



# PROGRAMACIÓN EN PYTHON

Desde los principios básicos  
hasta la visualización de datos



## Proyecto del Módulo 4 Astronomía y Exoplanetas

### Objetivo general

El objetivo de este proyecto es crear una aplicación para analizar las características de todos los exoplanetas descubiertos. En el desarrollo de esta aplicación pondrás en práctica los conceptos del módulo 4.

### Objetivos específicos

En el desarrollo de este proyecto mostrarás tu capacidad para:

1. Implementar algoritmos para construir y recorrer matrices.
2. Utilizar las librerías pandas y matplotlib, así como consultar los sitios web oficiales donde se encuentra toda la documentación.
3. Descomponer un problema en subproblemas e implementar las funciones que los resuelven.

### Instrucciones generales

La descripción de la aplicación te permitirá conocer el alcance, las funcionalidades esperadas y lo que debes realizar en este proyecto. Las tres actividades descritas posteriormente, detallan lo que debes realizar. Al final, se especifica su forma de entrega.

**Ten en cuenta que,** a lo largo de la descripción de la aplicación encontrarás secciones con el título "**ATENCIÓN**" con indicaciones para conseguir que el resultado de la valoración y calificación de tu proyecto corresponda con lo esperado. Es importante que las sigas cuidadosamente.

Antes de empezar, te sugerimos leer con atención todo el proyecto. Mientras lo lees, trata de reconocer los conceptos del módulo que tendrás que poner en práctica.

Recuerda que este proyecto debe realizarse de forma **completamente individual**.

### Descripción de la aplicación

La astronomía es la ciencia que se ocupa del estudio de los cuerpos celestes del universo cómo: los planetas y sus satélites, los cometas, los meteoritos, las estrellas, la materia interestelar, los sistemas de materia oscura, las galaxias y los cúmulos de galaxias. La mayoría de la información usada por los astrónomos es recogida por la observación remota, aunque en algunos casos se ha conseguido reproducir en laboratorio la ejecución de fenómenos celestes. También es una de las pocas ciencias en las que los aficionados aún pueden desempeñar un papel activo, especialmente

en el descubrimiento y el seguimiento de fenómenos como curvas de luz de estrellas variables, descubrimiento de asteroides y cometas, entre otros.<sup>1</sup>

En este proyecto trabajarás con los datos de todos los planetas extrasolares confirmados (llamados exoplanetas), descubiertos entre 1988 y 2018 (más de 3000). Esta información se encuentra recopilada en la base de datos llamada Open Exoplanet Catalogue (<https://www.kaggle.com/mrisdal/open-exoplanet-catalogue>). En el archivo “exoplanetas.csv” encontrarás una versión simplificada de los datos originales. Suprimimos de este catálogo los planetas que no han sido confirmados, eliminamos algunas columnas y modificamos las clasificaciones de valores numéricos por valores categóricos.

Trabajando con este conjunto de datos puedes convertirte en un explorador espacial, analizando las características de todos los exoplanetas descubiertos. Los campos de los datos incluyen atributos de estrellas y de planetas, métodos de descubrimiento y por supuesto, la fecha de los descubrimientos.

**Te deseamos la mejor de las suertes en esta exploración espacial!!!**

Tu aplicación debe tener las siguientes partes:

## Parte 1: Cargar los datos

### Requerimiento 0: Cargar datos

Lo primero que debes hacer es permitir que se carguen los datos desde un archivo csv a un DataFrame. Le debes preguntar al usuario el nombre del archivo, es decir, **la función que implemente este requerimiento debe recibir como parámetro el nombre del archivo y debe retornar un DataFrame.**

Las columnas del archivo y sus significados son las siguientes:

- NOMBRE: nombre del planeta que se descubrió.
- MASA: masa del planeta.
- DESCUBRIMIENTO: año de descubrimiento del planeta.
- ACTUALIZACION: última vez que se actualizó la información del planeta.
- ESTADO\_PUBLICACION: medio en el cual se publicó el descubrimiento.
- TIPO\_DETECCION: tipo de detección que se utilizó para descubrir el planeta.
- RA: hace referencia a la *Ascensión Recta*, que es una de las coordenadas del planeta visto desde la tierra equivalente al ángulo con respecto a un punto fijo sobre el ecuador celeste; es similar a la longitud.
- DEC: hace referencia a la *Declinación*, que es la otra coordenada del planeta visto desde la tierra, equivalente a la altura sobre el ecuador celeste; es similar a la latitud.
- DISTANCIA\_ESTRELLA: distancia a la estrella más cercana.
- MASA\_ESTRELLA: masa de la estrella más cercana.

**ATENCIÓN:** Cuando estés estudiando el problema y el archivo, recuerda que las funciones `describe()`, `unique()` y `value_counts()` pueden ser de mucha utilidad.

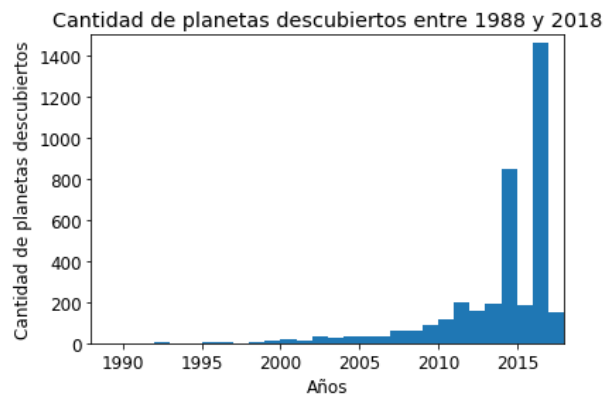
---

<sup>1</sup> *Astronomía*. Wikipedia.com. Tomado de: <https://es.wikipedia.org/wiki/Astronom%C3%ADa>

## Parte 2: Analizar el descubrimiento de los planetas

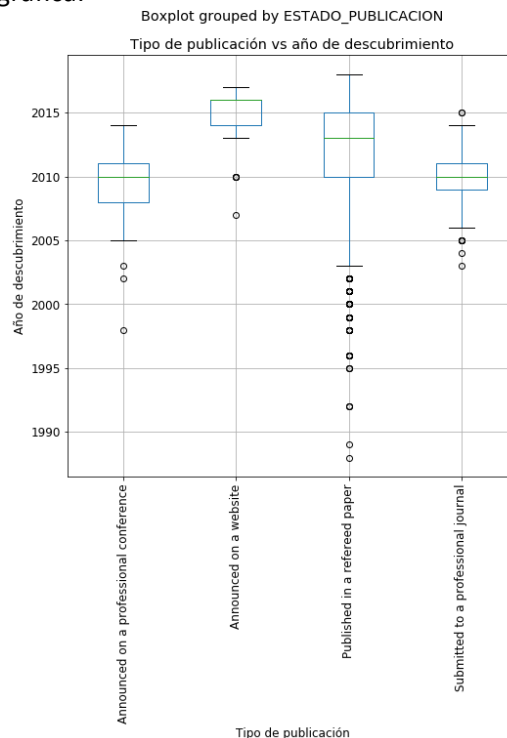
### Requerimiento 1: Número de descubrimientos por año

El primer requerimiento de la aplicación es calcular y graficar un histograma donde se muestre cuántos planetas fueron descubiertos a lo largo de los años. Para esto, debes crear una gráfica de tipo **histograma** (hist) con los valores de la columna DESCUBRIMIENTO. El histograma debe separar los datos en 30 grupos (bins). De esta forma podrás apreciar la cantidad de planetas que se descubrieron cada año, entre 1998 y 2018. La siguiente figura muestra la apariencia de esta gráfica:



### Requerimiento 2: Descubrimiento por estado de publicación

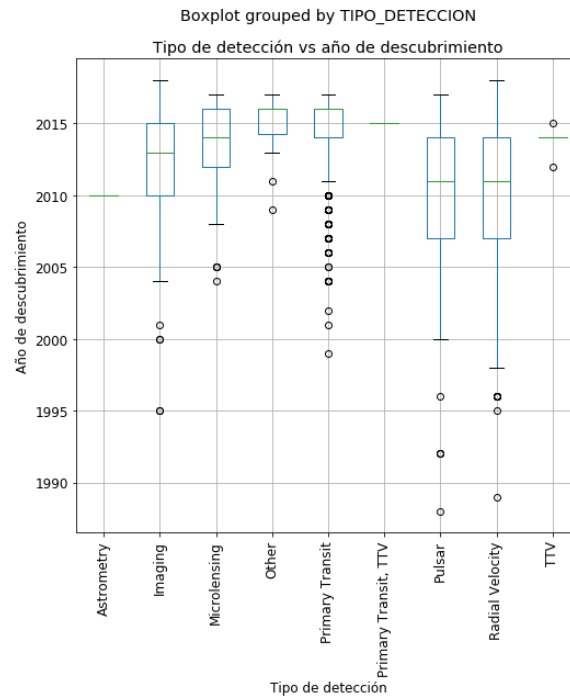
El segundo requerimiento es permitir relacionar los años de descubrimiento de los planetas y los tipos de publicación que se utilizaron para informar los descubrimientos. Para esto, debes crear una gráfica de tipo **boxplot**, usando la columna DESCUBRIMIENTO y agrupando los datos de acuerdo con la columna ESTADO\_PUBLICACION. La siguiente figura muestra la apariencia de esta gráfica:



**ATENCIÓN:** Para que los textos de los tipos de publicación aparezcan verticales y no horizontales (como lo es por defecto), utiliza el parámetro `rot=90` al momento de crear el boxplot. Adicionalmente, para que los títulos no se superpongan entre sí, configura el tamaño de la figura con el parámetro `figsize=(8,8)` al momento de crear el boxplot.

### Requerimiento 3: Descubrimiento por tipo de detección

El tercer requerimiento es permitir relacionar los años de descubrimiento de los planetas y los tipos de detecciones que se utilizaron. Para esto, debes crear una segunda gráfica de tipo boxplot, usando la columna DESCUBRIMIENTO y agrupando los datos de acuerdo con la columna TIPO\_DETECCION. La siguiente figura muestra la apariencia de esta gráfica:

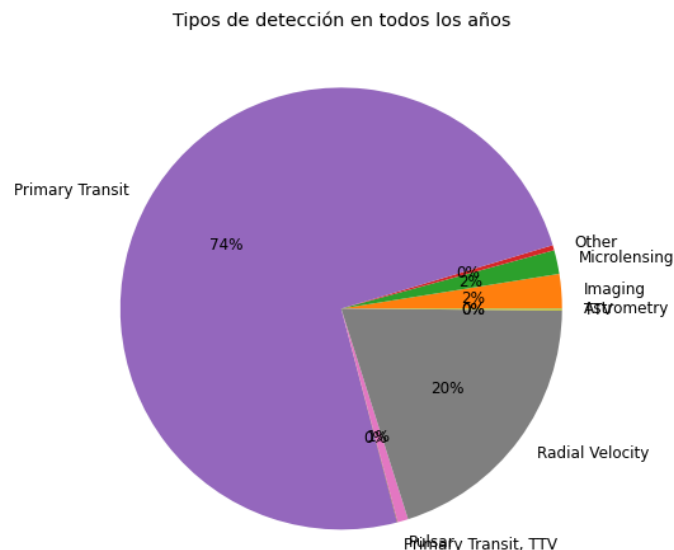


### Requerimiento 4: Tipo de detección por año

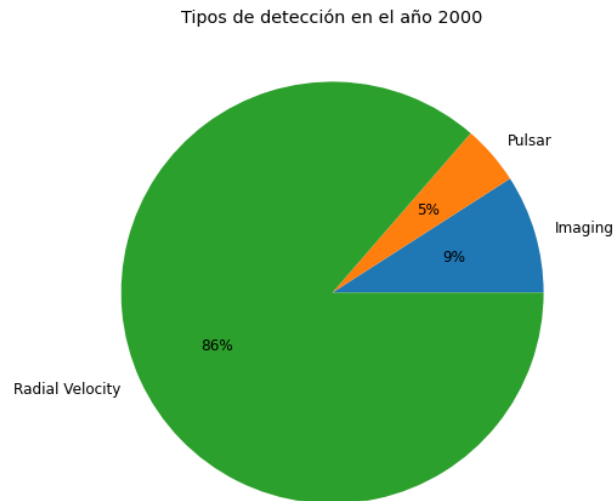
El cuarto requerimiento del programa es permitir generar dos gráficas de tipo pie que muestren:

- los tipos de detección para todos los años en los que se descubrió algún planeta.
- los tipos de detección para un año específico en el que se descubrió algún planeta.

Debes darle al usuario la posibilidad de escoger graficar todos los años o introducir un año específico de su elección. La siguiente figura muestra la apariencia de la gráfica de pie teniendo en cuenta todos los años:



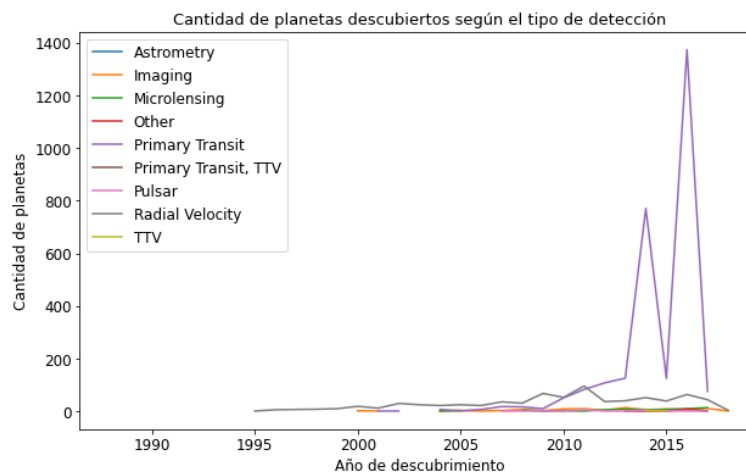
La siguiente figura muestra la apariencia de la gráfica de pie teniendo en cuenta únicamente el año 2000:



**ATENCIÓN:** La documentación sobre gráficas de tipo pie la puedes encontrar en el sitio oficial de Pandas. Puedes configurar el tamaño de la figura con el parámetro `figsize=(8,8)` al momento de crear la figura. Por ejemplo: `matplotlib.pyplot.figure(figsize=(8, 8))`. Para que los valores que se muestran en el pie correspondan a porcentajes, se puede usar el parámetro: `autopct='%1.1f%%'` al momento de crear el pie.

#### Requerimiento 5: Cantidad de descubrimientos por año según el tipo de detección

El siguiente requerimiento es explorar la relación entre la cantidad de planetas descubiertos, el tipo de detección utilizado y el año de descubrimiento. Primero, veamos la gráfica que se debes producir:



Para poder producir esta gráfica, tendrás que reorganizar la información del DataFrame de una forma que sea más coherente con el contenido de la gráfica. Te recomendamos construir un DataFrame donde las columnas sean los tipos de detección, las etiquetas de las filas sean los años y los valores sean la cantidad de planetas descubiertos en un año con un cierto método de detección.

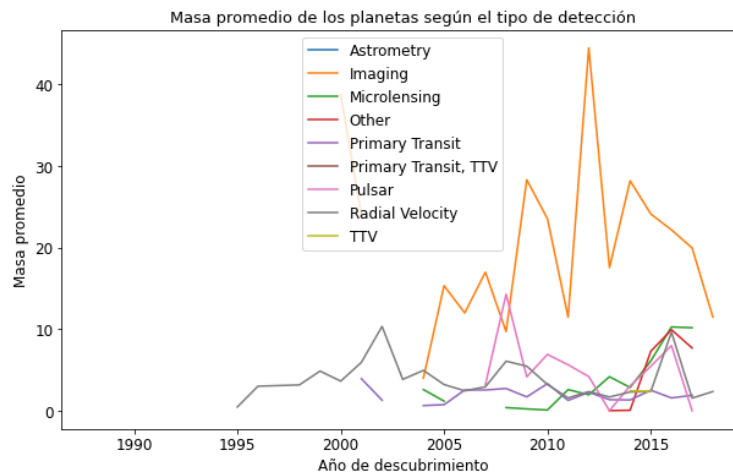
**ATENCIÓN:** Para construir este nuevo DataFrame a partir de los datos que ya tienes, podrías:

- Agrupar los datos por tipo de detección.
- Para cada grupo, agrupar los datos de acuerdo con el año de descubrimiento y contar la cantidad de elementos en cada grupo.

- Guardar esa información en un diccionario, donde las llaves sean los tipos de descubrimiento y los valores sean las series con las cantidades por año.
- Utilizar el diccionario para construir un nuevo DataFrame.

### Requerimiento 6: Masa promedio por año y por tipo de detección

El último requerimiento de esta parte es explorar la relación entre la masa promedio de los planetas descubiertos y el tipo de detección. Deberás construir una figura como la que se ve en la siguiente imagen:

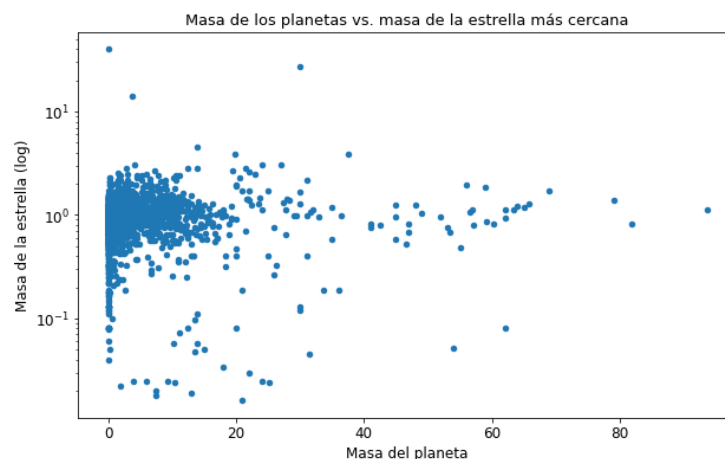


**ATENCIÓN:** Te recomendamos aplicar lo que aprendiste en el requerimiento pasada con respecto a la construcción de nuevos DataFrames para resolver este nuevo problema.

## Parte 3: Estudiar la masa de las estrellas más cercanas a los planetas

### Requerimiento 7: Masa de los planetas vs masa de la estrella más cercana

En esta parte de la aplicación, vas a generar una gráfica de dispersión para analizar la masa de los planetas vs. la masa de la estrella más cercana a cada uno. La siguiente figura muestra la apariencia de la gráfica que deberías obtener:



**ATENCIÓN:** Configura el eje Y para que los valores se presenten en escala logarítmica.

## Parte 4: Graficar y analizar el cielo

En esta última parte de la aplicación, vas a utilizar las coordenadas de las posiciones de los planetas en el cielo (ascensión recta y declinación, obtenidas a partir de las columnas RA y DEC del archivo) para graficar los planetas en una imagen representada por una matriz. Cada planeta será representado por un pixel en la imagen (una casilla de la matriz), cuyo color RGB (Red, Green, Blue) dependerá del tipo de detección que se utilizó para descubrirlo, de acuerdo con la siguiente convención:

Tipo de detección	Valores RGB	Color
Microlensing	[0.94,0.10,0.10]	Red
Radial Velocity	[0.1,0.5,0.94]	Blue
Imaging	[0.34,0.94,0.10]	Green
Primary Transit	[0.10,0.94,0.85]	Cyan
Other	[0.94,0.10,0.85]	Magenta
Astrometry	[0.94,0.65,0.10]	Orange
TTV	[1.0,1.0,1.0]	White

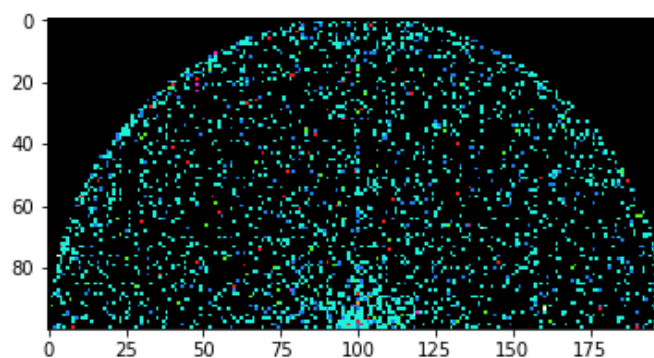
### Requerimiento 8: Graficar el cielo

Lo primero que debes hacer en esta parte, es crear una imagen de 100 filas x 200 columnas, con píxeles de color negro. Sobre esta imagen de base empezarás a cambiar los colores de los píxeles de acuerdo con la posición de cada planeta y el color que le corresponde. Graficar un planeta consiste entonces en cambiar el color del píxel en una casilla de la imagen, cuyas coordenadas (fila y columna) son calculadas a partir de las coordenadas RA-DEC del planeta, de la siguiente manera:

*Para un planeta  $p$  cuyas coordenadas RA-DEC son  $(RA_p, DEC_p)$ , las coordenadas del píxel correspondiente en la matriz que representa la imagen están dadas por las siguientes ecuaciones:*

$$\begin{aligned} Fila_p &= 99 - |\sin(RA_p) * \cos(DEC_p) * 100| \\ Columna_p &= [\cos(RA_p) * \cos(DEC_p) * 100] + 100 \end{aligned}$$

La siguiente figura muestra la apariencia de la imagen de los planetas:

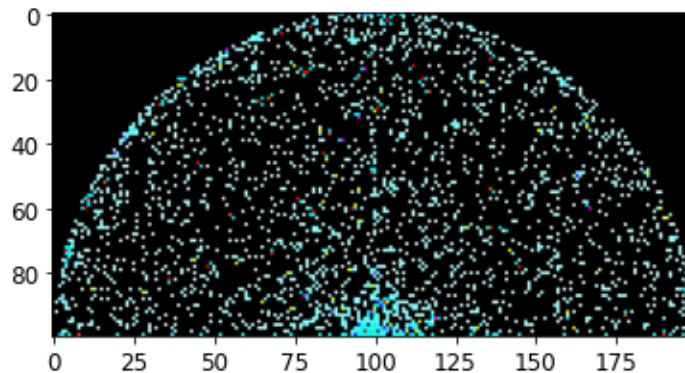


### Requerimiento 9: Afinar la imagen del cielo

Finalmente, vas a afinar la imagen aplicando una operación de convolución con una máscara predefinida. Para esto, debes implementar una función de convolución que reciba como parámetro la imagen de los planetas y retorne la imagen filtrada aplicando la siguiente máscara de convolución:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

La siguiente figura muestran la imagen afinada:



**ATENCIÓN:** Las convoluciones se estudiaron en uno de los videos de Manos a la Obra.

## Actividad 1: Preparación del ambiente de trabajo

1. Crea una carpeta de trabajo.
2. Descarga de Coursera el archivo con el “esqueleto” del proyecto (m4-exoplanetas-esqueleto.zip) que se encuentra en la guía de trabajo y descomprímelo en tu carpeta de trabajo. El esqueleto consiste en un conjunto de archivos que vas a usar o a modificar durante el desarrollo de tu proyecto.
3. Abre Spyder y reemplaza la carpeta de trabajo por la carpeta con el esqueleto.

## Actividad 2: Construir el módulo de funciones

4. Usando Spyder, abre el archivo con el nombre “exoplanetas.py”. En este archivo vas a implementar las funciones que responden a los requerimientos de la aplicación. Completa cada una de las funciones que se encuentran en el esqueleto. Si lo consideras necesario, puedes crear funciones adicionales.

## Actividad 3: Probar el correcto funcionamiento del programa

**ATENCIÓN:** Ten en cuenta que en este proyecto la interfaz basada en consola está completamente terminada así podrás concentrarte en los objetivos específicos de este proyecto.

5. **Ejecuta el programa y prueba cada una de las funciones para asegurarte que esté funcionando.**  
Puedes probar el correcto funcionamiento de tu programa cargando la información que se encuentra en el archivo “exoplanetas.csv” o creando tu propio archivo de prueba con un menor tamaño (respetando el mismo formato) que te permita corroborar que los resultados arrojados por tu programa son correctos.

## Entrega

6. Comprime tu carpeta de trabajo con tu proyecto resuelto como un .zip.
7. Entrega el archivo comprimido a través de Coursera en la actividad designada como “Evaluación por pares del Proyecto M4”.