

Walchand College of Engineering, Sangli

Computer Science & Engineering

Third Year

Course:Software Engg Tools Lab

PRN NO : 2020BTECS00207

Name : Kshirsagar Mayuri Manojkumar

Lab course coordinator : Ms.P.D.Mumdada

Batch: - T5

Assignment No-2

(Module 2- Software Development Frameworks)

Due date-16/02/2022

For every Google Collab given above provide the answers for below questions

1. Original author

Fernando Perez and Bria

2. Developers

Project Jupyter

3. Initial release

Google first started working with the Jupyter Development Team in 2014 to release an early version of the tool, , since then the tool has been constantly evolving

4. Stable release

Currently, access to Colab is managed by services without individual controls. Later in 2021, the individual ON/OFF control for Colab will be available and control access to C

5. Preview release

It was first publicly released in ea

6. Repository (with cloud support)

Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. When you create your own Colab notebooks, they are stored in your Google Drive account

7. Written in (Languages)

Three core programming languages supported by Jupyter, which are Julia, Python and R, and also a homage to Galileo's notebooks recording the discovery of the moons of Jupiter.

8. Operating System support

Colab is running Python 3.7, whereas you will run Python 3.8 on your machine through your Anaconda installation

9. Platform ,portability

Google Colab! It's an incredible online browser-based platform that allows us to train our models on machines for free . Sounds too good to be true, but thanks to Google, we can now work with large datasets, build complex models, and even share our work seamlessly with others

10.Available in (Total languages)

English

11.List of languages supported

Colab focuses on supporting Python and its ecosystem of third- party tools. We're aware that users are interested in support for other Jupyter kernels (eg R or Scala). We would like to support these, but don't yet have any ETA

12.Type (Programming tool, integrated development environment etc.)

Nonprofit organization

13.Website

jupyter.org

14.Features

To support interactive data science and scientific computing across all programming languages .

15.Size (in MB, GB etc.)

While 32 GB of RAM is available in Colab Pro, Pro+ users have 52 GB available with the high-memory option. That is about 1.6 times as much as pro users and 3.25 times as much as free users.

16.Privacy and Security

Trusted by industry leaders and Fortune 500 companies, CoLab is built to the highest standards of security and data protection adopting a security-first approach to product development and infrastructure design, while giving end users the safest possible method to review critical data with their team and external partners

17.Type of software (Open source/License)

Jupyter is the open source project on which Colab allows you to use and share Jupyter notebooks with others without having to download, install, or run anything.

18.If License- Provide details.

Google Colab now also provides a paid platform called Google Colab Pro, priced at \$9.99 a month. In this plan, you can get the Tesla T4 or Tesla P100 GPU, and an option of selecting an instance with a high RAM of around 27 GB

19.Latest version

Python 3.8

20.Cloud support (Yes/No)

Google Colaboratory is a free online cloud-based Jupyter notebook environment that allows us to train our machine learning and deep learning models on CPUs, GPUs, and TPU s

21.Applicability

Getting started with TensorFlow , Developing and training neural networks

22.Drawbacks (if any)

Overall usage limits as well as idle timeout periods, maximum VM lifetime, GPU types available, and other factors vary over time. Colab does not publish these limits, in part because they can (and sometimes do) vary quickly.

1. Implement linear regression problem using Google colab (Perform preprocessing, training and testing)

Dataset 7- <https://archive.ics.uci.edu/ml/datasets/KDD+Cup+1998+Data>

```

colabresearch.google.com/drive/12hUtaOGGsbBbRbIH_uZ8eTDC0hNtOFsrdTo-ZF8ZvaoVMin
SET209.ipynb
File Edit View Insert Runtime Tools Help Saving...
+ Code + Text
RAM 8 Disk
[2] from google.colab import drive
drive.mount('/content/drive')

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

[3] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

[4] print('Booting into Machine Learning....')

Booting into Machine Learning....

[12] data=pd.read_csv('/content/drive/MyDrive/day.csv')

[13] data.head(10)

```

	instant	registered	casual	humidity	windspeed	temp	atemp	casual	registered	cnt
0	1	654	331	0.805833	0.160446	2	0.344167	0.363625	0.805833	985
1	2	670	131	0.696087	0.248539	2	0.363478	0.353739	0.696087	801
2	3	1229	120	0.437273	0.248309	1	0.196364	0.189405	0.437273	1349
3	4	1454	108	0.590435	0.160296	1	0.200000	0.212122	0.590435	1562

0s completed at 3:32 PM

```

colabresearch.google.com/drive/12hUtaOGGsbBbRbIH_uZ8eTDC0hNtOFsrdTo-ZF8ZvaoVMin
SET209.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
RAM 8 Disk
[13] 5 6 2011-01-06 1 0 1 0 4 1 1 0.204348 0.233209 0.518261 0.089565 88 1518 1606
6 7 2011-01-07 1 0 1 0 5 1 2 0.196522 0.208839 0.498696 0.168726 148 1362 1510
7 8 2011-01-08 1 0 1 0 6 0 2 0.165000 0.162254 0.535833 0.266804 68 891 959
8 9 2011-01-09 1 0 1 0 0 0 1 0.138333 0.116175 0.434167 0.361950 54 768 822
9 10 2011-01-10 1 0 1 0 1 1 1 0.150833 0.150888 0.482917 0.223267 41 1280 1321

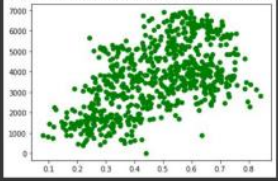
[13] print('defining variables')
x = data['atemp']
y = data['registered']

defining variables

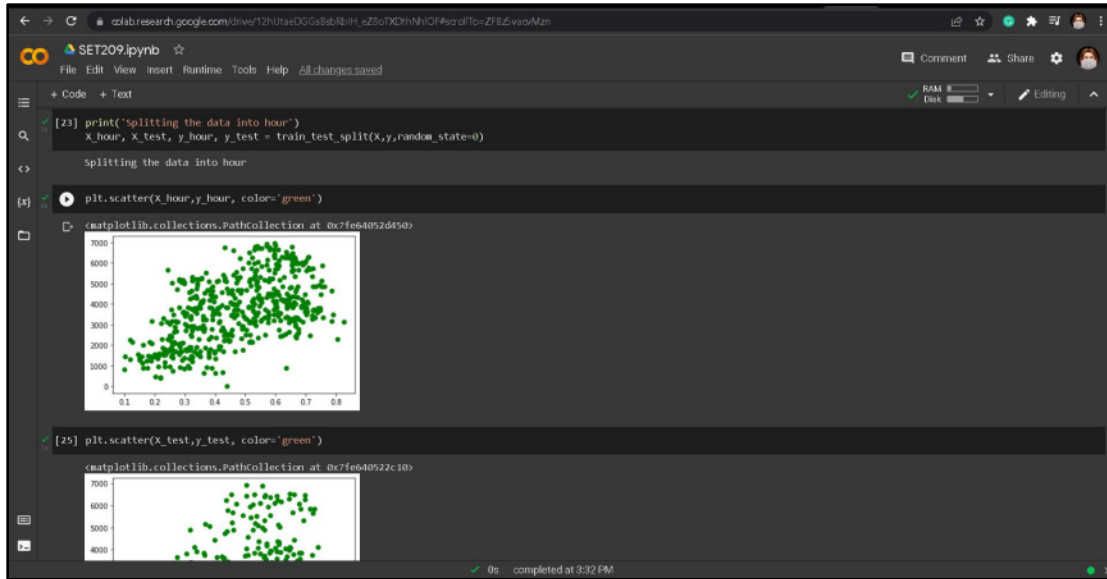
[22] plt.scatter(X,y, color='green')

matplotlib.collections.PathCollection at 0x7fe405ac9d80

```



0s completed at 3:32 PM



The screenshot shows a Jupyter Notebook titled 'SET209.ipynb'. The code cell contains the following Python code:

```
print('training the model using X_hour, y_hour')
lr = LinearRegression()

print(X_hour)
print(y_hour)
print(X_hour.values.reshape(-1,1))
lr.fit(X_hour.values.reshape(-1,1), y_hour)

print('Predicting using the trained model - X_hour')
y_pred = lr.predict(X_test.values.reshape(-1,1))

print(y_test) #Test data - actual data
print(y_pred) #model predicted dataset
```

The output displays the following text:

```
training the model using X_hour, y_hour
LinearRegression()

Predicting using the trained model - X_hour

196 3505
187 3838
14 1826
31 1313
390 3831
...
310 3662
21 808
650 6222
424 1769
35 905
Name: registered, Length: 183, dtype: int64
```

