

Diagramas de diagnóstico en galaxias observadas con espectroscopía de campo MaNGA del Sloan Digital Sky Survey

Renato Sánchez Loeza

25 de julio de 2024

INTRODUCCIÓN

Desde los inicios de la astronomía, nuestra visión del firmamento ha estado limitada por la capacidad tecnológica para observarlo. Pasando por etapas tempranas donde el ojo humano se apoyaba con los primeros telescopios ópticos para la definición de objetos celestes, tal como el holandés Hans Lippershey, quien construyó uno de los primeros telescopios en 1608, y Galileo Galilei, quien mejoró este modelo en 1610 Álvarez, 2015.

Posteriormente, la fotografía mejoró la precisión de las mediciones. El primero en hacerlo fue John William Draper, quien en 1840 utilizó un proceso de daguerrotipo¹ para capturar una imagen de la Luna. Cinco años más tarde, Foucault y Fizeau fotografiaron el Sol desde París Olsen, 2021. Pero los obstáculos se encuentran en la dificultad de compartir la información obtenida con base en este método.

Más recientemente, los CCD revolucionaron las observaciones astronómicas con una captura de entre el 90 % y el 95 % de la luz que llega a ellos, además de permitir la distribución de las fotografías obtenidas Observatory, s.f. A estos avances se incorpora el análisis del espectro electromagnético (ondas de radio, infrarrojo, ultravioleta, rayos X y Gamma) y la espectroscopia, siendo la segunda la encargada de extraer datos sobre la distribución de la energía en una región del plano con el método de la rendija, donde por esta se permite entrar la luz del segmento a estudiar para posteriormente estudiar su espectrograma.

En la búsqueda de métodos que combinen lo mejor de la fotometría y la espectroscopia, se han propuesto diversas técnicas. Entre ellas, destacan el barrido con rendijas, el interferómetro de Fabry-Pérot (Etalon) y la espectroscopia multiobjetos (MOS). El barrido con rendijas implica observar la luz de una fuente a través de una rendija estrecha. Esto permite analizar su espectro y obtener información sobre composición química, velocidad y temperatura, mientras que el Fabry-Pérot es una técnica que utiliza múltiples reflexiones entre dos superficies altamente reflectantes y cercanas entre sí HyperPhysics, s.f. Por otro lado, la MOS emplea máscaras con múltiples aperturas para capturar espectros de varios objetos simultáneamente O'Connell, s.f., a menudo utilizando una matriz de fibras para capturar parte del campo de visión. Estas fibras se encuentran en el plano focal del instrumento de imagen. Luego, las fibras se redirigen y se alinean en la ranura de entrada del espectrógrafo, dispersando la luz en un detector. Se emplea en estudios de objetos débiles, como cúmulos de galaxias. Ejemplo de estos sistemas son el *Low Resolution*

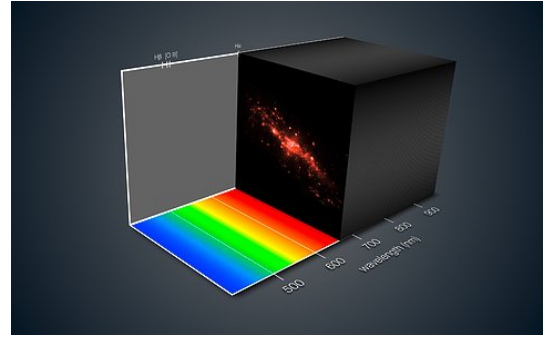


Fig. 1. Visualización de un cubo de datos resultante de una IFS. Crédito: ESO/MUSE consortium/R. Bacon/L. Calçada

Imaging Spectrometer (LRIS) en el Keck I y el *Visible MultiObject Spectrograph* (VIMOS)² en el VLT Oke et al., 1995.

ESPECTROSCOPIA INTEGRAL DE CAMPO

Sin embargo, el sistema que hoy en día se usa con más frecuencia es la espectroscopia integral de campo o IFS (Integral Field Spectrograph), la cual nos permite obtener un espectrograma de una región de 2 dimensiones del espacio. Un IFS genera un cubo de datos (Fig. 1), que contiene el campo de visión en 2 dimensiones y una dimensión espectral, formado por un bloque de imágenes de la misma región, cada una en una longitud de onda diferente Department of Astronomy, UW-Madison, s.f. Los astrónomos pueden medir, por ejemplo, el movimiento del gas en una galaxia lejana o las distancias a las diferentes galaxias detectadas en un campo de visión, utilizando la riqueza de la información de los espectrógrafos de campo integral.

En este proceso se encuentran las Unidades Integrales de Campo o IFU (Integral Field Units), los cuales son ampliamente utilizados para estudiar objetos de gran extensión, como cúmulos, galaxias o nebulosas, en una sola toma. Durante este proceso, la señal de cada una de las celdas se procesa en un espectrógrafo, el cual genera un espectro para cada uno de los píxeles, generando así un *spaxel*, y al final del proceso obtenemos un cubo de datos o *datacube*.

Métodos de IFS

Arreglo de lentes o Lenslet arrays

Utiliza imágenes de pupila para distribuir espectros cortos. La magnificación y espacio entre espectros es

¹Técnica fotográfica primitiva mediante la cual las imágenes captadas con la cámara oscura se fijan sobre una chapa metálica convenientemente preparada.

²VIMOS operates in three different modes: Imaging (IMG), Multi-Object Spectroscopy (MOS), and with Integral Field Unit (IFU).” webteam@eso.org, 1999-2008

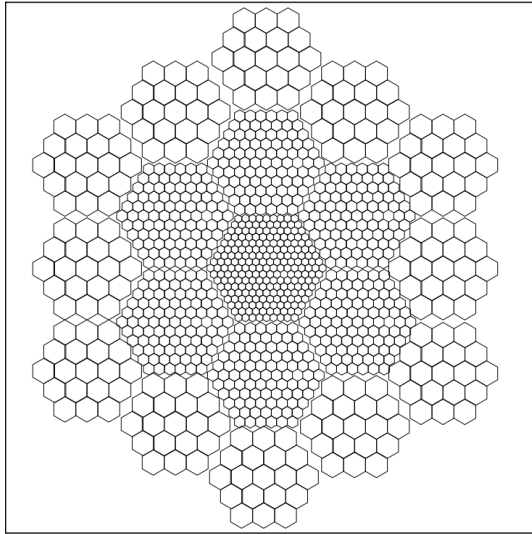


Fig. 2. Ejemplo de arreglo de lentes hexagonales con 3 tamaños de spaxels. Crédito: Schmoll et al., 2006

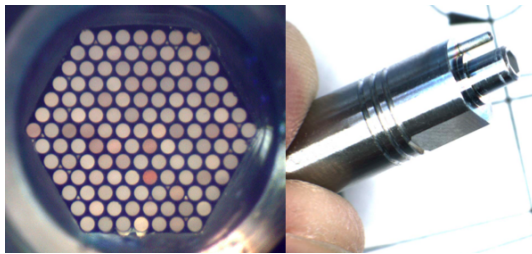


Fig. 3. A la izquierda se muestra la cara del primer prototipo de IFU de 127 fibras. A la derecha se muestra la carcasa de la férula que sostiene la IFU y permite enchufarla a la placa SDSS. “Bundle Ferrule Image”, 2014

importante para evitar mezcla de luz de diferentes longitudes de onda (Fig. 2) Allington-Smith, 2006.

Arreglo de fibras o Fibre arrays

Esta técnica consiste en acoplar una serie de fibras y un arreglo de microlentes³ que redirigen la luz a las rendijas para la generación de espectrogramas. Cada fibra recoge la luz de una ubicación específica en el campo de visión (Fig. 3). Además, requiere menos espacio muerto y, por lo tanto, tiene el potencial de aumentar la densidad de información Allington-Smith, 2006.

Cortador o Slicer

Un cortador de imágenes se compone generalmente de un conjunto de espejos cortadores ubicados en el plano de la imagen del telescopio y asociados a una fila de espejos pupilares y una fila de espejos de rendija Vivès y Prieto, 2005. Esta técnica busca dividir la imagen en tiras para después dispersarlas y formar un espectro (Fig. 4).

³ Los arreglos de microlentes son necesarios para las IFU acopladas a fibras. Adaptan el cono de luz entrante a la fibra, al mismo tiempo que optimizan el factor de llenado del conjunto de entrada. Schmoll et al., 2006

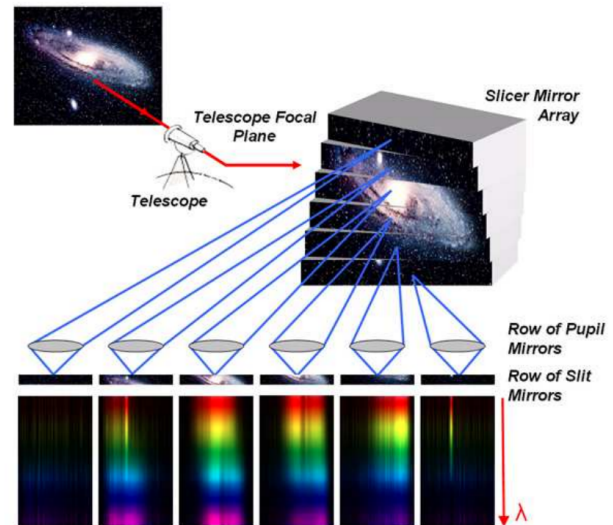


Fig. 4. Visualización de un IFU Slicer. Crédito: Vivès y Prieto, 2005

VISUALIZACIÓN Y ANÁLISIS

Si bien se ha avanzado significativamente en la mejora de nuestros métodos para observar las regiones espaciales, ahora es el momento de analizar dichos datos para obtener un aporte al conocimiento mucho más sofisticado. Hoy en día, podemos encontrar distintos software de procesamiento de datos de este tipo para modelos y usos muy específicos. Sin embargo, no podemos decir que exista una herramienta universal que nos brinde las herramientas suficientes para manipular datos de cualquier procedencia, con una interfaz intuitiva y una gran variedad de opciones para procesar los datos.

Soluciones existentes

Marvin

Actualmente se encuentra disponible Marvin Cherinka et al., 2019, un conjunto de herramientas que incluye un paquete de Python, una API y una aplicación web, diseñado para trabajar con los datos del proyecto MaNGA de SDSS-IV. Este software está equipado con datos generales de las galaxias, como las coordenadas celestes, la fecha de observación y una imagen de la galaxia observada, así como la visualización espectral en un spaxel específico (espectro observado y modelo de ajuste) con la opción de marcar los componentes químicos que refleja la espectroscopia. Incluye mapas de propiedades de la galaxia en cuestión, con datos como la velocidad estelar y el flujo de líneas de emisión (Fig. 5).

Starlink

También contamos con Starlink Berry et al., 2022, un proyecto de código abierto para el procesamiento de datos astronómicos. Aunque este proyecto terminó en 2005, el software sigue en desarrollo por la *East Asian Observatory*. Una desventaja de este software es que las tecnologías que utiliza están muy desactualizadas, como Fortran y C. Toda la referencia e instrucciones para la compilación del software se encuentran en el documento Starlink SSN/78 y

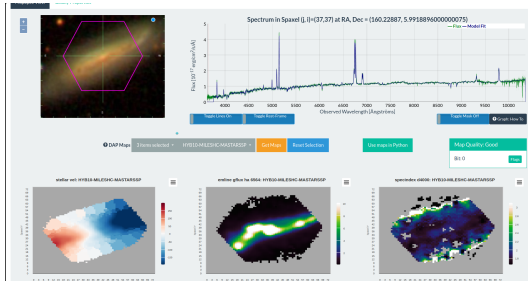


Fig. 5. Interfaz web de Marvin.

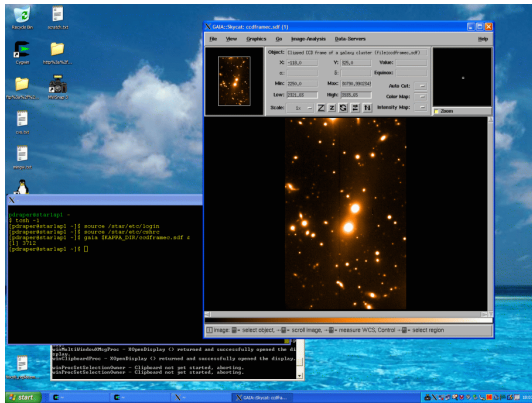


Fig. 6. Interfaz del software Starlink

en su repositorio de GitHub. Su interfaz, al igual que todo el sistema, está implementada en tecnologías obsoletas y difíciles de mantener hoy en día (Fig. 6).

METODOLOGIA

Una vez claro el marco de referencia en el que nos encontramos, se procede a iniciar el proceso de análisis de datos y exploración de tecnologías a utilizar. Para ello, nos dirigiremos en un principio a la página web de Marvin con el fin de hallar una galaxia que nos resulte interesante para comenzar a analizarla. Por ahora, tomaremos la galaxia manga 7495-6102 y descargaremos su cubo de datos **lineal**⁴ de la siguiente manera:

1. En el inicio de la página click en el botón **Go to SkyServer**
2. En la barra lateral derecha click al enlace **Explore**
3. Al final de la página click en **MaNGA observation(s)**
4. Click en **LIN Data Cube**

A continuación, comenzará la descarga de nuestro cubo de datos. Este archivo que se descargará será del tipo **.fits.gz**, lo que indica que nuestro archivo está comprimido, pero no será necesario descomprimirlo para el procesamiento.

Procesamiento con Python

Una vez descargado el cubo de datos, podremos analizarlo con la ayuda de **Python**, un lenguaje de programación interpretado multiplataforma, además de una librería llamada **Astropy**, enfocada en el análisis y procesamiento

⁴Optamos por el cubo de datos lineal ya que el modelo logarítmico requiere de más técnica para poder manipularse

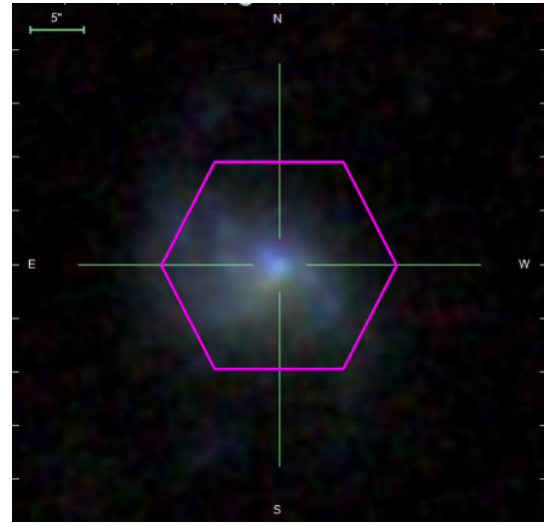


Fig. 7. Galaxia 7495-6102 - MaNGA - <https://magrathea.sdss.org/marvin/galaxy/7495-6102/>

astronómico. También utilizaremos **numpy** y **matplotlib** para la graficación de nuestros datos.

Implementación en código

Primero verificaremos cuáles son las dimensiones de nuestro cubo de datos.

```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt

fits_file = './manga-7495-6102-LINCUBE.fits.gz'
#ruta del archivo .fits
hdl = fits.open(fits_file) #abre el archivo
#fits
flux = hdl["FLUX"].data #lee los datos de flujo
print(flux.shape) #dimensiones del cubo de datos
#(espectral x espacial x espacial)
#resultado: (6732, 54, 54)
```

El resultado del código anterior indica que tenemos un cubo de datos de 54×5454×54 (espacial) = 2916 spaxels, con una profundidad de 6732 espectros, dando un total de 19,630,512 voxels.

```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
fits_file = './manga-7495-6102-LINCUBE.fits.gz'
#ruta del archivo .fits
hdl = fits.open(fits_file) #abre el archivo
#fits
flux = hdl["FLUX"].data #lee los datos de flujo
print(flux.shape) #dimensiones del cubo de datos
#(espectral x espacial x espacial)
#el nmero de espectros es (6732, 54, 54)

fig=plt.figure()
ax = fig.add_subplot(111)
ax.plot(flux[:,27,27])
ax.set_ymargin(0.1)
#ax.set_xlim(self.datacube.getData("WAVE")[0])
ax.xaxis.set_major_formatter(
    ticker.FuncFormatter( lambda x, pos:
        '{}{}'.format('',
            str(int(x)+self.datacube.getData("WAVE")[0]))))
```

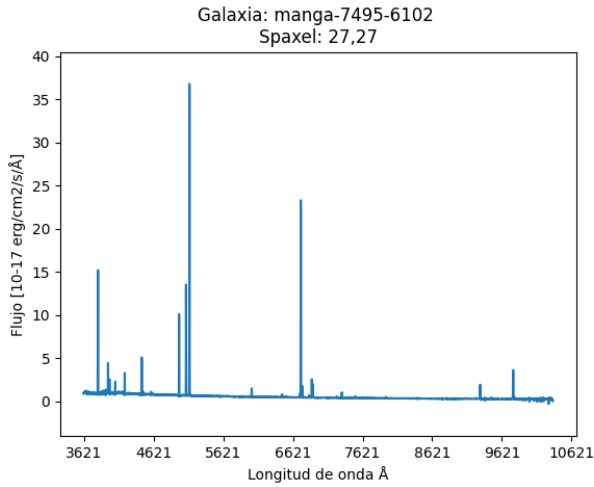


Fig. 8. Espectro en (:, 27, 27)

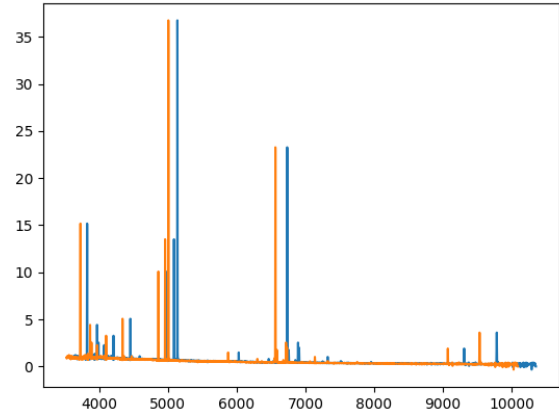


Fig. 10. Ajuste de longitud de onda por redshift

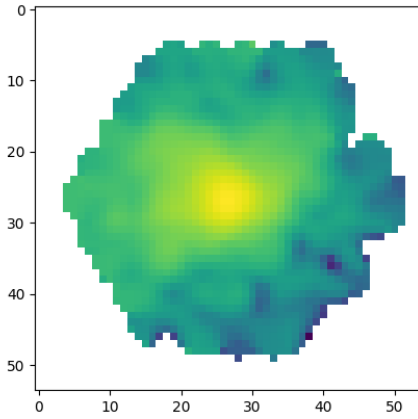


Fig. 9. Imagen en (900, :, :) con logaritmo base 10 aplicado

```
ax.set_xlabel("Longitud de onda Å")
ax.set_ylabel("Flujo [10-17 erg/cm2/s/Å]")
ax.set_title("Galaxia: manga-7495-6102 \nSpaxel:
27,27")

#extrayendo un corte del cubo en la longitud de
onda 900 a través de toda la imagen y
aplicando log10 - devuelve una imagen
fig=plt.figure()
bx=fig.add_subplot(111)
cx = bx.imshow(np.log10(flux[900,:,:]))

plt.show() #muestra los gráficos
```

Al ejecutar este código se desplegarán dos ventanas: una gráfica con el espectro del spaxel (espectro: todos, espacial: 27px, espacial: 27px) (**Fig. 8**), y una gráfica con la imagen de la galaxia en (espectro: 900, espacial: todos, espacial: todos) (**Fig. 9**).

Ajuste de longitud de onda en los datos

Dado que los datos de los espectros están situados en una longitud de onda específica y no comienzan en 0, debemos agregar al eje xx los datos que corresponden a las longitudes

de onda específicas para cada espectro. Estos datos se encuentran dentro del mismo archivo .fits en el apartado "WAVE".

Ajuste de redshift o corrimiento al rojo

Gracias al efecto Doppler, nuestros datos presentan un desfase en las longitudes de onda que nos son presentadas. Por lo tanto, debemos realizar otro ajuste a las longitudes de onda basándonos en la cantidad de corrimiento al rojo. Para ello, recurriremos al DAPall del SDSS, un resumen de las tablas finales de los resultados del MaNGA Data Analysis Pipeline (DAP), en el cual se encuentra la cantidad de redshift de nuestra galaxia en base al PLATEIFU.

Para ello se utilizará el siguiente código:

```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt
nombre_arch="./manga-7495-6102-LINCUBE.fits.gz"
hdu=fits.open(nombre_arch)
flujo = hdu["FLUX"].data
print("Dimensiones: ",flujo.shape)
hdu_catalog=fits.open(
"./redshift_catalog/dapall-v3_1_1-3.1.0.fits")
data_catalog=hdu_catalog[1].data
ix=np.where(data_catalog["plateifu"] ==
"7495-6102")
redshift=data_catalog[ "nsa_z"][ix][0]
longitud=hdu[ "WAVE"].data
wave=longitud/(1.0 + redshift)
fig=plt.figure()
ax=fig.add_subplot(111)
ax.plot(longitud, flujo[:,27,27])
ax.plot(wave,flujo[:,27,27])
plt.show()
```

Lo que nos da como resultado una gráfica con los dos espectros: el espectro observado (en azul) y el estado en reposo o rest frame (en naranja), ambos ya calibrados en la longitud de onda respectiva **Fig. 10**.

Para obtener una manipulación completa de los datos, me he basado en la información de la página del **Modelo de Datos del SDSS**, en la que se especifican los apartados de un archivo .fits.

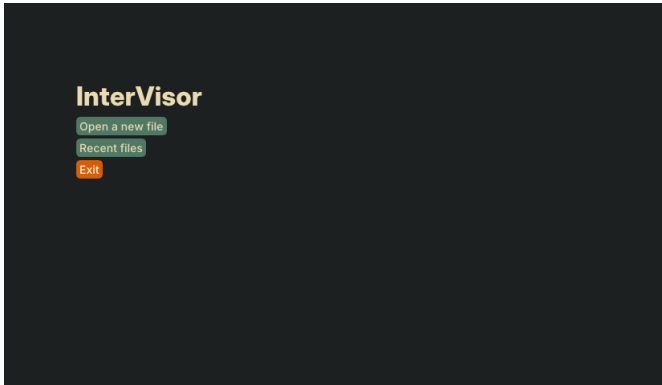


Fig. 11. Menu principal

Resultados

Al finalizar este proceso, logré comprender cómo se observan los objetos celestes usando espectroscopía integral de campo, incluyendo la estructura de un archivo proveniente del MaNGA / SDSS. Exploré temas de física, computación y análisis de datos.

Creación de interfaz

Con el fin de facilitar el uso de la implementación anterior, me propuse crear un entorno gráfico para esta. Entre las varias alternativas disponibles en Python, escogí **Dearpygui**, ya que es una librería gráfica multiplataforma, dinámica, acelerada por GPU y fácil de usar; un kit de herramientas de interfaz de usuario (GUI) para Python. Dado que en esta ocasión se manejarán grandes volúmenes de datos y requerimos una interfaz de alto rendimiento, cabe mencionar que esta librería también permite crear gráficas o "plots", así que para la sección del espectro me tomé la libertad de usar la alternativa de Dearpygui.

El programa de visualización creado, al que he denominado InterVisor, permite abrir un nuevo archivo `.fits`, `.gz` o `.fits.gz` desde el explorador de archivos con el botón **Open a new file**. Además, maneja un registro de los archivos abiertos con anterioridad; para ello está el botón **Recent files**. Para finalizar el programa, está la opción **Exit** (Fig. 11-13).

Al abrir algún archivo se procede a la siguiente pantalla en la cual se presentan las diferentes opciones para procesar el archivo **Fig. 14**.

Actualmente, se cuentan con los siguientes modos de procesamiento:

1. **Visor de espectro:** En este apartado se permite escoger el spaxel (i, j) y el rango de longitud de onda en el que se desea ver el espectro (**Fig. 15**).
2. **Extractor de imagen:** Después de ingresar una longitud de onda específica, se muestra la imagen del mapa. También está la opción de aplicar \log_{10} y \log_2 (**Fig. 16**).
3. **Más información:** Un apartado para observar más datos sobre el archivo `.fits` (**Fig. 17**).

Resultados

Como resultado, se obtiene un programa con las operaciones básicas de visualización de archivos `.fits`

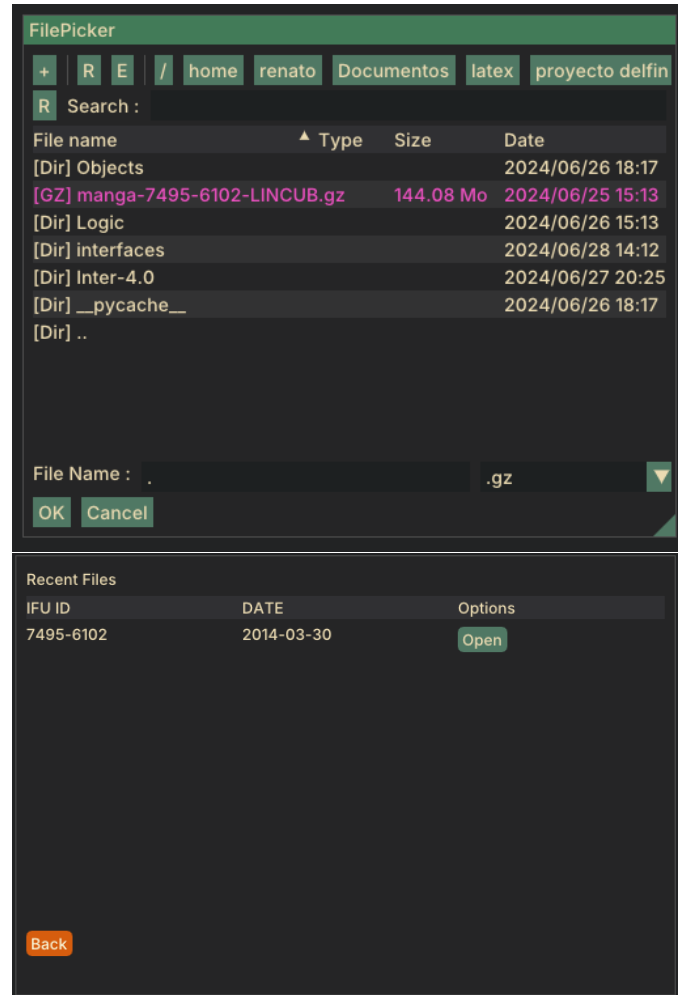


Fig. 12. Archivos Recientes y Explorador de archivos

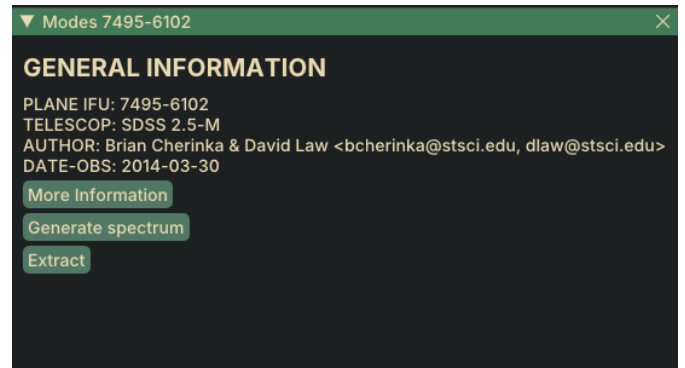


Fig. 13. Ventana de modos de procesamiento.

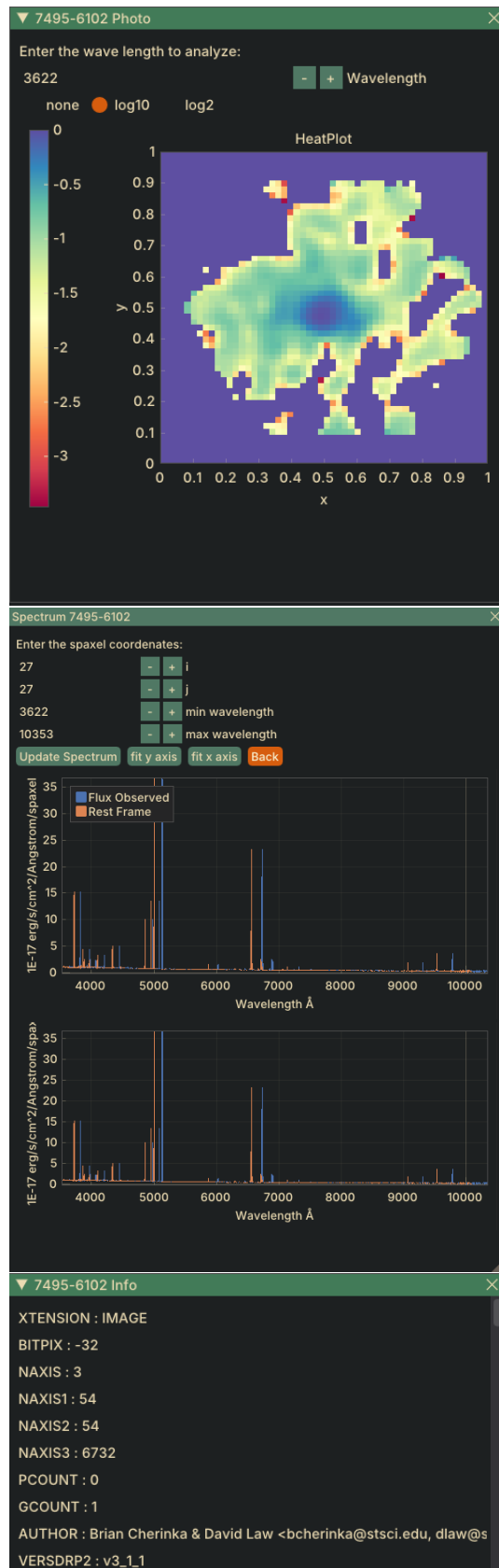


Fig. 14. Opciones de visualización en el programa con interfaz gráfica.

correspondientes al MaNGA. Utiliza las librerías Astropy, Numpy, Dearpygui y otras más. Además, permite la creación de rutinas de visualización consistentes y acoplables a otros métodos de análisis (visualización de espectros, mapas en una longitud de onda específica, información general del objeto observado).

OBSERVACIÓN DE LÍNEAS DE EMISIÓN

Lineas de emisión de interés

1. [OII] **3727** Oxígeno ionizado (singly ionized)
2. [NeIII] **3869** Neón ionizado (doubly ionized)
3. H δ **4102** Hidrógeno delta
4. H γ **4340** Hidrógeno gamma
5. [OIII] **4363** Oxígeno ionizado (doubly ionized)
6. [OIII] **4959** Oxígeno ionizado (doubly ionized)
7. [OIII] **5007** Oxígeno ionizado (doubly ionized)
8. HeI **5876** Helio neutro
9. [OI] **6300** Oxígeno neutro
10. [OI] **6364** Oxígeno neutro
11. [NII] **6584** Nitrógeno ionizado (singly ionized)
12. HeI **6678** Helio neutro
13. [SII] **6717** Azufre ionizado (singly ionized)
14. [SII] **6731** Azufre ionizado (singly ionized)
15. HeI **7065** Helio neutro
16. [ArIII] **7136** Argón ionizado (doubly ionized)
17. [OII] **7320** Oxígeno ionizado (singly ionized)
18. [ArIII] **7751** Argón ionizado (doubly ionized)

En base a estas líneas de emisión, se pueden obtener múltiples tipos de mapas:

- **Mapa de intensidad:** Área bajo la curva gaussiana ajustada a los datos.
- **Mapa cinético:** Movimiento de rotación de la galaxia.
- **Mapa de temperatura:** Diferencias entre los anchos de las líneas de emisión.
- **Estrellas antiguas:** Varianza entre las alturas del continuo.
- **Razón iónica:** [SII]6717/[SII]6731[SII]6717/[SII]6731.

Mapa de Intensidad

Aclaraciones

Antes de analizar los bloques de datos, he decidido realizar las siguientes operaciones:

1. **Discriminación de bloques de datos fuera del área observada:** Usando la función `numpy.all()`, se detecta si un arreglo de datos está lleno de ceros o si contiene valores infinitos no válidos para nuestro proceso. En estos casos, se procede a insertar `None` en la posición correspondiente en el arreglo bidimensional.
2. **Conversión de datos post-redshift:** Tras aplicar el redshift o corrimiento al rojo sobre el conjunto de datos de la longitud de onda del espectro, se procede a convertir estos datos a valores `int` para una mejor manipulación.

Cuando analizamos una línea de emisión debemos de examinar toda la curva de emisión ya que hasta ahora hemos visto y analizado solo un corte en el cubo de datos a lo largo del espectro. Por ejemplo tenemos la línea de emisión de la (Fig. 18) donde podemos apreciar que la formación de la línea de emisión en el fondo esta formada por un segmento del cubo de datos y no solo por una longitud de onda en específico.

Objetivo: Hallar el area bajo la curva gaussiana con ajuste a la línea de emisión n .

para iniciar hay que definir el método y nomenclatura que ocuparemos:

Extracción de datos de la línea de emisión

1. Definir el margen m usado para extraer el intervalo de datos referente a la línea de emisión n , que comprende `flujo[n - m : n + m]`.
2. Definir un submargen p lateral para calcular la media del continuo a los costados de la línea de emisión.
3. Calcular el FWHM (Full Width at Half Maximum) de la línea de emisión, lo que proporciona un sigma más preciso para la generación de una campana gaussiana.
4. Ajustar una gaussiana a los datos de la línea de emisión.
5. Integrar el área bajo la curva gaussiana (sin incluir el área entre 0 y la media del continuo) y almacenar este dato en una matriz bidimensional.
6. Generar la imagen con los datos de las líneas de emisión n .

Este proceso debe realizarse para cada píxel en la imagen del cubo de datos en cada una de las líneas de emisión que nos interesa analizar. De esta manera, obtendremos un mapa de líneas de emisión. Para generar la gaussiana, utilizamos la siguiente fórmula:

$$f(x) = a \cdot \exp\left(-\frac{(x - x_0)^2}{2\sigma^2}\right)$$

Además, usamos la función `curve_fit` de la librería `scipy.optimize` para ajustar la gaussiana a los datos. El código utilizado se encuentra en mi repositorio de [GitHub](#).

Una vez claro el proceso para un píxel específico en el cubo de datos, podemos proceder a automatizar este proceso

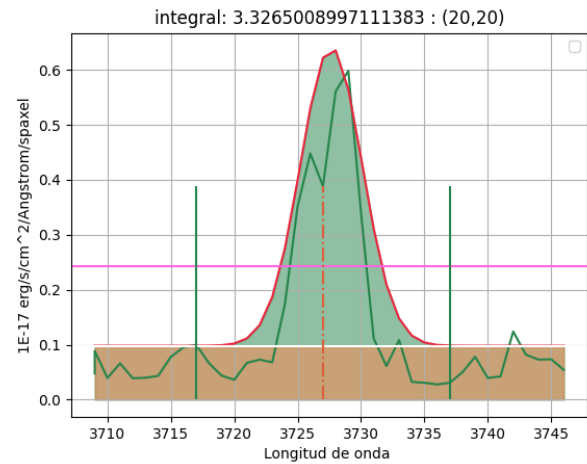


Fig. 15. Visualización del proceso de extracción de la intensidad de líneas de emisión.

para cada uno de los píxeles en el cubo, y almacenar los datos en un arreglo bidimensional para finalmente generar una imagen (mapa) de líneas de emisión.

Resultados

Durante el proceso de generación de estos mapas de líneas de emisión, encontré dificultades para **calcular el FWHM**. En el momento de ajustar la campana de Gauss, me resultó complicada la forma en que debía obtener dicho valor. Después de investigar, encontré una explicación clara en esta [página](#) (Suzuki, 2023).

De esta manera, obtuve resultados como los siguientes: imágenes con una gran cantidad de píxeles calculados y una barra de medición al costado. Además, en la Fig. 18, el área verde representa el valor de la integración de la campana de Gauss, el área naranja representa la zona que no debemos tomar en cuenta en este proceso, las barras laterales se interpretan como los submárgenes a partir de los cuales se calcula la media del continuo, y el intervalo central es la zona en la que se halla la información de la línea de emisión.

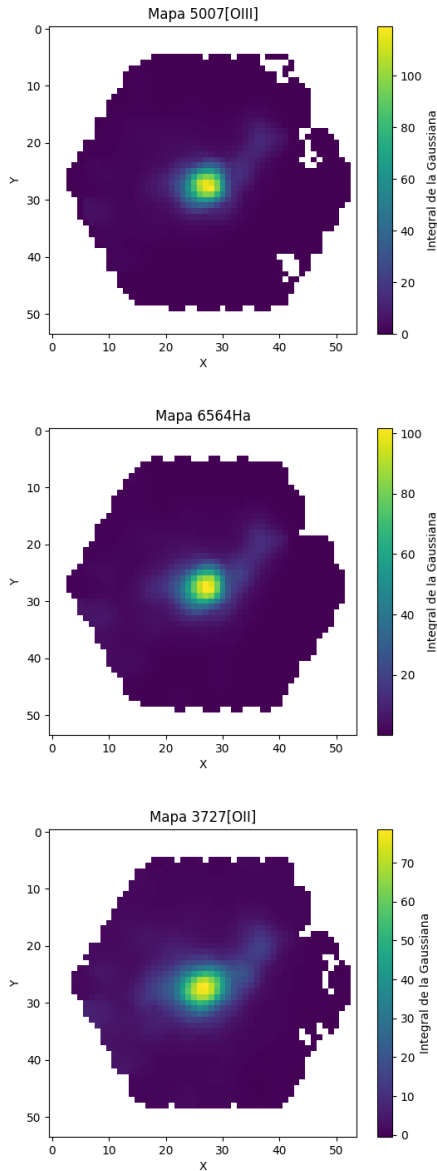


Fig. 16. PLATEIFU: 7495-6102

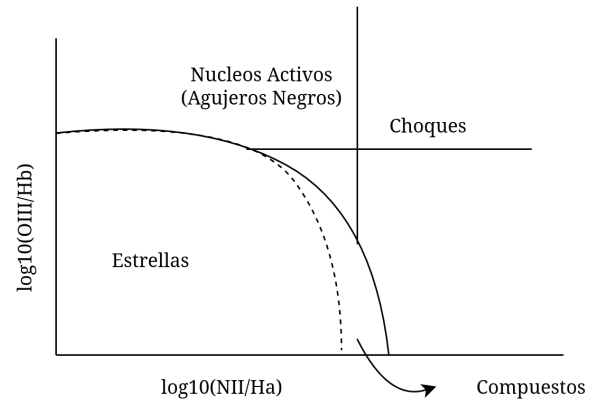


Fig. 17. Ejemplo visual de un diagrama de diagnóstico.

PROYECTO FINAL

Diagramas de diagnóstico / BPT

Los diagramas de diagnóstico ayudan a distinguir entre estrellas, núcleos activos de galaxias, choques, etc.

Como último proceso, se planea la generación de diagramas de diagnóstico, los cuales proporcionarán información sobre el objeto observado. Para esto, se debe realizar el siguiente proceso:

1. Una vez obtenidas las líneas de emisión y almacenadas en un archivo .fits, procederemos con la integración de estos datos⁵ de las líneas de emisión [NII] 6548, Ha 6564, Hb 4861, [OIII] 5007. Para la integración de la de los datos creé una función que verifica si la misma locación en ambos flujos contiene datos o no. De ser cierto, se realiza la suma de los datos como se puede observar.

```
def integrar(flujo):
    suma = 0
    for i in flujo:
        if np.all(flujo == np.nan):
            continue
        for k in i:
            if not np.isnan(k):
                suma += k
    return suma

hdl_NII = open("fits7495-6102_[NII]_6548_.fits")
hdl_Ha = open("fits7495-6102_Ha_6564_.fits")
hdl_Hb = open("fits7495-6102_Hb_4861_.fits")
hdl_OIII = open("fits7495-6102_[OIII]_5007_.fits")

integradoNII = integrar(flujoNII)
integradoHb = integrar(flujoHb)
integradoHa = integrar(flujoHa)
integradoOIII = integrar(flujoOIII)
```

2. A continuación, tomamos estos datos y realizamos las operaciones $\log_{10}(\text{NII}/\text{Ha})$ y $\log_{10}(\text{OIII}/\text{Hb})$.

⁵Con integrar nos referimos a sumar todos los datos que conforman la imagen de la línea de emisión, pasando de una matriz bidimensional a un solo valor

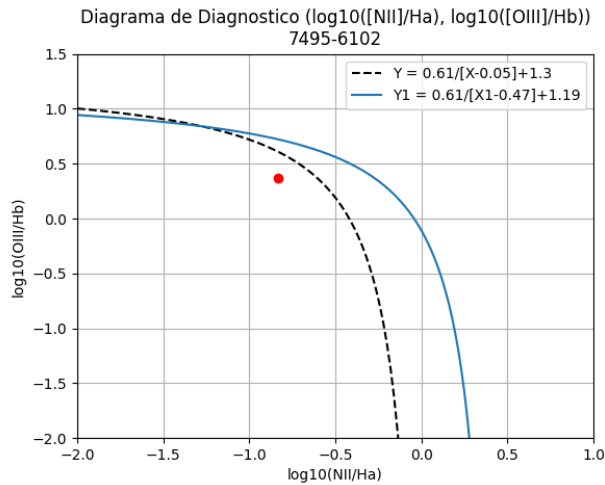


Fig. 18. Diagrama de diagnóstico: 7495-6102

```
NIIdivHa = integradoNII / integradoHa
OIIIdivHb = integradoOIII / integradoHb
Log10NIIdivHa = np.log10(integradoNII /
    integradoHa)
Log10OIIIdivHb = np.log10(integradoOIII /
    integradoHb)
```

3. Para determinar si nuestra galaxia contiene formación estelar, usaré las funciones (4) $0.61/(\log_{10}(\text{NII}/\text{Ha}) - 0.05) + 1.3$ (Kauffmann+03 line) y (5) $0.61/(\log_{10}(\text{NII}/\text{Ha}) - 0.47) + 1.19$ (Kewley+01 line) del artículo Kauffmann et al., 2006, las cuales graficaré sobre el plano al mismo tiempo que los datos obtenidos anteriormente.

```
X = np.linspace(-2, 0.049, 100)
Y = 0.61 / (X - 0.05) + 1.3
ax.plot(X, Y, c="k", linestyle="--", label="Y =
    0.61/[X-0.05]+1.3")

X1 = np.linspace(-2, 0.46, 1900)
Y1 = 0.61 / (X1 - 0.47) + 1.19
ax.plot(X1, Y1, label="Y1 =
    0.61/[X1-0.47]+1.19")

ax.plot(
    Log10NIIdivHa,
    Log10OIIIdivHb,
    "ro",
)
```

Este proceso lo he repetido con los cubos de datos manga-8456-6101, manga-8988-6104 y

DEFINICIONES

pixel

Un píxel se refiere a un píxel físico del detector que es leído por una serie de dispositivos electrónicos.

spaxel

Un spaxel (píxel espacial) se refiere a un elemento espacial de un cubo de datos reconstruido.

voxel

Un voxel (píxel volumétrico) se refiere a un elemento de volumen (espacial × espacial × espectral) de un cubo de datos.

redshift

El desplazamiento al rojo, como lo indica su nombre, ocurre cuando las ondas de luz se estiran debido al efecto Doppler y tienden hacia la región roja del espectro.

FWHM

La anchura completa a la mitad de la altura (FWHM, por sus siglas en inglés) es una medida del ancho de una distribución de intensidad en el punto donde la intensidad es máxima. A menudo se utiliza para determinar el ancho de una línea espectral o el rango de longitudes de onda en luz monocromática.

pársec

Unidad de medida de distancia utilizada en astronomía, equivalente a aproximadamente 3.26 años luz.

SDSS

El *Sloan Digital Sky Survey* es uno de los proyectos de investigación astronómica más grandes jamás emprendidos. Su objetivo es expandir el conocimiento sobre la evolución y estructura del espacio, y la formación de estrellas y galaxias.

MaNGA

Mapping Nearby Galaxies at APO es un estudio espectroscópico de campo integral de galaxias dentro del Sloan Digital Sky Survey de cuarta generación (SDSS-IV) Weijmans, 2015.

Unidad Astronómica / au

La unidad astronómica de distancia es una unidad de longitud utilizada en astronomía para las dimensiones típicas del Sistema Solar y se define como 1 au igual a exactamente 149,597,870.7 metros.

S/N

La relación señal-ruido (SNR, Signal-to-Noise Ratio) mide qué tan bien se mide un objeto. Valores típicos:

S/N = 2-3: objeto apenas detectado

S/N = 10: podemos comenzar a hacer mediciones

S/N = 100: medición excelente.

Archivos .fits

FITS (Flexible Image Transport System) es el formato de datos más utilizado en astronomía para transportar, analizar y archivar datos. FITS no es solo un formato de imagen (como JPG o GIF), sino que está diseñado principalmente para almacenar conjuntos de datos científicos que constan de matrices multidimensionales (imágenes) y tablas bidimensionales organizadas en filas y columnas de información.

Diagramas BPT

Los diagramas BPT (llamados así por "Baldwin, Phillips Terlevich") son un conjunto de diagramas de líneas de emisión nebular utilizados para distinguir el mecanismo de ionización del gas nebular. La versión más famosa muestra $[\text{NII}]\lambda 6584/\text{H}\alpha$ versus $[\text{OIII}]\lambda 5007/\text{H}\beta$ Baldwin et al., 1981.

REFERENCIAS

- Baldwin, J. A., Phillips, M. M., & Terlevich, R. (1981). Classification parameters for the emission-line spectra of extragalactic objects., 93, 5-19. <https://doi.org/10.1086/130766>
- Oke, J. B., Cohen, J. G., Carr, M., Cromer, J., Dingizian, A., Harris, F. H., Labrecque, S., Lucinio, R., Schaal, W., Epps, H., & Miller, J. (1995). The Keck Low-Resolution Imaging Spectrometer. *Publications Astronomical Society Pacific*, 107, 375. <https://doi.org/10.1086/133562>
- Vivès, S., & Prieto, E. (2005). An original image slicer designed for Integral Field Spectroscopy with NIRSpec/JSWT. *Physics > Optics*. <https://arxiv.org/abs/physics/0512002>
- Allington-Smith, J. (2006). Basic principles of integral field spectroscopy. *New Astronomy Reviews*, 50(4-5), 244-251. <https://doi.org/10.1016/j.newar.2006.02.024>
- Kauffmann, G., Heckman, T. M., De Lucia, G., Brinchmann, J., Charlot, S., Tremonti, C., White, S. D. M., & Brinkmann, J. (2006). Stellar masses and star formation histories for 10^5 galaxies from the Sloan Digital Sky Survey., 372, 961-987. <https://doi.org/10.1111/j.1365-2966.2006.10848.x>
- Schmoll, J., Robertson, D., & Ryder, D. (2006). Design and manufacture of micro-optical arrays using 3D diamond machining techniques. <https://arxiv.org/pdf/astro-ph/0606145>
- Bundle Ferrule Image [Image from the Sloan Digital Sky Survey (SDSS)]. (2014). https://www.sdss4.org/wp-content/uploads/2014/11/Bundle_ferrule_new.png
- Álvarez, E. J. (2015, octubre). ¿Quién inventó el telescopio? <https://www.iac.es/es/blog/vialactea/2015/10/quien-invento-el-telescopio>
- Weijmans, A.-M. (2015). MaNGA: Mapping Nearby Galaxies at Apache Point Observatory. *arXiv preprint arXiv:1508.04314*. <https://arxiv.org/abs/1508.04314>
- Cherinka, B., Andrews, B. H., Sánchez-Gallego, J., Brownstein, J., Argudo-Fernández, M., Blanton, M., Bundy, K., Jones, A., Masters, K., Law, D. R., Rowlands, K., Weijmans, A.-M., Westfall, K., & Yan, R. (2019). Marvin: A Tool Kit for Streamlined Access and Visualization of the SDSS-IV MaNGA Data Set., 158(2), Artículo 74, 74. <https://doi.org/10.3847/1538-3881/ab2634>
- Olsen, C. (2021, abril). *The Birth of Astrophotography - Vatican Observatory*. <https://www.vaticanobservatory.org/sacred-space-astronomy/the-birth-of-astrophotography/>
- Berry, D., Graves, S., Bell, G. S., Currie, M., Draper, P., & Jenness, T. (2022). Starlink - the original and best [n.d.]. *NASA/ADS*. <https://ui.adsabs.harvard.edu/abs/2022ASPC..532..559B/abstract>
- Suzuki, T. (2023). What is Full Width at Half Maximum (FWHM)? [Updated March 28, 2024]. <https://www.pulstec.net/what-is-full-width-at-half-maximum/>
- Department of Astronomy, UW–Madison. (s.f.). Integral Field Spectroscopy [Consultado en junio de 2024].
- HyperPhysics, R. N., M. Olmo. (s.f.). Fabry-Perot Interferometer [n.d.]. <http://hyperphysics.phy-astr.gsu.edu/hbasees/phyopt/fabry.html>
- Observatory, L. C. (s.f.). Astronomical cameras [n.d.]. <https://lco.global/spacebook/telescopes/cameras/>
- O’Connell, R. (s.f.). ASTR 511, Lecture notes [n.d.]. <https://rwoconne.github.io/rwoclass/ast511/lec10-f03.html>
- webteam@eso.org. (1999-2008). ESO - VIMOS [n.d.]. *ESO*. <https://www.eso.org/sci/facilities/paranal/decommissioned/vimos.html>