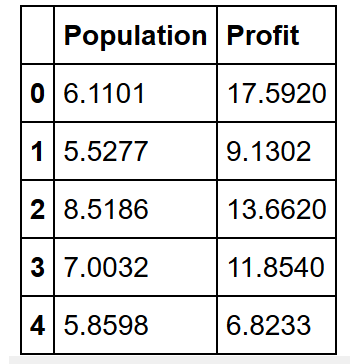
1. Load the iris dataset and find the correlation between variables of iris data except for the class column. Also, create a heatmap using Seaborn to present their relations.
2. Create a box plot (or box-and-whisker plot) that shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable of the iris dataset. Use seaborn.
3. Using beautiful soup and selenium scrap the parameters(Name of Product, Price, and Ratings) from

<https://www.flipkart.com/laptop-accessories/mouse/pr?sid=6bo%2Cai3%2C2ay&p%5B%5D=facets.serviceability%5B%5D%3Dtrue&p%5B%5D=facets.fulfilled_by%255B%255D%3DFlipkart%2BAssured&hpid=m3fhzodGd6WUwDzhx_TWI6p7_Hsxr70nj65vMAAFKlc=&fm=neo%2Fmerchandising&iid=M_28cdc0c1-fcbf-42fe-811c-1a9bc447c8a5_1.H7UUKY4SCA2X&ppt=None&ppn=None&ssid=ur9fhuix4w0000001643862607680&otracker=dynamic_omu_infinite_IT%2BAccessories_2_1.dealCard.OMU_INFINITE_H7UUKY4SCA2X&cid=H7UUKY4SCA2X>

and make a CSV file that should contain the columns and data.

1. Take the above data as input using pandas and predict the future value of profit at population 10.2324. Write a CSV file of the above data and also generate its scatter plot. (Hint: use linear regression).
2. Do as directed, write code for the lines that are commented in the code below:

|  |  |
| --- | --- |
|  | # Basic packages  import numpy as np  import pandas as pd  import matplotlib.pyplot as plt  # Sklearn modules & classes  from sklearn.linear\_model import Perceptron, LogisticRegression  from sklearn.svm import SVC  from sklearn.model\_selection import train\_test\_split  from sklearn.preprocessing import StandardScaler  from sklearn import datasets  from sklearn import metrics |

|  |  |  |
| --- | --- | --- |
|  | # Load the data set; In this example, the breast cancer dataset is loaded.  bc = datasets.load\_breast\_cancer()  X = bc.data  y = bc.target  # Create training and test split | |
|  | sc = StandardScaler()  sc.fit(X\_train)  X\_train\_std = sc.transform(X\_train)  X\_test\_std = sc.transform(X\_test) | |
|  | * # Instantiate the Support Vector Classifier (SVC) * # Fit the model | | |
|  | * # Make the predictions * # Measure the performance | | |