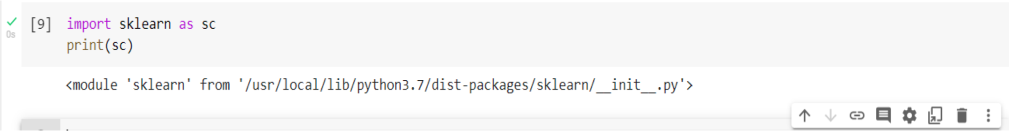
Aim: Write down to libraries used for the following machine learning applications

1. Scikit-learn(sklearn):

- Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.
- In python, sklearn is a machine learning package which includes a lot of ML algorithms.

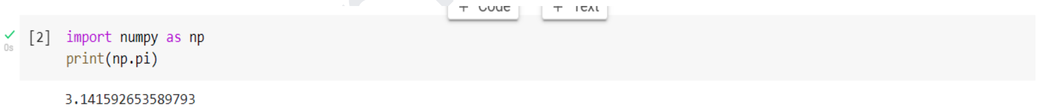


```
[9] import sklearn as sc
    print(sc)

<module 'sklearn' from '/usr/local/lib/python3.7/dist-packages/sklearn/__init__.py'>
```

2. NumPy:

- NumPy is a Python library used for working with arrays.
- It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.
- NumPy stands for Numerical Python.
- NumPy is a Python library and is written partially in Python, but most of the parts that require fast computation are written in C or C++.
- It is a numeric python module which provides fast maths functions for calculations.
- It is used to read data in numpy arrays and for manipulation purpose.

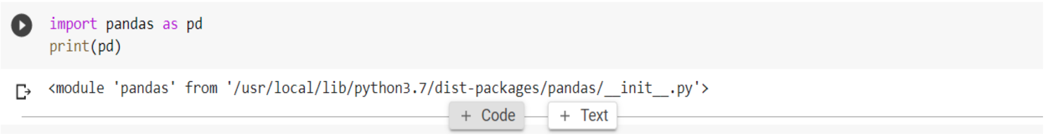


```
[2] import numpy as np
    print(np.pi)

3.141592653589793
```

3. Pandas:

- Pandas is a Python library used for working with data sets.
- It has functions for analyzing, cleaning, exploring, and manipulating data.
- The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.
- Used to read and write different files such as csv.
- Data manipulation can be done easily with dataframes.



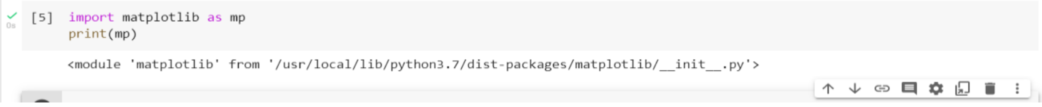
```
import pandas as pd
print(pd)

<module 'pandas' from '/usr/local/lib/python3.7/dist-packages/pandas/__init__.py'>
```

4. Matplotlib:

- Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

- One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc

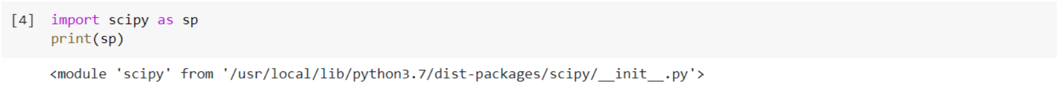


```
[5] import matplotlib as mp
    print(mp)

    <module 'matplotlib' from '/usr/local/lib/python3.7/dist-packages/matplotlib/__init__.py'>
```

5. SciPy:

- SciPy is an open-source Python library which is used to solve scientific and mathematical problems. It is built on the NumPy extension and allows the user to manipulate and visualize data with a wide range of high-level commands. As mentioned earlier, SciPy builds on NumPy and therefore if you import SciPy, there is no need to import NumPy.



```
[4] import scipy as sp
    print(sp)

    <module 'scipy' from '/usr/local/lib/python3.7/dist-packages/scipy/__init__.py'>
```

ALGORITHM NAME	NUMPY	PANDAS	SCIKT-LEARN (SKLEARN)	MATPLOTLIB	XGBOOST
REGRESSION					
LINEAR	Y	Y	Y	Y	
LOGISTIC	Y	Y	Y	Y	
KNN	Y	Y	Y	Y	
CLASSIFICATION					
ID3	Y	Y	Y	Y	
C4.5	Y	Y	Y	Y	
NAÏVE BAYES	Y	Y	Y	Y	
CLUSTERING					
K-MEANS	Y	Y	Y	Y	
K-MEDOID	Y	Y	Y	Y	
SVM	Y	Y	Y	Y	
RANDOM FOREST	Y	Y	Y	Y	
XGBOOST	Y	Y	Y	Y	Y

Numpy Applications:

1. An alternative for lists and arrays in Python

Arrays in Numpy are equivalent to lists in python. Like lists in python, the Numpy arrays are homogenous sets of elements. The most important feature of NumPy arrays is they are homogenous in nature.

This differentiates them from python arrays. It maintains uniformity for mathematical operations that would not be possible with heterogeneous elements. Another benefit of using NumPy arrays is there are a large number of functions that are applicable to these arrays.

These functions could not be performed when applied to python arrays due to their heterogeneous nature.

2. NumPy maintains minimal memory

Arrays in NumPy are objects. Python deletes and creates these objects continually, as per the requirements. Hence, the memory allocation is less as compared to Python lists. NumPy has features to avoid memory wastage in the data buffer.

It consists of functions like copies, view, and indexing that helps in saving a lot of memory. Indexing helps to return the view of the original array, that implements reuse of the data. It also specifies the data type of the elements which leads to code optimization.

3. Using NumPy for multi-dimensional arrays

We can also create multi-dimensional arrays in NumPy.These arrays have multiple rows and columns. These arrays have more than one column that makes these multi-dimensional. Multi-dimensional array implements the creation of matrices.

These matrices are easy to work with. With the use of matrices the code also becomes memory efficient. We have a matrix module to perform various operations on these matrices.

4. Mathematical operations with NumPy

Working with NumPy also includes easy to use functions for mathematical computations on the array data set. We have many modules for performing basic and special mathematical functions in NumPy.

There are functions for Linear Algebra, bitwise operations, Fourier transform, arithmetic operations, string operations, etc.

Numpy Array Applications

1. Shape Manipulations

Users can change array dimensions at runtime if the output produces the same number of elements. We apply np.reshape(…)function on the array. The reshape function is useful for performing various operations. For eg, we use it when we want to broadcast two dissimilar arrays.

2. Array Generation

We can generate array data set for implementing various functions. We can also generate a predefined set of numbers for the array elements using the np.arrange(…)function. Reshape function is useful to generate a different set of dimensions.

We can also use the random function to generate an array having random values. Similarly, we can use linspace function to generate arrays having similar spacing in elements.

We can create arrays with pre-filled ones or zeroes. The default data type is set to be float64 but we can edit the data type using dtype option.

3. Array Dimensions

Numpy consists of both one and multidimensional arrays. Some functions have restrictions on multidimensional arrays. It is then necessary to transform those arrays into one-dimensional arrays. We can transform multi-dimensional to single dimension using np.ravel(..)

Numpy Applications with Other Libraries

1. NumPy with Pandas

Pandas is one of the most important libraries in python for data analysis. Pandas provide high performance, fast analysis, and data cleaning. We use it to manipulate data structures and have data analysis tools.

It consists of a data frame object. It interoperates with NumPy for faster computations. When we use both the libraries together it is a very helpful resource for scientific computations.

2. NumPy with Matplotlib

Matplotlib is a module in NumPy. It is a very helpful tool to work with graphical representations. It consists of a wide range of functions to plot graphs and also manipulate them.

This combination can replace the functionalities of MatLab. It is used to generate the graphs of the results. We enhance it further with the use of graphic toolkits like PyQt and wxPython.

3. NumPy with SciPy

Scipy is an open-source library in Python. It is the most important scientific library in python. It has been built upon the functionalities of NumPy.There are advanced functionalities in SciPy for scientific computations.

We can combine it with NumPy for greater mathematical performance. The combination helps in the implementation of complex scientific operations.

ALGORITHMS:

A) **Linear Regression:**

It is one of the best statistical models that studies the relationship between a dependent variable (Y) with a given set of independent variables (X). The relationship can be established with the help of fitting a best line.

sklearn.linear_model.LinearRegression is the module used to implement linear regression.

B) **Logistic Regression:**

Logistic regression, despite its name, is a classification algorithm rather than regression algorithm. Based on a given set of independent variables, it is used to estimate discrete value (0 or 1, yes/no, true/false). It is also called logit or MaxEnt Classifier.

Basically, it measures the relationship between the categorical dependent variable and one or more independent variables by estimating the probability of occurrence of an event using its logistics function.

sklearn.linear_model.LogisticRegression is the module used to implement logistic regression.

C) **K-NN:**

k-NN (k-Nearest Neighbor), one of the simplest machine learning algorithms, is non-parametric and lazy in nature. Non-parametric means that there is no assumption for the underlying data distribution i.e. the model structure is determined from the dataset. Lazy or instance-based learning means that for the purpose of model generation, it does not require any training data points and whole training data is used in the testing phase.

sklearn.neighbors.NearestNeighbors is the module used to implement unsupervised nearest neighbor learning. It uses specific nearest neighbor algorithms named BallTree, KDTree or Brute Force. In other words, it acts as a uniform interface to these three algorithms.

D) **Decision Tree:**

Decisions tress (DTs) are the most powerful non-parametric supervised learning method. They can be used for the classification and regression tasks. The main goal of DTs is to create a model predicting target variable value by learning simple decision rules deduced from the data features. Decision trees have two main entities; one is root node, where the data splits, and other is decision nodes or leaves, where we got final output.

Decision Tree Algorithms

Different Decision Tree algorithms are explained below −

ID3 algorithm stands for Iterative Dichotomiser 3

It was developed by Ross Quinlan in 1986. It is also called Iterative Dichotomiser 3. The main goal of this algorithm is to find those categorical features, for every node, that will yield the largest information gain for categorical targets.

It lets the tree to be grown to their maximum size and then to improve the tree's ability on unseen data, applies a pruning step. The output of this algorithm would be a multiway tree.

C4.5

It is the successor to ID3 and dynamically defines a discrete attribute that partition the continuous attribute value into a discrete set of intervals. That's the reason it removed the restriction of categorical features. It converts the ID3 trained tree into sets of 'IF-THEN' rules.

In order to determine the sequence in which these rules should applied, the accuracy of each rule will be evaluated first.

C5.0

It works similar as C4.5 but it uses less memory and build smaller rulesets. It is more accurate than C4.5.

CART

It is called Classification and Regression Trees alsgorithm. It basically generates binary splits by using the features and threshold yielding the largest information gain at each node (called the Gini index).

Homogeneity depends upon Gini index, higher the value of Gini index, higher would be the homogeneity. It is like C4.5 algorithm, but, the difference is that it does not compute rule sets and does not support numerical target variables (regression) as well.

Classification with decision trees

In this case, the decision variables are categorical.

Sklearn Module – The Scikit-learn library provides the module name **DecisionTreeClassifier** for performing multiclass classification on dataset.

1	***criterion*** *– string, optional default= "gini"* It represents the function to measure the quality of a split. Supported criteria are "gini" and "entropy". The default is gini which is for Gini impurity while entropy is for the information gain.

E) KMeans:

This algorithm computes the centroids and iterates until it finds optimal centroid. It requires the number of clusters to be specified that's why it assumes that they are already known. The main logic of this algorithm is to cluster the data separating samples in n number of groups of equal variances by minimizing the criteria known as the inertia. The number of clusters identified by algorithm is represented by 'K. Scikit-learn have **sklearn.cluster.KMeans** module to perform K-Means clustering. While computing cluster centers and value of inertia, the parameter named **sample_weight** allows **sklearn.cluster.KMeans** module to assign more weight to some samples.

F) SVM:

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine

G) Random Forest:

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.

H) XGBoost:

XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way.

CONCLUSION:

From this practical, I have successfully learned about libraries which are used for the machine learning applications.