DOCUMENTATION:

***DAY 1:23-02-2024***

MODULE 1:PYTHON FOR DATA ANALYSIS

TASK 1:Review of core Python concepts:data types,operators,control flow,functions,modules,packages.

>IDE-Integrated Development Environment(Google Collab)

>GitHub Repository Creation(Folder Name:Data Analysis)

NUMPY

* Numpy is used for numerical or scientific computation in arrays,vectors and matrices.
* To import the numpy package we just have to use the import statement which is as follows: import numpy as np(our wish).

# Functions in numpy

## ARRAY CREATION:

* **np.array()**- creates an array from a list
* [**DAP\_FINAL**](https://docs.google.com/document/u/0/d/1xtNZKo0JUBNyEch8qn6-_ZMak7GzA4lagS153rdH5vo/edit)**np.zeros()**- creates an array filled with zeros of the specified size
* **np.ones()** - creates an array filled with ones of the specified shape.
* **np.arange()**-Generates numbers up to the given value and we can also starting value,ending value and step value
* For all these functions the default data type is “float” and we can change this using “dtype”.
* ARRAY MANIPULATION:



* + **reshape()**-Reshapes the array into desired shaped array
  + slicing-Slicing is done with **“:”** operator.It prints the elements from the given range.
  + **Transpose**-Transpose is done as follows:**”array\_variable.T”** As the name it returns the transpose.
  + **np.split(a,n)**-Split function splits the given array(a) into n arrays
  + **np.dot(a,b)-**Gives the dot product of matrix a and b
  + **np.linalg.eig(a)-**Gives the eigenvalues and vectors of given matrix
* **np.loadtxt(path,dtype)-**Loads any file from the path that we’ve given and we can give the datatype want like**”dtype=int”**
* **np.savetxt(path,file)-**Loads the file that we’ve given into the path
* **np.random.rand()-**Randomly produces numbers between 0 and 1
* **np.random.randint(a,b)-**Randomly produces between a and b
* **type(a)-**Gives the type of the matrix a
* ndim and shape are used to find the dimension of array and shape the array respectively
  + Invoked as **“Array\_variable.ndim”** and **“Array\_var.shape”**
* Element multiplication is done using **“\*”**operator and Matrix multiplication is done using “@”.
  + EX:a\*b and a@b
* **sum()-**Returns the sum of all the elements in the matrix
  + 
  + EX:a.sum() where a is the matrix
* **max()-**Returns the maximum element in the matrix
* **cumsum()-**Returns the cumulative sum of the elements
* The above three functions have special feature called **“axis”** if axis=1 respective operation is done along the column and if axis=0 then the operations are done along the rows
* **np.vstack(a**,**b) and np.hstack(a.b)-**Used to stack the given two matrices a and b vertically and horizontally respectively.
* **np.dstack(a)-**It used for some changes in matrix a.The changes are:
  + Number of rows become number of groups
  + Number of column become number of rows
  + Number of groups become number of columns



***DAY-2(24/02/2024)***

**●Pandas:**

➢Pandas is a Python library essential for data manipulation and analysis.

➢ Its implifiestaskslikereadingvariousdataformats,cleaning messy datasets, and performing complex data operations.

➢ It is richina set of statistical functions.

➢Pandasalsoplaysavital roleindatapreprocessingformachinelearning. ➢Wecanimportpandasintoourcodebyusingtheimportstatementwhichisimport pandasaspd(our wish).

**➢Functions in pandas:**

➔**pd.series(x,[index]):**Itproduces"x" as a series with provided index values. ➔**pd.read\_csv(path):**It load as csv file from the path that we gave. ➔**pd.read\_excel(path,sheet\_name):**It loadsaxlsxfilefromthepaththatwe gave.Sheetnameisoptionalwecangivethesheetnumberthatwewant to load. ➔**pd.head(n)**-BydefaultHeadgivesthefirst5rowsbutwecanchooseaspecific numberthatwewant(n).

➔**pd.tail(n)**-By default tail givesthelast5rowsbutwecanchooseaspecificnumber thatwewant(n). ➔pd.describe()-Itgivesboththefirst5andlast5rowsasoutput.

**◆FOR THEBELOWdf IS WHERE THE FILE LOADED**

➔**df.describe().**T-Italsogivesbothfirst5andlast5rowsasoutputbut it interchangestherowsandcolumns(Transpose).

➔ **df.shape[]**-By default it gives the number of rows and columns that the file contains but if we give ‘0’ as parameter then it gives only the row count and if we give ‘1’ it gives the count of columns

➔ **df.columns**-It gives the names of the columns.

➔ **df.copy()**-It copies the df file and stores it into some other.

◆ **EX:**- df2=df.copy() Here df is copied in to df2

➔ **df.loc[ ]**-”.loc[]” is primarily label based.

**◆ Allowed inputs are:**



● A single label, e.g. 5 or 'a', (note that 5 is interpreted as a label of the index, and

never as an integer position along the index).

● A list or array of labels, e.g. ['a', 'b', 'c'].

● A slice object with labels, e.g. 'a':'f'.

**➔ df.isnull()**-It returns boolean value if the given column or row is nan then it shows true or else false.

◆ Wecanalsouse this with .sum() function.It gives the number of nan values each column has.

● The syntax is df.isnull().sum()

**➔ df.index==val-**It gives the the row that we’ve given.

● val=row number

● We can also with the help giving range that is as follows ○ df.index.isin(range(n1,n2)) then it prints rows from n1 to n2.

**➔ df.dropna()-**It uses to drop the row that contains none value.

◆ Butwecandrop both rows and columns by using the below function.

● df3.dropna(inplace=True,how='all')

**➔ df[‘column’].mean()**-It is used to find the mean for specified column.

**➔df.fillna(val)**-It is used to fill the none value with whatever the value that we specified. .val=The value that we want to fill.

**➔ df.drop\_duplicates()**-It drops repeated column

**➔df.rename(columns = {'old name':'new name'})**-It renames the old column name with the given new column name.

◆ Wecandothis in other way which is

● df.columns=[new column name]

◆ Wecanalsocreate new column by the help other columns.

**➔ df.groupby(column)**-When you apply the .groupby() function to a column,it returns a GroupBy object. This object allows you to perform various operations on the grouped data such as mean,max and min etc.

**➔df.plot.line()**-It is going plot a graph for the specified column

◆ If we want plot graph for all the columns then we can use the following syntax:

● df.plot.line(subplots= True)



DAY 3:26-02-2024

MATPLOTLIB:

* Matplotlib is a versatile Python library used for data visualisation in fields like data analysis, scientific computing, and machine learning.
* It offers various plot types, including line plots, scatter plots, and histograms, making it suitable for exploratory data analysis.
* Matplotlib is also employed in machine learning projects for visualising model performance and feature distributions.
* Additionally, it supports geospatial data visualisation and interactive plotting capabilities, enabling users to create maps and dynamic visualisations.
* Overall, its flexibility, extensive functionality, and ease of use make it a popular choice.
* To import matplotlib library the syntax is:
  + **"import matplotlib.pyplot as plt” .**
* FUNCTIONS IN MATPLOTLIB:
* **plt.plot(parameters)-**It is used to plot for the given information as parameters.
  + It also accepts the “colour” parameter where we can give the colour of the graph.
* **plt.title(“title\_name”)-**This function is used to give a title to graph
  + The title is passed as a parameter in the quotes.
* **plt.legend(loc=”location”)-**There are more parameters to plot function such as labels where we can label the things for which we are plotting a graph.
  + This function works based on the labels.Here we can locate where we want to place the labels and the location is specified to the **“loc” .**
* **plt.show()-**This function is used to show our graph.
* **plt.subplot()-**This function gives the separate graph,without this function all the graphs will be shown in a single graph
* **plt.pie()-**This function is used to give pie chart based on the information given by us as the parameters
* **plt.xlabel()-**It is used to give the name or label to the x axis.
* **plt.ylabel()-**It is used to give the name or label to the y axis.

SEABORN:

* Seaborn, a Python data visualisation library, simplifies the creation of statistical graphics for data exploration and analysis.
* It offers high-level interfaces for visualising relationships between variables, handling categorical data, and creating heatmaps and matrices.
* It provides tools for visualising time series data and customising plot aesthetics.
* Seamlessly integrating with Pandas, Seaborn enables users to leverage its visualisation capabilities directly on DataFrame objects.
* To import seaborn the syntax is:
  + Import seaborn as sns.
* Seaborn has 14 -15 datasets in it as default.They are:

1. Anscombe: Anscombe's quartet dataset.

2. attention: Response times in a psychological experiment.

3. brain\_networks: Coordinates of networks in the human brain.

4. car\_crashes: Insurance data about car crashes.

5. diamonds: Characteristics of diamonds.

6. dots: Lateralized response times in a psychological experiment.

7. exercise: Measurements of exercise patterns.

8. flights: Data about flights.

9. fmri: Functional magnetic resonance imaging (fMRI) data.

10. gammas: Brain activity during exposure to gamma rays.

11. iris: Iris flower data.

12. mpg: Miles per gallon (MPG) and various car attributes.

13. planets: Exoplanets data.

14. tips: Restaurant tips data.

* To load these default datasets the command is as follows:
  + **var\_name=sns.load\_dataset(“dataset name”)**
* FUNCTIONS IN SEABORN:
* Seaborn is mainly used for plotting which have it consists many types of plotting functions.They are:
  + **sns.scatterplot()**
  + **sns.jointplot()**
  + **Sns.line plot()**
  + **sns.violinplot()**
  + **sns.lmplot()**
  + **sns.barplot()**
  + **sns.boxplot()**
  + **sns.countplot()**
  + **sns.FacetGrid()**
  + **sns.heatmap()**

All these functions gives different plotting style graphs as output.

DAY4:27-02-2024

Machine learning can be done in 3 steps

1)training

2)testing

3)processing

Types of machine learning

1)supervised machine learning : labelled

2)unsupervised machine learning : unlabelled

3)semi-supervised machine learning : both labelled and unlabelled

NEURAL NETWORK**:**  inter-connection of the neurons

CNN:

A convolutional neural network(CNN) is a type of deep learning.neural network architecture commonly used in computer vision.computer vision is a field of artificial intelligence that enables a computer to understand and interpret the image or visual data .

Types of layers:

1.input layer

2.hidden layer

convolutional layer

Activation layer

max pooling layer, average layer

dense layer

3.output layer

1.INPUT LAYER:

Its the layer in which we give input to our model.In CNN,generally,the input will be an image or a sequence of images.

2.CONVOLUTIONAL LAYER:

This is the layer,which is used to extract the feature from input dataset.it applies a set of learnable filters known as the kernels/filters to the input images.

the output of this layer is referred ad feature maps.suppose we use a total of 12 filters for this layer we’ll get an output volume of dimension 32x32x12

3.ACTIVATION LAYER:

By adding an activation function to the output of the preceding layer,activation layers add nonlinearity to the network.it will apply an element wise activation function to the output of the convolution.

**ACTIVATION FUNCTION:**

The activation function decides whether a neuron should be activated or not by calculating the weighted sum and further adding bias to it.the purpose of the activation function is to introduce non-linearity into the output of a neuron.

Activation function make the back-propagation possible since the gradients are supplied along with the error to update the weights and biases.

TYPES OF ACTIVATION FUNCTION:

**1)tanh -** range: -1 to +1,nature-nonlinear,hidden layer,back-propagation.

**2)sigmoid** - A=1/(1+e-x) formula,range=0 to 1,it is used in output layer of a binary classification.If it is less than 0.5 then it is considered as 0 else 1.

**3)relu -** formula A(x)=max(0,x),range : [0,infinity],nature:- non-linear, multiple layers of neuron being activated by the Relu function.it gives fast response and calculations(computation).it is the best for error corrected fastly. It is a rectified linear,hidden layer

**4)softmax -** nature : non-linear,output layer,it can handle multi-class classification problems,range: 0 to 1.it is very useful to predict the probability.

4.POOLING LAYER:

This layer is periodically inserted in the convnets and its main function is to reduce the size of volume which makes the computation fast, reduce memory and also prevents overfitting.

POOLING LAYERS are two types:

1.max pooling layer-16x16x12

2.average pooling layer

5.OUTPUT LAYER:

The output from the fully connected layers is then fed into a logistic function for classification tasks like sigmoid or softmax which converts the output of each class into the probability score of each class.

DAY 5:28-02-2024

LINEAR REGRESSION:

* It learns from labelled datasets and maps the data points to the most optimised linear functions.
* These points can be used for prediction on new datasets.
* We have two variables,they are dependent and independent.

The formulae is y=a+bx.To find

1. We have to calculate the mean for both the dependent and independent.
2. Find the differences between x point and x’(mean x).
3. Find the differences between y point and y’(mean y).
4. Find the total sum of squares of (x-x’).
5. Find the total sum of products of (x-x’) and (y-y’).

* b=sum of product of (x-x’) and (y-y’).

total sum of squares of (x-x’)

* Now with the help of **b** we can find **a** by replacing **x** with **x’** and **y** with **y’** in **y=ax+b.**
* The main two functions in this linear regression are fit() and predict().

LOGISTIC REGRESSION:

* It is used for binary classification and is learned from supervised algorithms.
* But the only difference is this will actually tells us which class does our prediction site in.

SIGMOID FUNCTION:

* When the model is confident then it shows a narrow decision boundary
* When the model is not confident the it shows a wide decision boundary.

**DAY 6:29-02-2024**

DECISION TREE:

* Decision trees in machine learning provide an effective method for making decisions.
* This is because they lay out the problem and all the possible outcomes.
* They have **nodes** and **leafs.**
  + **Nodes:** They have both the true and false branches.
  + **leaf:** It has either true or false in other words it shows us the result and where our outcome sits.
* Decision trees recurrently(continuously) split the data until it gets pure nodes(leaf).

DISADVANTAGES:

* It has no accuracy.
* It won't get to a conclusion until it recurrently splits the data which wastes time.

RANDOM FOREST:

* Collection of many decision trees is called Random Forest.
* We can frequently find **bootstrapping** and **aggregation.**
* Random forest creates a new dataset from the existing or the original dataset.
  + This process is called **“bootstrapping”.**
* This gives the more accurate answers when compared to the decision tree.

* CONDITIONS:
  + Select rows that are different and cover all the rows.
  + Select features that are different and cover all features.
  + Selecting the same data(rows) and same features have no use.

Day 7:1-03-2024

BIG DATA:

* Big Data refers to vast and complex datasets that cannot be effectively managed or processed using traditional data processing tools

Volume:

* Volume refers to the immense amount of data generated and collected from various sources such as social media,sensors,transactions and more.

Velocity:

* Velocity denotes the speed at which data is generated ,collected,processed.
* In today’s interconnected world,data is produced rapidly in real-time or near-rear-time from sources like social media updates,sensor data from IOT devices,online transactions and more.
* Efficient processing of this streaming data is crucial for real-time analytics and decision making.

Variety:

* Variety refers to diversity of data types and sources.
* Data comes in various formats including structured data(like databases),semi-structured data(like XML-Extensible markup language,JSON-java script object Notation)and unstructured data(like text,images,videos).
* Big data platforms must be able to handle this diverse range of data types efficiently.Veracity

Veracity:

emphasizes the quality and reliability of the data .

* Due to the sheer volume and variety of data sources,ensuring data quality and reliability can be challenging.
* Big data analytics often involves dealing with incomplete,inconsistent or inaccurate data,requiring robust mechanisms for data cleaning validation,and quality assurance.

STATISTICAL ANALYSIS WITH SCIPY AND STATS MODELS:Hypothesis Testing,Regression Analysis,ANOVA

SCIPY :

* SCIPY is an open-Source scientific computing library for python that builds on numpy.
* It provides many additional functionalities compared to numpy ,including Optimization,Integration,Interpolation,Egien-value problems,signal and Image Procesing,Statistical distributions and much more.

STAT MODELS:

* Stat model is a python library that provides classes and functions for estimating and testing statistical models.
* It is built in top of Numpy,SciPy and Matplotlib.And it integrates with Pandas for data handling.
* Stat models includes a wide range of statistical models and tests,making it a powerful tool for statistical analysis and hypothesis testing.

HYPOTHESIS TESTING:

**t\_statistic :**

* The t\_static is a measure of how many standard deviations the sample mean is away from the hypothesised population mean relative to that variability in the sample.
* It is calculated as the difference between the sample mean and the population mean divided by the standard error of the sample mean.
* The larger the absolute value of the t\_static the stronger the evidence against the null hypothesis.

**P\_value:**

* The value is the probability of observing a text statistic or one or more (extreme) theif null hypothesis is true.
* It quantifies the strength of the evidence against the null hypothesis.
* A small p value (typically less than 0.05) integrates that the observed data is unlikely under the assumption that the null hypothesis is true leading to the rejection of the null hypothesis in favour of the alternative hypothesis.
* Conversely a large p\_value suggests that the observed data is under the null hypothesis leading to a failure to reject the null hypothesis.

HYPOTHESIS TESTING:

* If the p\_value isb less than the predetermined significance value(suggest 0.05) it is typically interpreted as sufficient evidence to reject the null hypothesis.
* If the p\_value is greater than the significance level there is not enough evidence to reject the null hypothesis.
* Together t\_static and p\_value it is determining whether the observed sample data provides enough evidence to support a claim of hypothesis about the population.

Day 8:2-03-2024

TIME SERIES ANALYSIS:

* A time series is a sequence of observations recorded at regular time intervals. Depending on the frequency of observations, a time series can be hourly, daily, weekly, monthly, quarterly, or annual. It can even include seconds or minute-wise data, such as the number of clicks or user visits per minute. [Time series data is crucial for forecasting various aspects like demand, sales, stock prices, and website](https://www.machinelearningplus.com/time-series/time-series-analysis-python/) .

Importing Time Series Data in Python:

* df = pd.read\_csv('path/to/your/dataset.csv', parse\_dates=['date']).

Window Function:

* A window function operates on a group of rows (a “window”) and computes a value for each row based on that group.
* These functions allow us to perform calculations that involve neighboring rows, such as aggregations, ranking, and moving averages.
* The **“OVER”** clause defines the window over which the calculation should be performed.

Rolling Function:

* Rolling functions, also known as moving window functions, operate on a fixed-width subset (window) of time series data.
* The window slides through the data, calculating a value for each observation within the window.
* These functions allow us to analyze trends, volatility, correlations, and other aspects that change over time

BEAUTIFUL SOUP:

* Beautiful Soup is a Python library that is commonly used for web scraping purposes to pull the data out of HTML files. It provides Pythonic idioms for iterating, searching, and modifying the parse tree, making it easy to extract and manipulate data from web pages.

FUNCTIONS:-

**Requests:**

* Purpose: To send HTTP requests and retrieve the HTML content of web pages.
* requests.get() function in Python, it's a part of the requests library and is commonly used to send HTTP GET requests to a specified URL and retrieve the response.

**Pandas:**

* Purpose: To manipulate and analyze tabular data, often used to store and process data extracted from web pages.
* 3.In Beautiful Soup, the find() function is commonly used to locate the first occurrence of a specified tag or a group of tags based on various criteria. It's a powerful method for navigating and searching the parse tree. The find\_all() function is another commonly used method that returns a list of all occurrences of the specified tags.

**Day-9:4-03-2024**

HADOOP LIBRARY:

* + It is an open source framework designed to process and store big data in a distributed environment.
  + Programming model is a simple on called as map reduce and distributed file system it is called hdfs(hadoop distributed file system)

HDFS:

* + System that stores data across multiple machines provides high access to application data.

MAP REDUCE:

* + A programming model for processing and generating large data sets.
  + It involves two phases:
  + 1)MAP-PHASE:
  + Where data is divided into smaller chunks and processed in parallel.
  + 2)REDUCE-PHASE:
  + Where the result from the map phase are aggregated to produce the final output

YARN:(YET ANOTHER RESOURCE NEGOTIATOR)

* + It is a resource management layer that manages resources in hadoop clusters and scheduling user applications.

EXAMPLE1:

Word count using hadoop:

* + From the larger collection of documents we are going to count no of words by using hw

step1:Input Data

* + adoop.
  + e have a large collection of documents stored in HDFS.

step2:Map-Phase

* + In this phase each document is divided into words.Each word is emitted as a key value path.(word,1,where word is the key and 1 is the value).

step3:Shuffle and Sort-Phase

* + The output of the Map-Phase is sorted based on keys.
  + All values corresponding to the same key.all grouped together.

Step4:Reduce Phase

* + In this phase each unique word is passed to a reducer.
  + The reducer sum’s up the value corresponding to each word giving the count of occurrences of that word.
  + The output of the reduce phase is the final word count.

step5:Output Data

* + The final output containing the count of each word is stored in HDFS or any other desired location.

CONCLUSION:

* + Hadoop provides scalable and cost effective solution for processing and storing big data.by distributing data and computation across multiple nodes.it enables the processing of large dataset
  + ex:Word Count
  + principle:Hadoop mapreduce framework.