

CUMMULATIVE DISTRIBUTIVE FUNCTION & STANDARD NORMAL DISTRIBUTION

CUMMULATIVE DISTRIBUTIVE FUNCTION

```
: from statsmodels.distributions.empirical_distribution import ECDF
   ecdf = ECDF(dataset['ssc_p'])
   ecdf(40)    #40marks la no one take the marks

: 0.0
```

The ECDF class has a `__call__` method that can be used to evaluate the ECDF at a given value. For example, if `ecdf` is an instance of the ECDF class, then `ecdf(40)` returns the estimated probability that a random variable takes on a value less than or equal to 40 based on the observed data

So the summary here is given 40 marks. I need 40 marks, how much percentage of students took. The given output value is 0.0. So we conclude that no one student get 40 Marks.

STANDARD NORMAL DISTRIBUTION

```
def stdNBgraph(dataset):
    import seaborn as sns
    mean=dataset.mean()
    std=dataset.std()
    values=[i for i in dataset]

    z_score=[((j-mean)/std) for j in values]

    sns.distplot(z_score,kde=True)

    sum(z_score)/len(z_score)
```

The **stdNBgraph** function takes a dataset as input and creates a standard normal distribution plot of the data.

- Import the seaborn library for data visualization.
- Calculate the mean and standard deviation of the dataset using the mean and std methods, respectively.
- Create a list of values from the dataset using a list comprehension.
- Calculate the z-scores for each value in the dataset by subtracting the mean and dividing by the standard deviation.
- Create a standard normal distribution plot using seaborn with the distplot function, which takes the z-scores as input and sets the kde parameter to True to show the kernel density estimate.