Impact Factor (SJIF): 5.301

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444
Volume 5, Issue 6, June-2018

Statistical Analysis of Data in Research Methodology

Neelkumar N. Patel[1]

[1]*Information Technology Department, Birla Vishvakarma Mahavidhyalaya*

Abstract - This Research Paper is based on the statistical analysis of data used in the research methodology and how to manipulate the data for specific uses in the research purpose. Further it shows how to analyze the data using python and NumPy stack which is prerequisites for analysis of the data. Representation of the data using various presentation graphs and statistics.

Keywords – Analysis of data, Prediction of data, NumPy Stack, Deep Learning, Machine Learning, Statistical Analysis.

I. INTRODUCTION

The NumPy Stack is the prerequisites for the any topic related to the intelligence in the computer science for example Deep Learning, Machine Learning, Artificial Intelligence, Data Analysis etc. The analysis of the data is mainly done using the NumPy Stack. NumPy is nothing but framework which can easily manipulate the data in the computer. Further there are various framework available which uses NumPy for analysis. The framework we are going to discuss are as follows for the Statistical analysis of the data.

1. NumPy
2. Pandas
3. Matplotlib
4. SciPy

NumPy - These forms the basis for Statistical Analysis of Data. The main aspect NumPy is the NumPy array, on which you can do various operations. The key is that a NumPy array isn't just a regular array you'd see in a language like Java or C++, but instead is like a mathematical object like a vector or a matrix. That means you can do vector and matrix operations like addition, subtraction, and multiplication. The most important aspect of NumPy arrays is that they are optimized for speed.

Pandas - Pandas is. A framework which is used to create a datasets which can manipulate data easily. Pandas makes working with datasets a lot like R, if you're familiar with R. The main aspects Pandas is the DataFrame. Pandas perform operations like dataframe operations, like filtering by column, filtering by row, the apply function, and joins, which look a lot like SQL joins. Pandas is a set of labeled array data structures, the primary of which are Series and DataFrame. Pandas can easily convert data from one type to another for example .csv to .json and vice versa.

Matplotlib – Matplotlib is a python library which is used to plot 2D graphs which provides visualization of the statistical data analysis. Matplotlib is a standard Python Library which can be used for writing python script to analyse the standard dataset produces by pandas like DataFrame and series. Matplotlib can plot following type of the graphs

1. Plots
2. Histogram
3. Power Spectra
4. Bar charts
5. Error Charts
6. Scatter Plots

Matplotlib can be used in Python scripts, IPython Shells, Jupyter Notebook, Web Application Server, Graphical User Interface, and other development platform for visualization analysis of data.

SciPy – SciPy is Python based ecosystem for statistical analysis of data. SciPy is mostly a basic blocks for the computer science topics like Deep Learning, Machine Learning, Data Analysis, Artificial Intelligence. SciPy is nothing but addon to NumPy. NumPy provides basic building blocks, like vectors, matrices, and operations on them, SciPy uses those general building blocks to do specific things. For example, SciPy can do many common statistics calculations, including getting the PDF value, the CDF value, sampling from a distribution, and statistical testing. SciPy is signal processing tool, it can perform like convolution and the Fourier transform.

II. DETAIL ANALYSIS OF NUMPY STACK

1. NumPy

NumPy is the scientific the library for the scientific computing in the analysis of the data. As discussed it provides great multidimensional object know as Array. It can easily manipulate the data like matrix multiplication, vectors manipulation like addition, multiplication and also it can perform inverse of the matrix and various number of application.

- NumPy Arrays

A NumPy array is grid of same data type. It is indexed by the nonnegative number i.e. from 0 to the length of the data in the array. To use NumPy you need to import numpy in python script.

```
import numpy as np

a = np.array([1,2,3])
```

Multi-dimensional array can also be formed as follows

```
import numpy as np

a = np.array([1,2,3],[4,5,6],[7,8,9])
```

NumPy provides various function that creates zeros matrix, identity matrix, random matrix, one matrix, constant matrix

```
import numpy as np

a = np.array([1,2,3],[4,5,6],[7,8,9])  # Simple Matrix

b = np.zeros((3,3))  # it creates zero matrix of 3*3

c = np.ones((2,2)) ))  # it creates one matrix of 2*2 of "ones"

d = np.eye(3)  # it creates 3*3 identity matrix
```

- Array Math

Numpy array can perform various direct mathematics operations like Addition, Multiplication, Subtraction, Division of two array and Square root of array as follows.

```
import numpy as np

a = np.array([1,2],[3,4]) # Simple Matrix

b = np.array([5,6],[7,8])  # Simple Matrix

c = np.add(a,b) #addition

d = np.subtract(a,b) #subtraction

e = np.multiply(a,b) #multiplication

f = np.divide(a,b) #division

g = np.sqrt(a) #square root
```

2. Pandas

Main advantage of Pandas is the DataFrame which generates the data in frame and series. Pandas provides tools for loading data from different formats of file. Pandas also provides merging and joining of data like sql join. Pandas is high performance library for data manipulation.

```
import numpy as np
import pandas as pd

a = np.array([100,101,102,103])
b = pd.Series(a)
```

Above type of code generates Series type of data and assigned number as follows:

0 100
1 101
2 102
3 103

Following code shows how to create DataFrame using pandas

```
import numpy as np
import pandas as pd

a = [[‘Ramesh’,21],[‘Suresh’,22],[‘Hitesh’,23]]
b = pd.DataFrame(a)
```

DataFrame can take various inputs like as follows:

- List
- Dict
- Series
- NumPy arrays
- DataFrame itself

The output produces a DataFrame as follows:

	Name	Age
0	Ramesh	21
1	Suresh	22
2	Hitesh	23

Such type of DataFrame can be used to produce effective data in research methodology.

3. Matplotlib

Matplotlib provides various graphs method to provide visual relation of the data. It is used for making plots. It provides following type of the plots and show how to plot the data.

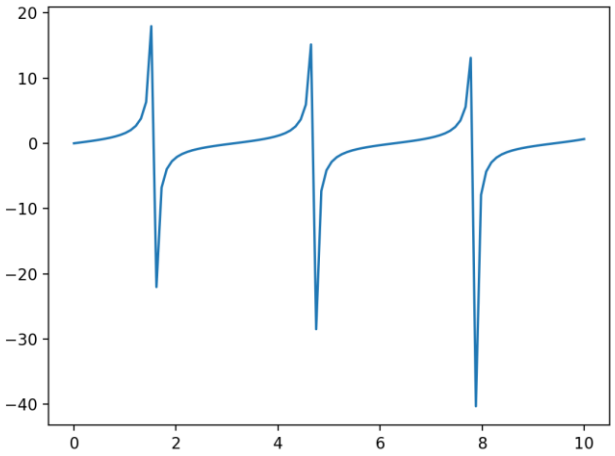
- Line Chart

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

a = np.linespace(0,10,100)
b = np.tan(a)

plt.plot(a,b)
plt.show()
```

It produces the tan graph as output as shown below:



Line Chart 2.1

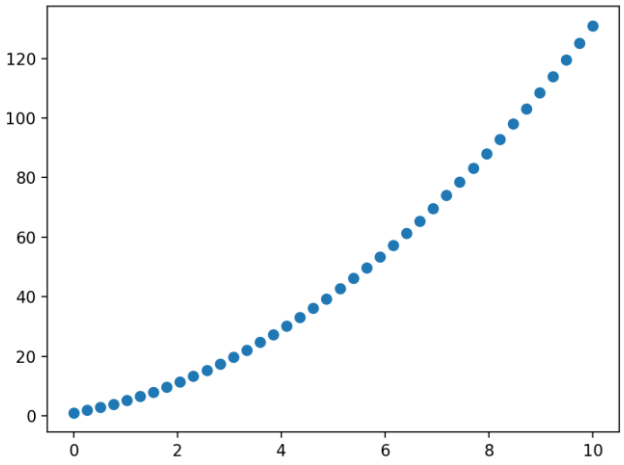
- Scatter Plot

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

a = np.linespace(0,10,40)
b = a**2 + 2*a + 1

plt.scatter(a,b)
plt.show()
```

As show in the code that b is the function of a and it produces scatter plot as follows



Scatter Plot 2.2

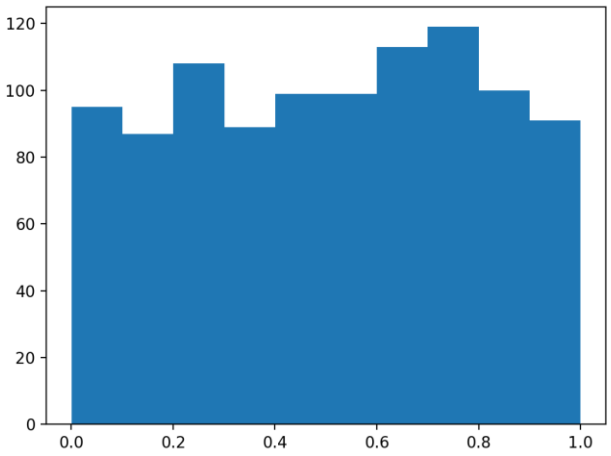
- Histogram

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

a = np.random.random(100)

plt.hist(a)
plt.show()
```

It produces the histogram as follows:



Histogram 2.3

III. CONCLUSION

Thus, after detail analysis we can conclude that the Statistical analysis of the data can be easily done with the NumPy Stack and the python programming language. Matplotlib can produces various graphs based on the values of the data and produces effective data for research methodology. Further this data can be used for training the program in the Artificial Intelligence and also to predict future results from the assumption which helps in analysing the data through statistics.

REFRENCES

[1] NumPy researchgate
(https://www.researchgate.net/publication/224223550_The_NumPy_Array_A_Structure_for_Efficient_Numerical_Computation)

[2] The NumPy Array
(https://dl.acm.org/citation.cfm?id=1957466)

[3] NumPy Recipes/ SciPy Recipies
(https://www.researchgate.net/publication/273133972_NumPy_SciPy_Recipes_for_Data_Science_Ordinary_Least_Squares_Optimization)

[4] NumPy
(http://www.numpy.org/)

[5] Matplotlib
(https://matplotlib.org/)

[6] Pandas
(https://pandas.pydata.org/)