In [ ]:
```
1
```

Expected Value :

- In probability theory, the expected value (also called expectation, expectancy, mathematical expectation, mean, average, or first moment) is a generalization of the weighted average. Informally, the expected value is the arithmetic mean of a large number of independently selected outcomes of a random variable.
- The expected value of a random variable with a finite number of outcomes is a weighted average of all possible outcomes. In the case of a continuum of possible outcomes, the expectation is defined by integration. In the axiomatic foundation for probability provided by measure theory, the expectation is given by Lebesgue integration.

We define the expected value (or expectation or mean) of a discrete random variable X by

$$E[X] = \sum_x xP(X = x).$$

```
1 Different notations for expected value of X: EX = E[X] = E(X) = μ
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

Sampling statistics

- In statistics, quality assurance, and survey methodology, sampling is the selection of a subset (a statistical sample) of individuals from within a statistical population to estimate characteristics of the whole population.
- Statisticians attempt to collect samples that are representative of the population in question. Sampling has lower costs and faster data collection than measuring the entire population and can provide insights in cases where it is infeasible to measure an entire population.

- In sampling, this includes defining the "population" from which our sample is drawn. A population can be defined as including all people or items with the characteristic one wishes to understand.
- Because there is very rarely enough time or money to gather information from everyone or everything in a population, the goal becomes finding a representative sample (or subset) of that population.

In [ ]:
```
1
```

Sample mean and Covariance

- The sample mean (or "empirical mean") and the sample covariance are statistics computed from a sample of data on one or more random variables.
- The sample mean is the average value (or mean value) of a sample of numbers taken from a larger population of numbers, where "population" indicates not number of people but the entirety of relevant data, whether collected or not. A sample of 40 companies' sales from the Fortune 500 might be used for convenience instead of looking at the population, all 500 companies' sales.
- The sample mean is used as an estimator for the population mean, the average value in the entire population, where the estimate is more likely to be close to the population mean if the sample is large and representative.

- ***The reliability of the sample mean is estimated using the standard error , which in turn is calculated using the variance of the sample.***
- If the sample is random, the standard error falls with the size of the sample and the sample mean's distribution approaches the normal distribution as the sample size increases.

In [ ]:
```
# from Wikipedia
```

Sample mean :

- The sample mean is the average of the values of a variable in a sample, which is the sum of those values divided by the number of values. Using mathematical notation, if a sample of N observations on variable X is taken from the population, the sample mean is:

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i.$$

Type *Markdown* and LaTeX: α^2

```
dice example :
    Random Variable X = {1,2,3,4,5,6}
                E(X) = 3.5   (from E(X) = sigma x*P(x))

    E(x-bar) = E[(x1+x2+x3+...+xn)/n]
              = (E[x1] + E[x2] + ... + E[xn]) / 2
             = (n * 3.5) / n
             = 3.5
             =  E(x)
             = actual mean
```

Sample mean :

$$\bar{X} = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}$$

$$\begin{aligned} E[\bar{X}] &= E\left[\frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}\right] \\ &= \frac{E[x1] + E[x2] + \cdots + E[Xn]}{n} \\ &= \frac{n\,E[x]}{n} \\ &= E[X] \end{aligned}$$

expectation of the sample mean is the true mean

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```

```

Sample Mean | Sampling Distribution | Sampling Variance

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
np.random.randint(1,7,100)
```

Out[2]:
```
array([6, 3, 5, 4, 6, 3, 6, 3, 2, 2, 1, 5, 1, 2, 1, 3, 1, 4, 6, 6, 1, 3,
       5, 2, 4, 1, 5, 1, 6, 2, 4, 2, 2, 2, 1, 3, 4, 6, 1, 1, 6, 1, 5, 3,
       2, 2, 4, 6, 3, 5, 6, 5, 6, 6, 1, 5, 6, 3, 3, 4, 4, 5, 5, 5, 2, 6,
       4, 4, 4, 1, 4, 1, 1, 6, 1, 6, 1, 1, 6, 3, 4, 3, 6, 1, 1, 2, 4, 4,
       3, 6, 4, 6, 5, 6, 5, 5, 4, 1, 6, 5])
```

In [ ]:
```

```

In [42]:
```
num_toss = 10000   #(each person tosses 1000 times)
num_people = 10000

means = []

for person in range(num_people):

    # toss 1000 times and give those 1000 tosses mean

    tosses = np.random.randint(1,7,num_toss) # array of number of tosses
    mean_person = np.mean(tosses)
    means.append(mean_person)

```

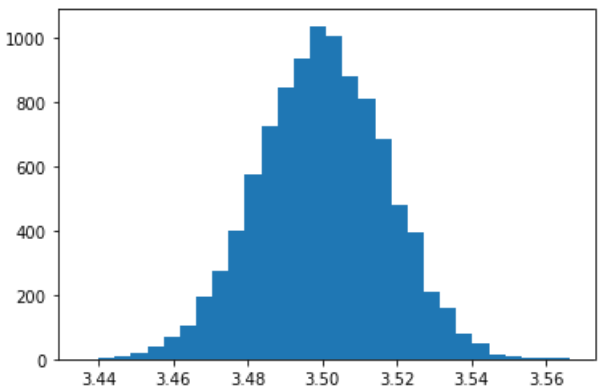
In [43]:
```
means
```

Out[43]:
```
[3.4808,
 3.5023,
 3.4871,
 3.4715,
 3.499,
 3.4811,
 3.4821,
 3.4694,
 3.4939,
 3.5102,
 3.5204,
 3.5258,
 3.4946,
 3.4987,
 3.4823,
 3.507,
 3.4939,
 3.4797,
 3.4732,
```

In [ ]:
```

```

In [44]:
```
# means # mean of all 100 people tosses 1000 times
```

In [46]:
```
plt.hist(means,bins=30)
plt.show()
```



In [ ]:
```

```

In [ ]:

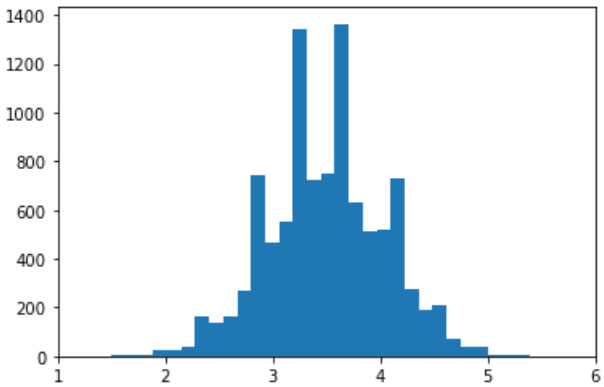
```
1
```

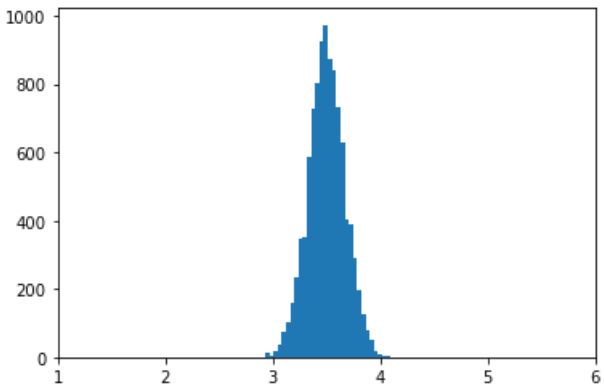
In [49]:

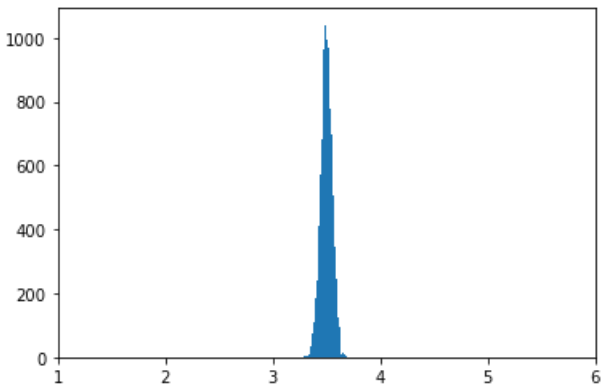
```
def whathappens_with_sample_size(num_toss):
    num_people = 10000

    means = []

    for person in range(num_people):

        # toss 10000 times and give those 1000 tosses mean

        tosses = np.random.randint(1,7,num_toss) # array of number of tosses
        mean_person = np.mean(tosses)
        means.append(mean_person)

    plt.hist(means, bins = 30)
    plt.xlim(1,6)
    plt.show()
```

In [50]:

```
whathappens_with_sample_size(10),  # 10000 people tosses 10 times # 10 samples
whathappens_with_sample_size(100), # 10000 people tosses 100 times
whathappens_with_sample_size(1000) # # 10000 people tosses 1000 times

```








```
If the sample is random, the standard error falls with the size of the sample and the sample mean's
distribution approaches the normal distribution as the sample size increases.
```

```
As sample size increases , distribution moves more towards the actual mean ,
and variance (spread) of the sampling distribution decreases (standard error decreases)
Var(x bar) reduces  as "n" increases
```

In [9]:
```
import ipywidgets

```

In [10]:
```
def plot_histogram(mean = 0 , sigma = 1, n = 100 , bins = 10):
    plt.xlim(-20,20)
    X = np.random.normal(loc = mean, scale = sigma ,size = n)
    plt.hist(X, bins = bins)
    plt.show()
```

In [11]:
```
ipywidgets.interact(plot_histogram, mean=(-10,10,0.3),
                    sigma = (0,10,0.2),
                    n = (10,10000,1),
                    bins = (1,100,1)

                    )
```

```
interactive(children=(FloatSlider(value=0.0, description='mean', max=10.0, min=-10.0, step=0.3), FloatSlider
(v…
```

Out[11]:
```
<function __main__.plot_histogram(mean=0, sigma=1, n=100, bins=10)>
```

In [ ]:
```

```

Sampling Distribution :

- a sampling distribution or finite-sample distribution is the probability distribution of a given random-sample-based statistic. If an arbitrarily large number of samples, each involving multiple observations (data points), were separately used in order to compute one value of a statistic (such as, for example, the sample mean or sample variance) for each sample, then the sampling distribution is the probability distribution of the values that the statistic takes on. In many contexts, only one sample is observed, but the sampling distribution can be found theoretically.
- Sampling distributions are important in statistics because they provide a major simplification en route to statistical inference. More specifically, they allow analytical considerations to be based on the probability distribution of a statistic, rather than on the joint probability distribution of all the individual sample values.

Expectation of the sample mean is the true exptectation

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i.$$

$$\mathrm{E}(\bar{x}) = \mu$$

- For each random variable, the sample mean is a good estimator of the population mean, where a "good" estimator is defined as being efficient and unbiased.
- Of course the estimator will likely not be the true value of the population mean since different samples drawn from the same distribution will give different sample means and hence different estimates of the true mean.
- Thus the sample mean is a random variable, not a constant, and consequently has its own distribution.
- The arithmetic mean of a population, or population mean,is often denoted μ
- For a random sample of n independent observations, the expected value of the sample mean is

  $$\mathrm{E}(\bar{x}) = \mu$$
- varinace of the sample mean is :

  $$\mathrm{var}(\bar{x}) = \frac{\sigma^2}{n}.$$

In [ ]:
```

```

Variance

```
var(X) = E[(X - E[X])^2]

```

$$\text{var}(X) = E[(X - E[X])^2] = \sum_x (x - E[X])^2 P(X = x)$$

- $\text{var}(X) = E[X]^2 - (E[X])^2$.

$\text{var}(X) =$

$$\sum_x (x - E[X])^2 P(X = x) = \sum_x \left(x^2 + (E[X])^2 - 2xE[X]\right) P(X = x)$$

$$= \sum_x x^2 P(X = x) + \sum_x (E[X])^2 P(X = x) - 2\sum_x xE[X]P(X = x)$$

$$= \sum_x x^2 P(X = x) + \underbrace{(E[X])^2 \sum_x P(X = x)}_{(E[X])^2 \text{ does not depend on } x} \underbrace{-2E[X]\sum_x xP(X = x)}_{\text{This is } E[X]}$$

$$= \sum_x x^2 P(X = x) + (E[X])^2 - 2(E[X])^2 = E[X^2] - (E[X])^2$$

Say you are looking at a linear function (or transformation) of your random variable X.

- $Y = aX + b$. Remember celsius to fahrenheit conversions? They are linear too!
- $E[Y] = E[aX + b] = aE[X] + b$, as simple as that! why?
  - $E[aX + b] = \sum_x (ax + b)P(X = x)$
    $= a\sum_x xP(X = x) + b\sum_x P(X = x)$
    $= aE[X] + b$

In [ ]:
```
# from University of Texas : Department of Data science Lecture notes:
```

what if we add constant to Random Variable, what happens to Var(x)

Let $Y = X + b$. What is $\mathrm{var}(Y)$?

- Proof:
- $$\begin{aligned}\mathrm{var}(X+b) &= E\left[(X+b)^2\right] - (E[X+b])^2 \\ &= E\left[X^2 + 2bX + b^2\right] - (E[X]+b)^2 \\ &= E[X^2] + 2bE[X] + b^2 - ((E[X])^2 + 2bE[X] + b^2) \\ &= E[X^2] - (E[X])^2 = \mathrm{var}(X)\end{aligned}$$

In [ ]:
```
1
```

Let $Y = aX$. Turns out $\mathrm{var}(Y) = a^2\mathrm{var}(X)$.

- Proof:
- $$\begin{aligned}\mathrm{var}(aX) &= E\left[(aX)^2\right] - (E[aX])^2 \\ &= E\left[a^2X^2\right] - (aE[X])^2 \\ &= a^2E\left[X^2\right] - a^2(E[X])^2 \\ &= a^2(E[X^2] - (E[X])^2) = a^2\mathrm{var}(X)\end{aligned}$$
- In general we can show that $\mathrm{var}(aX + b) = a^2\mathrm{var}(X)$.

In [ ]:
```
1
```

Variance is invariant with respect to changes in a location parameter. That is, if a constant is added to all values of the variable, the variance is unchanged:

$$\operatorname{Var}(X + a) = \operatorname{Var}(X).$$

If all values are scaled by a constant, the variance is scaled by the square of that constant:

$$\operatorname{Var}(aX) = a^2 \operatorname{Var}(X).$$

The variance of a sum of two random variables is given by

$$\operatorname{Var}(aX + bY) = a^2 \operatorname{Var}(X) + b^2 \operatorname{Var}(Y) + 2ab \operatorname{Cov}(X, Y),$$

$$\operatorname{Var}(aX - bY) = a^2 \operatorname{Var}(X) + b^2 \operatorname{Var}(Y) - 2ab \operatorname{Cov}(X, Y),$$

where $\operatorname{Cov}(X, Y)$ is the covariance.

$$\operatorname{Var}\left(\sum_{i=1}^{N} X_i\right) = \sum_{i=1}^{N} \operatorname{Var}(X_i).$$

$$\operatorname{Var}\left(\overline{X}\right) = \operatorname{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\sum_{i=1}^{n} \operatorname{Var}(X_i) = \frac{1}{n^2} n\sigma^2 = \frac{\sigma^2}{n}.$$

$$\begin{aligned}\operatorname{Var}(X+Y) &= \mathrm{E}\left[X^2\right] + 2\,\mathrm{E}[XY] + \mathrm{E}\left[Y^2\right] - \left(\mathrm{E}[X]^2 + 2\,\mathrm{E}[X]\,\mathrm{E}[Y] + \mathrm{E}[Y]^2\right) \\ &= \mathrm{E}\left[X^2\right] + \mathrm{E}\left[Y^2\right] - \mathrm{E}[X]^2 - \mathrm{E}[Y]^2 \\ &= \operatorname{Var}(X) + \operatorname{Var}(Y).\end{aligned}$$

In [ ]:
```
1
```

In [ ]:
```
1
```

In [44]:
```
1 x = np.array([7.07, 7.00, 7.10, 6.97, 7.00, 7.03 , 7.01, 8.02, 7.01, 5.38, 6.98, 7.08, 8.12])
```

In [45]:
```
1 x
```

Out[45]:
```
array([7.07, 7.  , 7.1 , 6.97, 7.  , 7.03, 7.01, 8.02, 7.01, 5.38, 6.98,
       7.08, 8.12])
```

In [46]:
```
1 x.var()
```

Out[46]:
```
0.3767147928994081
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [48]:
```
1 import math
```

In [ ]:
```
1
```

Suppose that 20% of all copies of a particular textbook fail a binding strength test. We select 15 copies at random for the binding strength test. What is the probability that at most 8 copies fail the test?

In [55]:
```
1 ans = 0
2 for i in range(0,9):
3     ans += (math.comb(15,i)) * ((0.20**i)) * ((0.80)**(15-i))
4 ans
```

Out[55]:
```
0.9992150146088968
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

If x = {1,2,3,4,5,6} # dice (1 throw)

Sample Mean :

$$\bar{X} = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}$$

Expected Value of Sample Mean

$$E[\bar{X}] = E\left[\frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}\right]$$
$$= \frac{E[x1] + E[x2] + \cdots + E[Xn]}{n}$$
$$= \frac{n\,E[x]}{n}$$
$$= E[X]$$
$$= \mu$$

$X \text{ is a random Variable}$:
$\boldsymbol{variance\ of\ X = Var[X]}$

$if\ Y = aX + b$
$Var[Y] = a^2 Var[X]$

$$if\ X = \frac{1}{2}x1 + \frac{1}{2}x2$$

$$Var\left[\frac{1}{2}x1 + \frac{1}{2}x2\right] = \frac{1}{4}Var[x1] + \frac{1}{4}Var[x2]$$

$$= \frac{1}{4}\ (2 * Var[X])$$

$$= \frac{Var[X]}{2}$$

: Variance of $[\bar{X}]$, n is the number of throw in code .

$$Var[\bar{X}] = Var\left[\ \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}\right]$$

$$= \frac{1}{n^2}\ (n * Var[X])$$

$$= \frac{Var[X]}{n}$$

$$= \frac{\sigma^2}{n}$$

as number of throw (n) increases, the variance of $Var[\bar{X}]$ decreases ,

And sample mean $E[\bar{X}]$ will be closer to the true mean μ

Sample Variance:

Variance formula : $= E(X - E[X])^2$

$$Var[X] = E[X^2] - (E[X])^2$$

Sample Variance

$$Var[\bar{X}] = \frac{(x_1 - \bar{X})^2 + (x_2 - \bar{X})^2 + \cdots \ldots \ldots + (x_n - \bar{X})^2}{n - 1}$$

$$S^2 = \frac{(x_1 - \bar{X})^2 + (x_2 - \bar{X})^2 + \cdots \ldots \ldots + (x_n - \bar{X})^2}{n - 1}$$

We want is : Expectation of sample variance to be σ^2 $Variance$

$Sample\ variance$: $E[S^2] = \sigma^2$

$Sample\ mean$: $E[\bar{X}] = \mu$

$$\bar{X} = \frac{\sum_{i=1}^{n} xi}{n}$$

$$\sum_{i=1}^{n} xi = n\,\bar{X}$$

$$Var[X] = E[X^2] - (E[X])^2$$

$$E[X^2] = Var[X] + (E[X])^2$$

$$E\left[\bar{X}^2\right] = Var[\bar{X}] + \left(E[\bar{X}]\right)^2$$

$$Var[\bar{X}] = \frac{\sigma^2}{n}$$

In [ ]:
```
1
```

$$\sum_{i=1}^{n} (x_i - \bar{X})^2 = \sum_{i=1}^{n} x_i^2 - 2\bar{X}\,x_i + \bar{X}^2$$

$$= \sum_{i=1}^{n} xi^2 - 2\bar{X} \sum_{i=1}^{n} xi + \sum_{i=1}^{n} \bar{X}^2$$

$$= \sum_{i=1}^{n} xi^2 - 2n\,\bar{X}^2 + n\,\bar{X}^2$$

$$= \sum_{i=1}^{n} xi^2 - n\,\bar{X}^2$$

$$E\left[\sum_{i=1}^{n} (x_i - \bar{X})^2\right] = E\left[\sum_{i=1}^{n} xi^2 - n\,\bar{X}^2\right]$$

$$= \sum_{i=1}^{n} E[X^2] - n\ \mathrm{E}\left[\bar{X}^2\right]$$

$$= \mathrm{n\,E}[\mathrm{x}^2] - \mathrm{n\,E}\left[\bar{X}^2\right]$$

$$= \mathrm{n}\, Var[X] + \mathrm{n}\,(E[X])^2 - \mathrm{n}\, Var[\bar{X}] - \mathrm{n}\,(E[\bar{X}])^2$$

$$= \mathrm{n}\,\sigma^2 + n\,\mu^2 - \mathrm{n}\frac{\sigma^2}{n} - \mathrm{n}\,\mu^2$$

$$= \mathrm{n}\,\sigma^2 - \mathrm{n}\frac{\sigma^2}{n}$$

$$\sum_{i=1}^{n} (x_i - \bar{X})^2 = = \mathrm{n}\,\sigma^2 - \sigma^2$$

$$\sum_{i=1}^{n} (x_i - \bar{X})^2 = = (\mathrm{n}-1)\,\sigma^2$$

$$E\left[\frac{\sum_{i=1}^{n} (x_i - \bar{X})^2}{(\mathrm{n}-1)}\right] = \sigma^2$$

$$\frac{\sum_{i=1}^{n} (x_i - \bar{X})^2}{(\mathrm{n}-1)} = S^2 \quad (Sample\ variance\ )$$

In [ ]:
```
1
```

```
 I have  samples like :  x1 , x2 , x3 , .... , xn  having varinace σ^2 and mean μ

Y = x1 + x2 + x3 + .... + xn

E[Y] =  E[x1] + E[x2] + ... + E[xn]
     = n * E[x]
     = n * μ

Var[Y] = Var[x1] + Var[x2] + ... + Var[xn]
        = n * Var(X)
       = n * σ^2


Sample mean of Y = Ybar = (x1+x2+...+yn)

```

In [ ]:
```
1
```

Dice example :

In [18]:
```
1 import numpy as np
```

lets say , throw a dice , every time we throw x on the top , we get x Rs.

if we throw , 6 on top we get 6 rs .

if we throw 100 times per person , and 1000 people are tossing

what is the expected amount that each person can get ?

In [5]:
```
num_sample = 100          # call a few people 1000(trials) and ask them to roll a dice for 100 times
samples = np.random.randint(1,7,size = num_sample)
```

In [6]:
```
samples
```

Out[6]:
```
array([6, 3, 6, 3, 6, 1, 3, 3, 4, 5, 2, 1, 3, 5, 5, 4, 5, 5, 2, 5, 6, 4,
       1, 5, 3, 2, 6, 3, 4, 1, 3, 4, 1, 3, 3, 3, 6, 4, 3, 6, 3, 6, 1, 1,
       4, 4, 2, 3, 2, 3, 6, 6, 2, 6, 4, 6, 4, 5, 2, 4, 3, 2, 6, 6, 6, 4,
       1, 5, 3, 4, 4, 3, 2, 3, 4, 5, 3, 6, 2, 4, 6, 2, 5, 1, 5, 6, 4, 3,
       1, 4, 3, 1, 6, 5, 2, 6, 2, 6, 6, 6])
```

In [7]:
```
num_sample = 100
mean = 3.5      # known from distribution            # dice example
variance = 35/12  # known from distribution
```

```
X = {1,2,3,4,5,6}


# E(X) = Σ(x*P(x))
= (1*(1/6))+(2*(1/6))+(3*(1/6))+(4*(1/6))+(5*(1/6))+(6*(1/6))
= 3.5

E(X) = 3.5 = 7/2

X^2 = {1,4,9,16,25,36}
E(X^2) = (1*(1/6)) + (4*(1/6))  + (9*(1/6)) + (16*(1/6)) + (25*(1/6)) + (36*(1/6))

E(x^2) = 15.166 = 91/6

Var(X) = E[ ( X - E(X) )^2 ]
       = E(X2) - (E(X))2
       =  91/6 - (7/2)^2
       = 2.91
       = 35/12
```

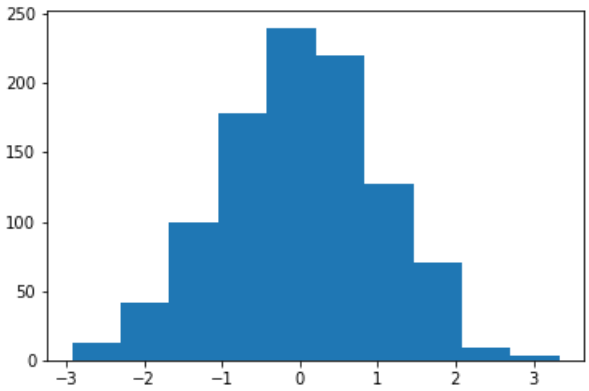
In [8]:
```
trial_values = []
for trial in range(1000):                          # 1000 people tossing 100 times each person
    samples = np.random.randint(1,7,size = 100)   # generating samples : 100 tosses per person
    val = np.sum(samples)                        # Y = x1+ x2 +x3 ... +x100
    trial_values.append(val)                    # appending all sums in trials_values
trial_values # Y1 Y2 Y3. ... Y1000
```

Out[8]:
```
[359,
 357,
 347,
 343,
 364,
 365,
 370,
 340,
 368,
 336,
 325,
 335,
 335,
 331,
 341,
 315,
 355,
 352,
 322,
```

In [19]:
```
plt.hist(trial_values)
plt.show()
# historgram shows , # of calling  1000 people and asking people dice 100 times.
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.Javascript object>
```



In [10]:
```
# num_sample = 100
# mean = 3.5                  # dice example
# variance = 35/12
```

In [11]:
```
np.mean(trial_values)    # n*mean

# np.mean(trial_values) = mean of [Y1 Y2 ... Y1000]
```

Out[11]: 349.953

In [12]:
```
num_sample* mean # expectation
```

Out[12]: 350.0

In [13]:
```
np.var(trial_values)   # observed
```

Out[13]: 287.51479100000006

In [14]:
```
num_sample * variance             # expecatation n*variance

```

Out[14]: 291.66666666666663

```
sample mean = sum of all samples / size

mean value of 100 tosses by 1000 people =
           sum of all samples : sum(trial_values)  / 1000 people

```

In [40]:
```
sum(tri  al_values)/ 1000
```

Out[40]: 351.532

In [ ]:
```

```

$$\begin{aligned}\operatorname{Var}(X+Y) &= \mathrm{E}\left[X^2\right] + 2\,\mathrm{E}[XY] + \mathrm{E}\left[Y^2\right] - \left(\mathrm{E}[X]^2 + 2\,\mathrm{E}[X]\,\mathrm{E}[Y] + \mathrm{E}[Y]^2\right) \\ &= \mathrm{E}\left[X^2\right] + \mathrm{E}\left[Y^2\right] - \mathrm{E}[X]^2 - \mathrm{E}[Y]^2 \\ &= \operatorname{Var}(X) + \operatorname{Var}(Y).\end{aligned}$$

```
Var[x + x] = Var[x] + Var[x] = 2*Var[x]
```

```
samples like :  x1 , x2 , x3 , .... , xn  having varinace σ^2 and mean μ

Y = x1 + x2 + x3 + .... + xn

E[Y] =  E[x1] + E[x2] + ... + E[xn]
     = n * E[x]
     = n * μ

Var[Y] = Var[x1] + Var[x2] + ... + Var[xn]   # Var[x1]=Var[x2]=Var[x3]=....=Var[xn]
        = n * Var(X)
       = n * σ^2


Sample mean of Y = Ybar = (x1+x2+...+yn)
```

from Central limit Theorem

says that {x1,x2,x3,...,xn} samples have variance σ^2 and mean μ

- x1+x2+x3....+xn is approximately normally distributed with

    - mean = n * μ
    - variance = n * σ^2

$$X : \{x1, x2, x3, \ldots \ldots, xn\}$$
$$has, variance\ \sigma^2$$
$$mean\ \mu$$

$$mean\ of\ X = E[X] = n\mu$$
$$variance\ of\ X = \ Var[X] = \ n\sigma^2$$

Standardisation :

X = x1 + x2 ....+ xn are approximately normally distributed having

- mean = n * μ
- variance = n * σ^2

$$X \sim N(n\mu, n\sigma^2)$$
$$Z = \frac{X - n\mu}{\sqrt{Var[X]}}$$

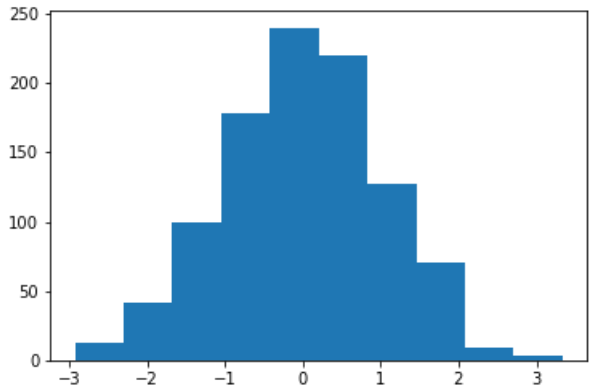
$$Z = \frac{\bar{X} - n\mu}{\sigma\sqrt{n}}$$

In [ ]:
```
```

In [15]:
```
num_sample = 100
mean = 3.5      # known from distribution            # dice example
variance = 35/12  # known from distribution


trial_values = []
for trial in range(1000):
    samples = np.random.randint(1,7,size = 100)
#     val = np.sum(samples)

    val = (np.sum(samples) - num_sample*mean)   /  (np.sqrt(num_sample * variance))
    #          (  x    -    n     *    u  )   /   squareroot(   n      *  sigmasq)
    trial_values.append(val)
plt.hist(trial_values)
plt.show()
# historgram shows , # of calling  1000 people and asking people dice 100 times.
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.Javascript object>
```



In [16]:
```
np.var(trial_values)  # this will become close to 1
```

Out[16]:
```
0.997614085714286
```

In [17]:
```
np.mean(trial_values)   # this will bcome close to 0
```

Out[17]:
```
0.010832490809728713
```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```

```

Example :

Insurance company : has given 25000 policies' , yearly claim's mean = 320 and standard deviation = 540 "

we want to know approximately what is the probability that claim is greater than 8.3 million

```
mean = 320
std = 540

X = x1 + x2 + .. x25000

X : total yealy claim:
        P[X > 8.3 million ]  = P[sum of the claim of 25000 people >8.3 million]



 (25000*320)
```

$$Z = \frac{\bar{X} - n\mu}{\sigma\sqrt{n}}$$

In [20]:
```
8300000 , 25000*320
```

Out[20]: (8300000, 8000000)

In [21]:
```
(8300000 - 8000000)  / ((540)*np.sqrt(25000))

```

Out[21]: 3.513641844631532

In [83]:
```
stats.norm.cdf(3.51)
```

<IPython.core.display.Javascript object>

Out[83]: 0.999775946653009

In [84]:
```
1-stats.norm.cdf(3.51)
```

<IPython.core.display.Javascript object>

Out[84]: 0.00022405334699104884

In [ ]:
```

```

In [ ]:
```

```

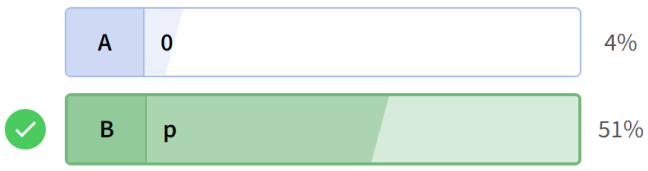
Binomial Random variable :

```
n trials
k success
k <= n

Y = x1+ x2 + x3 +  ..... + xn

p : probability of success
1-p : probabiliy of failure m
```

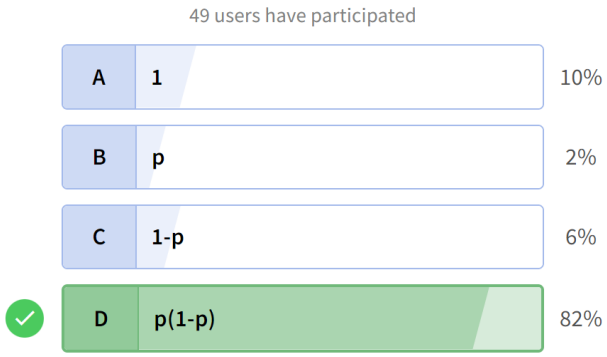
In [ ]:
```

```

For the Bernoulli random variable with parameter p, what is the expectation

55 users have participated



For a Bernoulli random variable with parameter p, what is the variance




```
toss example :

    xi :  0 : ith toss is tails T
          1 : ith toss is heads H

            E[xi] = p

            Y = x1+ x2 + x3 +  ..... + xn
        E[Y] = n *E(xi) =  n*p


    Var[Y] = n * Var(xi)
           = n * (E(x^2)-((E(x))^2)
           = n * (p     - np^2)
           = n * p * (1-p)
```

Gaussian Approximation to Binomial :

$$Z = \frac{Y - np}{\sqrt{np(1-p)}} \sim N(0,1)$$

Binomial R V :

$$Y = x1 + x2 + \cdots + xn$$
$$E(Y) = E(x1) + E(x2) + \cdots + E(xn)$$
$$E(Y) = nE(x)$$

$$E(x) = p \; for \; binomial \; RV$$
$$E(Y) = np$$
$$Var(Y) = Var(x1) + Var(x2) + \cdots + Var(xn)$$
$$= nVar(x)$$
$$= nE(x^2) - n\left(E(x)\right)^2$$
$$= n(p) - n(p^2)$$
$$= np(1-p)$$
$$= npq$$
$$q = 1 - p$$

In [ ]:
```
1
```

In [ ]:
```
1
```

150 student capacity in class

among those accepted , only 30 % student attend the class and others drop out

college accepts 450 students

what is the probability of more than 150 people attend

In [54]:
```
1 from scipy.stats import binom
```

In [ ]:
```
1 # binomial formula
```

In [63]:
```
1 ans = 0
2 for i in range(151,451):
3     ans += binom.pmf(i,450,0.30)
4 ans
```

Out[63]: 0.056516944458631466

In [64]:
```
1 np.round(ans,2)
```

Out[64]: 0.06

In [ ]:
```
1 # using gaussian
```

In [28]:
```
1 from scipy.stats import norm
```

$$Z = \frac{X - n\mu}{\sigma\sqrt{n}}$$

In [116]:
```
1 1-norm.cdf((150-(450*0.30))/np.sqrt(450*0.30*0.7),0,1)
2 #            X -   n * p  /    (sq(np(1-p)))
3           #   X - mean /   sq(varinace)
```

Out[116]: 0.06141132405069627

In [66]:
```
1 np.round(1-norm.cdf((150-(450*0.30))/np.sqrt(450*0.30*0.7),0,1),2)
```

Out[66]: 0.06

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

airline case of overbooking :

50 seats are aavilbale in aircraft 52 tickets has been sold 5% of people dont show up

Q: P[all who show up gets the seat] = ?

probability of all 52 people show up : P[all of people who bought ticket shows up]

```
p[one person show up] = 1- P[person dont shw up]
                      = 1- 5%
                      = 0.95


P[all of people who bought ticket shows up] = (0.95)**52



P[51 people show up] = C(52,51 ) * ((0.95)**51) * (0.05)
                                51 show-up       1 dont show up


so , probability of 50 people show up so they all have seat is :

    = 1 - P[52 people show up] - P[51 people showup]
    = 1 - (0.95**52) - (52 * (0.95**51)*0.05)
```

P[all who show up gets the seat]

In [58]:
```
1 - (0.95**52) - (52 * (0.95**51)*0.05)
```

Out[58]: 0.7405030708792849

In [ ]:
```
# coding way
```

In [59]:
```
ans = 0
for i in range(51,53):
    ans += binom.pmf(i,52,0.95)
ans    #  =0.25949692912071526
print(1-ans)
```

```
0.7405030708792848
```

In [ ]:
```

```

In [62]:
```
# 1-norm.cdf((150-(450*0.30))/np.sqrt(450*0.30*0.7),0,1)
```

In [ ]:
```

```

In [ ]:
```

```

Sample Mean

$$\bar{X} = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}$$

$$E[\bar{X}] = E\left[\frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}\right]$$
$$= E[X]$$
$$= \mu$$

Variance of $\bar{X}$

$$Var[\bar{X}] = Var\left[\frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}\right]$$
$$= \frac{1}{n^2}(n * Var[X])$$
$$= \frac{Var[X]}{n}$$
$$= \frac{\sigma^2}{n}$$

$$Z = \frac{\bar{X} - E[\bar{X}]}{\sqrt{Var(\bar{X})}}$$

$$Z = \frac{\bar{X} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

In [ ]:
```
1
```

Confidence Interval

weights : population mean =167 pounds , std = 27 pounds

a sample of 36 people are taken :

mean for those 36 people from 163 to 170

$$\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{n}} = \frac{27}{\sqrt{36}} = 4.5$$

$$P[163 < \overline{X} < 170] = P\left[\frac{163-167}{4.5} < \frac{\overline{X}-167}{4.5} < \frac{170-167}{4.5}\right]$$
$$= P[-0.889 < Z < 0.6667]$$

what is the probability of mean be in 163 to 170 pounds when taken sample of 36 people weights

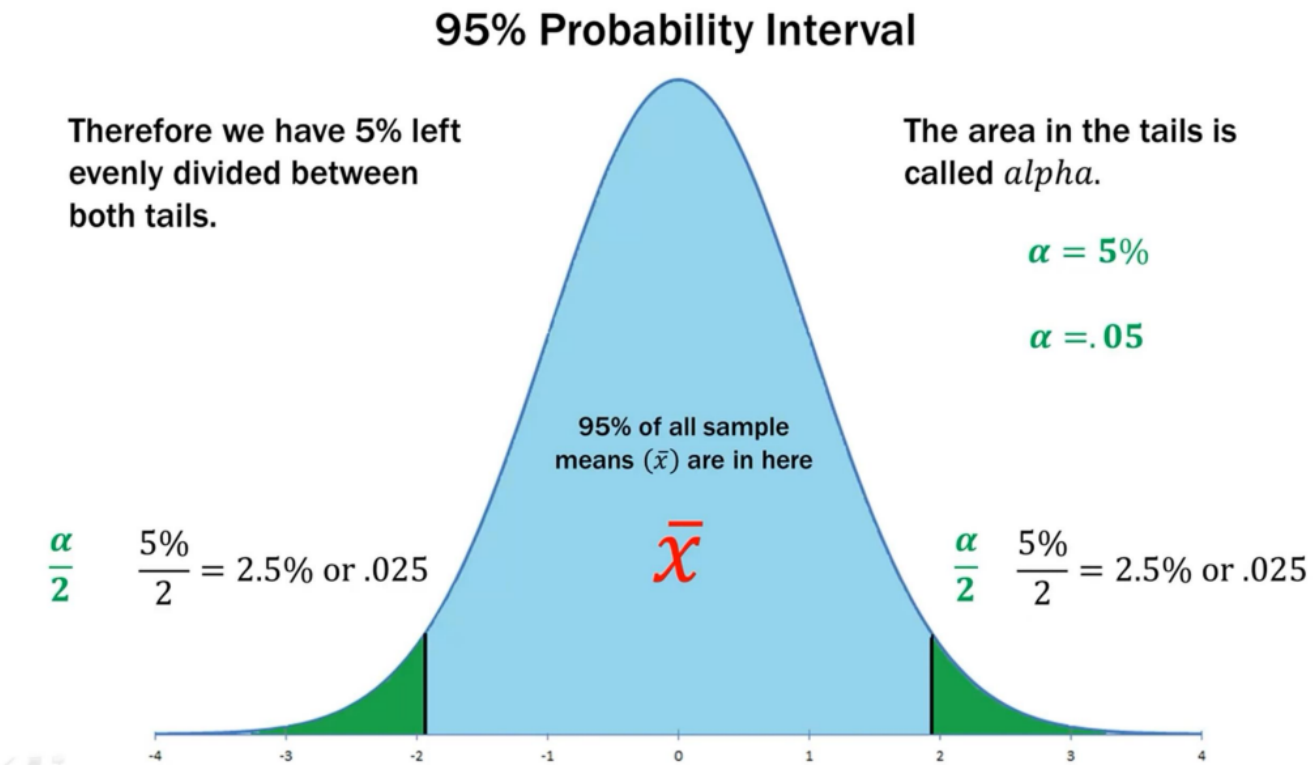
In [33]:
```
from scipy.stats import norm
```

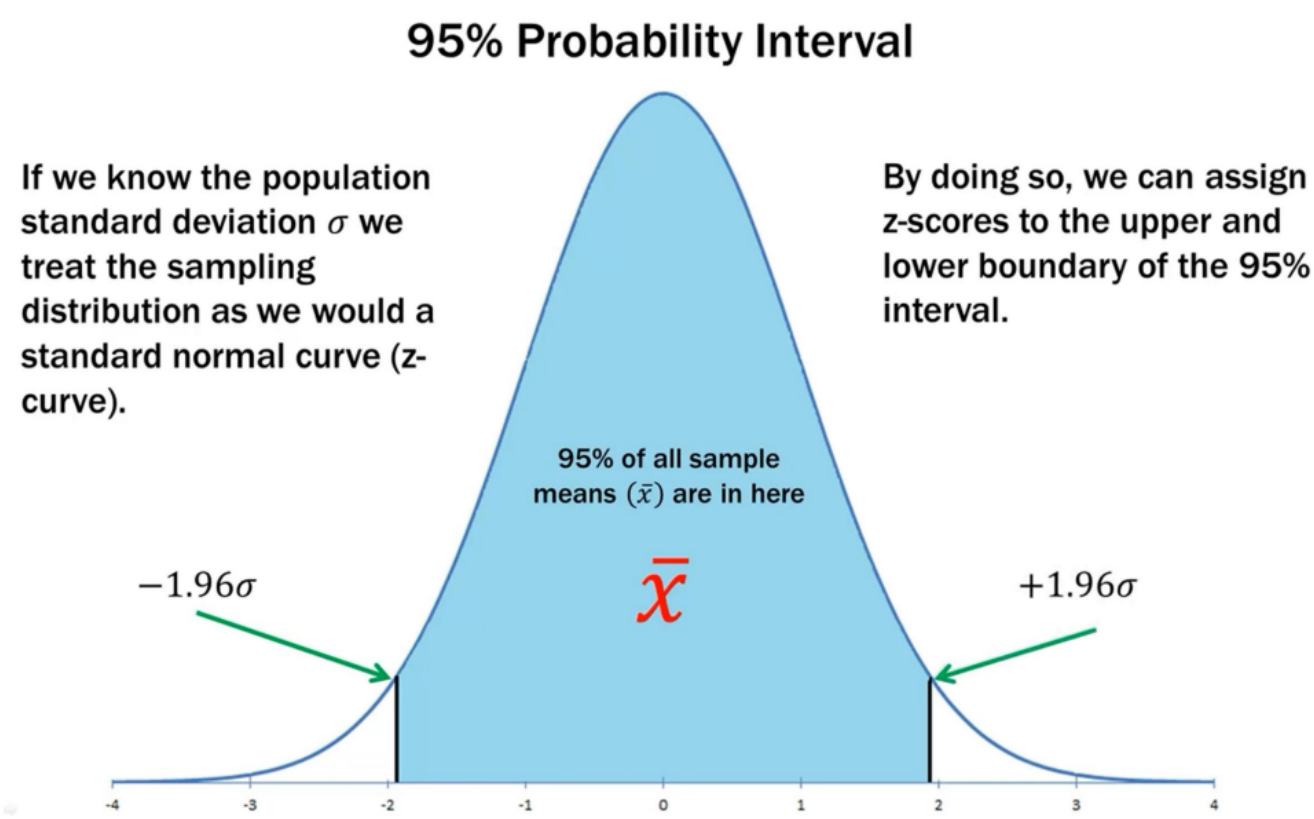
```
P[163 < xbar < 170]  = cdf P[z< 0.6667] - cdf P[Z< -0.889]
```

In [47]:
```
norm.cdf(0.6667) - norm.cdf(-0.8889)
```

Out[47]:
```
0.5604896978732206
```

In [ ]:
```

```





In [50]:
```
1 norm.cdf(1.96) , norm.cdf(-1.96)
```

Out[50]: (0.9750021048517795, 0.024997895148220435)

In [51]:
```
1 norm.cdf(1.96) - norm.cdf(-1.96)
```

Out[51]: 0.950004209703559

In [53]:
```
1 norm.ppf(0.975),norm.ppf(0.025)
```

Out[53]: (1.959963984540054, -1.9599639845400545)

In [8]:
```
1 -8+(1.96*0.510)
```

Out[8]: -7.0004

$$P[-1.96 < Z < 1.96] = 0.95$$

$$P\left[-1.96 < \frac{\overline{X}-\mu}{\frac{\sigma}{\sqrt{n}}} < 1.96\right] = 0.95$$

$$P\left[-1.96\frac{\sigma}{\sqrt{n}} < \overline{X} - \mu < 1.96\frac{\sigma}{\sqrt{n}}\right] = 0.95$$

$$P\left[-1.96\frac{\sigma}{\sqrt{n}} < \mu - \overline{X} < 1.96\frac{\sigma}{\sqrt{n}}\right] = 0.95$$

$$P\left[\left(\overline{X} - 1.96\frac{\sigma}{\sqrt{n}}\right) < \mu < \left(\overline{X} + 1.96\frac{\sigma}{\sqrt{n}}\right)\right] = 0.95$$

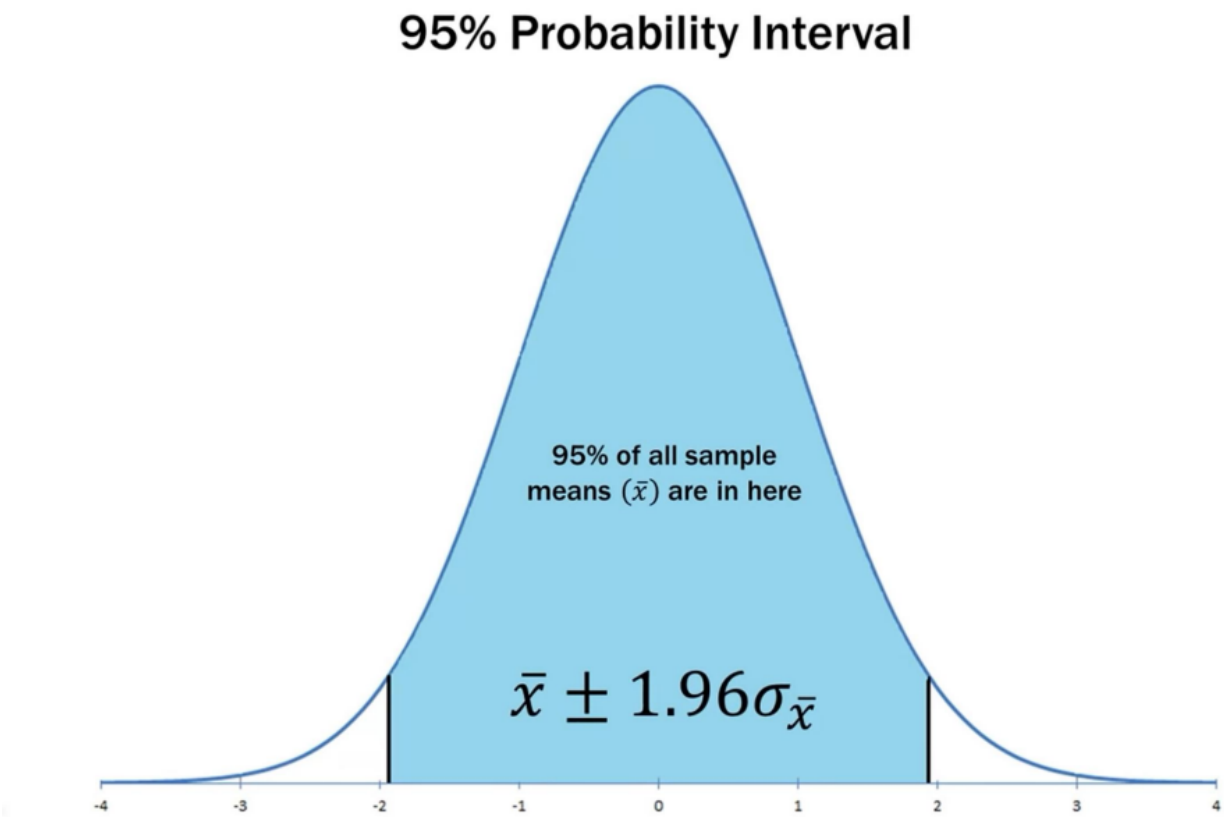
True/ Population mean , lies in an interval of

$$\left[\left(\overline{X} - 1.96\frac{\sigma}{\sqrt{n}}\right), \left(\overline{X} + 1.96\frac{\sigma}{\sqrt{n}}\right)\right]$$

with 95% Confidence

$$\sigma_{\overline{X}} = \frac{\sigma}{\sqrt{n}}$$

$$\left[\left(\overline{X} \pm 1.96\,\sigma_{\overline{X}}\right)\right]$$



In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [ ]:
```
1
```