Introduction to EDA

Understanding Descriptive Statistics

The main idea of descriptive statistics is to use a quantitative approach to numerically describe and summarize the data. Also, it takes advantage of the bar plots, histograms, charts and other graphs to visualize the dataset.

Types of Descriptive Analysis

The methods of descriptive statistics can be applied to one or more variables. Based on this, the analysis of variables can be classified into three types, namely:

1. **Univariate Analysis**: When only one variable is considered for describing and analyzing.
2. **Bivariate Analysis**: When a pair of variables are used for statistical relationships.
3. **Multivariate Analysis**: When multiple variables at once are used for statistical analysis.

Types of measures in Descriptive Statistics:

1. **Central Tendency**: It helps in studying the central measure of the dataset. It is mainly done with the help of mean, median and mode.
2. **Variability**: It helps in studying the variation or spread of the dataset. It can be done with the help of standard deviation and variance.
3. **Correlation**: It helps in studying the relationship between a set of variables. The methods used to find a correlation between various variables are covariance and correlation coefficient.

You will learn about these Measures of Descriptive Statistics using Python.

The Measure of Central Tendency

The measures of central tendency show the central or middle values of datasets.

Mean: The sample mean, also called the sample arithmetic mean or simply the average, is the arithmetic average of all the items in a dataset. The mean of a dataset x is mathematically expressed as $\Sigma_i x_i/n$, where $i = 1, 2, \ldots, n$. In other

words, it's the sum of all the elements x_i divided by the number of items in the dataset x.

```
x = [8.0, 1, 2.5, 4, 28.0]
mean_ = sum(x) / len(x)
mean_
8.7
```

Another way to calculate the mean is by using the "statistics" library.

```
import statistics
mean_ = statistics.mean(x)
mean_
8.7
```

Median: The sample median is the middle element of a sorted dataset. The dataset can be sorted in increasing or decreasing order. If the number of elements n of the dataset is odd, then the median is the value at the middle position: $0.5(n + 1)$. If n is even, then the median is the arithmetic mean of the two values in the middle, that is, the items at the positions $0.5n$ and $0.5n + 1$.

```
n = len(x)
if n % 2:
   median_ = sorted(x)[round(0.5*(n-1))]
else:
   x_ord, index = sorted(x), round(0.5 * n)
   median_ = 0.5 * (x_ord[index-1] + x_ord[index])
median_
4
```

Median using Statistics Library

```
median_ = statistics.median(x)
median_
4
```

Mode: The sample mode is the value in the dataset that occurs most frequently. If there isn't a single such value, then the set is multimodal since it has multiple modal values. For example, in the set that contains points 2, 3, 2, 8, and 12, the number 2 is the mode because it occurs twice, unlike the other items that occur only once.

```
u = [2, 3, 2, 8, 12]
mode_ = max((u.count(item), item) for item in set(u))[1]
mode_
2
```

Mode using statistics Library

```
mode_ = statistics.mode(u)
mode_
2
```

The Measure of Variability

The measures of central tendency aren't sufficient to describe data. You'll also need measures of variability that quantify the spread of data points.

Variance: The sample variance quantifies the spread of the data. It shows numerically how far the data points are from the mean. You can express the sample variance of the dataset x with n elements mathematically as $s^2 = \Sigma_i(x_i - \text{mean}(x))^2 / (n - 1)$, where $i = 1, 2, ..., n$ and $\text{mean}(x)$ is the sample mean of x.

```
n = len(x)
mean_ = sum(x) / n
var_ = sum((item - mean_)**2 for item in x) / (n - 1)
var_
123.19999999999999
```

Variance using statistics Library

var_ = statistics.variance(x)
var_
123.2

Standard Deviation: The sample standard deviation is another measure of data spread. It's connected to the sample variance, as standard deviation, s, is the positive square root of the sample variance. The standard deviation is often more convenient than the variance because it has the same unit as the data points.

std_ = var_ ** 0.5
std_
11.099549540409287

Standard Deviation using Statistics Library

std_ = statistics.stdev(x)
std_
11.099549540409287