Software Engineering Tools Lab

Assignment No-2

(Module 2- Software Development Frameworks)

Name: SumedhMilindBhatkar

PRN: 2019BTECS00045

Batch: T6

Name: Prathamesh Ramdas Chavhan

PRN: 2019BTECS00044

Batch: T6

Anaconda

1. Original author
   - Peter Wang and Travis Oliphant
2. Developers
   - Anaconda, Inc.(previously Continuum Analytics)
3. Initial release
   - 0.8.0/17 July 2012
4. Stable release
   - 2021.11/17 November 2021
5. Preview release
   - JupyterLab alpha preview 0.27.0 is included
6. Repository (with cloud support)
   - A cloud package repository hosting service at https://www.anaconda.org. With a free account, you can publish packages you create to be used publicly.Open source repository of hundreds of popular data science packages, along with the conda package and virtual environment manage
7. Written in (Languages)
   - Anaconda is a distribution of **the Python and R programming languages** for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.

8. Operating System support
    - Windows 8 or newer, 64-bit macOS 10.13+, or Linux, including Ubuntu, RedHat, CentOS 7+ etc.
9. Platform ,portability
    - A software platform for developing, governing, and automating data science and AI pipelines from laptop to production.Packages can be encapsulated into Anaconda projects for easy portability.
10. Available in (Total languages)
    - Total 2 languages that is python and R programming language.
11. List of languages supported
    - It also supports other programming languages like C, C++, FORTRAN, Java, Scala, Ruby, and Lua.

12. Type (Programming tool, integrated development environment etc.)
    - We can use the following IDEs with Anaconda:
        - ➢ Eclipse and PyDev
        - ➢ IDLE
        - ➢ IntelliJ
        - ➢ Ninja IDE
        - ➢ Python Tools for Visual Studio (PTVS)
        - ➢ Python for Visual Studio Code
        - ➢ Spyder
        - ➢ Sublime Text
        - ➢ Microsoft Visual Studio Code (VS Code)
        - ➢ Wing IDE

13. Website
    - https://www.anaconda.com/products/individual

14. Features
    - **Better Reliability:** The reliability of Anaconda has been improved in the latest release by capturing and storing the package metadata for installed packages.

    • **Enhanced CPU Peformance:** The Intel Math Kernel Library 2019 for Deep Neural Networks(MKL 2019) has been introduced in Anaconda 5.3 distribution. Users deploying Tensorflow can make use of MKL 2019 for Deep Neural Networks. These Python binary packages are provided to achieve high CPU performance.

• **New packages are added:** There are over 230 packages which has been updated and added in the new release.

• **Work in Progress:** There is a casting bug in Numpy with Python 3.7 but the team is currently working on patching it until Numpy is updated.

15. Size (in MB, GB etc.)

- Minimum 5 GB disk space to download and install.

16. Privacy and Security

- Security Risk :- Due to the sheer number of Python and R packages and their dependencies, packages are still not free of security vulnerabilities and exposures.
- Privacy :-Under the Privacy Shield, Anaconda is responsible for the processing for personal data we receive and subsequently transfer to a third party service provider acting for or on our behalf.

17.Type of software (Open source/License)

-

18.If License- Provide details

- Terms of Service require commercial users to purchase a license, so that we may continue to support innovation in open-source data science, maintaining and improving Anaconda Individual Edition, Conda packages, and our repositories.

19.Latest version

- Anaconda Individual Edition 2021.11 includes a new release of Anaconda Navigator - version 2.1. 1.

20.Cloud support (Yes/No)

- Yes

21.Applicability

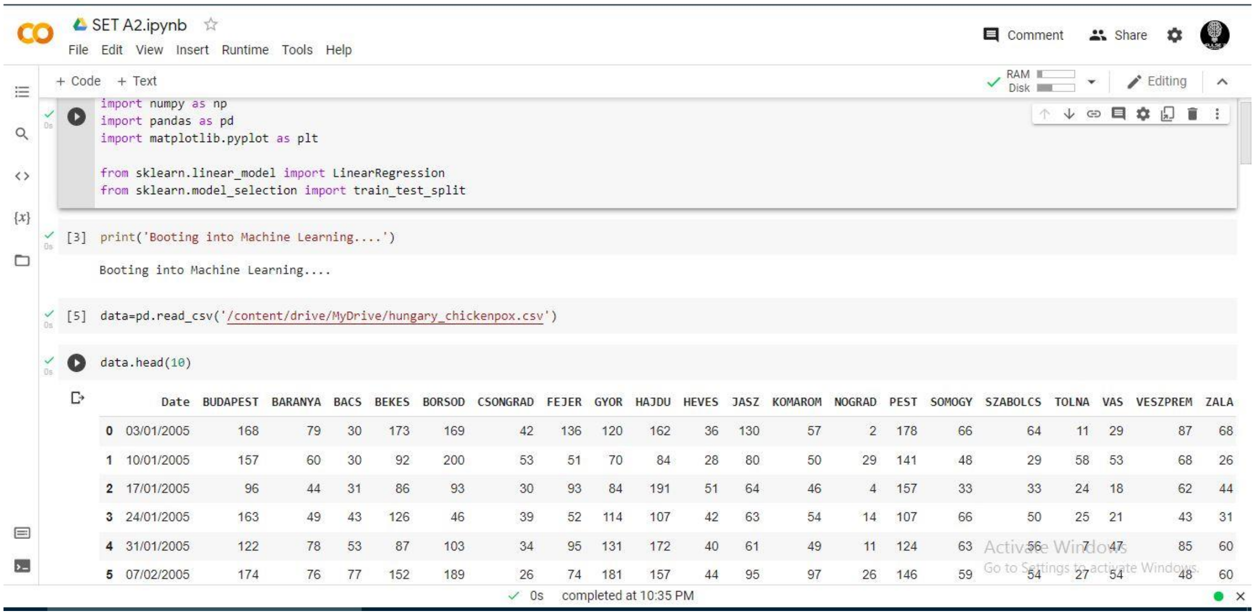
- Anaconda Individual Edition is the world’s most popular Python distribution platform with over 25 million users worldwide. You can trust in our long-term commitment to supporting the Anaconda open-source ecosystem, the platform of choice for Python data science.

22.Drawbacks (if any)

- The application takes a lot of time to load the first time.
- Sometimes, it stops working because it consumes more ram.

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

Used Dataset no. 6 i.e.https://archive.ics.uci.edu/ml/datasets/Hungarian+Chickenpox+Cases for implementation of linear regression problem using Google colab.



```
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
```

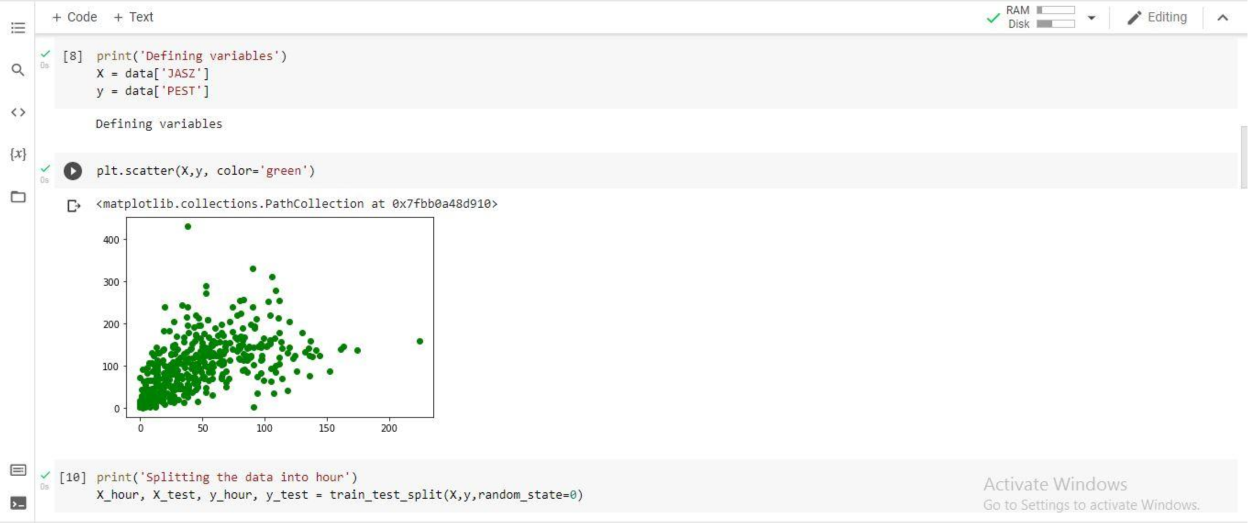
```
[3] print('Booting into Machine Learning....')

Booting into Machine Learning....
```

```
[5] data=pd.read_csv('/content/drive/MyDrive/hungary_chickenpox.csv')
```

```
data.head(10)
```

	Date	BUDAPEST	BARANYA	BACS	BEKES	BORSOD	CSONGRAD	FEJER	GYOR	HAJDU	HEVES	JASZ	KOMAROM	NOGRAD	PEST	SOMOGY	SZABOLCS	TOLNA	VAS	VESZPREM	ZALA
0	03/01/2005	168	79	30	173	169	42	136	120	162	36	130	57	2	178	66	64	11	29	87	68
1	10/01/2005	157	60	30	92	200	53	51	70	84	28	80	50	29	141	48	29	58	53	68	26
2	17/01/2005	96	44	31	86	93	30	93	84	191	51	64	46	4	157	33	33	24	18	62	44
3	24/01/2005	163	49	43	126	46	39	52	114	107	42	63	54	14	107	66	50	25	21	43	31
4	31/01/2005	122	78	53	87	103	34	95	131	172	40	61	49	11	124	63	56	7	47	85	60
5	07/02/2005	174	76	77	152	189	26	74	181	157	44	95	97	26	146	59	54	27	54	48	60



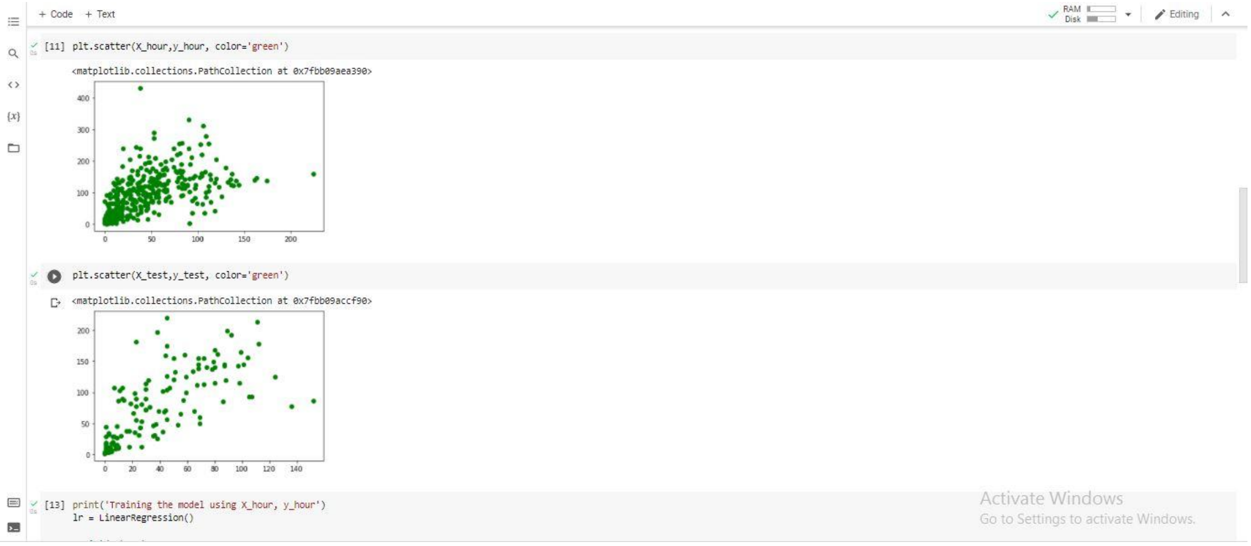
```
[8] print('Defining variables')
    X = data['JASZ']
    y = data['PEST']
```

```
Defining variables
```

```
plt.scatter(X,y, color='green')
```

```
<matplotlib.collections.PathCollection at 0x7fbb0a48d910>
```

```
[10] print('Splitting the data into hour')
     X_hour, X_test, y_hour, y_test = train_test_split(X,y,random_state=0)
```



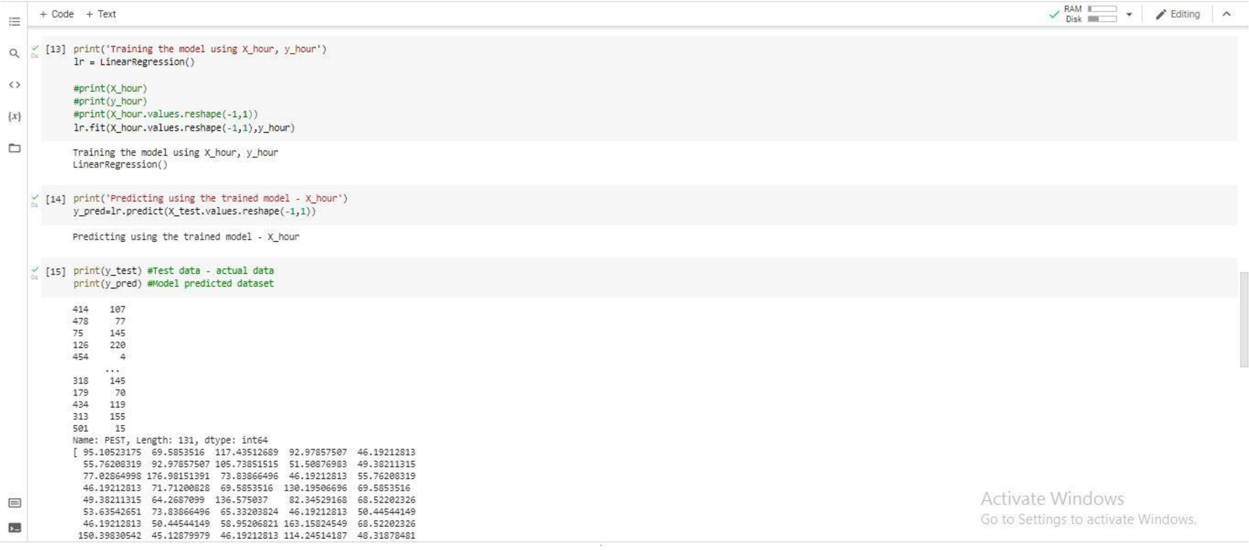
```
[11] plt.scatter(X_hour,y_hour, color='green')
```

```
<matplotlib.collections.PathCollection at 0x7fbb09aea390>
```

```
plt.scatter(X_test,y_test, color='green')
```

```
<matplotlib.collections.PathCollection at 0x7fbb09accf90>
```

```
[13] print('Training the model using X_hour, y_hour')
     lr = LinearRegression()
```



```
[13] 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)
```

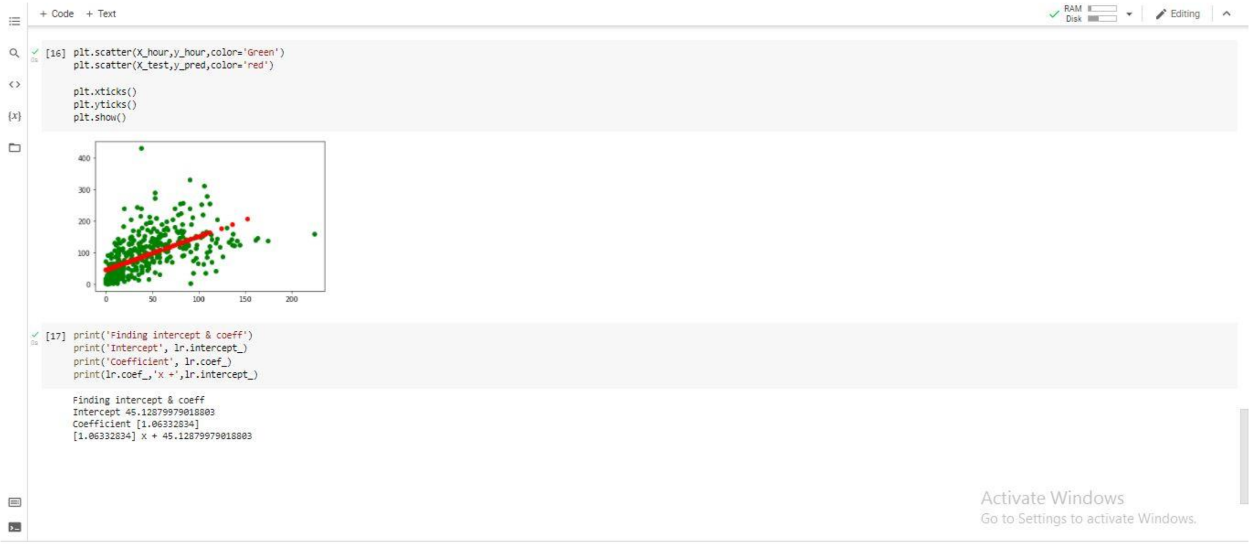
```
Training the model using X_hour, y_hour
LinearRegression()
```

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

```
Predicting using the trained model - X_hour
```

```
[15] print(y_test) #Test data - actual data
     print(y_pred) #Model predicted dataset
```

```
414    107
478     77
75     145
126    220
454      4
      ...
318    145
179     70
434    119
313    155
501     15
Name: PEST, Length: 131, dtype: int64
[ 95.10523175  69.5853516  117.43512689  92.97857507  46.19212813
  55.76208319  92.97857507 105.73851515  51.50876983  49.38211315
  77.02864998 176.98151391  73.83866496  46.19212813  55.76208319
  46.19212813  71.71200828  69.5853516  130.19506696  69.5853516
  49.38211315  64.2687099  136.575037    82.34529168  68.52202326
  53.63542651  73.83866496  65.33203824  46.19212813  50.44544149
  46.19212813  50.44544149  58.95206821 163.15824549  68.52202326
 150.39830542  45.12879979  46.19212813 114.24514187  48.31878481
```



```
[16] plt.scatter(X_hour,y_hour,color='Green')
     plt.scatter(X_test,y_pred,color='red')

     plt.xticks()
     plt.yticks()
     plt.show()
```

```
[17] print('Finding intercept & coeff')
     print('Intercept', lr.intercept_)
     print('Coefficient', lr.coef_)
     print(lr.coef_,'x +',lr.intercept_)
```

```
Finding intercept & coeff
Intercept 45.12879979018803
Coefficient [1.06332834]
[1.06332834] x + 45.12879979018803
```