Univariate Analysis

The term univariate analysis refers to the analysis of one variable. You can remember this because the prefix “uni” means “one.”

There are three common ways to perform univariate analysis on one variable:

1. Summary statistics – Measures the center and spread of values.
2. Frequency table – Describes how often different values occur.
3. Charts – Used to visualize the distribution of values.

This tutorial provides an example of how to perform univariate analysis with the following pandas DataFrame:

```
#import libraries

import pandas as pd
```

```
# loading and reading dataset
ds_path = r"D:\APHRC\Projects\Mental Health\data\MH.csv"
df = pd.read_csv(ds_path)
df.head()
```

	Unnamed: 0	start	end	_submission_time	FA_Code	StudyID	_index	locationid	villagenam	re
0	1	2023-06-29	2023-06-29	2023-06-30	ANE	12737	1735	I41607189001	NABIDONGHAC	P
1	2	2023-09-13	2023-09-13	2023-09-14	GYH	14487	7027	I31501165003	NAKIGO II A	
2	3	2023-09-28	2023-09-28	2023-09-28	BGO	12027	9019	M20901008002	MBALE TC	
3	4	2023-07-27	2023-07-27	2023-07-27	MGA	5802	3667	I31201305001	BUSEYI B	
4	5	2023-08-14	2023-08-14	2023-08-21	MLF	7449	5101	I10503309001	BUSOWOBI CENTRA	

5 rows × 71 columns

1. Summary statistics – Measures the center and spread of values.

```
#calculate mean of 'age'
df['age'].mean()
```

```
52.060088202866595
```

```
#calculate median of 'age'
df['age'].median()
```

```
50.0
```

```
#calculate standard deviation of 'points'
df['age'].std()
```

```
14.922865665928079
```

You can also use describe to get summary fo all continous columns

```
#Summary
df.describe()
```

	Unnamed: 0	StudyID	_index	age	ANXIETY	Overthelast2weekshowoften	
count	1814.000000	1814.000000	1814.000000	1814.000000	0.0	0.0	1814.0
mean	907.500000	8431.299338	4580.059537	52.060088	NaN	NaN	3.0
std	523.801012	4788.627705	2670.713830	14.922866	NaN	NaN	3.5
min	1.000000	37.000000	7.000000	19.000000	NaN	NaN	0.0
25%	454.250000	4770.250000	2221.250000	42.000000	NaN	NaN	0.0
50%	907.500000	7998.500000	4493.000000	50.000000	NaN	NaN	2.0
75%	1360.750000	12746.750000	6891.750000	61.000000	NaN	NaN	5.0
max	1814.000000	17013.000000	9302.000000	105.000000	NaN	NaN	21.0

Let’s understand what each of the value means.

count is Total number of entries

mean is Average of all the entries

std is standard deviation

min is minimum value

25% is 25 percentile mark

50% is 50 percentile mark (median)

75% is 75 percentile mark

max is maximum value

2. Frequency table – Describes how often different values occur.

```
#create frequency table for 'age'
df['age'].value_counts()
```

```
age
43     73
53     64
54     57
46     56
45     56
       ..
23      1
20      1
98      1
19      1
103     1
Name: count, Length: 82, dtype: int64
```

Gives you the age and its frequency. eg 73 people are 43 years old

```
#Two way frequency table
pd.crosstab(index=df['age'], columns=df['gender'])
```

gender	Female	Male
age		
19	0	1
20	0	1
21	1	2
23	0	1
24	0	2
...	...	...
98	0	1
99	0	2
100	0	2
103	0	1
105	1	0

82 rows × 2 columns

For chats check visualisation tutorial