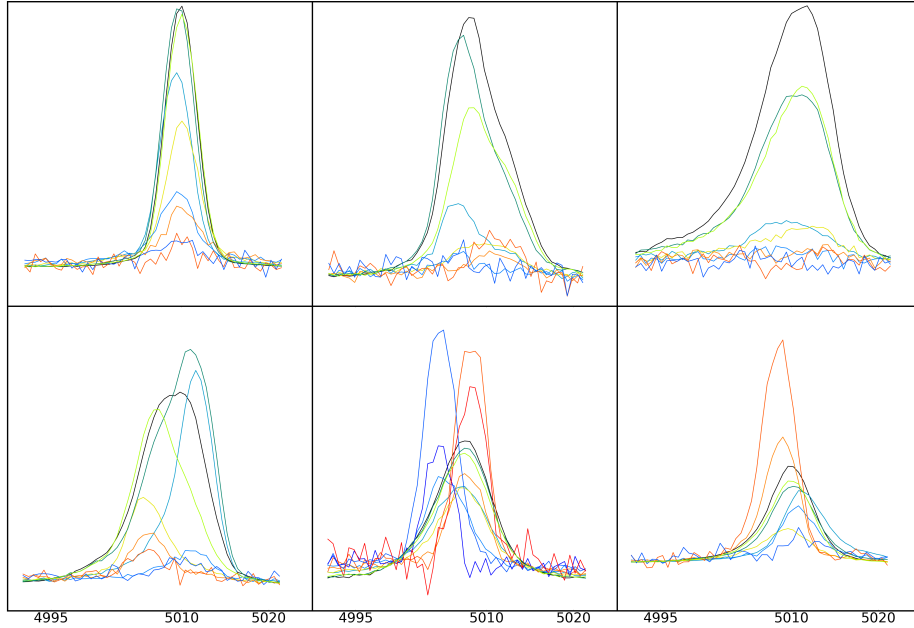
Figure 2: The [OIII] emission line of six selected galaxies. The x-axis shows wavelength in Å, the y-axis shows flux in arbitrary units. The central emission line is shown in black. Outer spectra are colored-coded as follows: spectra extracted on one side of the galaxy center start at turquoise and get more blue towards the edge, on the other side starts lime green and goes to yellow, orange and red. Top row from left to right, names are abreviated RA/Dec: 1043+1105: A typical emission line with no wing, 1606+3324: an [OIII] line with a red wing, 1419+0754: an emission line with a blue wing. Bottom row: 1554+3238: A galaxy with spatial emission-line profile changes, from a red wing on one side of center to a blue wing on the other side, 0206-0017: a galaxy with gas undergoing galactic rotation, 1355+3834: a galaxy where the strength of the [OIII] emission line increases further away from the center, indicating an HII star forming region.

3.3 IDL Code

3.3.1 Goal of the code

To measure the width of the [OII] emission line, we use an IDL program to fit a double Gaussian profile to the blended doublet line. The IDL code does take into consideration the underlying AGN and host-galaxy continuum, but does not need to factor in other emission lines as the [OII] line (unlike the [OIII] line) is isolated from other strong lines. The [OIII] line, for comparison, has strong closely neighboring lines such as broad Hβ and underlying broad FeII emission that need to be fitted simultaneously (for a detailed description of the IDL code used in that case, see Bennert et al. 2015). The downside to the [OII] line is that it is a weaker line so it has a lower signal-to-noise ratio to [OIII].

3.3.2 Fitting process

The IDL program uses an expected wavelength range to limit the scope of the fit with a default range of 3715 to 3740 angstroms. For certain objects, this range was either widened or narrowed to obtain the best fit. It is worth noting that changing the expected range did not significantly affect the width achieved from the fit, only the likelihood of a successful fit. The program fits a linear continuum and two Gaussian functions. The Gaussian functions are separated by the known wavelength difference between the two blended [OII] lines of 2.7 angstroms and are forced to have the same width. We are able to make the assumption that the widths are the same because both components of the line originate from the same region and thus any broadening should be uniform. The heights can vary freely, since the ratio of the flux in the doublet depends on temperature and density (Osterbrock and Ferland. 2006). The program then plots the observed spectrum, the continuum fit, each Gaussian fit, and the resulting combined fit, as well as the residual (observed spectrum minus total fit). It also prints the width of the [OII] fit. The resulting plot and output data can be seen in figure 3. At this point the user can verify the fit looks good before moving on to the next object. The data, including the fits, the raw data, and numerical information about the fits is stored in a ".sav" file. A separate Python program was used to access and re-plot the data.
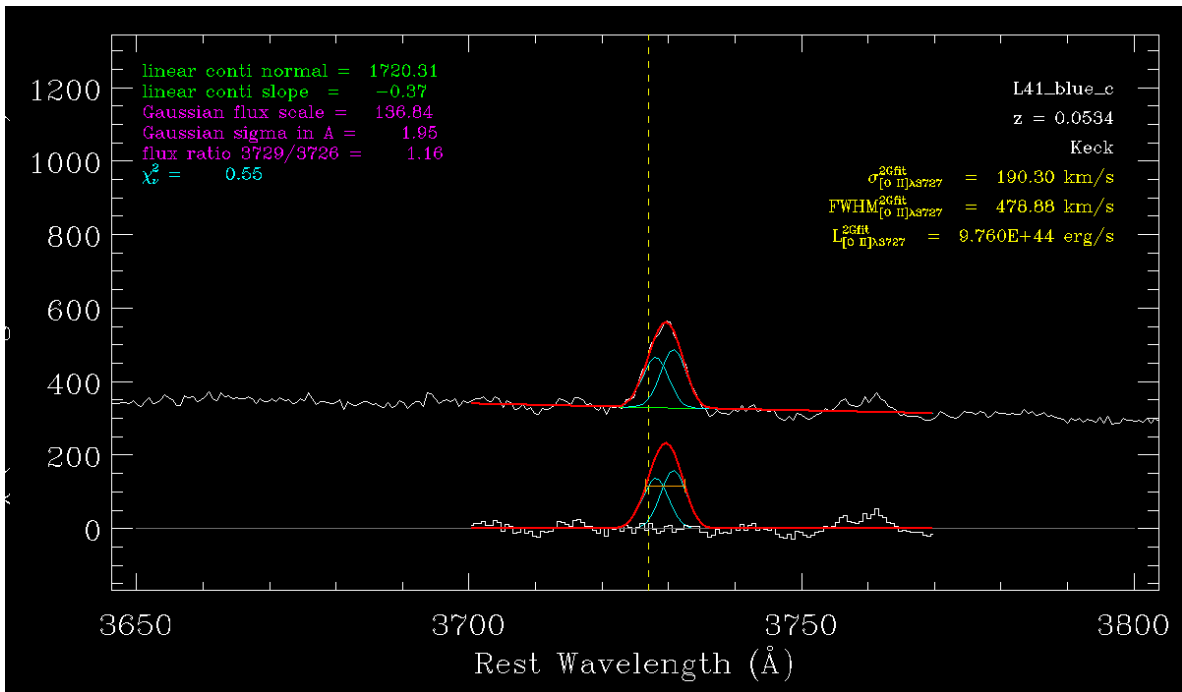
Figure 3: Sample output plot from the IDL fitting process. The blue and green lines are the individual components and the red line is the combined fit. The white line on the top graph is the true spectra and the white line on the bottom is the residual.

4   Results

To evaluate our results, we directly compare our [OII] data to the data we had for [OIII] (research done by Cal Poly Physics major Donald Loveland) and the true stellar velocity dispersion (Harris et al. 2012). The results are plotted with the width of the [OII] line (sigma) on the x-axis and the width of the [OII] line divided by the width of the [OIII] line or the stellar velocity dispersion on the y-axis, as seen in figure 4 below. The best fit is between the [OII] width and the [OIII] double Gaussian width. At low sigma values (100-150 km/s) there is a very small difference between the two fits. At higher widths however, [OII] over-fit the [OIII] double Gaussian width. This suggests that the [OII] wings, although less prominent, still lead to over-fitting at higher sigma values. When compared to the [OIII] single Gaussian and Gauss-Hermite polynomials however, the [OII] value is consistently lower. This supports our original hypothesis that [OII] has less prominent wings. When directly compared to the stellar velocity dispersion, [OII] had a somewhat larger error with a similar over-fitting by [OII] at higher sigma values. In conclusion, although the wings observed in the [OII] line are less prominent than the wings in [OIII], they are still significant enough to eliminate a simple double Gaussian fit of [OII] as the best candidate for a surrogate for stellar velocity dispersion.
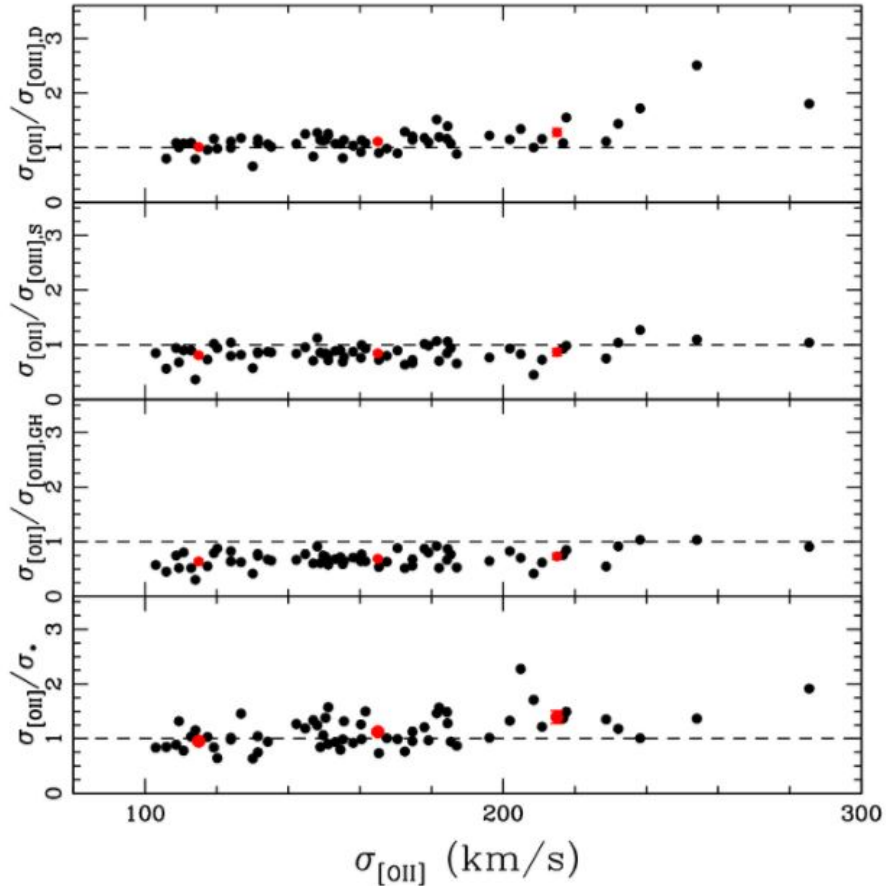
Figure 4: [OII] sigma results in comparison to previous results plotted against [OII] sigma. [OII] is divided by the previous result for each sub-graph. From top to bottom: [OIII] double Gaussian, [OIII] single Gaussian, [OIII] Gauss-Hermite, stellar velocity dispersion. The red data points are the mean ratio from three bins designed to show the trend as the [OII] sigma increases. Results were from Donald Loveland's research, figure is from Bennert et al. 2018

5   Discussion and Summary

In this project we found that [OII] can be fit with two Gaussian profiles to give an estimation of the stellar velocity dispersion for active galaxies. The [OII] emission lines appear to still have wings, especially at higher sigma values, although to a lesser extent than [OIII]. There appears to be a strong correlation between the [OII] emission line, the [OIII] emission line, the stellar velocity dispersion, and the black hole mass. In both cases the wings are observed and must be accounted for. As for using the [OII] line, a better method may be fitting each component of the blended doublet with a double Gaussian, with fixed separation and fixed widths for the two narrow components and fixed widths for the two wider components that represent the wings. Another possibility is that the [OII] and [OIII] lines could be compared to each other to give an estimation for the strength of the wings. This would only be true if the [OII] wings were smaller than the [OIII] wings by a predictable amount. With the usefulness of the [OII] line not fully determined, the best known method is fitting the double-Gaussian for [OIII]. Hopefully further research can determine the underlying mechanism behind the apparent relationship of black hole mass, stellar velocity dispersion, and broadening of emission lines.

6   Acknowledgements

7 Bibliography

Bennert, V. N., Auger, M. W., Treu, T., Woo, J.-H., Markan, M. A., The Astrophysical Journal, 2011, 726:59, *A Local Baseline of the Black Hole Mass Scaling Relations for Active Galaxies. I. Methodology and Results of Pilot Study*

Bennert, V. N., Treu, T., Auger, M. W., Cosens, M., Park, D., Rosen, R., Harris, C. E., Markan, M. A., Woo, J.-H., The Astrophysical Journal, 2015, 809-820, *A Local Baseline of the Black Hole Mass Scaling Relations for Active Galaxies. III. The MBH-σ Relation*

Bennert, V. N., Loveland, D., Donogue, E., Cosens, M., Lewis, S., Komossa, S., Treu, T., Malkan, M. A., Milgram, N., Flatland, K., Auger, M. W., Park, D., Lazarova, M. S., 2018, Monthly Notices of the Royal Astronomical Society, 481, 138, *Studying the spatially-resolved [OIII] 5007 A emission-line profile in a sample of ~80 local active galaxies*

Cosens, M. *Using an IDL Code to Fit AGN Spectra.* Cal Poly (2015).

Flatland, K. *Determining Stellar Velocity Dispersion in Active Galaxies: Is the [OIII] Width a Valid Surrogate?* Cal Poly (2012).

Harris, C. E., Bennert, V. N., Auger, M. W., Treu, T., Woo, J.-H., Malkan, M. A., *A Local Baseline of the Black Hole Mass Scaling Relations for Active Galaxies. II. Measuring Stellar Velocity Dispersion in Active Galaxies.* The Astrophysical Journal Supplement, August 2012, Volume 201, Issue 2.

Line, A. *Probing the Fitting Accuracy of Active Galaxy Spectra.* Cal Poly (2016).

Milgram, N. *Determining the Relationship Between the [OIII] 5007 A Emission Line Profile and the Stellar Velocity Dispersion in Active Galaxies.* Cal Poly (2015).

Osterbrock, Donald E., and Gery J. Ferland. *Astrophysics of Gaseous Nebulae and Active Galactic Nuclei.* 2nd ed., University Science Books, 2006.

Peterson, Bradley M. *An Introduction to Active Galactic Nuclei.* Cambridge University Press, 1997.

Urry, C. M., Padovani, P., *Unified Schemes for Radio-Loud Active Galactic Nuclei.* Publications of the Astronomical Society of the Pacific, v.107, p.803, 1995.

8 Appendix

8.1 Table of Results

The table on the following page details the results from this project and the related research presented in Bennert et al. 2018. Col. (1): Right ascension. Col. (2): Declination. Col. (3): Redshift. Col. (4): Logarithm of mass of black hole in solar units. Col. (5): Effective spherical radius in kpc. Col (6): Stellar velocity dispersion of effective radius measured from CaH K dark absorption lines. Col. (7): [OIII] width within effective radius measured with a double Gaussian fit. Col. (8): [OIII] width within effective radius measured with a single Gaussian fit. (9): [OIII] width within effective radius measured with a Gaussian-Hermite polynomial fit. (10): [OII] width within effective radius.

R.A. (J2000)	Dec. (J2000)	z	$\log(M_{BH}/M_\odot)$	R-eff kpc	σ km s^{-1}	$\sigma_{\rm [OIII]}D$ km s^{-1}	$\sigma_{\rm [OIII]}S$ km s^{-1}	$\sigma_{\rm [OIII]}GH$ km s^{-1}	$\sigma_{\rm [OII]}$ km s^{-1}
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
00 13 35.38	-09 51 20.9	0.0615	7.85	4.8	96	123	212	261	151
00 26 21.29	+00 09 14.9	0.06	7.05	1.8	172	190	190	193	171
00 38 47.96	+00 34 57.5	0.0805	8.23	1.9	127	174	212	249	160
01 09 39.01	+00 59 50.4	0.0928	7.52	0.3	183	144	263	310	175
01 21 59.81	-01 02 24.4	0.054	7.75	1.8	90	152	247	290	205
01 50 16.43	+00 57 01.9	0.0847	7.25	4.5	176	131	174	245	149
02 06 15.98	-00 17 29.1	0.043	8.00	6.2	225	183	229	307	165
02 12 57.59	+14 06 10.0	0.0618	7.32	1.0	171	152	181	223	158
03 01 44.19	+01 15 30.8	0.0747	7.55	2.7	99	144	312	375	114
03 36 02.09	-07 06 17.1	0.097	7.53	12.9	236	138	188	230	238
03 53 01.02	-06 23 26.3	0.076	7.50	1.6	175	113	155	177	131
08 11 10.28	+17 39 43.9	0.0649	7.17	2.5	142	103	124	138	111
08 13 19.34	+46 08 49.5	0.054	7.14	1.0	122	100	116	145	109
08 45 56.67	+34 09 36.3	0.0655	7.37	1.4	123	...	121	179	103
08 57 37.77	+05 28 21.3	0.0586	7.42	2.5	126	124	156	194	124
09 04 36.95	+55 36 02.5	0.0371	7.77	4.0	194	144	173	216	155
09 21 15.55	+10 17 40.9	0.0392	7.45	2.6	...	109	161	211	109
09 23 43.00	+22 54 32.7	0.0332	7.69	0.9	149	158	275	316	285
09 23 19.73	+29 46 09.1	0.0625	7.56	4.2	142	102	117	151	119
09 27 18.51	+23 01 12.3	0.0262	6.94	7.1	196	172	198	241	185
09 32 40.55	+02 33 32.6	0.0567	7.44	0.7	126	121	152	169	131
10 29 25.73	+14 08 23.2	0.0608	7.86	3.0	185	163	182	224	179
10 29 01.63	+27 28 51.2	0.0377	6.92	2.6	112	133	169	213	142
10 29 46.80	+40 19 13.8	0.0672	7.68	2.0	166	170	210	266	168
10 42 52.94	+04 14 41.1	0.0524	7.14	3.2	...	133	157	207	135
10 58 28.76	+52 59 29.0	0.0676	7.50	1.3	122	116	152	187	145
11 01 01.78	+11 02 48.8	0.0355	8.11	5.8	197	161	224	253	232
11 04 56.03	+43 34 09.1	0.0493	7.04	1.1	...	108	155	203	127
11 16 07.65	+41 23 53.2	0.021	7.23	1.6	108	149	174	252	162
11 37 04.17	+48 26 59.2	0.0541	6.74	1.1	155	152	241	257	175
11 43 44.30	+59 41 12.4	0.0629	7.51	3.8	122	111	119	150	124
11 44 29.88	+36 53 08.5	0.038	7.73	1.0	168	120	190	229	151
11 45 45.18	+55 47 59.6	0.0534	7.22	1.4	118	136	201	241	156
11 47 55.08	+09 02 28.8	0.0688	8.39	3.4	147	151	175	204	178
12 05 56.01	+49 59 56.4	0.063	8.00	2.4	152	175	217	244	202
12 10 44.27	+38 20 10.3	0.0229	7.80	0.6	141	133	179	200	150
12 23 24.14	+02 40 44.4	0.0235	7.10	3.4	124	120	170	198	181
12 31 52.04	+45 04 42.9	0.0621	7.32	1.5	169	205	306	417	229
12 41 29.42	+37 22 01.9	0.0633	7.38	1.7	144	132	174	212	184
12 46 38.74	+51 34 55.9	0.0668	6.93	3.9	119	116	132	162	148
13 06 19.83	+45 52 24.2	0.0507	7.16	2.3	114	122	161	212	117
13 12 59.59	+26 28 24.0	0.0604	7.51	1.7	109	103	126	217	113
13 23 10.39	+27 01 40.4	0.0559	7.45	0.9	124	158	219	276	184
14 05 14.86	-02 59 01.2	0.0541	7.04	0.6	125	132	189	235	106
14 16 30.82	+01 37 07.9	0.0538	7.26	3.6	173	182	291	342	211
14 19 08.30	+07 54 49.6	0.0558	8.00	5.4	215	211	285	354	187
14 34 52.45	+48 39 42.8	0.0365	7.66	0.9	109	132	178	207	150
15 35 52.40	+57 54 09.3	0.0304	8.04	2.8	110	175	208	244	147
15 43 51.49	+36 31 36.7	0.0672	7.73	3.8	146	140	221	257	218
15 45 07.53	+17 09 51.1	0.0481	8.03	1.1	163	143	172	225	153
15 54 17.42	+32 38 37.6	0.0483	7.87	1.7	158	200	235	287	217
16 05 02.46	+33 05 44.8	0.0532	7.82	1.6	187	122	128	137	120
16 06 55.94	+33 24 00.3	0.0585	7.54	1.7	157	192	226	263	155
16 11 56.30	+52 11 16.8	0.0409	7.67	1.3	116	152	259	349	182
16 36 31.28	+42 02 42.5	0.061	7.86	9.7	205	197	227	314	130
22 21 10.83	-09 06 22.0	0.0912	7.77	6.1	142	126	154	198	134
22 22 46.61	-08 19 43.9	0.0821	7.66	1.7	122	208	464	500	209
22 33 38.42	+13 12 43.5	0.0934	8.11	2.1	193	160	257	303	196
23 27 21.97	+15 24 37.4	0.0458	7.52	6.6	225	133	271	335	173
23 51 28.75	+15 52 59.1	0.0963	8.08	2.5	186	101	232	245	254

8.2 Python Codes

The following Python programs are versions of the programs used to plot and extract data from image files and files generated from the IDL code. The programs have been simplified for readability but are still functional.

8.2.1 fits_file_displayer.py Code and Output

Code:

```
#Program for Python 3 to display data from fits files

#import fits from astropy
from astropy.io import fits

#open the fits file as "fits_file"
fits_file = fits.open("data/L1_blue_c.fits")

#display the information in the fits file
#in this case we want to use the first ImageHDU, index "1"
print(fits_file.info())

#we will use matplotlib to display the spectrum
import matplotlib
matplotlib.use('agg')
import matplotlib.pyplot as plt

#plot the data from the "1" index from the fits file with matplotlib
plt.plot(fits_file[1].data)

#the axes are in arbitrary units so we will remove them
axes = plt.gca()
axes.axes.get_xaxis().set_visible(False)
axes.axes.get_yaxis().set_visible(False)

plt.savefig('spectrum', dpi=200)
```

Output:

```
Filename: Desktop/data/L1_blue_c.fits
No.    Name     Ver    Type      Cards   Dimensions   Format
  0  PRIMARY      1 PrimaryHDU      6    ()
  1               1 ImageHDU       13    (3253,)   float64
  2               1 ImageHDU       13    (3253,)   float64
  3               1 ImageHDU       13    (3253,)   float64
None
```
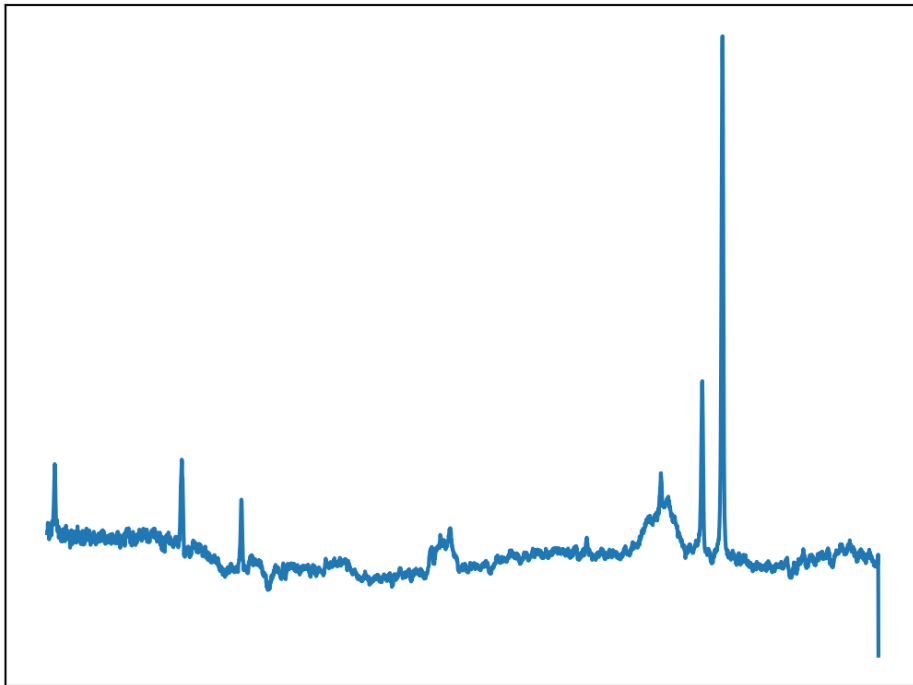

Figure 5: Spectrum extracted from a fits file and plotted

8.2.2   sav_file_displayer.py Code and Output

Code:

```
#Python 3 program to read and plot spectral data saved in the IDL format ".sav"

#to read an IDL ".sav" files we use a function from the scipy package
import scipy.io
data = scipy.io.readsav("oii/allmodelfits_L1_blue_c_Keck.sav")

#the sav file is converted into a dicionary, the only entry here is "allmodels"
#"allmodels" corresponds to a recarray which is a type of numpy array
#we can learn about the contents of the array from the following:
print(data['allmodels'].dtype)

#here we will use the following indecies of the recarray:
#  4:   'flux', the raw spectral data in arbitrary units of flux
#  7:   'modelpl', a linear fit of the local continuum around the [OII] line,
#       this is added to the following three plots so they match the raw data
#  8:   'modeloiifull' the two Gaussians combined that make up the [OII] doublet line
#  9:   'modeloii3726' the Gaussian that makes up the [OII] 3726 A line
#  10:  'modeloii3726' the Gaussian that makes up the [OII] 3729 A line

#we will use matplotlib to display our plots
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt

#in the sample image 4 is blue, 8+7 is orange, 9+7 is green, 10+7 is red
plt.plot(data['allmodels'][0][4][500:555])
plt.plot(data['allmodels'][0][8][500:555]+data['allmodels'][0][7][500:555])
plt.plot(data['allmodels'][0][9][500:555]+data['allmodels'][0][7][500:555])
plt.plot(data['allmodels'][0][10][500:555]+data['allmodels'][0][7][500:555])

#the axes are in arbitrary units so we will remove them
axes = plt.gca()
axes.axes.get_xaxis().set_visible(False)
axes.axes.get_yaxis().set_visible(False)

plt.savefig('oii_fit', dpi=200)
```

Output:

```
(numpy.record, [(('objectname', 'OBJECTNAME'), 'O'), (('tobjname', 'TOBJNAME'), 'O'),
(('redshift', 'REDSHIFT'), '>f8'), (('wavelength', 'WAVELENGTH'), 'O'),
(('flux', 'FLUX'), 'O'), (('error', 'ERROR'), 'O'), (('params', 'PARAMS'), 'O'),
(('modelpl', 'MODELPL'), 'O'), (('modeloiifull', 'MODELOIIFULL'), 'O'),
(('modeloii3726', 'MODELOII3726'), 'O'), (('modeloii3729', 'MODELOII3729'), 'O')])
```
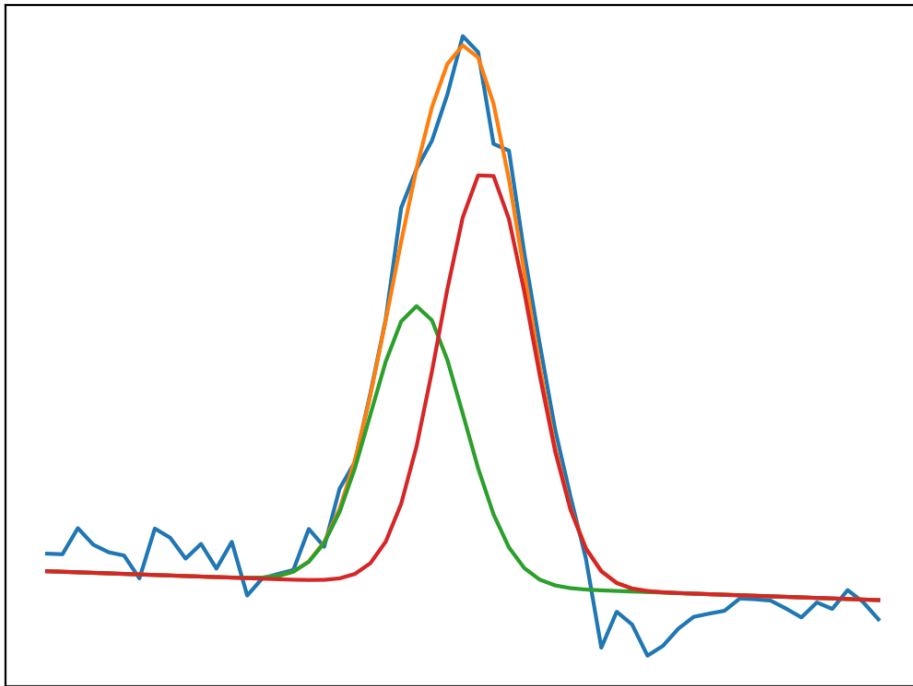

Figure 6: IDL fit of [OII] and individual components with original spectrum also displayed

8.2.3 sav_file.py Code and Output

Code:

```
#Python 3 program to read the data on the [OII] fit found from IDL

#to read an IDL ".sav" files we use a function from the scipy package
import scipy.io
data = scipy.io.readsav("oii/measurements_L1_blue_c_Keck.sav")

#the sav file is converted into a dicionary containting all the important information
print(data)
print('\n')

#the most importaint value obtained is the width expressed as sigma
print('Sigma value: ' + str(data['sigma_oiifullprofile']))
```

Output:

```
{'tobjname': b'L1_blue_c', 'objectname': b'L1_blue_c',
'sigma_oiifullprofile': 181.64165528214406, 'fwhm_oiifullprofile': 456.72106200585944,
'loiifull_fit': 3.926865070506631e+43, 'f_oiifullprofile': 100.07789063324614,
'f_unit': -13, 'oii3726_sigma_kms': 147.24414169844968,
'oii3729_sigma_kms': 147.13752316632517, 'oii3726_fwhm_kms': 346.7334563848868,
'oii3729_fwhm_kms': 346.4823889282682, 'p_1g3726':
array([8.73321917e+00, 3.72747256e+03, 1.83008739e+00]),
'p_1g3729': array([1.30828439e+01, 3.73017256e+03, 1.83008739e+00])}


Sigma value: 181.64165528214406
```

8.3 IDL Code

The IDL fitting program written by Dr. Daeseong Park. Modifications were made to the width of the fit and a large section containing the redshift data for each object was added. The code is displayed with long lines being broken up and indented.

```
;
; fitting [OII] doublet with a simple model
;

function Ez, z
common share, c,H0,Om,Ol
        return, 1D/sqrt(Om*(1D + z)^3D + Ol)
end

pro gaussian, X, P, YMOD
        YMOD = P[0]*exp(-(X-P[1])^2/(2D*P[2]^2))
end

function get_gaussian, X, P
        gaussian, X,P,YMOD
        return,YMOD
end




function calculate_centroid, X,Y
        return, total(X*Y,/double)/total(Y,/double)
end

function calculate_sigma, X,Y,line
common share, c,H0,Om,Ol
common resolCorrInfo, ResolutionCorrection, FWHM_inst_resol
        sigma = sqrt(total(X^2*Y,/double)/total(Y,/double) -
            calculate_centroid(X,Y)^2)
        if (ResolutionCorrection eq 'On') then sigma = sqrt(sigma^2 - (
            FWHM_inst_resol/(2D*sqrt(2D*alog(2D))))^2)
        sigma *= c/line ; in km/s
        return, sigma
end




function get_Lconti, X, P
        return,P[0] + P[1]*X
end


pro linear_gaussian, X, P, YMOD
        ;——————————————————————————————————————————————
        ;local linear comtinuum
        Lmodel = get_Lconti(X,P[0:1])
```

```
        ; single gaussian
        Gmodel = get_gaussian(X,P[2:4])

        YMOD = Lmodel + Gmodel
end


function get_doublegaussians, X, P
        comp1 = P[0]*exp(−(X−(3726.1D − P[1]))^2/(2D*P[2]^2))
        comp2 = P[3]*P[0]*exp(−(X−(3728.8D − P[1]))^2/(2D*P[2]^2))

        return, comp1 + comp2
end


pro linear_doublegaussian, X, P, YMOD
        ;——————————————————————————————————————
        ; local linear comtinuum
        Lmodel = get_Lconti(X,P[0:1])
        ; single gaussian
        GGmodel = get_doublegaussians(X,P[2:5])

        YMOD = Lmodel + GGmodel
end

function newtfunc, X
common bestfitp, pars,peakHeight2Gfit
        return, get_doublegaussians(X, pars[2:5]) − 0.5D*peakHeight2Gfit
end




##################################################################
; begin main procedure
##################################################################

PRO fit_keckspec_oii, eps=eps

;——————————————— extra jobs ————————————————————{{{
; journal ,'log_meanfit.txt'
; profiler ,/ reset & profiler ,/ system & profiler                    ;
start_mem = memory(/current)     ; to know how much memory required ;
itime = systime(/sec)            ; to compute elapsed time          ;
;——————————————————————————————————————————————}}}
common bestfitp, pars,peakHeight2Gfit
;**********************************************************
; constant
;**********************************************************
common share, c,H0,Om,Ol
; conversion for velocity in km/s
c = 299792.458D; (km/s)
```

```
H0 = 70D  ; (km/s  /Mpc)
Om = 0.3D
Ol = 0.7D

@angstrom.txt
;*********************************************************


;_____
;read data
;_____
filehead = COMMAND_LINE_ARGS()
read,'Enter object (ex. L100): ',filehead
filename = filehead + '_blue_c.fits'




objectName = strmid(filename,0,strpos(filename,'.fits'))

wavelength = get_wavelength('data/'+filename,1,/sil)
flux = mrdfits('data/'+filename,1,hdr)
;error = flux*0.1D ;mrdfits('data/'+filename,3)

ind = where(wavelength gt 5100 - 20 and wavelength lt 5100 + 20)
SNRat5100 = mean(flux[ind])/stddev(flux[ind])
error = flux/SNRat5100




;_____
common resolCorrInfo, ResolutionCorrection, FWHM_inst_resol
;_____
;objectName = ''

tobjName = objectName
datafrom = 'Keck'
ResolutionCorrection = 'Off' ;'On' or 'Off'
        FWHM_inst_resol = 3D ; ? in A ( ?? **please check this out!!** )
;_____

;        redshift_lit = redshift



journal,'log_fit_'+tobjName+'_'+datafrom+'.txt'


;( **please check this out!!** )
;;_____
;;Galactic extinction correction
;ebv = 0.248D - 0.188D ;E(B-V) = A_B - A_V (mag) from NED
;FM_UNRED, wavelength, flux, ebv, funred
;SNR = flux/error
```

```
;error = funred/SNR ;to preserve S/N
;flux = funred
;;————————————————————————————————————————————————————


;flux unit (in units of 10^−13)
f_unit = −13
flux *= 10D^(−f_unit)
error *= 10D^(−f_unit)




;————————————————————————————————————————————————————————


;————————————————————————————————————————————————————————
;adjust exact position of line peak to get redshift
;————————————————————————————————————————————————————————
if objectName eq 'L1_blue_c'   then redshift = 0.041D
if objectName eq 'L11_blue_c'   then redshift = 0.054D
if objectName eq 'L2_blue_c'   then redshift = 0.043D
if objectName eq 'L4_blue_c'   then redshift = 0.051D
if objectName eq 'L6_blue_c'   then redshift = 0.076D
if objectName eq 'L8_blue_c'   then redshift = 0.085D
if objectName eq 'L13_blue_c'   then redshift = 0.021D
if objectName eq 'L57_blue_c'   then redshift = 0.0304D
if objectName eq 'L58_blue_c'   then redshift = 0.0481D
if objectName eq 'L59_blue_c'   then redshift = 0.0483D
if objectName eq 'L60_blue_c'   then redshift = 0.0465D
if objectName eq 'L61_blue_c'   then redshift = 0.0532D
if objectName eq 'L62_blue_c'   then redshift = 0.0585D
if objectName eq 'L63_blue_c'   then redshift = 0.0409D
if objectName eq 'L15_blue_c'   then redshift = 0.038D
if objectName eq 'L32_blue_c'   then redshift = 0.0524D
if objectName eq 'L33_blue_c'   then redshift = 0.0475D
if objectName eq 'L34_blue_c'   then redshift = 0.055D
if objectName eq 'L35_blue_c'   then redshift = 0.0355D
if objectName eq 'L43_blue_c'   then redshift = 0.0229D
if objectName eq 'L46_blue_c'   then redshift = 0.047D
if objectName eq 'L49_blue_c'   then redshift = 0.0559D
if objectName eq 'L50_blue_c'   then redshift = 0.0501D
if objectName eq 'L51_blue_c'   then redshift = 0.0541D
if objectName eq 'L53_blue_c'   then redshift = 0.0558D
if objectName eq 'L54_blue_c'   then redshift = 0.0365D
if objectName eq 'L5_blue_c'   then redshift = 0.06D
if objectName eq 'L9_blue_c'   then redshift = 0.097D
if objectName eq 'L70_blue_c'   then redshift = 0.0458D
if objectName eq 'L71_blue_c'   then redshift = 0.0615D
if objectName eq 'L73_blue_c'   then redshift = 0.0805D
if objectName eq 'L74_blue_c'   then redshift = 0.0928D
if objectName eq 'L76_blue_c'   then redshift = 0.0847D
if objectName eq 'L77_blue_c'   then redshift = 0.0618D
if objectName eq 'L78_blue_c'   then redshift = 0.0715D
```

```
if  objectName  eq  'L79_blue_c '    then  redshift  =  0.0747D
if  objectName  eq  'L80_blue_c '    then  redshift  =  0.0801D
if  objectName  eq  'L81_blue_c '    then  redshift  =  0.0921D
if  objectName  eq  'L82_blue_c '    then  redshift  =  0.0962D
if  objectName  eq  'L83_blue_c '    then  redshift  =  0.0882D
if  objectName  eq  'L88_blue_c '    then  redshift  =  0.0841D
if  objectName  eq  'L91_blue_c '    then  redshift  =  0.0722D
if  objectName  eq  'L96_blue_c '    then  redshift  =  0.0805D
if  objectName  eq  'L99_blue_c '    then  redshift  =  0.0838D
if  objectName  eq  'L100_blue_c '    then  redshift  =  0.0992D
if  objectName  eq  'L102_blue_c '    then  redshift  =  0.0912D
if  objectName  eq  'L103_blue_c '    then  redshift  =  0.0821D
if  objectName  eq  'L106_blue_c '    then  redshift  =  0.0934D
if  objectName  eq  'L108_blue_c '    then  redshift  =  0.0907D
if  objectName  eq  'L109_blue_c '    then  redshift  =  0.0963D
if  objectName  eq  'L16_blue_c '    then  redshift  =  0.0409D
if  objectName  eq  'L10_blue_c '    then  redshift  =  0.054D
if  objectName  eq  'L21_blue_c '    then  redshift  =  0.0506D
if  objectName  eq  'L22_blue_c '    then  redshift  =  0.0392D
if  objectName  eq  'L26_blue_c '    then  redshift  =  0.0567D
if  objectName  eq  'L27_blue_c '    then  redshift  =  0.059D
if  objectName  eq  'L28_blue_c '    then  redshift  =  0.0218D
if  objectName  eq  'L156_blue_c '    then  redshift  =  0.0672D
if  objectName  eq  'L157_blue_c '    then  redshift  =  0.0631D
if  objectName  eq  'L162_blue_c '    then  redshift  =  0.0676D
if  objectName  eq  'L36_blue_c '    then  redshift  =  0.0493D
if  objectName  eq  'L39_blue_c '    then  redshift  =  0.0541D
if  objectName  eq  'L174_blue_c '    then  redshift  =  0.0612D
if  objectName  eq  'L187_blue_c '    then  redshift  =  0.063D
if  objectName  eq  'L18_blue_c '    then  redshift  =  0.0508D
if  objectName  eq  'L19_blue_c '    then  redshift  =  0.0586D
if  objectName  eq  'L138_blue_c '    then  redshift  =  0.0625D
if  objectName  eq  'L23_blue_c '    then  redshift  =  0.0332D
if  objectName  eq  'L24_blue_c '    then  redshift  =  0.0262D
if  objectName  eq  'L25_blue_c '    then  redshift  =  0.057D
if  objectName  eq  'L29_blue_c '    then  redshift  =  0.0469D
if  objectName  eq  'L30_blue_c '    then  redshift  =  0.0517D
if  objectName  eq  'L31_blue_c '    then  redshift  =  0.0377D
if  objectName  eq  'L155_blue_c '    then  redshift  =  0.0608D
if  objectName  eq  'L38_blue_c '    then  redshift  =  0.0599D
if  objectName  eq  'L14_blue_c '    then  redshift  =  0.044D
if  objectName  eq  'L40_blue_c '    then  redshift  =  0.0348D
if  objectName  eq  'L180_blue_c '    then  redshift  =  0.0688D
if  objectName  eq  'L196_blue_c '    then  redshift  =  0.0621D
if  objectName  eq  'L197_blue_c '    then  redshift  =  0.0633D
if  objectName  eq  'L202_blue_c '    then  redshift  =  0.0668D
if  objectName  eq  'L126_blue_c '    then  redshift  =  0.0655D
if  objectName  eq  'L20_blue_c '    then  redshift  =  0.0371D
if  objectName  eq  'L177_blue_c '    then  redshift  =  0.0629D
if  objectName  eq  'L41_blue_c '    then  redshift  =  0.0534D
if  objectName  eq  'L42_blue_c '    then  redshift  =  0.052D
if  objectName  eq  'L44_blue_c '    then  redshift  =  0.0308D
```

```
if objectName eq 'L47_blue_c '    then redshift = 0.0507D
if objectName eq 'L204_blue_c '   then redshift = 0.0604D
if objectName eq 'L213_blue_c '   then redshift = 0.0667D
if objectName eq 'L207_blue_c '   then redshift = 0.0626D
if objectName eq 'L209_blue_c '   then redshift = 0.0639D
if objectName eq 'L205_blue_c '   then redshift = 0.061D
if objectName eq 'L64_blue_c '    then redshift = 0.0253D
if objectName eq 'L114_blue_c '   then redshift = 0.0649D
if objectName eq 'L208_blue_c '   then redshift = 0.0635D
if objectName eq 'L130_blue_c '   then redshift = 0.0654D
if objectName eq 'L143_blue_c '   then redshift = 0.06D
if objectName eq 'L37_blue_c '    then redshift = 0.0421D
if objectName eq 'L45_blue_c '    then redshift = 0.0235D
if objectName eq 'L210_blue_c '   then redshift = 0.064D
if objectName eq 'L48_blue_c '    then redshift = 0.049D
if objectName eq 'L52_blue_c '    then redshift = 0.0538D
if objectName eq 'L55_blue_c '    then redshift = 0.0519D
if objectName eq 'L56_blue_c '    then redshift = 0.0358D
if objectName eq 'L214_blue_c '   then redshift = 0.0672D


;————————————————————————————————————————————————
; conversion to rest−frame
;————————————————————————————————————————————————
        wavelength /= (1D + redshift)
        flux *= (1D + redshift)^1
        error *= (1D + redshift)^1


;————————————————————————————————————————————————
;  basic setting on rest frame
;————————————————————————————————————————————————
OII3727 = 3727D
OIId3726 = 3726.1D
OIId3729 = 3728.8D



; model fit Limits
        lineLimit1 = 3715D
        lineLimit2 = 3740D

if objectName eq ' ' then lineLimit2 = 0D


;————————————————————————————————————————————————



;————————————————————————————————————————————————
```

```
;            ———————— spectrum  plot  &  fitting  ————————
;—————————————————————————————————————————————————

!x.margin = [6,1.5]
!y.margin = [3.5,1]
!x.ticklen = 0.03

xr = [3300,4100]
xr = [3650,3800]

;xr = [contiWinL1,contiWinR2]
yr = [−2,15]
idx = where(wavelength gt xr[0] and wavelength lt xr[1])
yr = [−median(flux[idx])*0.5,  median(flux[idx])*4]

if objectName eq 'L78_blue_c' or objectName eq 'L22_blue_m12' or
    objectName eq 'L37_blue_c'    then yr = [−median(flux[idx])*0.25,median
    (flux[idx])*2.5]
if objectName eq 'L30_blue_m5'    then yr = [−median(flux[idx])*0.5,median
    (flux[idx])*6]

;xr=[4990,5020]
;yr=[−2,12]

thick = 1
charsize = 3
charthick = 1

VSYM,24,/FILL

if (keyword_set(eps)) then begin
        thick = 4
        charsize = 2
        charthick = 3
        symsize = 2

        VSYM,4,thick=thick,/FILL

        SET_PLOT,'PS'

        ;!p.multi = [0,1,2]
         !x.margin = [12,1.5]
         !y.margin = [6,0.5]
         ;!P.FONT = −1
         !x.thick = 3
         !y.thick = 3
         !x.ticklen = 0.03
         !y.ticklen = 0.02

        DEVICE,/PORTRAIT,/COLOR,BITS=8,/ISOLatin1,$
                xsize=35,ysize=20,/ENCAPSUL  ;/Times/Helvetica/Palantino
```

```
                    DEVICE, FileName=datafrom+'_'+tobjName+'_OIIFit.eps'

endif else begin

            WINDOW,0 , xs=1200,ys=700;,/pixmap

endelse


            xh = wavelength
            yh = flux
            err = error
            delX = [xh[1:*], xh[n_elements(xh)−1]] − xh

;————————————————————————————————————————————
;0) raw spectrum
;————————————————————————————————————————————

            PLOT,xh,yh,/nodata,$
                        ;psym=−8,symsize=1,thick=thick , xsty=3,ysty=3,$
                        linesty=0,thick=thick , xsty=3,ysty=1,xtickinterval=50,$
                        charsize=charsize , charthick=charthick , xr=xr , yr=yr , yminor
                            =4,$
                        ; position =[0.08 ,0.15 ,0.65 ,0.975] ,/ normal ,$
                        tit='!6',$
                        xtit=textoidl('Rest  Wavelength  ('+angstrom+'_{ })') ,$
                        ytit=textoidl('f_{\lambda}  (10^{'+string(f_unit , format='(
                            I0) ')+'}  erg  s^{−1}  cm^{−2} '+angstrom+'^{−1})')

            oPLOT,xh,yh,linesty=0,thick=thick

            ;[OII]  3727
            PLOTs,[OII3727 , OII3727] , yr , linesty=2,thick=thick −1,color=cgCOLOR
                ('yellow')


;————————————————————————————————————————————
;1) [OII] emission line fit
;————————————————————————————————————————————
                        i_lineLimit = where(xh ge lineLimit1 and xh le lineLimit2
                            ,complement=i_lineLimit_out)

                        ;————————————————————————————————————————————

                        ; linear  continuum + a  Gaussian
                        ;————————————————————————————————————————————

                        X = xh[i_lineLimit]
                        Y = yh[i_lineLimit]
                        E = err[i_lineLimit]
                        WEIGHTS = 1D/E^2
```

```
##################################################################
P = [ mean(Y) ,  0D,  max(Y)−mean(Y) , OII3727 , stddev (X) ]
##################################################################


parinfo = replicate ({ value:0D, fixed :0 , limited :[0 ,0] ,
    limits :[0D,0D]} ,  n_elements (P))
parinfo [∗]. value = P ; starting  value
parinfo [[2 ,4]]. limited [0]  = 1 ; set  lower  bound  of  height
parinfo [[2 ,4]]. limits [0]   = 0D;  should  be  positive

YFIT = MPCURVEFIT(X,  Y,  WEIGHTS,  P,  FUNCTION_NAME=’
    linear_gaussian ’ ,PARINFO=parinfo , $
        /NODERIVATIVE,CHISQ=chisq ,DOF=dof ,STATUS=STA , /
            QUIET)



##################################################################
P = [P[0] ,P[1] , $
        P[2]∗0.5 ,0D,P[4] ,  1D]
##################################################################


parinfo = replicate ({ value:0D, fixed :0 , limited :[0 ,0] ,
    limits :[0D,0D]} ,  n_elements (P))
parinfo [∗]. value = P ; starting  value
parinfo [[2 ,4]]. limited [0]  = 1 ; set  lower  bound  of  height
parinfo [[2 ,4]]. limits [0]   = 0D;  should  be  positive

parinfo [5]. limited [∗]  = 1 ; set  lower  bound  of  height
parinfo [5]. limits [∗]   = [0.35D,1.5D];  should  be  positive

YFIT = MPCURVEFIT(X,  Y,  WEIGHTS,  P,  FUNCTION_NAME=’
    linear_doublegaussian ’ ,PARINFO=parinfo , $
        /NODERIVATIVE,CHISQ=chisq ,DOF=dof ,STATUS=STA , /
            QUIET)

        print
        print ,’————————————————————————————’
        print , ’CHISQ=’ ,STRTRIM( chisq ,2)
        print , ’DOF=’ ,STRTRIM( dof ,2)
        print , ’CHISQ/DOF=’ ,STRTRIM( chisq /dof ,2)
        print , ’STATUS=’ ,STRTRIM( sta ,2)
        print , ’P = ’ ,P

        ; fitted  line
        print ,’∗normal = ’ , strtrim (P[0] ,2)
        print ,’∗ slope = ’ , strtrim (P[1] ,2)

        print ,’∗ height = ’ , strtrim (P[2] ,2)
```

27

```
                    print ,'*deviation = ',strtrim(P[3],2)
                    print ,'*width = ',strtrim(P[4],2)
                    print ,'*ratio = ',strtrim(P[5],2)
                    print,'--------------------------------'
                    print


Lconti = get_Lconti(X,P[0:1])
comp1 = get_gaussian(X,[P[2],  OIId3726 - P[3],  P[4]])
comp2 = get_gaussian(X,[P[5]*P[2],  OIId3729 - P[3],  P[4]])

OIIfullprofile_X = X
OIIfullprofile_Y = comp1+comp2

OII3726_sigma_kms = P[4]*c/OIId3726 ; in km/s
OII3729_sigma_kms = P[4]*c/OIId3729 ; in km/s

OII3726_FWHM_kms = (2D*sqrt(2D*alog(2D)))*P[4]*c/OIId3726 ; in km
    /s
OII3729_FWHM_kms = (2D*sqrt(2D*alog(2D)))*P[4]*c/OIId3729 ; in km
    /s

        print ,'FWHM([OII]3726) from a gaussian fit = ', ((2D*sqrt
            (2D*alog(2D)))*P[4]) ,' A', OII3726_FWHM_kms,' km/s'
        print ,'FWHM([OII]3729) from a gaussian fit = ', ((2D*sqrt
            (2D*alog(2D)))*P[4]) ,' A', OII3729_FWHM_kms,' km/s'


oPLOT,X,Lconti ,linesty=0,thick=1,color=cgCOLOR('green')
        oPLOT,X, Lconti+comp1 ,linesty=0,thick=1,color=cgCOLOR('
            cyan')
        oPLOT,X, Lconti+comp2 ,linesty=0,thick=1,color=cgCOLOR('
            cyan')

oPLOT,x, yfit ,linesty=0,thick=2,color=cgCOLOR('red')
        oPLOT,X,comp1 ,linesty=0,thick=1,color=cgCOLOR('cyan')
        oPLOT,X,comp2 ,linesty=0,thick=1,color=cgCOLOR('cyan')


fullmodel_Lconti = get_Lconti(xh,P[0:1])
fullmodel_Lconti[i_lineLimit_out] =      0D

        P_1G3726 = [ P[2],        OIId3726 - P[3], P[4] ]
        P_1G3729 = [ P[5]*P[2], OIId3729 - P[3], P[4] ]

fullmodel_OII3726 = get_gaussian(xh,P_1G3726)
fullmodel_OII3726[i_lineLimit_out] =      0D

fullmodel_OII3729 = get_gaussian(xh,P_1G3729)
fullmodel_OII3729[i_lineLimit_out] =      0D

fullmodel_OIIfull = fullmodel_OII3726 + fullmodel_OII3729
```

```
AL_LEGEND, [ $
        string (P[0], format='(" linear  conti  normal = ",D8.2)') ,$
        string (P[1], format='(" linear  conti  slope   = ",D8.2)') ,$
        ;
        string (P[2], format='(" Gaussian  flux  scale  = ",D8.2)') ,$
        string (P[4], format='(" Gaussian  sigma  in A = ",D8.2)') ,$
        ;
        string (P[5], format='(" flux  ratio  3729/3726 = ",D8.2)') ,$
        ;
        TEXTOIDL('\chi^{2}_{\nu} = ')+string (chisq/dof, format='(
            D8.2)')] ,$
        box=0,/TOP,/LEFT, charsize=charsize −1.25, charthick=
            charthick ,$
        textcolors=[cgCOLOR('green') ,cgCOLOR('green') ,cgCOLOR('
            magenta') ,cgCOLOR('magenta') ,cgCOLOR('magenta') ,
            cgCOLOR('cyan') ]


; residual
oPLOT, X, Y−yfit ,  psym=10, thick =1, color=cgCOLOR('white')

; baseline
PLOTs, xr ,[0 ,0] , linesty =0, thick =1, color=cgCOLOR('Dark  Gray')


        print
        print ,'* sigma & FWHM calculations  for  emission  line
            profile '

        ;—————————————————————————————————————————

        ; fitted  line  profile  of  [OII]  line  ( full  line  profile )
        ;—————————————————————————————————————————

        ;——— [OII]  doublet  line  only  by  subtracting  local
            continuum
        ; OIIfullprofile_X
        ; OIIfullprofile_Y
        OIIfullprofile_delX = delX[i_lineLimit]
        oPLOT, OIIfullprofile_X , OIIfullprofile_Y , linesty =0, thick
            =2, color=cgCOLOR('red')

                ;—————————————————————————————————————

                ; line  flux  from  fit  profile
                ;—————————————————————————————————————

                f_OIIfullprofile = total(OIIfullprofile_Y *
                    OIIfullprofile_delX ,/ double)
```

```
;————————————————————————————
;sigma calculation
;————————————————————————————

print
print ,'sigma calculation for fit line profile'
centroid_OIIfullprofile = calculate_centroid(
    OIIfullprofile_X , OIIfullprofile_Y ) ; in
    angstrom
sigma_OIIfullprofile = calculate_sigma(
    OIIfullprofile_X , OIIfullprofile_Y ,OII3727) ;
    in km/s
print ,'sigma_OIIfullprofile = ' ,
    sigma_OIIfullprofile , ' km/s'


;————————————————————————————

;FWHM calculation
;————————————————————————————

print
print ,'FWHM calculation for fit line profile'

;real peak of fit
peakHeight2Gfit = max( OIIfullprofile_Y ,
    i_2Gfitpeak )
peakLocation2Gfit = OIIfullprofile_X[ i_2Gfitpeak ]

pars = P

i_temp = where(newtfunc(OIIfullprofile_X) gt 0D)
;initial guess
ig_left = OIIfullprofile_X[ i_temp [0]]
ig_right = OIIfullprofile_X[ i_temp [ n_elements(
    i_temp) −1]]

FWHMleft = NEWTON( ig_left ,'newtfunc',CHECK=check
    ,/DOUBLE)
FWHMright = NEWTON( ig_right ,'newtfunc',CHECK=
    check ,/DOUBLE)
print ,'FWHM_OIIfullprofile = lamR − lamL = ',
    FWHMright,' − ',FWHMleft,' = ',FWHMright −
    FWHMleft,format='(A,D7.2 ,A,D7.2 ,A,D7.2) '
PLOTs,[ FWHMleft,FWHMright ] ,[0.5D∗peakHeight2Gfit
    ,0.5D∗peakHeight2Gfit ] , thick=thick , linesty=0,
    color=cgCOLOR('orange')
PLOTs,[ FWHMleft,FWHMleft ] ,[0.5D∗peakHeight2Gfit
    ∗0.85 ,0.5D∗peakHeight2Gfit ∗1.15] , thick=thick ,
    linesty=0,color=cgCOLOR('orange')
PLOTs,[ FWHMright,FWHMright ] ,[0.5D∗peakHeight2Gfit
    ∗0.85 ,0.5D∗peakHeight2Gfit ∗1.15] , thick=thick ,
```

```
                              linesty=0,color=cgCOLOR('orange')

                     FWHM_OIIfullprofile_angs = abs(FWHMright −
                         FWHMleft) ; in angstrom
                     if (ResolutionCorrection eq 'On') then
                         FWHM_OIIfullprofile_angs = sqrt(
                         FWHM_OIIfullprofile_angs^2 − FWHM_inst_resol
                         ^2)
                     FWHM_OIIfullprofile = FWHM_OIIfullprofile_angs*c/
                         OII3727 ; in km/s
                     print ,'FWHM_OIIfullprofile = ',
                         FWHM_OIIfullprofile , ' km/s'




;####################################################################
 redshift_lit = redshift; literature value
;####################################################################
 ;comoving radial distance
 DC = c/H0*QPINT1D('Ez',0D,redshift_lit)
 print ,'luminosity distance check (Mpc) −> ',strtrim(DC*(1D +
     redshift_lit),2)

 ;luminosity distance
 DL = DC*(1D + redshift_lit)
 Mpc = 3.08568D+24;cm
 DL *= Mpc ; in cm


;————————————————————————————————————————————
;OII line luminosity
;————————————————————————————————————————————
 ; − from fit
 LOIIfull_fit = 4D*!DPI*DL^2*f_OIIfullprofile *10D^f_unit
 print ,'LOIIfull_fit = ',LOIIfull_fit , format='(A,E10.3)'




AL_LEGEND,[$
        objectName ,$
        string(redshift ,format='("z = ",D6.4)') ,$
        datafrom ,$
        ;
        TEXTOIDL('\sigma_{[O II]\lambda3727}^{2Gfit}')+string(
            sigma_OIIfullprofile ,format='(" = ",D7.2," km/s")') ,$
        TEXTOIDL('FWHM_{[O II]\lambda3727}^{2Gfit}')+string(
            FWHM_OIIfullprofile ,format='(" = ",D7.2," km/s")') ,$
        TEXTOIDL('L_{[O II]\lambda3727}^{2Gfit}')+string(
            LOIIfull_fit ,format='(" = ",E10.3," erg/s")')] ,$
        box=0,/TOP,/RIGHT,charsize=charsize −1.25,charthick=
            charthick ,spacing=3,pspacing=3,$
```

31

```
                    textcolors=[cgCOLOR('white'),cgCOLOR('white'),cgCOLOR('
                        white'),cgCOLOR('yellow'),cgCOLOR('yellow'),cgCOLOR('
                        yellow')]
                    ;pos=[0.65,0.995],/normal

        ;null = GET_KBRD()
        ;stop

if (keyword_set(eps)) then begin
        DEVICE,/Close_File
        CLEANPLOT,/Silent
        SET_PLOT,'X'
endif else begin
        WRITE_PNG,datafrom+'_'+tobjName+'_OIIFit.png',tvrd(true=1)
endelse

print,'————————————————————————————————————————',
print,'                        END of PLOT'
print,'————————————————————————————————————————',


;endfor



;——————————————— extra jobs ———————————————————————{{{
print
etime = (systime(/sec)−itime)/60D
if etime ge 1D then print,'*Elapsed time= ',strtrim(etime,2), ' min' $
        else print,'*Elapsed time= ',strtrim(etime*60D,2), ' sec'
print,'*Memory used= ',strtrim((memory(/highwater)−start_mem)/2D^20D,2),'
    MB'
;profiler,/report,filename='profiler_report.txt' & profiler,/reset
print
journal
;————————————————————————————————————————————}}}


SAVE,$
        tobjName,$
        objectName,$
        sigma_OIIfullprofile,$
        FWHM_OIIfullprofile,$
        LOIIfull_fit,$
        f_OIIfullprofile,$
        f_unit,$
        OII3726_sigma_kms,$
        OII3729_sigma_kms,$
        OII3726_FWHM_kms,$
        OII3729_FWHM_kms,$
        P_1G3726,$
        P_1G3729,$
```

```
                 filename='measurements_'+tobjName+'_'+datafrom+'.sav'


;————————————————————————————————————————————————
;SAVE all spectra (data and modelfit)
;————————————————————————————————————————————————
allmodels = { $
        ;————————————————————————————————————————————
        objectName : objectName ,$
        tobjName : tobjName ,$
        redshift : redshift ,$
        ;————————————————————————————————————————————
        ;raw
        wavelength : wavelength ,$
        flux : flux ,$
        error : error ,$
        ;————————————————————————————————————————————
        ;model fits
        params : P, $
        modelPL : fullmodel_Lconti , $
        ;————————————————————————————————————————————
        modelOIIfull : fullmodel_OIIfull , $
        modelOII3726 : fullmodel_OII3726 , $
        modelOII3729 : fullmodel_OII3729 $
        }

SAVE, allmodels ,$
        filename='allmodelfits_'+tobjName+'_'+datafrom+'.sav'


;stop ,' *** END of CODE *** '

END
```