Experiment-1

Aim:- Introduction to PyTorch

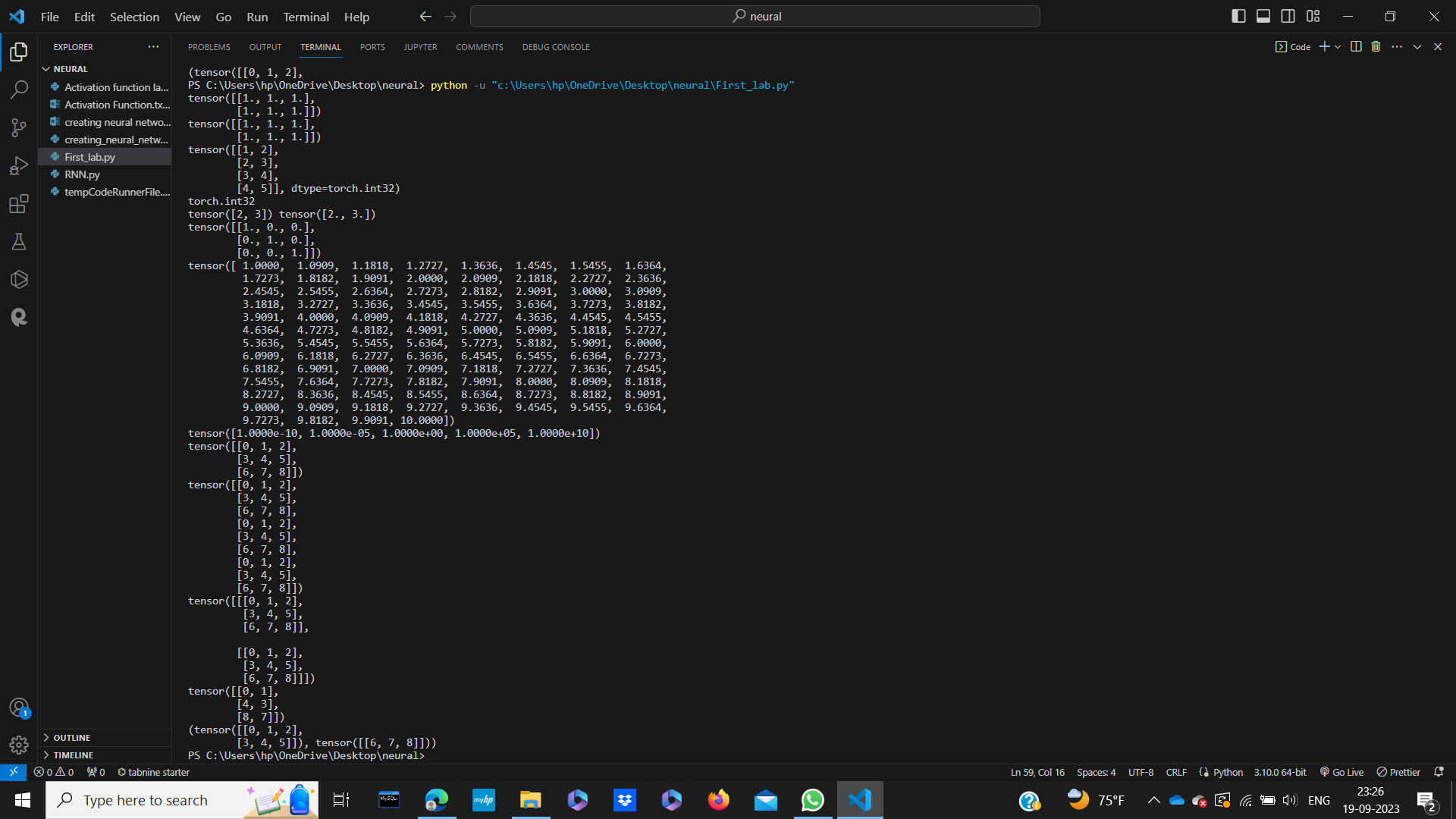
PyToch:- PyTorch is an open-source machine learning library developed by Facebook's AI Research lab (FAIR). It is a library of python language used for building deep learning project. It provide core data structure and tensor which work similar to numpy array. PyTorch provides a flexible and dynamic computational graph, which makes it popular among researchers and developers for building and training neural networks.

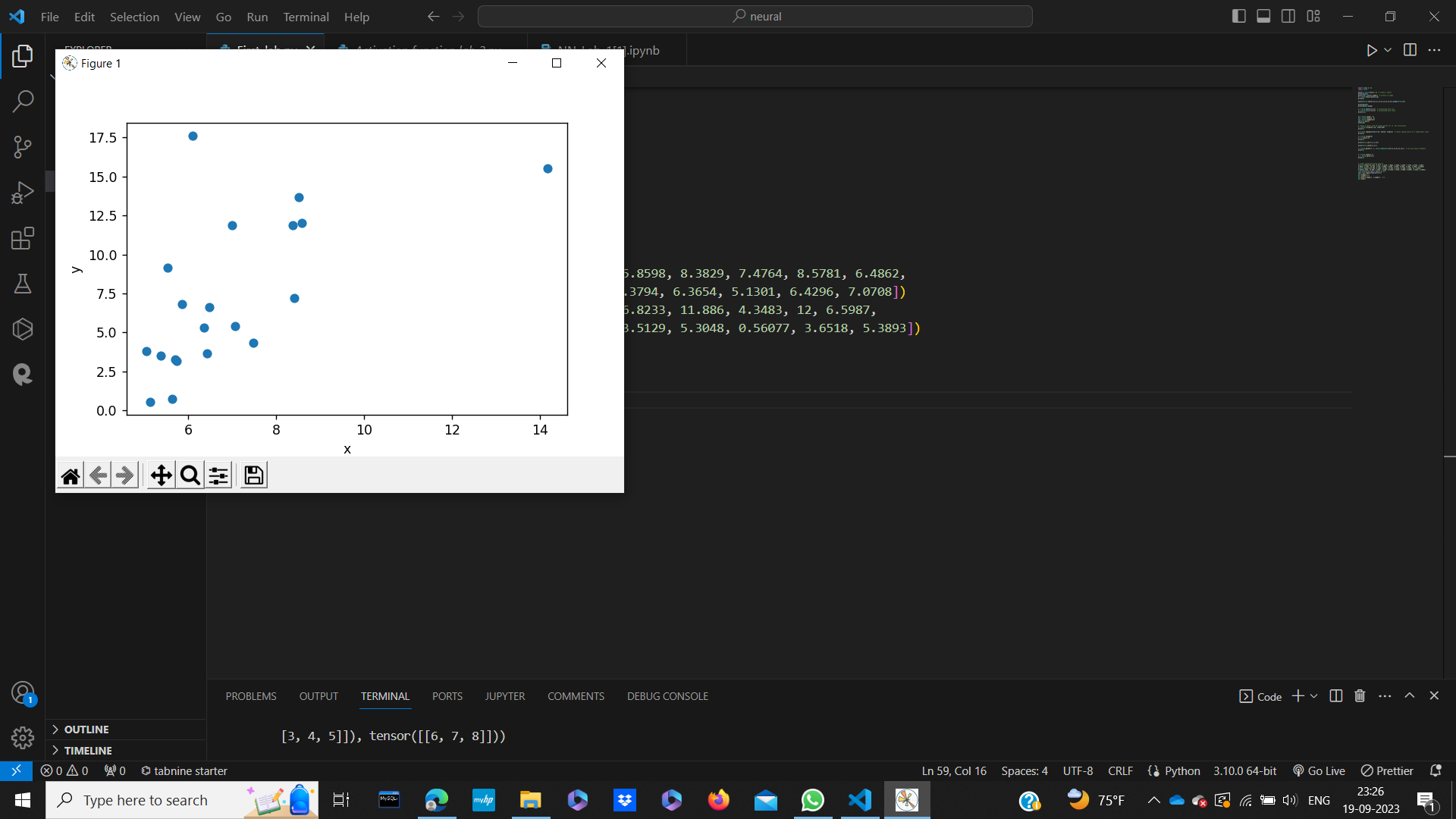
Tensor:- It is a array or data structure which is use to store or collect number which is accessible individual using index. PyToch tensor can convert in numpy and vice versa.

**Code:-**

|  |
| --- |
| **import numpy as np**  **import torch**  **points = torch.ones(2, 3) # create a tensor**  **print(points)**  **points\_np = points.numpy() # convert to numpy**  **b = torch.tensor(points\_np)**  **print(b)**  **points=torch.tensor([[1,2],[2,3],[3,4],[4,5]],dtype=torch.int)**  **print(points)**  **print(points.dtype)**  **v = torch.tensor([2,3]) # initialised with list**  **s = torch.Tensor([2,3]) # initialised with float**  **print(v,s)**  **v1 = torch.rand(2, 3)**  **v2 = torch.randn(2, 3)**  **v3 = torch.randperm(4)**  **id=torch.eye(3)**  **print(id)**  **# Create a Tensor with 10 linear points for (1, 10) inclusively**  **v = torch.linspace(1,10, steps=100)**  **print(v)**  **a = torch.logspace(start=-10, end=10, steps=5) # values spaced evenly on a logarithmic scale**  **print(a)**  **v = torch.arange(9)**  **v = v.view(3,3)**  **print(v)**  **print(torch.cat((v,v,v),0))**  **print(torch.stack((v,v)))**  **r = torch.gather(v, 1, torch.LongTensor([[0,1],[1,0],[2,1]])) # row wise select elements**  **print(r)**  **r = torch.chunk(v,2)**  **s = torch.split(v,2)**  **print(r)**  **# linear regression using pytorch**  **x=torch.tensor([6.1101, 5.5277, 8.5186, 7.0032, 5.8598, 8.3829, 7.4764, 8.5781, 6.4862,**  **5.0546, 5.7107, 14.164, 5.734, 8.4084, 5.6407, 5.3794, 6.3654, 5.1301, 6.4296, 7.0708])**  **y=torch.tensor([17.592, 9.1302, 13.662, 11.854, 6.8233, 11.886, 4.3483, 12, 6.5987,**  **3.8166,3.2522, 15.505, 3.1551, 7.2258, 0.71618, 3.5129, 5.3048, 0.56077, 3.6518, 5.3893])**  **from matplotlib import pyplot as plt**  **fig = plt.figure(figsize=(6,4))**  **plt.xlabel("x")**  **plt.ylabel("y")**  **plt.plot(x.numpy(), y.numpy(), 'o')**  **plt.show()** |

**Output:-**

****

****