CS3361 DATA SCIENCE LABORATORY L T P C
0 0 4 2

LIST OF EXPERIMENTS:

1. Download, install and explore the features of NumPy, SciPy, Jupyter, Statsmodels and Pandas packages.
2. Working with Numpy arrays
3. Working with Pandas data frames
4. Reading data from text files, Excel and the web and exploring various commands for doing descriptive analytics on the Iris data set.
5. Use the diabetes data set from UCI and Pima Indians Diabetes data set for performing the following:
    a. Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis.
    b. Bivariate analysis: Linear and logistic regression modeling
    c. Multiple Regression analysis
    d. Also compare the results of the above analysis for the two data sets.
6. Apply and explore various plotting functions on UCI data sets.
    a. Normal curves
    b. Density and contour plots
    c. Correlation and scatter plots
    d. Histograms
    e. Three dimensional plotting
7. Visualizing Geographic Data with Basemap

Ex.No: 1 Download, Install and Explore the Features of NumPy, Scipy, Jupyter, Statsmodels, and Pandas Package

Aim:

To know the procedure of how to download and install Python and various modules and also explore the features of NumPy, SciPy, Jupyter, Statsmodels and Pandas packages.

To check whether Python is installed or not?

Open **Command Prompt**-> Type **Python** or **Py** and then hit enter. If python is installed it will show the version details.

```
C:\Users\Student>cd\
C:\>python
Python 2.7.18 (v2.7.18:8d21aa21f2, Apr 20 2020, 13:25:05) [MSC v.1500 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

PIP

Pip stands for '**Pip Installs Packages**' and it can be used as a standard package manager. We can install it on any operating system.

Check if PIP is installed or not?

Navigate your command line to the location of Python's script directory and type **pip –version**

```
C:\Users\Student>cd\
C:\>cd python27
C:\Python27>cd scripts
C:\Python27\Scripts>pip --version
pip 19.2.3 from c:\python27\lib\site-packages\pip (python 2.7)
C:\Python27\Scripts>
```

Steps to Change system variable:

On the **Windows taskbar, right-click the Windows icon** and select **System**. In the **Settings** window, under **Related Settings**, click **Advanced System Settings**. On the **Advanced** tab, click **Environment variables.** In the **Environment Variables window** in **System Variables** session select **PATH** and then click **Edit.** In the **Edit Environment Variable** window select **new** and paste the location of Python (C:\Python27) and the press OK.

NumPy

NumPy stands for **Numerical Python**. It is a **Python library** used for working with an array. It is the core library for scientific computing, which contains a powerful n-dimensional array object. In Python, we use the **list** for purpose of the array but it's slow to process. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

Why is NumPy Faster than Lists?

We use python NumPy array instead of a list because of the below three reasons:

1. Less Memory

2. Fast
3. Convenient

NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently. This behavior is called **locality of reference** in computer science. This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.

How do I install NumPy?

If you have **Python** and **PIP** already installed on a system, then installation of NumPy is very easy. Go to your **command prompt** and type **"pip install numpy"**

Import NumPy

Once NumPy is installed, import it in your applications by adding the import keyword: **"import numpy".** Now NumPy is imported and ready to use.

```
import numpy
arr = numpy.array([1, 2, 3, 4, 5])
print(arr)
```

OUTPUT:
[1 2 3 4 5]

SciPy

SciPy is a free and open-source library in Python that is used for scientific and mathematical computations. It is pronounced as Sigh Pie. This is an extension of NumPy. It contains a wide range of algorithms and functions to do mathematical calculations, manipulating, and visualizing data.

Using this module we can perform the following operations:

- Linear algebra
- Integration
- Optimization
- Interpolation
- Signal and Image processing
- ODE solvers

Advantages of using Python SciPy

- It is Open-source
- It is fast and easy to use.
- Contains high-level commands and classes to do visualization and manipulation of data.
- It also includes classes, and web and database routines that support parallel programming.

Installing Python SciPy using pip

Using pip we can install SciPy using the command **"pip install scipy".** SciPy provides a function called find() that returns the value of constants.

Displaying various constant values:

```
import scipy.constants as constant
print("The value of the constant pi is: {}".format(constant.pi))
print("The valu of the Golden ratio is: {}".format(constant.golden_ratio))
print("Speed of light in vacuum is: {}".format(constant.c))
```

```
print("Gravitational Constant is: {}".format(constant.G))
print("Molar Gas Constant is: {}".format(constant.R))
print("Boltzman Constant is: {}".format(constant.k))
print("Proton mass Constant is: {}".format(constant.proton_mass))
```

OUTPUT:

The value of the constant pi is: 3.14159265359
The valu of the Golden ratio is: 1.61803398875
Speed of light in vacuum is: 299792458.0
Gravitational Constant is: 6.67408e-11
Molar Gas Constant is: 8.3144598
Boltzman Constant is: 1.38064852e-23
Proton mass Constant is: 1.672621898e-27

Calculating the area of the circle:

```
import scipy.constants as constant
def Area_of_Circle(r):
   return (constant.pi * r * r)
r = int(input("Enter the radius of the circle: "))
print("Area of Circle with radius: {}".format(Area_of_Circle(r)))
```

OUTPUT:

Enter the radius of the circle: 7
Area of Circle with radius: 153.938040026

Solving Two variables systems of Linear Equations

The **linalg.solve** function is used to solve the given linear equations. It is used to evaluate the equations automatically and find the values of the unknown variables.

```
import scipy
from scipy import linalg
import numpy as np
arrlist=[]
alist=[]
alist1=[]
print("Solving Two Variable SYSTEMS OF LINEAR EQUATIONS:")
value = int(input("Enter the Equation 1 x coefficient:"))
arrlist.append(value)
value = int(input("Enter the Equation 1 y coefficient:"))
arrlist.append(value)
a=np.array(arrlist)
print(a)
value  = int(input("Enter the Equation 2 x coefficient:"))
alist.append(value)
value = int(input("Enter the Equation 2 y coefficient:"))
alist.append(value)
b=np.array(alist)
print(b)
c=np.array([arrlist,alist])
```

```
value=int(input("Enter the Equation 1 constant term:"))
alist1.append(value)
value=int(input("Enter the Equation 2 constant term:"))
alist1.append(value)
d=np.array(alist1)
print(d)
res=linalg.solve(c,d)
print("The value of x and y is: {}".format(res))
```

OUTPUT:

```
Solving Two Variable SYSTEMS OF LINEAR EQUATIONS:
Enter the Equation 1 x coefficient:2
Enter the Equation 1 y coefficient:1
[2 1]
Enter the Equation 2 x coefficient:4
Enter the Equation 2 y coefficient:-5
[ 4 -5]
Enter the Equation 1 constant term:5
Enter the Equation 2 constant term:7
[5 7]
The value of x and y is: [2.28571429 0.42857143]
```

Scipy.linalg.inv → This function is used to find the inverse of the matrix.
Scipy.linalg.det → This function is used to find the determinant of the matrix.
Scipy.linalg.eig →This function takes a complex or real matrix M whose Eigen values and Eigen vectors are to be evaluated.

Jupyter

Project Jupyter is a comprehensive software suite for interactive computing, that includes various packages such as Jupyter Notebook, QtConsole, nbviewer, JupyterLab. Jupyter Notebook is an open-source web-application that can be used to write, execute, store, and share code in a single document file. Although using Jupyter notebook we can write and execute code for 40 different programming languages.

Cell types

At the top of your notebook, there is a dropdown that allows you to change the cell type.

Python

A Python cell contains Python code. This can be any valid code, so all these are allowed:

- One simple line of code
- A Python import statement
- A function definition
- A complete class definition

Markdown

Markdown is a lightweight markup language for creating formatted text. You can use it to create headers, lists, code snippets. Markdown is great for adding documentation to your notebook.

Formulas

You can enter nice-looking formulas using Markdown as well. Internally, Jupyter uses **MathJax** to format and display formulas. A formula starts and ends with either a single or a double dollar sign:

$ .... $: in-line; the formula is in line with the rest of the current sentence
$$ .. $$: display mode; the formula stands out from the rest of the text
Inside, you need to enter a TeX-based equation.

Statsmodels

Statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration. Extensive lists of result statistics are available for each estimator. The results are tested against existing statistical packages to ensure that they are correct.

This library or package is created on top of the SciPy and NumPy packages and also makes the data handling by using pandas and has the patsy interface for the formula that resembles the R-like. The matplotlib is the library from which the graphics functions are used. Many other Python packages consider this one the base for creating statistics libraries. Even now, many of the statistical models, tools for plotting, and new models are emerging and introduced in the market with continuous development by the team of statsmodel.

Why StatsModels?

We can work with statistics in a way that no other platform will allow us as the statsmodel itself is made, keeping the purpose of hardcore statistics in mind. It has more inclination towards R and is a perfect tool when analyzing statistical things. Most of the developers who program in R can make use of this and can easily make their move to Python using this package.

Installing Python Statsmodels using pip

Using pip we can install Statsmodels using the command **"python -m pip install statsmodels".**

Pandas

A panda 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.

Why Use Pandas?

Pandas allow us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant.
Relevant data is very important in data science.
Data Science: is a branch of computer science where we study how to store, use and analyze data for deriving information from it.

What Can Pandas Do?

Pandas give you answers about the data. Like:

- Is there a correlation between two or more columns?
- What is average value?
- Max value?
- Min value?

Pandas are also able to delete rows that are not relevant, or contain wrong values, like empty or NULL values. This is called *cleaning* the data.

Installation of Pandas

If you have Python and PIP already installed on a system, then installation of Pandas is very easy. Install it using this command: **"pip install pandas"**

Example program:

```
import pandas
mydataset = {
  'cars': ["BMW", "Volvo", "Ford"],
  'passings': [3, 7, 2]
}
myvar = pandas.DataFrame(mydataset)
print(myvar)
```

OUTPUT:

```
cars  passings
0    BMW        3
1  Volvo        7
2   Ford        2
```

What is a Series?

A Pandas Series is like a column in a table. It is a one-dimensional array holding data of any type.

```
import pandas as pd
a = [1, 7, 2]
myvar = pd.Series(a)
print(myvar)
```

OUTPUT:

```
0    1
1    7
2    2
dtype: int64
```

Labels

If nothing else is specified, the values are labeled with their index number. First value has index 0, second value has index 1 etc. This label can be used to access a specified value.

```
import pandas as pd
a = [1, 7, 2]
myvar = pd.Series(a)
print(myvar[0])
```

OUTPUT:

```
1
```

Create Labels

With the index argument, you can name your own labels.

```
import pandas as pd
a = [1, 7, 2]
myvar = pd.Series(a, index = ["x", "y", "z"])
```

print(myvar)

OUTPUT:
x 1
y 7
z 2
dtype: int64

DataFrames

Data sets in Pandas are usually multi-dimensional tables, called DataFrames. Series is like a column, a DataFrame is the whole table.

What is a DataFrame?

A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns.

```
import pandas as pd
data = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}
df = pd.DataFrame(data)
print(df)
```

OUTPUT:

	calories	duration
0	420	50
1	380	40
2	390	45

Locate Row

DataFrame is like a table with rows and columns. Pandas use the loc attribute to return one or more specified row(s).

```
import pandas as pd
data = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}
df = pd.DataFrame(data)
print(df.loc[[0, 1]])
```

OUTPUT:

	calories	duration
0	420	50
1	380	40

Result:

Thus the procedure of how to download and install Python and its various modules has been learned. Also explored the features of NumPy, SciPy, Jupyter, Statsmodels and Pandas packages have been done successfully.

Ex.No: 2(a) Working with NumPy Arrays

Aim:

To know various Python NumPy array operations.

Creating 1-D array:

```
import numpy as np
a=np.array([1,2,3])
print (a)
```

OUTPUT:

```
[1 2 3]
```

Creating 2-D array:

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
print (a)
```

OUTPUT:

```
[[1 2 3]
 [4 5 6]]
```

Initializing Numpy Arrays:

Initialize an array x X y dimension with 0

- It creates an array of size 3X4 with all values as zeros.

```
import numpy as np
a=np.zeros((3,4))
print(a)
```

OUTPUT:

```
[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]
```

Arranging the numbers between x and y with an interval of z

- This prints the numbers between 1 and 10 with an interval of 2.

```
import numpy as np
a=np.arange(1,10,2)
print(a)
```

OUTPUT:

```
[1 3 5 7 9]
```

- It prints all the even numbers between 10 and 20.

```
import numpy as np
a=np.arange(10,20,2)
print(a)
```

OUTPUT:

[10 12 14 16 18]

Arranging z numbers between x and y

- This prints 6 numbers between 5 and 10

```
import numpy as np
a=np.linspace(5,10,6)
print(a)
```

OUTPUT:

[5. 6. 7. 8. 9. 10.]

```
import numpy as np
a=np.linspace(5,10,10)
print(a)
```

OUTPUT:

[5. 5.5555555 6 6.11111111 6.6666666 7 7.22222222 7.7777777 8 8.33333333 8.8888888 9 9.44444444 10.]

Filling same number in an array of dimension x X y

```
import numpy as np
a=np.full((4,3),6)
print(a)
```

OUTPUT:

[[6 6 6]
 [6 6 6]
 [6 6 6]
 [6 6 6]]

ndim

- It returns the dimension of the array.

```
import numpy as np
a = np.array([(1,2,3),(4,5,6)])
print(a.ndim)
```

OUTPUT:

2

itemsize:

- Calculate the byte size of each element.

```
import numpy as np
a = np.array([(1,2,3)])
print(a.itemsize)
```

OUTPUT:

4

dtype

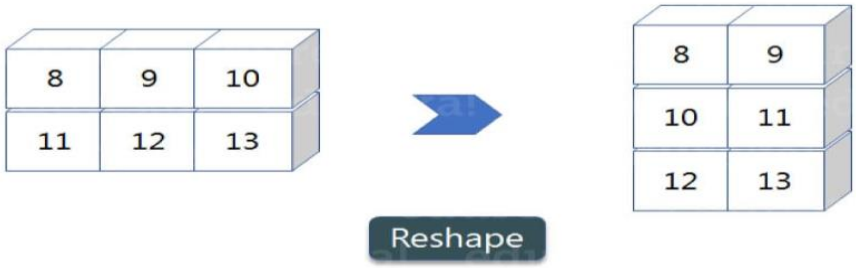
- It displays the data type of the elements that are stored in an array.

import numpy as np
a = np.array([(1,2,3)])
print(a.dtype)

OUTPUT:
int32

reshape

- Reshape is when you change the number of rows and columns which gives a new view to an object.



As we have 3 columns and 2 rows which has converted into 2 columns and 3 rows.

import numpy as np
a = np.array([(8,9,10),(11,12,13)])
print(a)
a=a.reshape(3,2)
print(a)

OUTPUT:
[[8 9 10] [11 12 13]] [[8 9] [10 11] [12 13]]

Slicing

- Slicing is extracting particular set of elements from an array.

import numpy as np
a=np.array([(1,2,3,4),(3,4,5,6)])
print(a[0,2])

OUTPUT:
3

Here, the array (1,2,3,4) is your index 0 and (3,4,5,6) is index 1 of the python numpy array. Therefore, we have printed the second element from the zeroth index.

min/max

- Find the minimum, maximum as well the sum of the numpy array.

```
import numpy as np
a= np.array([1,2,3])
print(a.min())
print(a.max())
print(a.sum())
```

OUTPUT:

1 3 6

Arithmetic Operations

Addition

```
import numpy as np
a = np.sum([10,20])
print(a)
```

OUTPUT:

30

```
import numpy as np
a,b=25,30
c = np.sum([a,b])
print(c)
```

OUTPUT:

55

- This program adds all the values 1+2+5+6=14

```
import numpy as np
c = np.sum([[1,2],[5,6]])
print(c)
```

OUTPUT:

14

- This program prints the added value 1+5=6 and 2+6=8

```
import numpy as np
c = np.sum([[1,2],[5,6]],axis=0)
print(c)
```

OUTPUT:

[6 8]

- This program prints the added value of 1+2=3 and 5+6=11

```
import numpy as np
c = np.sum([[1,2],[5,6]],axis=1)
print(c)
```

OUTPUT:
[3 11]

Subtraction

```
import numpy as np
a=np.array([2,4,6])
b=np.array([1,2,4])
c=np.subtract(a,b)
print(c)
```

OUTPUT:
[1 2 2]

Multiplication

```
import numpy as np
a=np.array([2,4,6])
b=np.array([1,2,4])
c=np.multiply(a,b)
print(c)
```

OUTPUT:
[2 8 24]

Division

```
import numpy as np
a=np.array([2,4,6])
b=np.array([1,2,4])
c=np.divide(a,b)
print(c)
```

OUTPUT:
[2 2 1]

Square Root & Standard Deviation:

```
import numpy as np
a=np.array([(1,2,3),(3,4,5,)])
print(np.sqrt(a))
print(np.std(a))
```

OUTPUT: [[1. 1.41421356 1.73205081] [1.73205081 2. 2.23606798]]
1.29099444874

Matrix Arithmetic Operation

- **Addition Operation:** Addition, subtraction, multiplication and division of the two matrices.

```
import numpy as np
x= np.array([(1,2,3),(3,4,5)])
y= np.array([(1,2,3),(3,4,5)])
print(x+y)
print(x-y)
print(x*y)
print(x/y)
```

OUTPUT: [[2 4 6] [6 8 10]]
[[0 0 0] [0 0 0]]
[[1 4 9] [9 16 25]]
[[1. 1. 1.] [1. 1. 1.]]

Vertical & Horizontal Stacking

- Concatenate two arrays and not just add them; you can perform it using two ways – *vertical stacking* and *horizontal stacking*.

```
import numpy as np
x= np.array([(1,2,3),(3,4,5)])
y= np.array([(1,2,3),(3,4,5)])
print(np.vstack((x,y)))
print(np.hstack((x,y)))
```

OUTPUT: [[1 2 3] [3 4 5] [1 2 3] [3 4 5]]
[[1 2 3 1 2 3] [3 4 5 3 4 5]]

Ravel

- Convert one numpy array into a single column.

```
import numpy as np
x= np.array([(1,2,3),(3,4,5)])
print(x.ravel())
```

OUTPUT: [1 2 3 3 4 5]

Result:

Thus the various Python Numpy array operations have been learned successfully.

Ex.No: 2(b) Working with NumPy Arrays to Find the Sum and Average

Aim:

To write a python program to work with NumPy arrays to find the sum and average of the array elements.

Algorithm:

1. Start the program.
2. Get the total number of array elements and assign it to the variable Number.
3. Repeat step 3 for Number+1 times
    a. Get the array elements one at a time and append it to the array name of arrlist.
4. Assign the array elements into arr.
5. Calculate the total and average by using array elements.
6. Print the Sum of array elements and average of array elements.
7. Stop the program.

Program:

```
import numpy as np
arrlist = []
Number = int(input("Total Array Elements to enter = "))
for i in range(1, Number + 1):
   value = int(input("Please enter the %d Array Value = "  %i))
   arrlist.append(value)
arr = np.array(arrlist)
total = 0
for i in range(len(arr)):
   total = total + arr[i]
avg = total / len(arr)
print("The sum of array elements={}".format(total))
print("The Average of Array Elements={}".format(avg))
```

OUTPUT:

```
Total Array Elements to enter = 5
Please enter the 1 Array Value = 6
Please enter the 2 Array Value = 5
Please enter the 3 Array Value = 4
Please enter the 4 Array Value = 3
Please enter the 5 Array Value = 2
The sum of array elements=20
The Average of Array Elements=4
```

Result:

Thus the python program to work with NumPy arrays to find the sum and average of the array elements is executed and output is verified.

Ex.No: 3 Working with Pandas Data Frames

Aim:

To write a python program to work with Pandas data frame to access the dataset which is stored in a file and load them into a data frame.

Load Files Into a DataFrame

If your data sets are stored in a file, Pandas can load them into a DataFrame.

Storing data in a file:

Open Notepad type data set with fields Duration,Pulse,Maxpulse,Calories and click Save As from File menu. In the Save As dialog box enter the file name as "data" with extension ".csv("Comma Seperted Values")" that is "data.csv".

Algorithm:

1. Start the program.
2. Import pandas module as pd.
3. Read the data set which is stored in the file named as "data.csv".
4. Print the data set.
5. Stop the program.

Program:

```
import pandas as pd
df = pd.read_csv('C:\Python27\data.csv')
print(df)
```

OUTPUT:

	Duration	Pulse	Maxpulse	Calories
0	60	110	130	409.1
1	60	117	145	479.0
2	60	103	135	340.0
3	45	109	175	282.4
4	45	117	148	406.0
..	...	...	...	...
164	60	105	140	290.8
165	60	110	145	300.4
166	60	115	145	310.2
167	75	120	150	320.4
168	75	125	150	330.4

[169 rows x 4 columns]

Result:

Thus the program to work with Pandas data frame to access the dataset which is stored in a file and load them into a data frame is executed and output is verified.

Ex.No: 4(a) Reading Data from Text Files and Exploring Various Commands for doing Descriptive Analytics on the Iris Data Set

Aim:

To write a python program to read data from text files and explore various commands for doing descriptive analytics on the Iris data set.

Exploratory Data Analysis (EDA):

It is a technique to analyze data using some visual Techniques. With this technique, we can get detailed information about the statistical summary of the data. Also we are able to deal with the duplicates values, outliers, and also see some trends or patterns present in the dataset.

Iris Dataset

Iris Dataset is considered as the Hello World for data science. It contains five columns namely – Petal Length, Petal Width, Sepal Length, Sepal Width, and Species Type. Iris is a flowering plant; the researchers have measured various features of the different iris flowers and recorded them digitally.

Algorithm (Program 1):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in read mode.
4. Print the content of the text file.
5. Close the file.
6. Stop the program.

Algorithm (Program 2):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in read mode.
4. Read first 5 characters from the text file and print it.
5. Close the file.
6. Stop the program.

Algorithm (Program 3):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in read mode.
4. Read first 15 characters from the text file and print it.
5. Close the file.
6. Stop the program.

Algorithm (Program 4):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in read mode.
4. Read first line from the text file and print it.
5. Close the file.
6. Stop the program.

Algorithm (Program 5):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in read mode.
4. Read first three lines from the text file and print it.
5. Close the file.
6. Stop the program.

Algorithm (Program 6):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in append mode.
4. Write the line “Now the file has more content!” at the end of the text file.
5. Close the file.
6. Open the text file “iris.txt” in read mode.
7. Print the content of the text file.
8. Stop the program.

Algorithm (Program 7):

1. Start the program.
2. Import pandas module as pd
3. Open the text file “iris.txt” in write mode.
4. Write the line “I have deleted the content!” on the text file.
5. Close the file.
6. Open the text file “iris.txt” in read mode.
7. Print the content of the text file.
8. Stop the program.

Program 1:

```
import pandas as pd
f = open("c:\Python27\iris.txt","r")
print(f.read())
f.close()
```

Program 2:

```
import pandas as pd
f = open("c:\Python27\iris.txt","r")
print(f.read(5))
```

```
f.close()
```

Program 3:

```
import pandas as pd
f = open("c:\Python27\iris.txt","r")
print(f.read(15))
f.close()
```

Program 4:

```
import pandas as pd
f = open("c:\Python27\iris.txt","r")
print(f.readline())
f.close()
```

Program 5:

```
import pandas as pd
f = open("c:\Python27\iris.txt","r")
print(f.readline())
print(f.readline())
print(f.readline())
f.close()
```

Program 6:

```
import pandas as pd
f = open("c:\Python27\iris.txt","a")
f.write("Now the file has more content!")
f.close()
f = open("c:\Python27\iris.txt","r")
print(f.read())
```

Program 7:

```
import pandas as pd
f = open("c:\Python27\iris.txt","w")
f.write("I have deleted the content!")
f.close()
f = open("c:\Python27\iris.txt","r")
print(f.read())
```

OUTPUT 1:

```
sepallength,sepalwidth,petallength,petalwidth,class
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
```

4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
…….
6.5,3.0,5.2,2.0,Iris-virginica
6.2,3.4,5.4,2.3,Iris-virginica
5.9,3.0,5.1,1.8,Iris-virginica

OUTPUT 2:
Sepal

OUTPUT 3:
sepallength,sep

OUTPUT 4:
sepallength,sepalwidth,petallength,petalwidth,class

OUTPUT 5:
sepallength,sepalwidth,petallength,petalwidth,class

5.1,3.5,1.4,0.2,Iris-setosa

4.9,3.0,1.4,0.2,Iris-setosa

OUTPUT 6:
ssepallength,sepalwidth,petallength,petalwidth,class
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
…….
6.2,3.4,5.4,2.3,Iris-virginica
5.9,3.0,5.1,1.8,Iris-virginica
Now the file has more content!

OUTPUT 7:
I have deleted the content!

Result:

Thus the program to read data from text files and exploring various commands for doing descriptive analytics on the Iris data set is executed and output is verified.

Ex.No: 4(b) Reading Data from Excel Files and Exploring Various Commands for doing Descriptive Analytics on the Iris Data Set

Aim:

To write a python program to read data from excel files and explore various commands for doing descriptive analytics on the Iris data set.

How to install xlrd

Navigate your command line to the location of Python's script directory and type **"python –m pip install xlrd"**

C:\Users\Student>cd
C:\>cd python27
C:\Python27>cd scripts
C:\Python27\Scripts> python –m pip install xlrd

Algorithm:

1. Start the program.
2. Import numpy module as np.
3. Import pandas module as pd
4. Read the excel file from the location.
5. Print the data from the excel file.
6. The function df.head() displays the first 5 rows by default.
7. The function df.tail() displays the last 5 rows by default.
8. The function df.shape() displays the number of rows and number of columns.
9. The function df.info() displays the information about the data frame.
10. The function df[0:10][‘petal_length’] displays the 10 rows with the column petal length.
11. The function df["sepal_length"].mean() displays the mean value of the sepal length.
12. The function df["sepal_length"].mode()[0] displays the mode of the column sepal length.
13. The function df["sepal_length"].max() displays the maximum value of the sepal length.
14. The function df["sepal_length"].min() displays the minimum value of the sepal length.
15. The function df.corr() displays the relationship between each column in the data set. The number varies from -1 to +1.
16. Stop the program.

Program:

```
import numpy as np
import pandas as pd
df=pd.read_excel("C:\Python27\iris.xls")
print(df)
print("By default head() displays first 5 rows:\n{}".format(df.head()))
print("By default tail() displays last 5 rows:\n{}".format(df.tail()))
print("The number of rows and columns in the Excel file is:{}".format(df.shape))
print("Information about the Data Frame is:")
print(df.info())
print("Displays first 10 rows of the column Petal_length:")
```

```
print(df[0 : 10]['petal_length'])
print("The Mean value of the Sepal length is:{}".format(df["sepal_length"].mean()))
print("The Mode value of the Sepal length is:{}".format(df["sepal_length"].mode()[0]))
print("The Maximun Sepal length is:{}".format(df["sepal_length"].max()))
print("The Maximun Sepal length is:{}".format(df["sepal_length"].min()))
print(df.corr())
```

OUTPUT:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa
15	5.7	4.4	1.5	0.4	setosa
16	5.4	3.9	1.3	0.4	setosa
17	5.1	3.5	1.4	0.3	setosa
18	5.7	3.8	1.7	0.3	setosa
19	5.1	3.8	1.5	0.3	setosa
20	5.4	3.4	1.7	0.2	setosa
21	5.1	3.7	1.5	0.4	setosa
..	...	...	...	...	...
141	6.9	3.1	5.1	2.3	virginica
142	5.8	2.7	5.1	1.9	virginica
143	6.8	3.2	5.9	2.3	virginica
144	6.7	3.3	5.7	2.5	virginica
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

[150 rows x 5 columns]

By default head() displays first 5 rows:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

By default tail() displays last 5 rows:

	sepal_length	sepal_width	petal_length	petal_width	species
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

The number of rows and columns in the Excel file is:(150, 5)
Information about the Data Frame is:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
sepal_length 150 non-null float64
sepal_width 150 non-null float64
petal_length 150 non-null float64
petal_width 150 non-null float64
species 150 non-null object
dtypes: float64(4), object(1)
memory usage: 5.9+ KB
None
Displays first 10 rows of the column Petal_length:
0 1.4
1 1.4
2 1.3
3 1.5
4 1.4
5 1.7
6 1.4
7 1.5
8 1.4
9 1.5
Name: petal_length, dtype: float64
The Mean value of the Sepal length is:5.84333333333
The Mode value of the Sepal length is:5.0
The Maximun Sepal length is:7.9
The Maximun Sepal length is:4.3

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	1.000000	-0.109369	0.871754	0.817954
sepal_width	-0.109369	1.000000	-0.420516	-0.356544
petal_length	0.871754	-0.420516	1.000000	0.962757
petal_width	0.817954	-0.356544	0.962757	1.000000

Result:

Thus the program to read data from excel file and exploring various commands for doing descriptive analytics on the Iris data set is executed and output is verified.

Ex.No: 4(c) Reading Data from Web and Exploring Various Commands for doing Descriptive Analytics on the Iris Data Set

Aim:

To write a python program to read data from web and explore various commands for doing descriptive analytics.

How to install xlrd

Navigate your command line to the location of Python's script directory and type **"python –m pip install urllib3"**

C:\Users\Student>cd
C:\>cd python27
C:\Python27>cd scripts
C:\Python27\Scripts> python –m pip install urllib3

Algorithm:

1. Start the program.
2. Import urllib3 module.
3. Get the URL of the website which we want to access.
4. Print the status.
5. Print the data of the website.
6. Stop the program.

Program:

```
import urllib3
http = urllib3.PoolManager()
r = http.request('GET','https://www.google.com/')
print(r.status)
print(r.data)
```

OUTPUT:

200

<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="en-IN"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-Type"><meta content="/images/branding/googleg/1x/googleg_standard_color_128dp.png" itemprop="image"><title>Google</title><script nonce="rE7U4SA-nMzRBl01B670ww">(function(){window.google={kEI:'GwNYY8nyMtGF3LUP69WXqA8',kEXPI:'0,1302536,56873,6059,206,4804,2316,383,246,5,5367,1123753,1197755,646,380090,16114,19397,9287,22431,1361,12312,17587,4998,13228,3847,10622,22741,6674,1279,2742,149,1103,840,6297,3514,606,2023,1777,520,14670,3227,2845,7,4811,28959,4465,13142,3,346,230,1014,1,5444,151,11321,2652,4,1528,2304,7039,22023,5708,7357,18095,16786,5818,2539,4094,4052,3,3541,1,11942,30212,2,14022,2715,3534,7867,9865,1758,5679,1020,2381,28742,4567,6259,23418,1252,5835,14968,4332,8,7476,445,2,2,1,17312,9320,8155,7381,2,1478,14490,873,19633,7,1922,5784,3995,8213,10917,1509,15515,17015,123,700,4,1,2,2,2,2,8652,928,4337,2551,978,3022,6299,2,2020,14,82,1531,226,1119,1014,482,269,202,1866,

643,5981,933,51,2685,922,613,249,1079,341,213,259,1150,782,2824,382,15,1171,456,1125,1538,2249,4326,3,15,857,702,185,829,39,224,2628,1295,964,101,207,1284,77,1437,2346,84,597,213,227,1,714,230,403,588,219,326,1287,156,61,2,44,230,204,393,49,486,443,7,240,247,96,148,570,793,61,707,1446,144,12,287,242,4,102,1,134,979,342,346,261,2,38,275,86,609,424,311,213,483,2,337,51,73,12,141,490,1058,265,1531,5317326,106,5995768,2803376,3311,141,795,19736,299,4,44,1754,468,17,20,18,23948208,2770167,1271975,1964,3094,13578,3102,304,5595,11,5713,1681',kBL:'lHUN'};google.sn='webhp';google.kHL='en-IN';})();(function(){
var f=this||self;var h,k=[];function l(a){for(var b;a&&(!a.getAttribute||!(b=a.getAttribute("eid")));)a=a.parentNode;return b||h}function m(a){for(var b=null;a&&(!a.getAttribute||!(b=a.getAttribute("leid")));)a=a.parentNode;return b}
function n(a,b,c,d,g){var e="";c||-1!==b.search("&ei=")||(e="&ei="+l(d),-1===b.search("&lei=")&&(d=m(d))&&(e+="&lei="+d));d="";!c&&f._cshid&&-1===b.search("&cshid=")&&"slh"!==a&&(d="&cshid="+f._cshid);c=c||"/"+(g||"gen_204")+"?atyp=i&ct="+a+"&cad="+b+e+"&zx="+Date.now()+d;/^http:/i.test(c)&&"https:"===window.location.protocol&&(google.ml&&google.ml(Error("a"),!1,{src:c,glmm:1}),c="");return c};h=google.kEI;google.getEI=l;google.getLEI=m;google.ml=function(){return null};google.log=function(a,b,c,d,g){if(c=n(a,b,c,d,g)){a=new Image;var e=k.length;k[e]=a;a.onerror=a.onload=a.onabort=function(){delete k[e]};a.src=c}};google.logUrl=n;}).call(this);(function(){
google.y={};google.sy=[];google.x=function(a,b){if(a)var c=a.id;else{do c=Math.random();while(google.y[c])}google.y[c]=[a,b];return!1};google.sx=function(a){google.sy.push(a)};google.lm=[];google.plm=function(a){google.lm.push.apply(google.lm,a)};google.lq=[];google.load=function(a,b,c){google.lq.push([[a],b,c])};google.loadAll=function(a,b){google.lq.push([a,b])};google.bx=!1;google.lx=function(){};}).call(this);google.f={};(function(){
document.documentElement.addEventListener("submit",function(b){var a;if(a=b.target){var c=a.getAttribute("data-submitfalse");a="1"===c||"q"===c&&!a.elements.q.value?!0:!1}else a=!1;a&&(b.preventDefault(),b.stopPropagation())},!0);document.documentElement.addEventListener("click",function(b){var
a;a:{for(a=b.target;a&&a!==document.documentElement;a=a.parentElement)if("A"===a.tagName){a="1"===a.getAttribute("data-nohref");break a}a=!1}a&&b.preventDefault()},!0);}).call(this);</script><style>#gbar,#guser{font-size:13px;padding-top:1px !important;}#gbar{height:22px}#guser{padding-bottom:7px !important;text-align:right}.gbh,.gbd{border-top:1px solid #c9d7f1;font-size:1px}.gbh{height:0;position:absolute;top:24px;width:100%}@media all{.gb1{height:22px;margin-right:.5em;vertical-align:top}#gbar{float:left}}a.gb1,a.gb4{text-decoration:underline !important}a.gb1,a.gb4{color:#00c !important}.gbi .gb4{color:#dd8e27 !important}.gbf .gb4{color:#900 !important}
</style><style>body,td,a,p,.h{font-family:arial,sans-serif}body{margin:0;overflow-y:scroll}#gog{padding:3px 8px 0}td{line-height:.8em}.gac_m td{line-height:17px}form{margin-bottom:20px}.h{color:#1558d6}em{font-weight:bold;font-style:normal}.lst{height:25px;width:496px}.gsfi,.lst{font:18px arial,sans-serif}.gsfs{font:17px arial,sans-serif}.ds{display:inline-box;display:inline-block;margin:3px 0 4px;margin-left:4px}input{font-family:inherit}body{background:#fff;color:#000}a{color:#4b11a8;text-decoration:none}a:hover,a:active{text-decoration:underline}.fl
a{color:#1558d6}a:visited{color:#4b11a8}.sblc{padding-top:5px}.sblc

a{display:block;margin:2px 0;margin-left:13px;font-size:11px}.lsbb{background:#f8f9fa;border:solid 1px;border-color:#dadce0 #70757a #70757a #dadce0;height:30px}.lsbb{display:block}#WqQANb a{display:inline-block;margin:0 12px}.lsb{background:url(/images/nav_logo229.png) 0 -261px repeat-x;border:none;color:#000;cursor:pointer;height:30px;margin:0;outline:0;font:15px arial,sans-serif;vertical-
align:top}.lsb:active{background:#dadce0}.lst:focus{outline:none}</style><script nonce="rE7U4SA-
nMzRBl01B670ww">(function(){window.google.erd={jsr:1,bv:1670,de:true};
var h=this||self;var k,l=null!=(k=h.mei)?k:1,n,p=null!=(n=h.sdo)?n:!0,q=0,r,t=google.erd,v=t.jsr;google.ml=function(a,b,d,m,e){e=void 0===e?2:e;b&&(r=a&&a.message);if(google.dl)return google.dl(a,e,d),null;if(0>v){window.console&&console.error(a,d);if(-2===v)throw a;b=!1}else b=!a||!a.message||"Error loading script"===a.message||q>=l&&!m?!1:!0;if(!b)return null;q++;d=d||{};b=encodeURIComponent;var c="/gen_204?atyp=i&ei="+b(google.kEI);google.kEXPI&&(c+="&jexpid="+b(google.kEXPI));c+="&srcpg="+b(google.sn)+"&jsr="+b(t.jsr)+"&bver="+b(t.bv);var f=a.lineNumber;void 0!==f&&(c+="&line="+f);var g=
a.fileName;g&&(0<g.indexOf("-extension:/")&&(e=3),c+="&script="+b(g),f&&g===window.location.href&&(f=document.documentElement.outerHTML.split("\n")[f],c+="&cad="+b(f?f.substring(0,300):"No script found.")));c+="&jsel="+e;for(var u in d)c+="&",c+=b(u),c+="=",c+=b(d[u]);c=c+"&emsg="+b(a.name+": "+a.message);c=c+"&jsst="+b(a.stack||"N/A");12288<=c.length&&(c=c.substr(0,12288));a=c;m||google.log(0,"",a);return a};window.onerror=function(a,b,d,m,e){r!==a&&(a=e instanceof Error?e:Error(a),void 0===d||"lineNumber"in a||(a.lineNumber=d),void 0===b||"fileName"in a||(a.fileName=b),google.ml(a,!1,void 0,!1,"SyntaxError"===a.name||"SyntaxError"===a.message.substring(0,11)||-1!==a.message.indexOf("Script error")?3:0));r=null;p&&q>=l&&(window.onerror=null)};})();</script></head><body bgcolor="#fff"><script nonce="rE7U4SA-nMzRBl01B670ww">(function(){var src='/images/nav_logo229.png';var iesg=false;document.body.onload = function(){window.n && window.n();if (document.images){new Image().src=src;}
if (!iesg){document.f&&document.f.q.focus();document.gbqf&&document.gbqf.q.focus();}
}
})();</script> <nobr>Search Images Maps Play YouTube News Gmail Drive <a class=gb1 style="text-decoration:none"
href="https://www.google.co.in/intl/en/about/products?tab=wh"><u>More</u>
»</nobr> <nobr>Web History | Settings | <a target=_top id=gb_70

href="https://accounts.google.com/ServiceLogin?hl=en&passive=true&continue=https://www.google.com/&ec=GAZAAQ" class=gb4>Sign in</nobr> <br clear="all" id="lgpd"> <img alt="Google" height="92" src="/images/branding/googlelogo/1x/googlelogo_white_background_color_272x92dp.png" style="padding:28px 0 14px" width="272" id="hplogo">

 <form action="/search" name="f"><table cellpadding="0" cellspacing="0"><tr valign="top"><td width="25%"> </td><td align="center" nowrap=""><input name="ie" value="ISO-8859-1" type="hidden"><input value="en-IN" name="hl" type="hidden"><input name="source" type="hidden" value="hp"><input name="biw" type="hidden"><input name="bih" type="hidden"> <input class="lst" style="margin:0;padding:5px 8px 0 6px;vertical-align:top;color:#000" autocomplete="off" value="" title="Google Search" maxlength="2048" name="q" size="57"> <br style="line-height:0"><input class="lsb" value="Google Search" name="btnG" type="submit"><input class="lsb" id="tsuid_1" value="I'm Feeling Lucky" name="btnI" type="submit"><script nonce="rE7U4SA-nMzRBl01B670ww">(function(){var id='tsuid_1';document.getElementById(id).onclick = function(){if (this.form.q.value){this.checked = 1;if (this.form.iflsig)this.form.iflsig.disabled = false;} else top.location='/doodles/';};})();</script><input value="AJiK0e8AAAAAY1gRK5l_w0C5WrkmOEWv7Y4H6KtHlgbi" name="iflsig" type="hidden"></td><td class="fl sblc" align="left" nowrap="" width="25%">Advanced search</td></tr></table><input id="gbv" name="gbv" type="hidden" value="1"><script nonce="rE7U4SA-nMzRBl01B670ww">(function(){
var a,b="1";if(document&&document.getElementById)if("undefined"!=typeof XMLHttpRequest)b="2";else if("undefined"!=typeof ActiveXObject){var c,d,e=["MSXML2.XMLHTTP.6.0","MSXML2.XMLHTTP.3.0","MSXML2.XMLHTTP","Microsoft.XMLHTTP"];for(c=0;d=e[c++];)try{new ActiveXObject(d),b="2"}catch(h){}}a=b;if("2"==a&&-1==location.search.indexOf("&gbv=2")){var f=google.gbvu,g=document.getElementById("gbv");g&&(g.value=a);f&&window.setTimeout(function(){location.href=f},0)};}).call(this);</script></form>
 <style>#gws-output-pages-elements-homepage_additional_languages__als{font-size:small;margin-bottom:24px}#SIvCob{color:#3c4043;display:inline-block;line-height:28px;}#SIvCob a{padding:0 3px;}.H6sW5{display:inline-block;margin:0 2px;white-space:nowrap}.z4hgWe{display:inline-block;margin:0 2px}</style> Google offered in: हिन्दी বাংলা తెలుగు <a href="https://www.google.com/setprefs?sig=0_kHoC7jahVesilKxhHg57m365mns%3D&

p;hl=mr&source=homepage&sa=X&ved=0ahUKEwiJ7L2r2_v6AhXRArcAHevqBfUQ2ZgBCAg">मराठी தமிழ் ગુજરાતી ಕನ್ನಡ മലയാളം ਪੰਜਾਬੀ
Advertising ProgramsBusiness SolutionsAbout GoogleGoogle.co.in © 2022 - Privacy - Terms <script nonce="rE7U4SA-nMzRBl01B670ww">(function(){window.google.cdo={height:757,width:1440};(function(){var a=window.innerWidth,b=window.innerHeight;if(!a||!b){var c=window.document,d="CSS1Compat"==c.compatMode?c.documentElement:c.body;a=d.clientWidth;b=d.clientHeight}a&&b&&(a!=google.cdo.width||b!=google.cdo.height)&&google.log("","","/client_204?&atyp=i&biw="+a+"&bih="+b+"&ei="+google.kEI);}).call(this);})();</script> <script nonce="rE7U4SA-nMzRBl01B670ww">(function(){google.xjs={ck:'xjs.hp.iHfCprUw_lc.L.X.O',cs:'ACT90oGUabG-jqyM7vztwfEcZ-EtZrFQeg',excm:[]};})();</script> <script nonce="rE7U4SA-nMzRBl01B670ww">(function(){var
u='/xjs/_/js/k\x3dxjs.hp.en.OXyHHoQ5vB8.O/am\x3dAAB0AgBQAKAC/d\x3d1/ed\x3d1/rs\x3dACT90oEVwJR5DpgprhAi94VZWMVfvNGcYA/m\x3dsb_he,d';
var d=this||self,e=function(a){return a};
var g;var l=function(a,b){this.g=b===h?a:""};l.prototype.toString=function(){return this.g+""};var h={};function n(){var a=u;google.lx=function(){p(a);google.lx=function(){}};google.bx||google.lx()}
function p(a){google.timers&&google.timers.load&&google.tick&&google.tick("load","xjsls");var b=document;var c="SCRIPT";"application/xhtml+xml"===b.contentType&&(c=c.toLowerCase());c=b.createElement(c);if(void 0===g){b=null;var k=d.trustedTypes;if(k&&k.createPolicy){try{b=k.createPolicy("goog#html",{createHTML:e,createScript:e,createScriptURL:e})}catch(q){d.console&&d.console.error(q.message)}g=b}else g=b}a=(b=g)?b.createScriptURL(a):a;a=new l(a,h);c.src=a instanceof l&&a.constructor===l?a.g:"type_error:TrustedResourceUrl";var

f,m;(f=(a=null==(m=(f=(c.ownerDocument&&c.ownerDocument.defaultView||window).document).querySelector)?void 0:m.call(f,"script[nonce]"))?a.nonce||a.getAttribute("nonce")||"":"")&&c.setAttribute("nonce",f);document.body.appendChild(c);google.psa=!0};google.xjsu=u;setTimeout(function(){n()},0);})();function _DumpException(e){throw e;}
function _F_installCss(c){}
(function(){google.jl={blt:'none',chnk:0,dw:false,dwu:true,emtn:0,end:0,ine:false,injs:'none',injt:0,injth:0,injv2:false,lls:'default',pdt:0,rep:0,snet:true,strt:0,ubm:false,uwp:true};})();(function(){var
pmc='{\x22d\x22:{},\x22sb_he\x22:{\x22agen\x22:true,\x22cgen\x22:true,\x22client\x22:\x22heirloom-hp\x22,\x22dh\x22:true,\x22dhqt\x22:true,\x22ds\x22:\x22\x22,\x22ffql\x22:\x22en\x22,\x22fl\x22:true,\x22host\x22:\x22google.com\x22,\x22isbh\x22:28,\x22jsonp\x22:true,\x22msgs\x22:{\x22cibl\x22:\x22Clear Search\x22,\x22dym\x22:\x22Did you mean:\x22,\x22lcky\x22:\x22I\\u0026#39;m Feeling Lucky\x22,\x22lml\x22:\x22Learn more\x22,\x22oskt\x22:\x22Input tools\x22,\x22psrc\x22:\x22This search was removed from your \\u003Ca href\x3d\\\x22/history\\\x22\\u003EWeb History\\u003C/a\\u003E\x22,\x22psrl\x22:\x22Remove\x22,\x22sbit\x22:\x22Search by image\x22,\x22srch\x22:\x22Google Search\x22},\x22ovr\x22:{},\x22pq\x22:\x22\x22,\x22refpd\x22:true,\x22rfs\x22:[],\x22sbas\x22:\x220 3px 8px 0 rgba(0,0,0,0.2),0 0 0 1px rgba(0,0,0,0.08)\x22,\x22sbpl\x22:16,\x22sbpr\x22:16,\x22scd\x22:10,\x22stok\x22:\x22OIYWeCLDw57PIelVqpOlIKUh0FM\x22,\x22uhde\x22:false}}';google.pmc=JSON.parse(pmc);})();</script> </body></html>

Result:

Thus the program to read data from web and exploring various commands for doing descriptive analytics is executed and output is verified.

Ex.No: 5(a) Univariate Analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis

Aim:

To write a python program to perform Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis using the Pima Indians diabetes data set.

Skewness is a statistical term and it is a way to estimate or measure the shape of a distribution. It is an important statistical methodology that is used to estimate the asymmetrical behavior rather than computing frequency distribution. Skewness can be two types:

Symmetrical: A distribution can be called symmetric if it appears the same from the left and right from the center point.

Asymmetrical: A distribution can be called asymmetric if it doesn't appear the same from the left and right from the center point.

Distribution on the basis of skewness value:

- **Skewness = 0:** Then normally distributed.
- **Skewness > 0:** Then more weight in the left tail of the distribution.
- **Skewness < 0:** Then more weight in the right tail of the distribution.

Kurtosis:

It is also a statistical term and an important characteristic of frequency distribution. It determines whether a distribution is heavy-tailed in respect of the normal distribution. It provides information about the shape of a frequency distribution.

- Kurtosis for normal distribution is equal to 3.
- For a distribution having kurtosis < 3: It is called playkurtic.
- For a distribution having kurtosis > 3, It is called leptokurtic and it signifies that it tries to produce more outliers rather than the normal distribution.

Algorithm:

1. Start the program.
2. Import numpy, pandas and scipy modules.
3. Open Diabetes.csv file to get the data frame.
4. Frequency of the element age is calculated and displayed it.
5. Average of the element age is calculated and displayed.
6. Median of the element age is calculated and displayed.
7. Mode of the element age is calculated and displayed it.
8. Variance of the element age is calculated and displayed it.
9. Standard deviation of the element age is calculated and displayed it.
10. Skewness for the data set is calculated and displayed it.
11. Kurtosis for the data set is calculated and displayed it.
12. Stop the program.

Program:

```
import numpy
import pandas as pd
from scipy.stats import skew
from scipy.stats import kurtosis
df=pd.read_csv('C:\Python27\Diabetes.csv')
df1 = pd.Series(df["Age"]).value_counts().sort_index().reset_index().reset_index(drop=True)
df1.columns = ['Element', 'Frequency']
print("The list frequency of elements is :\n{}".format(df1))
print("Average age is:{}".format(df["Age"].mean()))
print("Median of the field age is:{}".format(df["Age"].median()))
print("Mode is:{}".format(df["Age"].mode()[0]))
print("Variance is:{}".format(df["Age"].var()))
print("Standard Deviation is:{}".format(df["Age"].std()))
print("Skewness for the data set:\n [Pregnancies Glucose BloodPressure SkinThickness Insulin BMI DiabetesPedigreeFunction Age Outcome \n{}".format(skew(df, axis=0, bias=True)))
print("Kurtosis for the data set:\n [Pregnancies Glucose BloodPressure SkinThickness Insulin BMI DiabetesPedigreeFunction Age Outcome \n{}".format(kurtosis(df, axis=0, bias=True)))
```

OUTPUT:

```
The list frequency of elements is :
    Element  Frequency
0        21         63
1        22         72
2        23         38
3        24         46
4        25         48
5        26         33
6        27         32
7        28         35
8        29         29
9        30         21
10       31         24
11       32         16
12       33         17
13       34         14
14       35         10
15       36         16
16       37         19
17       38         16
18       39         12
19       40         13
20       41         22
21       42         18
```

22 43 13
23 44 8
24 45 15
25 46 13
26 47 6
27 48 5
28 49 5
29 50 8
30 51 8
31 52 8
32 53 5
33 54 6
34 55 4
35 56 3
36 57 5
37 58 7
38 59 3
39 60 5
40 61 2
41 62 4
42 63 4
43 64 1
44 65 3
45 66 4
46 67 3
47 68 1
48 69 2
49 70 1
50 72 1
51 81 1

Average age is:33.2408854167
Median of the field age is:29.0
Mode is:22
Variance is:138.30304589
Standard Deviation is:11.7602315407
Skewness for the data set:
[Pregnancies Glucose BloodPressure SkinThickness Insulin BMI DiabetesPedigreeFunction Age Outcome
[0.89991194 0.17341396 -1.84000523 0.10915876 2.26781046 -0.42814328 1.9161592 1.12738926 0.6337757]
Kurtosis for the data set:
[Pregnancies Glucose BloodPressure SkinThickness Insulin BMI DiabetesPedigreeFunction Age Outcome
[0.15038274 0.62881333 5.13869066 -0.52449449 7.15957492 3.26125742 5.55079205 0.63117694 -1.59832836]

Result:

Thus the python program to perform Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis using the Pima Indians diabetes data set are executed and output is verified.

Ex.No: 5(b)(i) Bivariate Analysis: Linear and logistic Regression Modeling

Aim:

To write a python program to perform Bivariate analysis: Linear and logistic regression modeling.

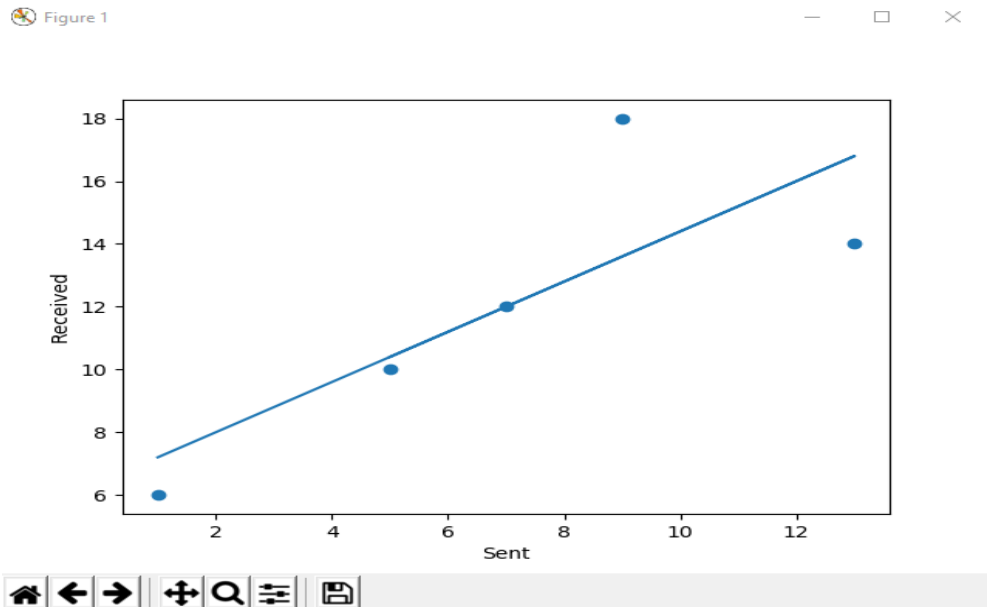
Algorithm:

1. Start the program.
2. Initialize array x and y.
3. Calculate the slope intercept, correlation coefficient and standard error using the linregression() function.
4. Plot the scatter plot with label Sent in X-axis and Received in Y-axis.
5. Show the Scatter plot.
6. Stop the program.

Program:

```
import matplotlib.pyplot as plt
from scipy import stats
x=[5,7,13,9,1]
y=[10,12,14,18,6]
slope,intercept,r,p,std_err=stats.linregress(x,y)
def myfunc(x):
   return slope *x+intercept
mymodel=list(map(myfunc,x))
plt.scatter(x,y)
plt.plot(x,mymodel)
plt.xlabel('Sent')
plt.ylabel('Received')
plt.show()
```

OUTPUT:



Result:

Thus the python program to perform Bivariate analysis: Linear and logistic regression modeling are executed and output is verified.

Ex.No: 5(b)(ii) Bivariate Analysis: Linear Regression Modeling

Aim:

To write a python program to perform Bivariate analysis: Linear regression modeling using Pima Indians diabetes data set.

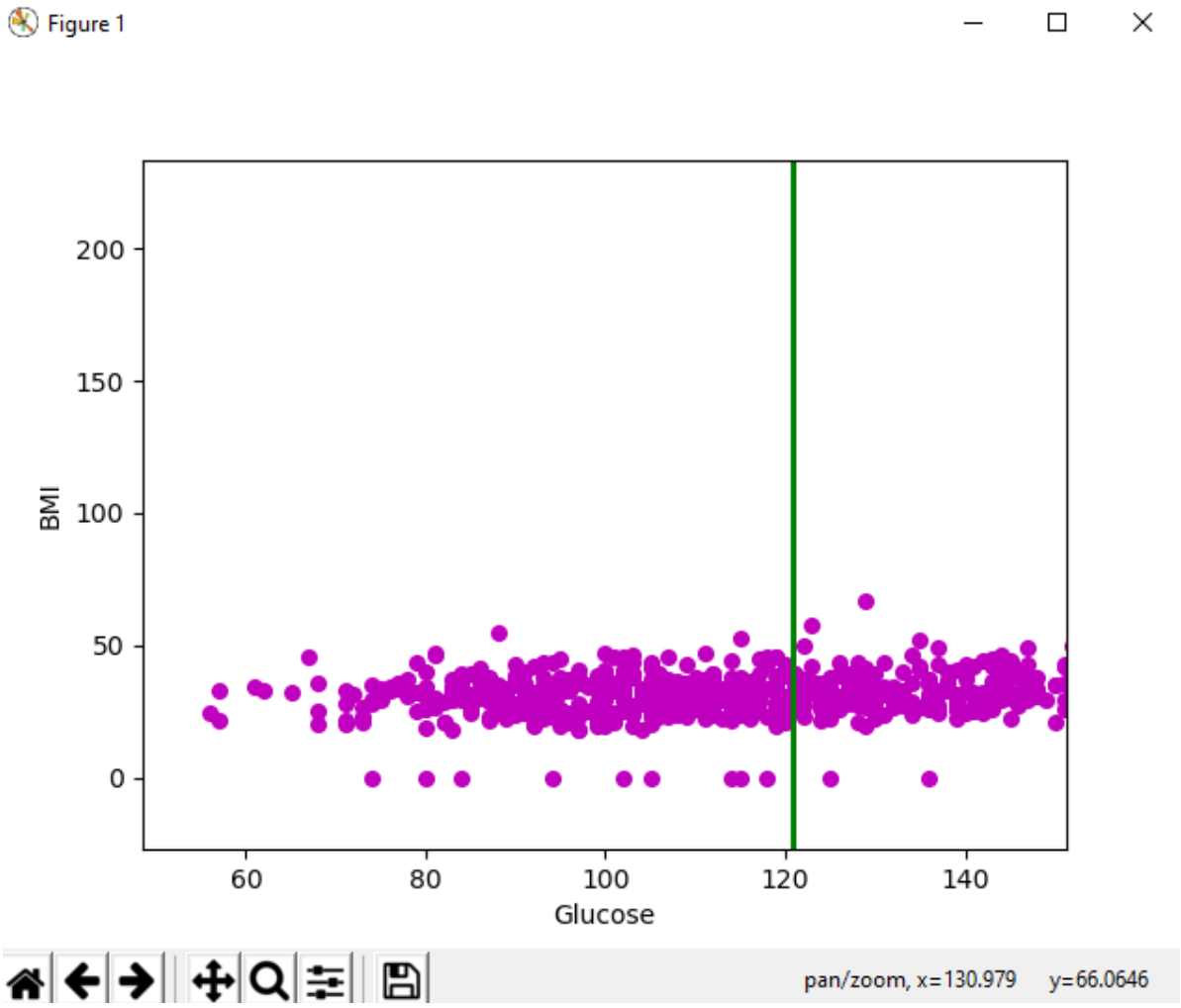
Algorithm:

1. Start the program.
2. Import numpy and pandas modules.
3. Open the Diabetes.csv file.
4. Coefficients are drawn against the field “Glucose” and “BMI” field.
5. Regression line is drawn.
6. Stop the program.

Program:

```
import numpy as np
import pandas
import matplotlib.pyplot as plt

def estimate_coef(x, y):
  n = np.size(x)
  m_x = np.mean(x)
  m_y = np.mean(y)
  SS_xy = np.sum(y*x) - n*m_y*m_x
  SS_xx = np.sum(x*x) - n*m_x*m_x
  b_1 = SS_xy / SS_xx
  b_0 = m_y - b_1*m_x
  return (b_0, b_1)

def plot_regression_line(x, y, b):
  plt.scatter(x, y, color = "m",marker = "o", s = 30)
  y_pred = b[0] + b[1]*x
  plt.plot(x, y_pred, color = "g")
  plt.xlabel('Glucose')
  plt.ylabel('BMI')
  plt.show()

def main():
    df = pandas.read_csv("c:\python27\Diabetes.csv")
    c = df[['Glucose']]
    x=np.array(df[['Glucose']])
    d = df['BMI']
    y=np.array([df['BMI']])
    samplecount=20
    b = estimate_coef(x, y)
    print("Estimated coefficients:\nb_0 = {} \nb_1 = {}".format(b[0], b[1]))
    plot_regression_line(x, y, b)

if __name__ == "__main__":
  main()
```

OUTPUT:
Estimated coefficients:
b_0 = -351259.285763
b_1 = 2905.76649505



Result:

Thus the python program to perform Bivariate analysis: Linear regression modeling using the Pima Indians diabetes data set are executed and output is verified.

Ex.No: 5(b)(iii) Bivariate Analysis: Logistic regression modeling

Aim:

To write a python program to perform Bivariate analysis: Logistic regression modeling using Pima Indians diabetes data set.

Algorithm:

1. Start the program.
2. Import numpy and pandas modules.
3. Open the file “Diabetes.csv”.
4. Read the field “Glucose” and “DiabetesPedigreeFunction”
5. Calculate the probability of that the person has diabetes from the glucose level.
6. Stop the program.

Program:

```
import pandas
import numpy
from sklearn import linear_model

df = pandas.read_csv("c:\python27\Diabetes.csv")
X = df[['Glucose']]
c=numpy.array(df[['Glucose']])
y = df['DiabetesPedigreeFunction']

regr = linear_model.LinearRegression()
regr.fit(X, y)
def logit2prob(regr, X):
  log_odds = regr.coef_ * X + regr.intercept_
  odds = numpy.exp(log_odds)
  probability = odds / (1 + odds)
  return(probability)

a=numpy.array(logit2prob(regr, X))
b=a*100
for i in range(0,len(c)):
   print("If the Glucose level is:{}then the probability of that the person has diabetes is:
{}%".format(c[i],b[i]))
```

OUTPUT:

If the Glucose level is:[148]then the probability of that the person has diabetes is: [62.4912685]%
If the Glucose level is:[85]then the probability of that the person has diabetes is: [60.36724703]%
If the Glucose level is:[183]then the probability of that the person has diabetes is: [63.65139637]%
If the Glucose level is:[89]then the probability of that the person has diabetes is: [60.50336875]%
If the Glucose level is:[137]then the probability of that the person has diabetes is: [62.12360133]%
If the Glucose level is:[116]then the probability of that the person has diabetes is: [61.41783998]%
If the Glucose level is:[78]then the probability of that the person has diabetes is: [60.12864834]%
……..
If the Glucose level is:[101]then the probability of that the person has diabetes is: [60.91075014]%
If the Glucose level is:[122]then the probability of that the person has diabetes is: [61.61999088]%
If the Glucose level is:[121]then the probability of that the person has diabetes is: [61.58632673]%
If the Glucose level is:[126]then the probability of that the person has diabetes is: [61.75453588]%
If the Glucose level is:[93]then the probability of that the person has diabetes is: [60.63932777]%

Result:

Thus the python program to perform Bivariate analysis: Logistic regression modeling using Pima Indians diabetes data set are executed and output is verified.

Ex.No: 5(c) Multiple Regression Analysis

Aim:

To write a python program to perform multiple regression analysis using Pima Indians diabetes data set.

Algorithm:

1. Start the program.
2. Import pandas modules.
3. Open the file “Diabetes.csv”.
4. Read the Pregnancies and BloodPressure field
5. Find the linear regression equation.
6. Predict the Blood Pressure and print the predicted value.
7. Stop the Program.

Program:

```
import pandas
from sklearn import linear_model
df = pandas.read_csv("c:\python27\Diabetes.csv")
X = df[['Pregnancies', 'Glucose']]
y = df['BloodPressure']
regr = linear_model.LinearRegression()
regr.fit(X, y)
predictedBloodpressure = regr.predict([[6, 148]])
print("Number of Pregnancies is : 6 and the Glucose level is : 148 and the Predicted Blood
Pressure is: {}".format(predictedBloodpressure))
```

OUTPUT:

Number of Pregnancies is : 6 and the Glucose level is : 148 and the Predicted Blood Pressure is: [72.87678249]

Result:

Thus the python program to perform multiple regression analysis using Pima Indians diabetes data set are executed and output is verified.