Module 3: Reading data files in Python

contributed by

Mr. Anik Paul
Intern, R Team, FOSSEE, IIT Bombay
Mathematics Department, IIT Bombay

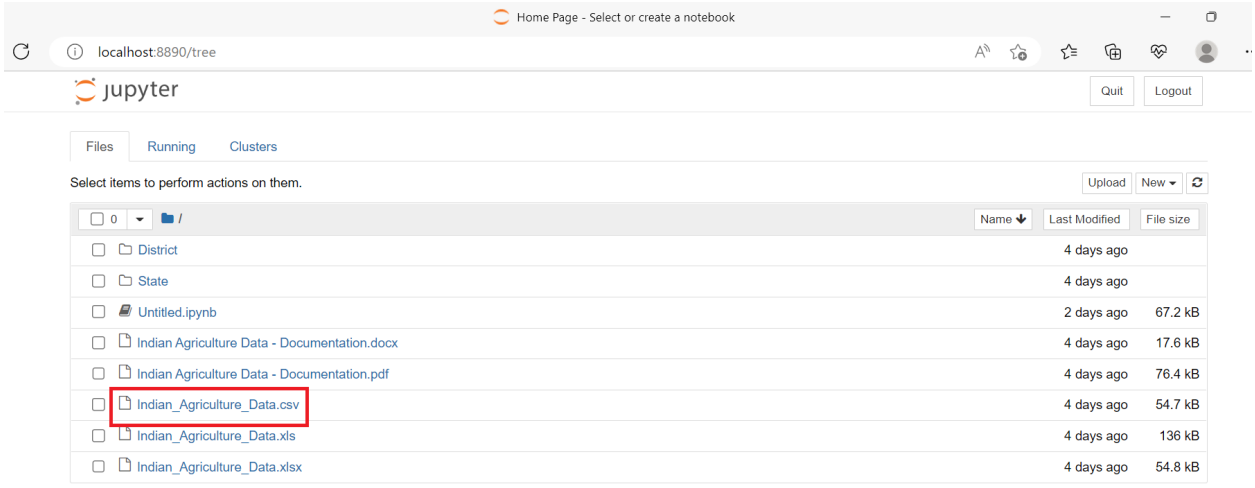
Ms. Usha Viswanathan
Sr. Project Manager
FOSSEE, IIT Bombay

5 December 2022

Steps to read data files in Python

This module explains the methods for importing data from a file present in the working directory to Python. Before importing data, set the folder containing the data file as the working directory by following the instructions given below.

Data from a **.csv**, **.xlsx** or **.xls** file can be read in Python. To read data from a **.csv** file in Python follow the steps given below. 'Indian_Agriculture_Data.csv' is the .csv file used in this module.



Using Pandas module:

Step 1: Import Pandas module.

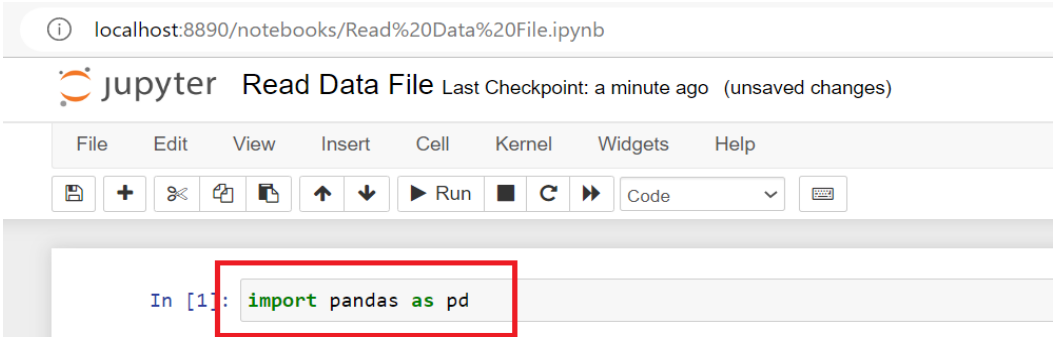


Figure 2: Import Pandas module.

Remember that, here for the sake of simplicity, the pandas module was imported as 'pd'. The name can be chosen as per the user's choice.

Step 2: Read the data file by using the read_csv() function in the pandas module. The syntax is given by:

```
data=pd.read_csv('filenames.csv')
```

Figure 3: Read the csv data file and store it as 'Agriculture_data'.

Step 3: To have an overview of the data, use the head() function. By default, it shows the first 5 rows. The number of rows to be returned can be passed as an argument to the head function inside the braces.

```
n [3]: Agriculture_data=pd.read_csv('Indian_Agriculture_Data.csv')
n [4]: Agriculture_data.head()
```

ut[4]:

	State	Zone	GSDP	Area	Prod.	Prod./Area	NFS	PFS	KFS	Dist.	SML	SMV	SDN	SDP	SDK
0	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	AMRITSAR	6.84	17.20	91.67	61.93	23.83
1	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	BARNALA	3.21	15.76	89.10	55.37	31.51
2	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	BATHINDA	4.86	13.83	91.66	67.61	29.96
3	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	FARIDKOT	0.49	14.56	0.00	54.91	19.87
4	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	FATEHGARH SAHIB	5.49	21.54	100.00	56.73	23.33

Figure 4: Use of head() function for an overview of the data.

```
Agriculture_data.head(6)
```

	State	Zone	GSDP	Area	Prod.	Prod./Area	NFS	PFS	KFS	Dist.	SML	SMV	SDN	SDP	SDK
0	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	AMRITSAR	6.84	17.20	91.67	61.93	23.83
1	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	BARNALA	3.21	15.76	89.10	55.37	31.51
2	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	BATHINDA	4.86	13.83	91.66	67.61	29.96
3	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	FARIDKOT	0.49	14.56	0.00	54.91	19.87
4	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	FATEHGARH SAHIB	5.49	21.54	100.00	56.73	23.33
5	PUNJAB	Northern	51251093	391.71	7342.12	18.74	LOW	LOW	MEDIUM	GURDASPUR	21.02	20.13	100.00	53.10	30.08

Figure 5: Get an overview of the data consisting of a specified number of rows.

Step 4: One can get the number of rows and columns all together by using the shape() function.

```
Agriculture_data.shape
(525, 15)
```

Figure 6: number of rows and columns using shape function.

The same steps given above can be used to read data from an **.xlsx** or **.xls** file. To read the data from a **.xlsx** file use read_excel() function.