

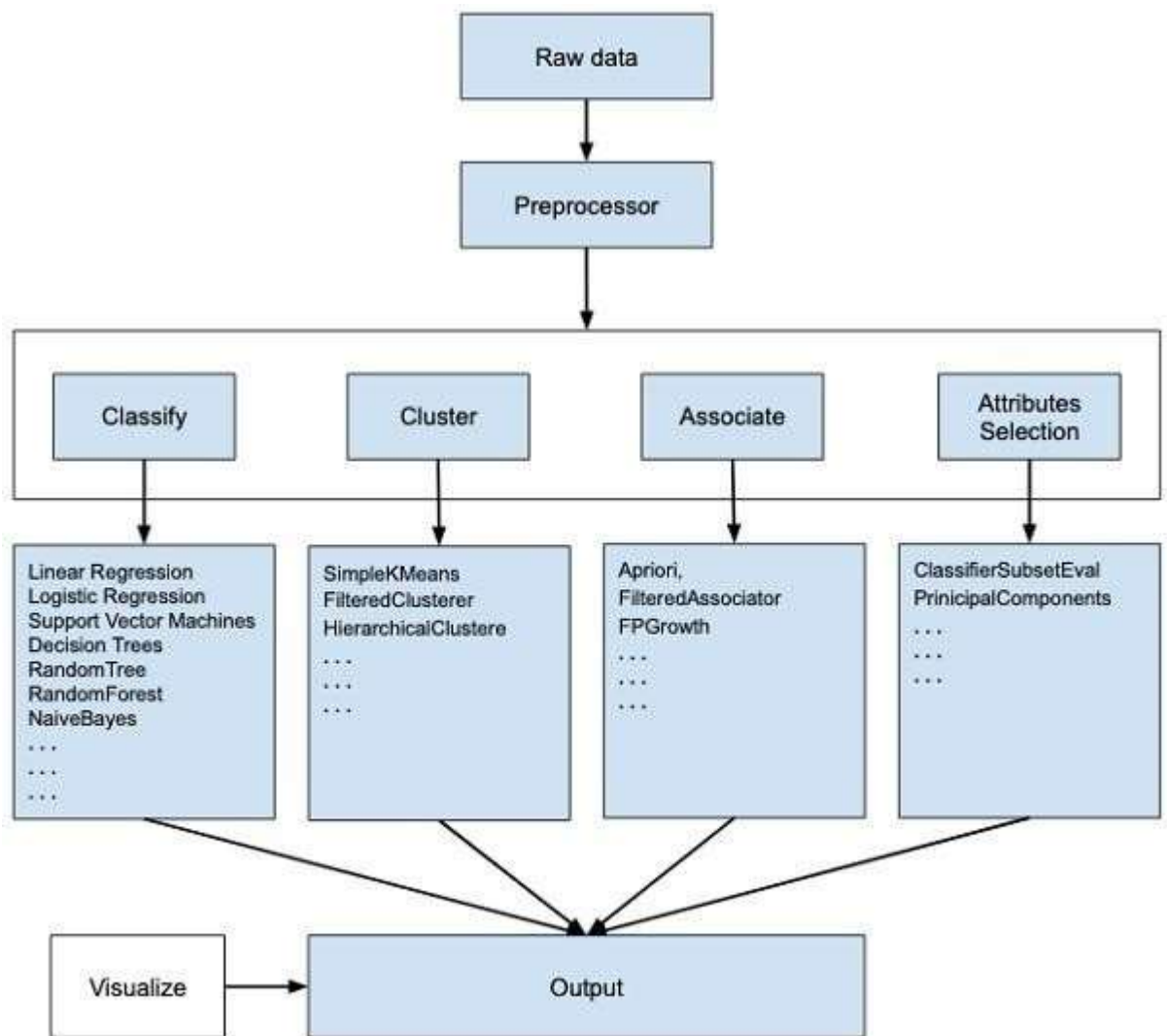
**ANSH AWASTHI**

**2100290139002**

## **DATA ANALYTICS LAB-1**

### **1. Introduction to Weka Tool**

WEKA is an open source software that provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems.



Observing the beginning of the flow of the image, we understand that there are many stages in dealing with Big Data to make it suitable for machine learning –

- i. First, we start with the raw data collected from the field. This data may contain several null values and irrelevant fields. We use the data preprocessing tools provided in WEKA to cleanse the data.
- ii. Then, we would save the preprocessed data in our local storage for applying ML algorithms.
- iii. Next, depending on the kind of ML model that we are trying to develop we would select one of the options such as **Classify**, **Cluster**, or **Associate**. The **Attributes Selection** allows the automatic selection of features to create a reduced dataset.
- iv. Note that under each category, WEKA provides the implementation of several algorithms. We can select an algorithm of our choice, set the desired parameters and run it on the dataset.
- v. Then, WEKA would give us the statistical output of the model processing. It provides us a visualization tool to inspect the data.

- vi. The various models can be applied on the same dataset. We can then compare the outputs of different models and select the best that meets our purpose.
- vii. Thus, the use of WEKA results in a quicker development of machine learning models on the whole.

## **2. Get the input from the user and perform numerical operations (MAX, MIN, AVG, SUM, SQRT, ROUND) using Python Programming Language.**

```
from statistics import mean
from numpy import sqrt

a=list(map(float,input().split()))
ans=[]
print('MAX:',max(a))
print('MIN:',min(a))
print('AVERAGE:',mean(a))
print('SUM:',sum(a))
for i in range(len(a)):
    ans.append(sqrt(a[i]))
print('SQUARE ROOT:',ans)
for i in range(len(a)):
    ans[i]=round(a[i],1)
print('ROUND:',ans)
```

```
main.py
1 '''Get the input from the user and perform numerical operations
2 (MAX, MIN, AVG, SUM, SQRT, ROUND) using Python Programming Language.'''
3 from statistics import mean
4 from numpy import sqrt
5 a=list(map(float,input().split()))
6 ans=[]
7 print('MAX:',max(a))
8 print('MIN:',min(a))
9 print('AVERAGE:',mean(a))
10 print('SUM:',sum(a))
11 for i in range(len(a)):
12     ans.append(sqrt(a[i]))
13 print('SQUARE ROOT:',ans)
14 for i in range(len(a)):
15     ans[i]=round(a[i],1)
16 print('ROUND:',ans)
17
```

input

```
4.345 5.456 2.756 8.123 4.32 7.1 9
MAX: 9.0
MIN: 2.756
AVERAGE: 5.871428571428571
SUM: 41.1
SQUARE ROOT: [2.084466358567583, 2.3358082113050
29, 1.6601204775557705, 2.850087717948344, 2.078
460969082653, 2.6645825188948455, 3.0]
ROUND: [4.3, 5.5, 2.8, 8.1, 4.3, 7.1, 9.0]
```

### 3. Perform Statistical analysis operation (Mean, Median, Mode and Standard Deviation) in Python.

```
import statistics as st
x=[2,7,4,9,2,8,3,7,4,2,9,1,6,3,8,7,9,8,8]
st.mean(x)
st.median(x)
st.mode(x)
st.stdev(x)
```

```
In [14]: import statistics as st
```

```
In [15]: x=[2,7,4,9,2,8,3,7,4,2,9,1,6,3,8,7,9,8,8]
```

```
In [16]: st.mean(x)
```

```
Out[16]: 5.631578947368421
```

```
In [17]: st.median(x)
```

```
Out[17]: 7
```

```
In [18]: st.mode(x)
```

```
Out[18]: 8
```

```
In [19]: st.stdev(x)
```

```
Out[19]: 2.8128776549566434
```