Ans 1.

1. **Comments:** # symbol is being used for comments in python.

Example: print “Hello World” *#* *this is comment*

1. **Type Function:** These python commands are used to check the type of variable and used inbuilt functions to check.

Example: v = 24.35 *# here v is variable*

Print (type(v)) *# here we are printing variable type*

1. **Strings:** It is mainly enclosed in double quotes (“”)

Example: type (“hello world”)

<type ‘str’>

1. **Lists:** Lists are mainly enclosed in square brackets **[ ]**

Example: type ([1, 2]), type is list

type ([1, 2, 3])

<type ‘List’>

1. **Tuple:** Tuple are mainly enclosed in parenthesis. ( )

Example: type (1, 2, 3)

type ((1, 2, 3))

Ans 2.

1. Machine learning (ML) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence.
2. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so.
3. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.
4. A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning.
5. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning.
6. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

Ans 3.

1. **Pandas:** Pandas is primarily used for data analysis, and it is one of the most commonly used Python libraries. It provides you with some of the most useful set of tools to explore, clean, and analyse your data. With Pandas, you can load, prepare, manipulate, and analyse all kinds of structured data. Machine learning libraries also revolve around Pandas DataFrames as an input.
2. **NumPy:** NumPy is mainly used for its support for N-dimensional arrays. NumPy is also used by other libraries such as TensorFlow for their internal computation on tensors. NumPy also provides fast precompiled functions for numerical routines, which can be hard to manually solve. To achieve better efficiency, NumPy uses array-oriented computations, so working with multiple classes becomes easy.
3. **MatplotLib:** This library is responsible for plotting numerical data. And that’s why it is used in data analysis. It is also an open-source library and plots high-defined figures like pie charts, histograms, scatterplots, graphs, etc.

Ans 5.

1. **Mean:** The mean is the average of all numbers and is sometimes called the arithmetic mean.

# Python program to print mean of elements

# list of elements to calculate mean

n\_num = [1, 2, 3, 4, 5]

n = len(n\_num)

get\_sum = sum(n\_num)

mean = get\_sum / n

print("Mean / Average is: " + str(mean))

Output:

Mean / Average is: 3.0

1. **Median:** The median is the middle number in a group of numbers.

# Python program to print median of elements

# list of elements to calculate median

n\_num = [1, 2, 3, 4, 5]

n = len(n\_num)

n\_num.sort()

if n % 2 == 0:

median1 = n\_num[n//2]

median2 = n\_num[n//2 - 1]

median = (median1 + median2)/2

else:

median = n\_num[n//2]

print("Median is: " + str(median))

Output:

Median is: 3

Ans 7.

1. **Training Set:**
   1. A training data set is a data set of examples used during the learning process and is used to fit the parameters (e.g., weights) of, for example, a classifier.
   2. For classification tasks, a supervised learning algorithm looks at the training data set to determine, or learn, the optimal combinations of variables that will generate a good predictive model.
   3. The goal is to produce a trained (fitted) model that generalizes well to new, unknown data.
   4. The fitted model is evaluated using “new” examples from the held-out datasets (validation and test datasets) to estimate the model’s accuracy in classifying new data. To reduce the risk of issues such as over-fitting, the examples in the validation and test datasets should not be used to train the model.
   5. Most approaches that search through training data for empirical relationships tend to overfit the data, meaning that they can identify and exploit apparent relationships in the training data that do not hold in general.
2. **Test Set:**
   1. A test data set is a data set that is independent of the training data set, but that follows the same probability distribution as the training data set. If a model fit to the training data set also fits the test data set well, minimal overfitting has taken place. A better fitting of the training data set as opposed to the test data set usually points to over-fitting.
   2. A test set is therefore a set of examples used only to assess the performance (i.e. generalization) of a fully specified classifier. To do this, the final model is used to predict classifications of examples in the test set. Those predictions are compared to the examples' true classifications to assess the model's accuracy.
   3. In a scenario where both validation and test datasets are used, the test data set is typically used to assess the final model that is selected during the validation process. In the case where the original data set is partitioned into two subsets (training and test datasets), the test data set might assess the model only once (e.g., in the holdout method). Note that some sources advise against such a method.
   4. However, when using a method such as cross-validation, two partitions can be sufficient and effective since results are averaged after repeated rounds of model training and testing to help reduce bias and variability.

Ans 8. Hyperparameter

1. In machine learning, a hyperparameter is a parameter whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are derived via training.
2. Hyperparameters can be classified as model hyperparameters, that cannot be inferred while fitting the machine to the training set because they refer to the model selection task, or algorithm hyperparameters, that in principle have no influence on the performance of the model but affect the speed and quality of the learning process.
3. Different model training algorithms require different hyperparameters, some simple algorithms (such as ordinary least squares regression) require none. Given these hyperparameters, the training algorithm learns the parameters from the data. For instance, LASSO is an algorithm that adds a regularization hyperparameter to ordinary least squares regression, which has to be set before estimating the parameters through the training algorithm.

Ans 11.

1. Pandas: Pandas is primarily used for data analysis, and it is one of the most commonly used Python libraries. It provides you with some of the most useful set of tools to explore, clean, and analyse your data. With Pandas, you can load, prepare, manipulate, and analyse all kinds of structured data. Machine learning libraries also revolve around Pandas DataFrames as an input.

import pandas as pd

# Calling DataFrame constructor

df = pd.DataFrame()

print(df)

# list of strings

lst = ['NFSU', 'IS', 'A', 'UNIVERSITY',

'OF', 'NATIONAL', 'IMPORTANCE']

# Calling DataFrame constructor on list

df = pd.DataFrame(lst)

print(df)

Output:

Empty DataFrame

Columns: []

Index: []

0

0 NFSU

1 IS

2 A

3 UNIVERSITY

4 OF

5 NATIONAL

6 IMPORTANCE

1. NumPy: NumPy is mainly used for its support for N-dimensional arrays. NumPy is also used by other libraries such as TensorFlow for their internal computation on tensors. NumPy also provides fast precompiled functions for numerical routines, which can be hard to manually solve. To achieve better efficiency, NumPy uses array-oriented computations, so working with multiple classes becomes easy.

# using numpy

import numpy

arr = numpy.array ([1,2,3,4,5,6,7,8,9,10])

print (arr)

Output:

[ 1 2 3 4 5 6 7 8 9 10]