Shashank Patel
9369617861

Data Science & ML Task Details

Task 1

NumPy

NumPy (Numerical Python) is a very popular Python library used for working with arrays. It is an open source project that has functions for working in the domain of arrays, linear algebra, fourier transform, and matrices.

Uses:

Lists used in Python instead of arrays are slow to process. NumPy provides an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. **Vectorization** describes the absence of any explicit looping and indexing in the code as these things are taking place in abstraction without us knowing in optimized, pre-compiled C code. Hence it is so much faster than using normal Python code.

Therefore, Numpy is one of the most commonly used packages in Data Science, Machine Learning and Deep Learning as speed and resources are very important in these tasks.

NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently and also because it is optimized to work with the latest CPU architectures.

Arrays in Numpy:

A numpy array is a grid of values of the same data type that is indexed by a tuple (ordered list) of positive numbers. The number of dimensions is called the rank of the array while the shape of an array is a tuple of integers giving the size of the array along each dimension. The elements can be referenced using square brackets [index_of_element] after the variable name.

Import numpy: import numpy as np

Array Creation: a = np.array([1, 2, 3, 4], [10, 11, 12, 3])

Printing Shape: print(a.shape) *// prints (2,4)*

Slicing: print(a[: , 1:3]) *// prints [[2 3]*

[11 12]]

Common Functions:

np.zeros()	np.append()	np.insert()
np.random()	np.resize()	np.split()
np.reshape()	np.squeeze()	np.concatenate()
np.flatten()	np.broadcast()	variable_name.T

Math Functions:

np.sin()	np.cos()	np.tan()	
np.arcsin()	np.arccos()	np.arctan()	
np.degrees()	np.around()	np.ceil()	np.floor()

Linear Algebra:

np.dot()	np.vdot()	np.determinant()
np.inv()	np.inner()	np.matmul()
np.add()	np.inner()	np.multiply()
np.add()	np.divide()	np.subtract()
np.mod()	np.power()	np.reciprocal()

Broadcasting: If the dimensions of two arrays are not same then for element-to-element operations, the smaller array is broadcasted to the size of the larger array so that they have compatible shapes.

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' is a reference to both 'Panel Data' and 'Python Data Analysis'.

Uses:

Pandas allows us to analyze big data and make conclusions based on statistical theories. We can clean messy data sets, and make them readable and relevant with its help. Hence it is an integral part of most data science and Machine Learning pipelines.

Importing: import pandas as pd

Creating DataFrame: pd.DataFrame({'Yes': [50, 21], 'No': [131, 2]})

Common Functions:

df.describe() df.mean() df.unique() df.head()

df.groupby() df.sort_values() df.reset_index()

Functions for Missing Data:

df.nan() df.dropna() df.fillna()

Functions for Transformation:

df.groupby() pd.merge() pd.concat() df.stack()

df.duplicated() df.unstack() df.drop_duplicated()

df.rename() pd.get_dummies()

Getting Data:

pd.read_csv('----')

pd.DataFrame(json.dumps (json.load('---')) ['data'], columns=['field'])