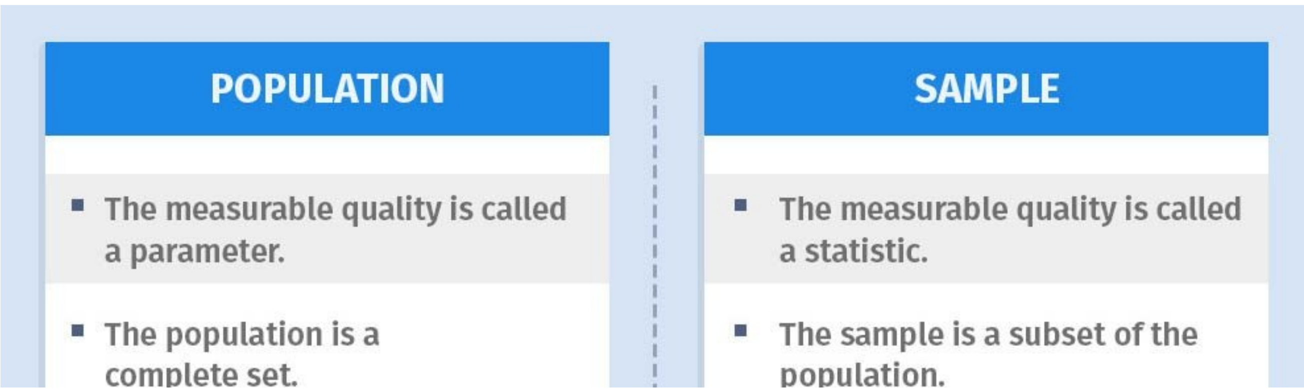
1. Difference between descriptive statistics and Inferential?

Descriptive Statistics describes the characteristics of a data set. It is a simple technique to describe, show and summarize data in a meaningful way. You simply choose a group you're interested in, record data about the group, and then use summary statistics and graphs to describe the group.

Inferential statistics involves drawing conclusions about populations by examining samples.

	Descriptive Statistics	**Inferential Statistics**
Purpose	Describe and summarize data	Make inferences and draw conclusions about a population based on sample data
Data Analysis	Analyzes and interprets the characteristics of a dataset	Uses sample data to make generalizations or predictions about a larger population
Population vs Sample	Focuses on the entire population or dataset	Focuses on a subset of the population (sample) to draw conclusions about the entire population
Measurements	Provides measures of central tendency and dispersion	Estimates parameters, tests hypotheses, and determines the level of confidence or significance in the results
Examples	Mean, median, mode, standard deviation, range, frequency tables	Hypothesis testing, confidence intervals, regression analysis, ANOVA (analysis of variance), chi-square tests, t-tests, etc.

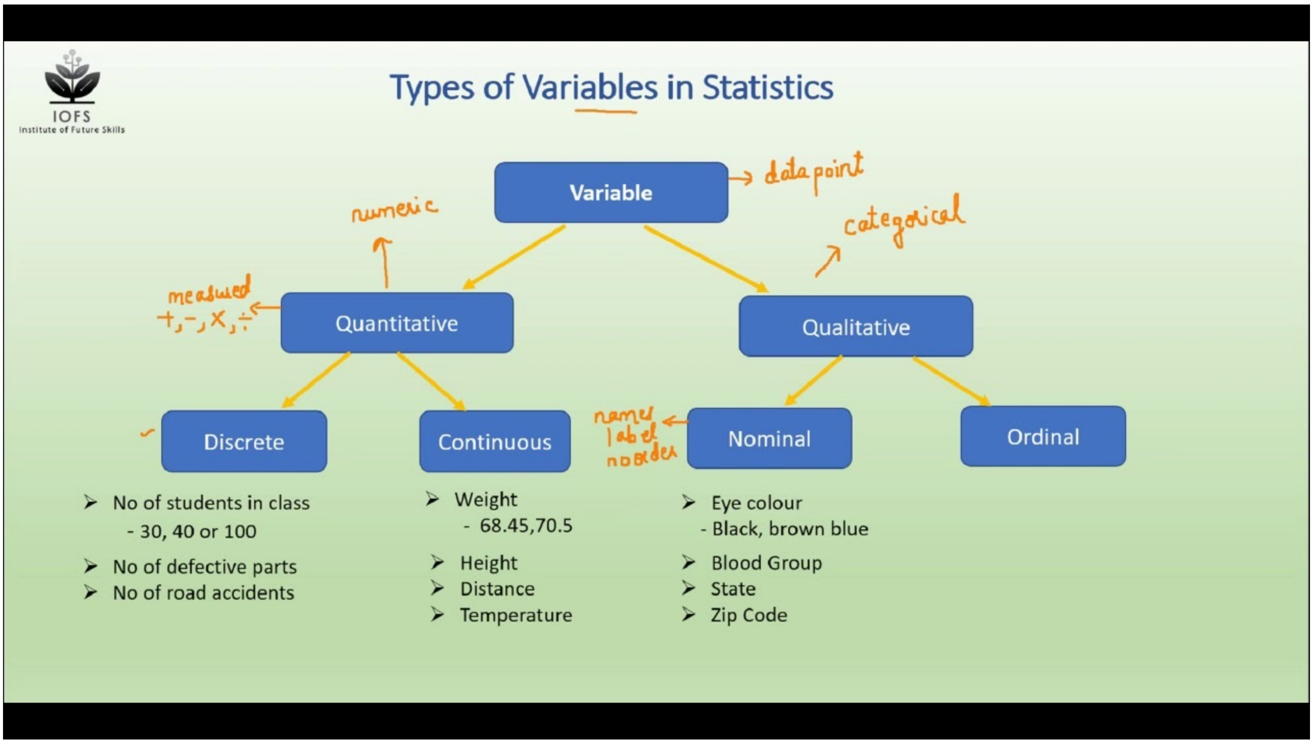
1. Sample vs Population



POPULATION	SAMPLE
The measurable quality is called a parameter.	The measurable quality is called a statistic.
The population is a complete set.	The sample is a subset of the population.

Variable in stat

Variable is a property that can take many values



Discrete only contain whole number (not decimal).

Measure of Central Tendency - Maen, Median and mode

Central tendency is measure of central of distribution of data.

Population Mean	Sample Mean
$\mu = \frac{\sum_{i=1}^{N} x_i}{N}$	$\overline{X} = \frac{\sum_{i=1}^{n} x_i}{n}$
N = number of items in the population	n = number of items in the sample

Mean

summation xi is nothing the summation or addition of total data/number of data.

In [1]:
```
# Example; 1,2,2,3,6,8,9,2
mean=(1+2+2+3+6+8+9+2)/8
```

In [2]:
```
print(mean)
```

```
4.125
```

4.125 is mean, but if we add some big number here:

In [3]:
```
# Example; 1,2,2,3,6,8,9,2,100
mean=(1+2+2+3+6+9+8+2+100)/9
```

In [4]:
```
print(mean)
```

```
14.777777777777779
```

See the difference in mean with addition of one num, so it may not giving accurate central distribution of number. In such case, we say 100 as outliner, in such case we use median.

In [6]:
```
l=(1,2,2,3,6,8,9,2,100)
L=tuple(sorted(l))
```

In [7]:
```
print(L)
```

```
(1, 2, 2, 2, 3, 6, 8, 9, 100)
```

median=3 is median.But, what if total number is even in count?

In [8]:
```
L=(1,2,2,2,3,6,8,9,100,110)
```

In [9]:
```
median=(3+6)/2
```

In [10]:
```
print(median)
```

```
4.5
```

see there is not such big difference as in mean so median is mainly used when mean cannot decide central distribution of data due to outliners.

Mode? most occurance of frequency

Example: mainly used in categorical, suppose one column of table is gender and there is M,F,M,F,F,F,-,M one value is missing, in such case we use mode, largest repeating frequency is Female so we provide F to missing values.

In [11]:
```
#Numerically, in above exaple 2 is mode.
```

Lets see in pythonic way:

In [19]:
```
import statistics
import numpy as np
```

In [13]:
```
l=(1,2,2,3,6,8,9,2,100)
```

In [15]:
```
mean_a=np.mean(l)
print(mean)
```

```
14.777777777777779
```

In [16]:
```
median_a=np.median(l)
print(median)
```

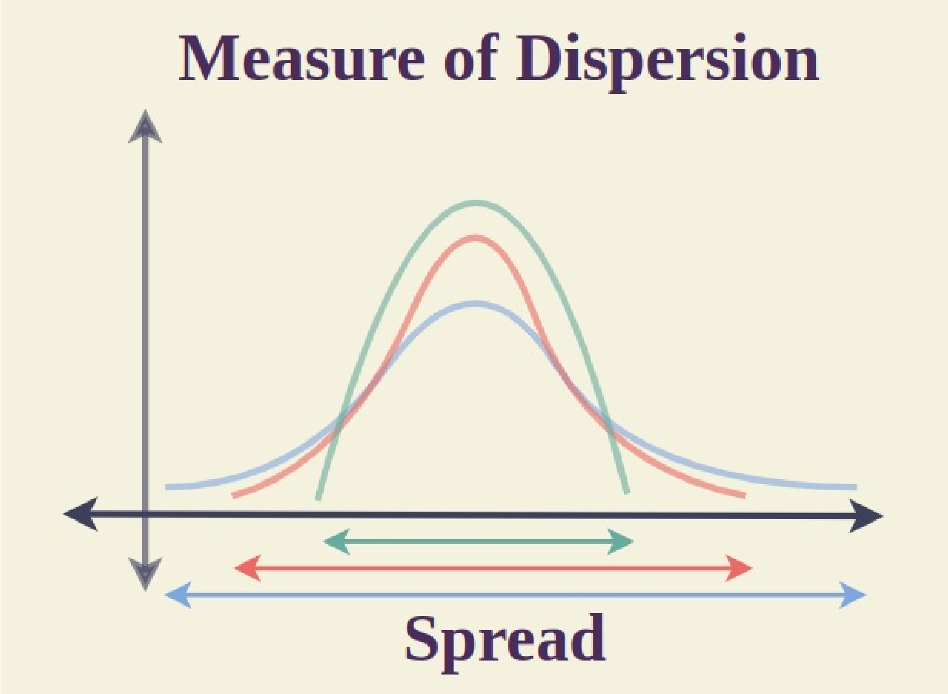
```
4.5
```

In [21]:
```
mode_a=statistics.mode(l)
print(mode_a)
```

```
2
```

Measure of dispersion - Variance and Statndar Deviation

Measure of dispersion talk about spread of data.



Population Variance	Sample Variance
$\sigma^2 = \frac{\sum_{i=1}^{N}(x_i-\mu)^2}{N}$	$s^2 = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n-1}$
$\sigma^2 =$ population variance $x_i =$ value of i^{th} element $\mu =$ population mean $N =$ population size	$s^2 =$ sample variance $x_i =$ value of i^{th} element $\bar{x} =$ sample mean $n =$ sample size

Note: n-1 is called bessel correction or degree of freedom

Example of variance calculation:

Example 2: Find the sample variance of the data set {2, 6, 12, 15}

Solution: Variance is a measure of dispersion given by

$\sum_1^n \frac{(X_i-\overline{X})^2}{n-1}$

n = 4

$\overline{X}$ = (2 + 6 + 12 + 15) / 4 = 8.75

Variance = $\frac{(2-8.75)^2+(6-8.75)^2+(12-8.75)^2+(15-8.75)^2}{3}$ = 34.25

Answer: Variance = 34.25

Actaully what is 34.25? - It is spreadness of data

In [4]:
```
# Now, lets use same data from above, and lets find variance using numpy

import numpy as np
x=[2,6,12,15]
```

In [5]:
```
variance_of_x=np.var(x)
print(variance_of_x)
```

```
25.6875
```

Why answer is different, because python is calculating population variance automatically, lets calculate sample variance using degree of freedom.

In [6]:
```
variance_of_x_sample=np.var(x, ddof=1)
print(variance_of_x_sample)
```

```
34.25
```

Okay, but when we use variance in real example in data science. Suppose we are calculating variance of two cricket player of this worldcup

In [7]:
```
#lets look babar azam and virat kholi data of worldcup 2023. later in hardcore python, we will do web scrapping
#just an example
import random
BA=[random.randint(1, 100) for _ in range(5)]
VK=[random.randint(1,150) for _ in range(5)]
```

In [8]:
```
print(BA)
print(VK)
```

```
[41, 92, 83, 36, 97]
[11, 43, 113, 34, 142]
```

In [9]:
```
#lets calculate variance to calculate consitent player, like not good or bad player but who is scoring similar
#in all 5 games
BA_variance=np.var(BA)
VK_variance=np.var(VK)
```

In [10]:
```
print(BA_variance)
print(VK_variance)
```

```
675.76
2505.84
```

In [11]:
```
#low variance means low spread and near to mean, it shows BA is more consistent player.
```

Standar Deviation

Formula for standard deviation=root of variance.

In [12]:
```
#previous above example - 2,6,12,15
variance_of_x_sample=np.var(x, ddof=1)
print(variance_of_x_sample)
```

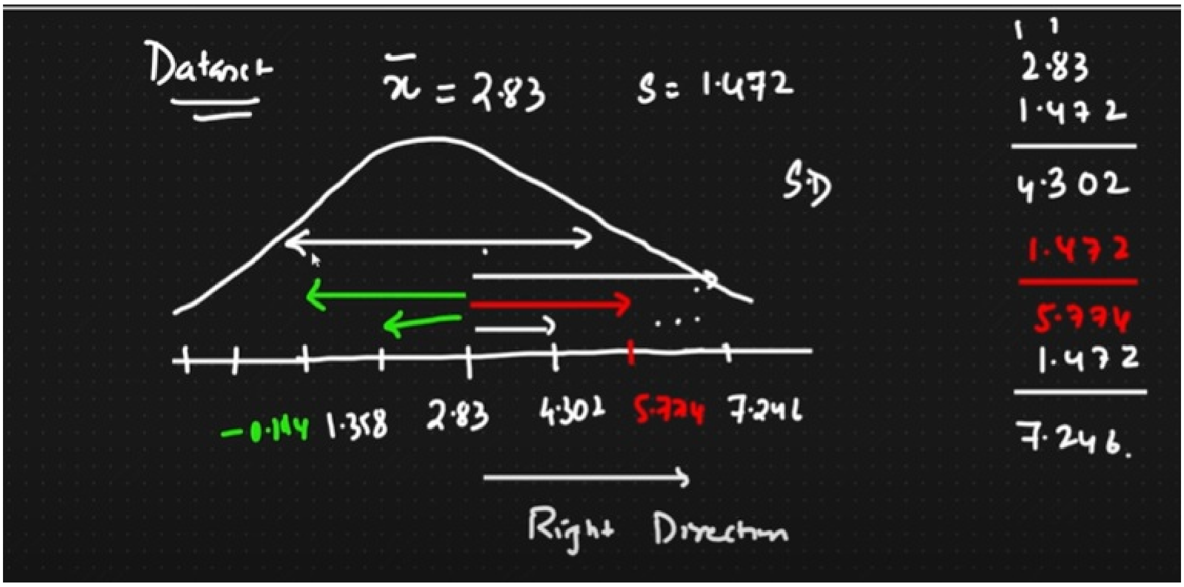
```
34.25
```

In [16]:
```
#lets find SD
#lets check manually what is square root of 34.25 = 5.85
# pythonic way
sd=np.std(x,ddof=1)
print(sd)
```

```
5.852349955359813
```

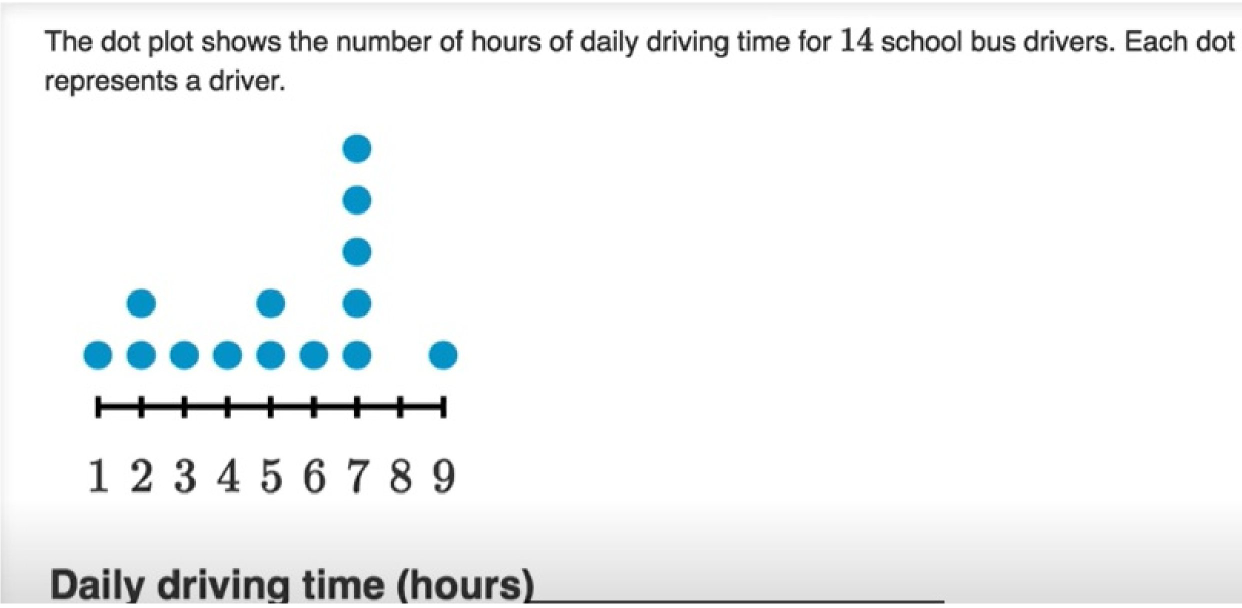
Always remember for sample, we have to use degree of freedom but not for population

How figure is related to SD and variance?



Suppose 2.83 is mean then 1.472 is SD then on right side we add mean + SD = 4.302 (First SD right side), we deduct for left side

PERCENTILE AND QUARTILE



In [1]:
```
#Now find percentile of driver who drive 6 hour a day?
```

In [2]:
```
#manual formula
# percentile= (value below 6/n)*100
percentile=(5/9)*100
print(percentile)
```

```
55.55555555555556
```

It means driver driver 6 hr daily is better than 55% of total driver driving

Quartile

25 percentile= Q1 (first quartile) 75 percentile = Q3 (Third quartile

Quartile is mainly used to find outliner.

CONSTRUCT BOXPLOT AND OUTLINER ...

Five Number Summary

In [36]:
```
data=[12,15,17,19,20,22,23,24,25,28,30,32,35,40,45,50,60,70,80,100]
len(data)
```

Out[36]:
```
20
```

1. Calculate Q1 (1st quartile) and Q3 (3rd quartile) from the dataset.
2. Calculate IQR: $IQR = Q3 - Q1$

2. Calculate IQR: $IQR = Q3 - Q1$
3. Calculate LF: $LF = Q1 - 1.5 \times IQR$
4. Calculate UF: $UF = Q3 + 1.5 \times IQR$

In [37]:
```
import numpy as np
```

In [38]:
```
q1 = np.percentile(data, 25)
q3 = np.percentile(data, 75)
k=1.5

# Calculate the interquartile range (IQR)

iqr = q3 - q1

# Calculate the lower fence (LF) and upper fence (UF)
lf = q1 - k * iqr
uf = q3 + k * iqr
```

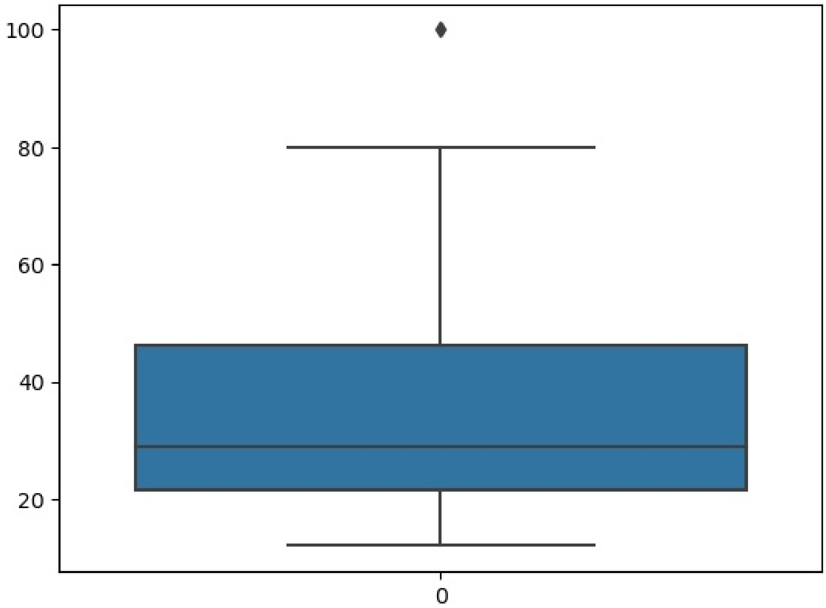
In [39]:
```
print(iqr)
print(lf)
print(uf)
```

```
24.75
-15.625
83.375
```

In [40]:
```
#so for above data 100 is outlier
```

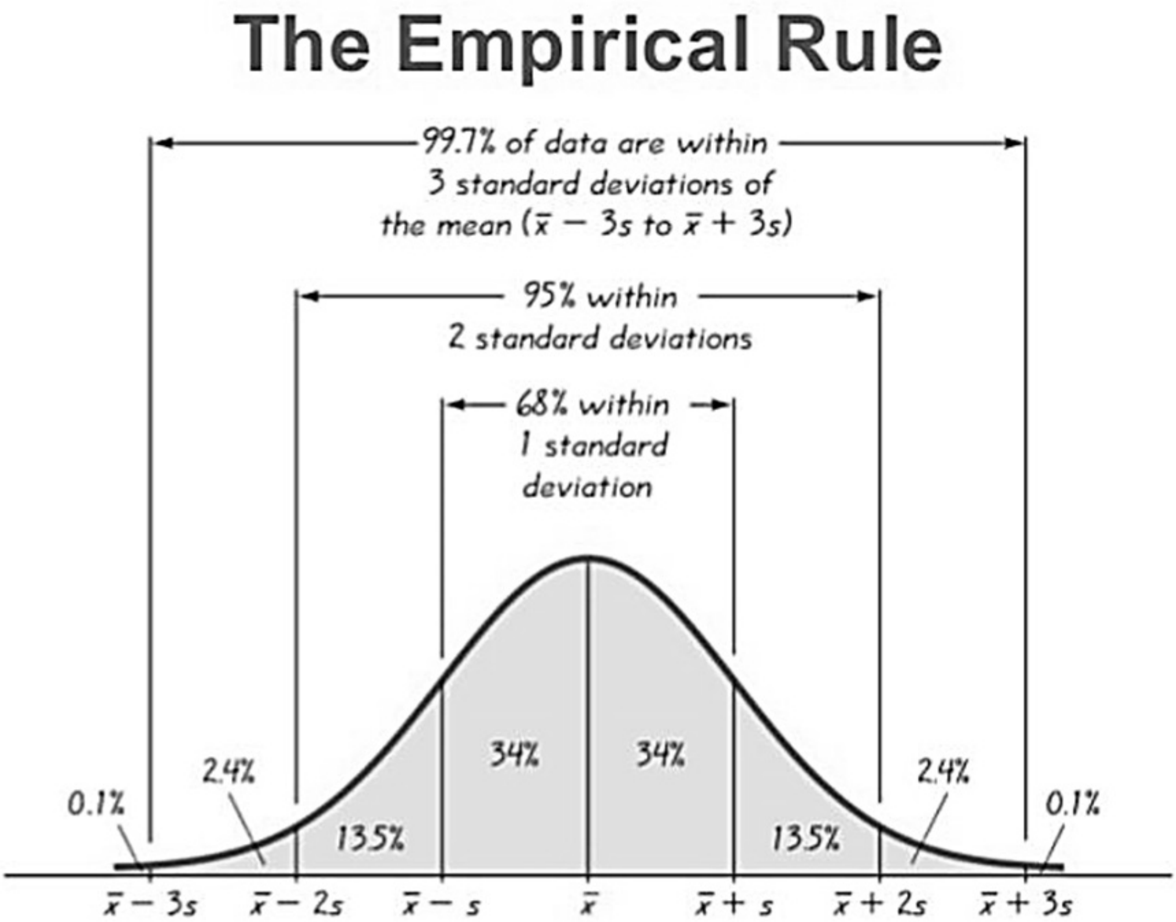
In [41]:
```
#If want to see visually, lets import seaborn
import seaborn as sns
sns.boxplot(data)
```

Out[41]: <Axes: >



Normal Distribution/Gussian Distribution And Its Empirical Formula

Empirical Rule or 3 sigma rule (Properties of gussian)



Note; above figure is symmetrical figure thatswhy empirical rule will effectively apply.

CENTRAL LIMIT THEOREM

Suppose we have left or right sekewed curve, but now we take or increase sample size, n>30 then graph will move towards normal distribution