If a random variable X is normally distributed with mean μ and standard deviation σ, then

$$(X - \mu) / \sigma$$

is distributed as a normal random variable with mean 0 and standard deviation 1

(i.e. is a standard normal distribution)

—

The Central Limit Theorem:

Suppose that a sample from a population is obtained containing many observations of fixed size, say, n, each observation being randomly generated in a way that does not depend on the values of the other observations, and that the arithmetic mean (average) statistic, say $\overline{X}$, of the observed values is computed.

If this procedure is performed many times, the central limit theorem says that the distribution of the arithmetic mean statistic, $\overline{X}$, will be closely approximated by a normal distribution with

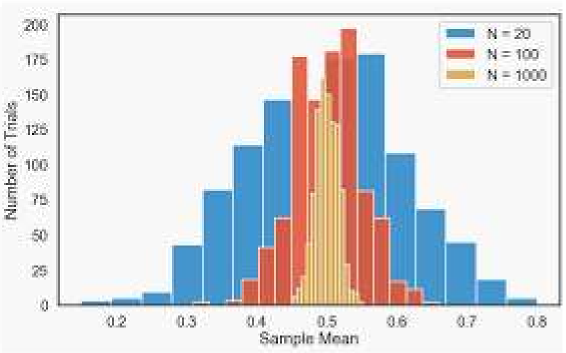
mean given by μ, the mean of the population, and with
standard deviation given by $\sigma/\sqrt{n}$ where σ is the standard deviation of the population.

Equivalently, the distribution of

$$(\overline{X} - \mu) / \sigma/\sqrt{n}$$

will be approximately the standard normal distribution (i.e. with mean 0 and standard deviation 1)

The larger the value of n, the “better” the approximation to normal.
Also note that the larger the value of n, the smaller the standard deviation, $\sigma/\sqrt{n}$, becomes.



B Testing Part 1: The Central Limit Theorem
medium.com

From https://stackoverflow.com/questions/20626994/how-to-calculate-the-inverse-of-the-normal-cumulative-distribution-function-in-p

Assuming the distribution of X (house price) is normal
with mean $\mu = 60$ and standard deviation $\sigma = 40$

Python Code:

```
import scipy as sc
import scipy.stats as sct
sc.version.full_version # 0.15.1

#a. Find P(X<50)
print(sct.norm.cdf(x=50,loc=60,scale=40)) # 0.4012936743170763

#b. Find P(X>=50)
print(sct.norm.sf(x=50,loc=60,scale=40)) # 0.5987063256829237

#c. Find P(60<=X<=80)
print(sct.norm.cdf(x=80,loc=60,scale=40)- sct.norm.cdf(x=60,loc=60,scale=40))

#d. how much top most 5% expensive house cost at least? or find x where P(X>=x) = 0.05
print(sct.norm.isf(q=0.05,loc=60,scale=40))

#e. how much top most 5% cheapest house cost at least? or find x where P(X<=x) = 0.05
print(sct.norm.ppf(q=0.05,loc=60,scale=40))
```

Output:

```
0.4012936743170763
0.5987063256829237
0.19146246127401312
125.79414507805892
-5.794145078058918
```

The acronyms

cdf	cummulative distribution function
sf	survival function
isf	inverse suvival function
ppf	percent point function