python-tutorial

July 11, 2025

1 Python and Jupyter Notebook Tutorial

1.0.1 Notebooks and Notebook Architecture

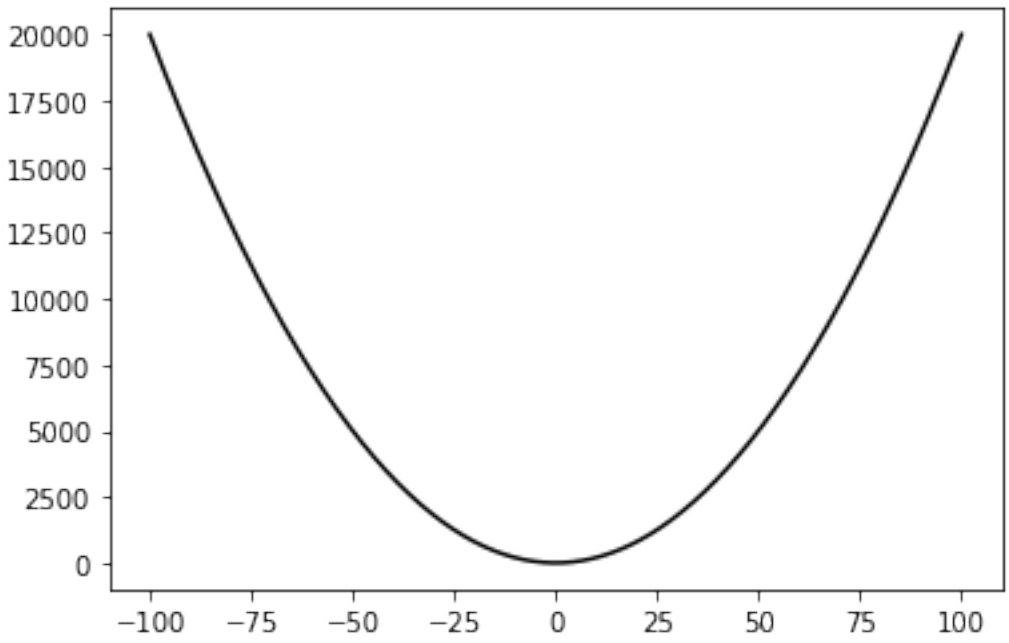
- Interactive documents where you can write small portions code along with rich text (formatted text and images).
- Jupyter Notebook is the default python notebook format
- Cloud based options: Jupyter Lab and Google Colab

[ ]:
```
#Importing libraries and aliasing them
import numpy as np # numpy for Numerical Analysis
import matplotlib.pyplot as plt # for plotting Graphs
```

[ ]:
```
print('Hello World') # prints the output. i.e., inside the ''
a,b = 3,4
print('Product of a and b is ',a*b) # will print the product of a and b
```

```
Hello World
Product of a and b is  12
```

[ ]:
```
#plotting
'''
commenting is essential to guide the user through the code
'''
a = np.linspace(-100,100,1000) #to create an array of 1000 points between -100␣
 ↪and 100 all evenly spaced
b = 2*a**2 + 5
plt.plot(a,b,'-k')
plt.show() # to show the plot
```



2 Some Cool Stuff Python can do

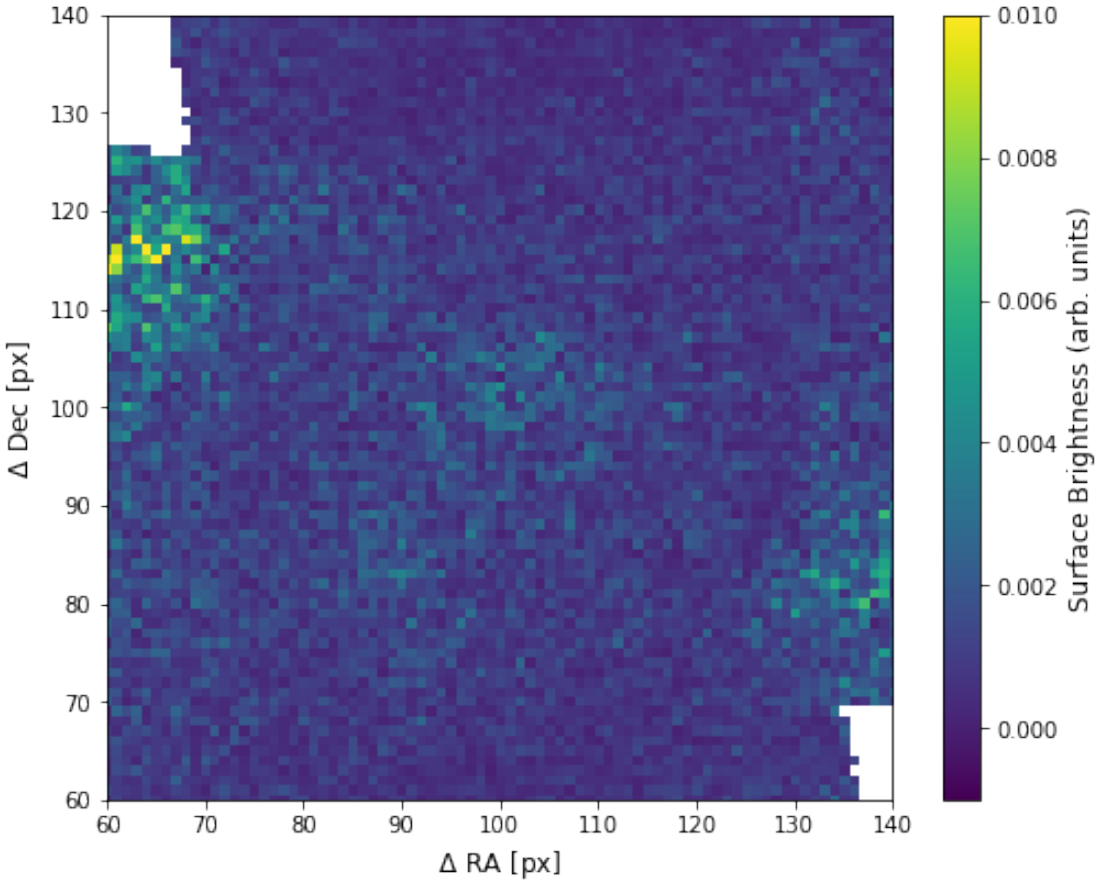
In astronomy, observed data is typically stored in fits files, that can be plotted here using the same packages to plot data points. Here, I am going to give an example using a published image of a popular disk, that recently garnered attention for a newly discovered protoplanet in the system, AB Aur b

[ ]:
```
#import astronomy package
from astropy.io import fits # to read and write FITS files
```

[ ]:
```
#loading the image
#remember to alter the image path accordingly
file_dir = 'data/' # data directory where the FITS file is stored
img_name = 'pdi_pi_collapsed.fits' # The FITS file
img = fits.getdata(file_dir+img_name,ext=1)
```

[ ]:
```
# create a figure
fig = plt.figure(figsize=(8,8)) # sets the size of figure in inches
#plot in the figure
image = plt.imshow(img, origin = 'lower', vmin=-0.001, vmax=0.0cbar = fig.
 ↪colorbar(image,shrink=0.82)
cbar.set_label('Surface Brightness (arb. units)', rotation=90,fontsize=12)e=12)
plt.xlim([60,140]) # sets the x-axis limits
plt.ylim([60,140]) # sets the y-axis limits
plt.xlabel(r'$\Delta$ RA [px]',fontsize=12) # sets the x-axis label
```

```
plt.ylabel(r'$\Delta$ Dec [px]',fontsize=12); # sets the y-axis label
```



[ ]:
```
import pandas as pd # panda library is very useful for manipulating data in a␣
↪talble format and reading csv files
```

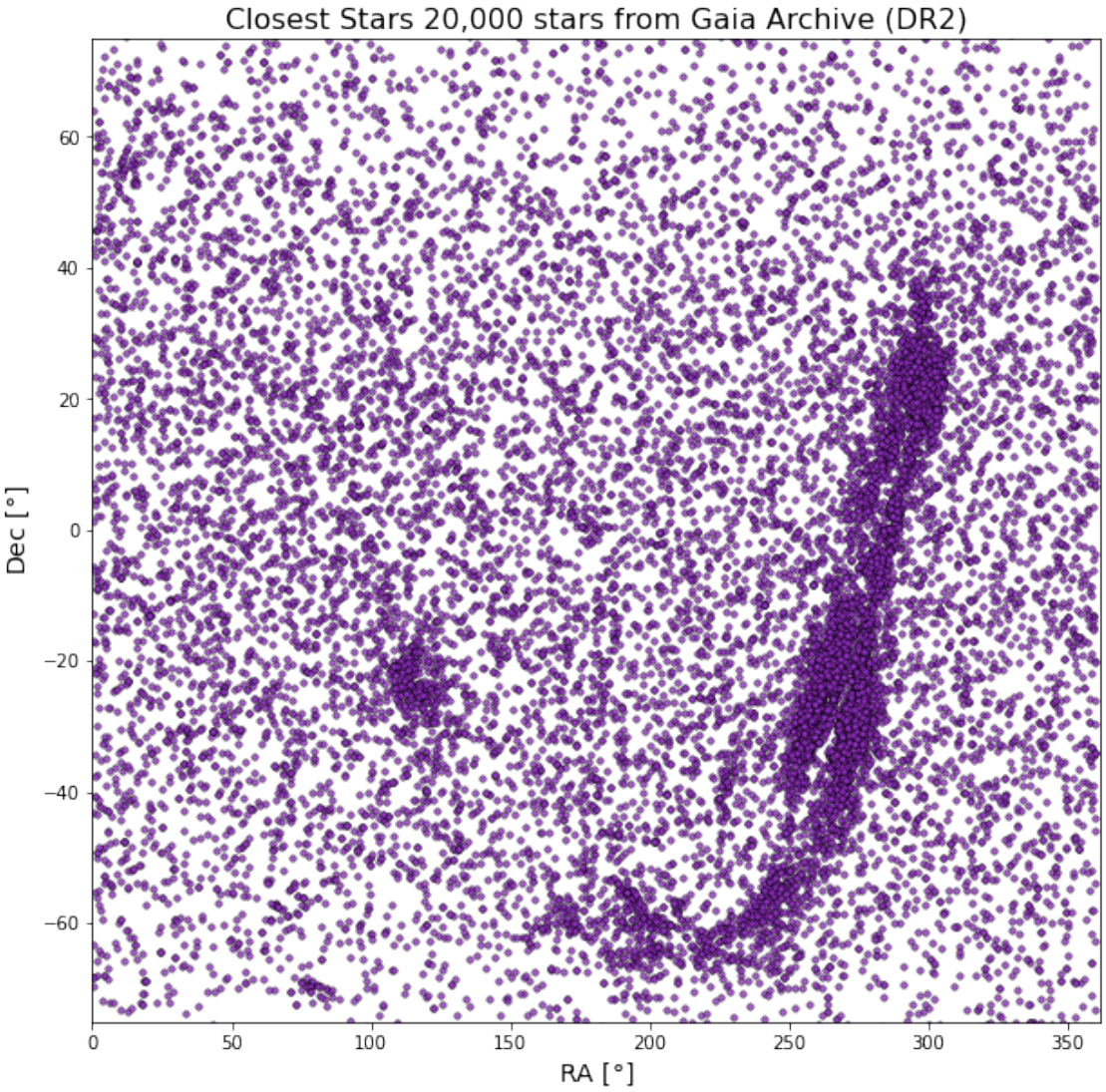
```
/opt/anaconda3/lib/python3.9/site-
packages/pandas/core/computation/expressions.py:21: UserWarning: Pandas requires
version '2.8.4' or newer of 'numexpr' (version '2.7.3' currently installed).
  from pandas.core.computation.check import NUMEXPR_INSTALLED
/opt/anaconda3/lib/python3.9/site-packages/pandas/core/arrays/masked.py:60:
UserWarning: Pandas requires version '1.3.6' or newer of 'bottleneck' (version
'1.3.2' currently installed).
  from pandas.core import (
```

[ ]:
```
# Reading 20,000 Rows*97 columns of Data- Closest 20k stars from Gaia Archive␣
↪(DR2)
stellar=pd.read_csv(file_dir+'closest20kstars.csv') # reads the csv file and␣
↪stores it in a pandas dataframe called stellar
```

```
# Creating a matplotlib (pyplot) figure
fig = plt.figure(figsize = [10,10])

# Plotting the scatter graph with RA on x-axis and Dec on y-axis. alpha tells␣
 ↪the opacity, s is the size
plt.scatter(stellar['ra'], stellar['dec'], alpha=0.8, s=12, lw = 0.5, ec = 'k',␣
 ↪color='darkorchid') # plots the scatter plot of RA vs Dec
# Adding labels and a title for the figure
plt.xlim([0,362])
plt.ylim([-75,75])
plt.xlabel('RA [$\degree$]', fontsize = 14)
plt.ylabel('Dec [$\degree$]', fontsize = 14)
plt.title('Closest Stars 20,000 stars from Gaia Archive (DR2)', fontsize = 16)␣
 ↪# makes the title of the plot
```

[ ]: Text(0.5, 1.0, 'Closest Stars 20,000 stars from Gaia Archive (DR2)')



2.1 For your Assignment

1. Go through the tutorial and repeat all of this on Google Colab to make sure you understand, and add your own comments along the way.
2. One thing that is important is to know how to save these figures. So why don't you try that?
3. Print the Fibonacci Sequence (a series of numbers in which each number is the sum of the two preceding. Starts with 0,1) (Hint: Use a for loop).

1. done in above

2. in Vs Code there are two icons on the top right of the plot. one is to copy the plot and the other is to save the graph to a desired location. or you can use plt.savefig("plot.png", format='png', dpi=300) right after the code for plotting the graph to save it.

[ ]:
```python
# printing Fibonacci series
def print_fibonacci_series(n):
    a, b = 0, 1
    for _ in range(n):
        print(a, end=' ')
        a, b = b, a + b
    print()

print_fibonacci_series(10)  # prints the first 10 Fibonacci numbers
```

[ ]:
```
```