

How can the Normal CDF be used in R, and what are some examples of implementing it?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can the Normal CDF be used in R, and what are some examples of implementing it?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165468>

The Normal CDF (Cumulative Distribution Function) is a statistical tool used to calculate the probability of a random variable falling within a certain range. In R, the Normal CDF can be implemented using the "pnorm()" function, which takes in parameters for the value, mean, and standard deviation of the normal distribution. It can also be used to find the area under a normal curve, which can be useful in various statistical analyses. For example, one can use the Normal CDF in R to determine the probability of a stock price falling within a certain range, or to calculate the probability of a student achieving a certain score on a standardized test. Additionally, the Normal CDF can be used in hypothesis testing to determine the likelihood of obtaining a certain result by chance. Overall, the Normal CDF is a powerful tool in R for analyzing and interpreting data in a wide range of fields.

Use the Normal CDF in R (With Examples)

You can use the following methods to work with the normal CDF (cumulative distribution function) in R:

Method 1: Calculate Normal CDF Probabilities

#calculate probability that random value is less than 1.96 in normal CDF

```
pnorm(1.96)
```

#calculate probability that random value is greater than 1.96 in normal CDF

```
pnorm(1.96, lower.tail=FALSE)
```

Method 2: Plot the Normal CDF

#define sequence of x-values

```
x <- seq(-4, 4, .01)
```

```
#calculate normal CDF probabilities
```

```
prob <- pnorm(x)
```

```
#plot normal CDF
```

```
plot(x, prob, type="l")
```

The following examples show how to use these methods in practice.

Example 1: Calculate Normal CDF Probabilities

The following code shows how to calculate the probability that a random variable takes on a value less than 1.96 in a standard normal distribution:

```
#calculate probability that random value is less than  
1.96 in normal CDF
```

```
pnorm(1.96)
```

```
0.9750021
```

The probability that a random variables takes on a value less than 1.96 in a standard normal distribution is 0.975.

We can also find the probability that a random variable takes on a value greater than 1.96 by using the `lower.tail` argument:

#calculate probability that random value is greater than 1.96 in normal CDF

```
pnorm(1.96, lower.tail=FALSE)
```

0.0249979

And we can use the following syntax to find the probability that a random variable takes on a value between two values in a standard normal distribution:

#calculate probability that random value takes on value between -1.96 and 1.96

```
pnorm(1.96) - pnorm(-1.96)
```

0.9500042

The probability that a random variable takes on a value between -1.96 and 1.96 in a standard normal distribution is 0.95.

Example 2: Plot the Normal CDF

The following code shows how to plot a normal CDF:

```
#define sequence of x-values
```

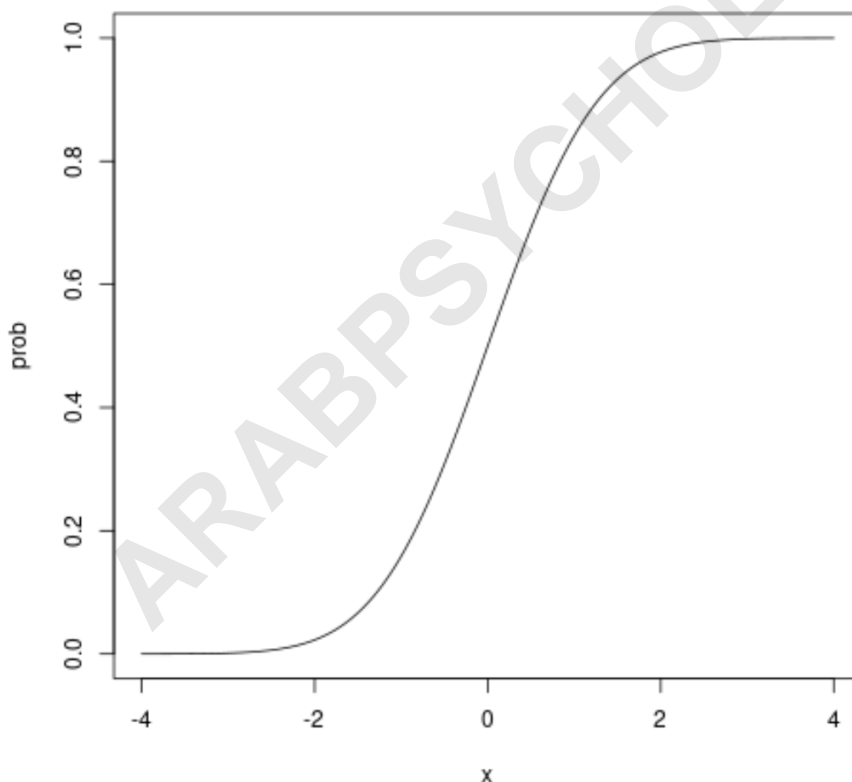
```
x <- seq(-4, 4, .01)
```

```
#calculate normal CDF probabilities
```

```
prob <- pnorm(x)
```

```
#plot normal CDF
```

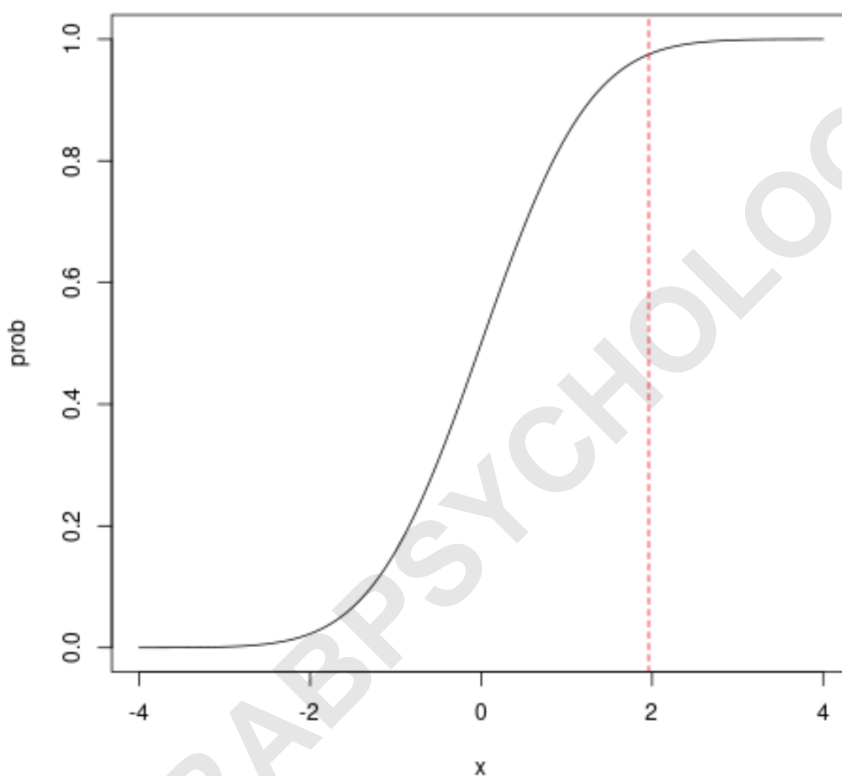
```
plot(x, prob, type="l")
```



The x-axis shows the values of a random variable that follows a standard normal distribution and the y-axis

shows the probability that a random variable takes on a value less than the value shown on the x-axis.

For example, if we look at $x = 1.96$ then we'll see that the cumulative probability that x is less than 1.96 is roughly 0.975:



Note that you can modify the aesthetics of the normal CDF plot as well:

#define sequence of x-values

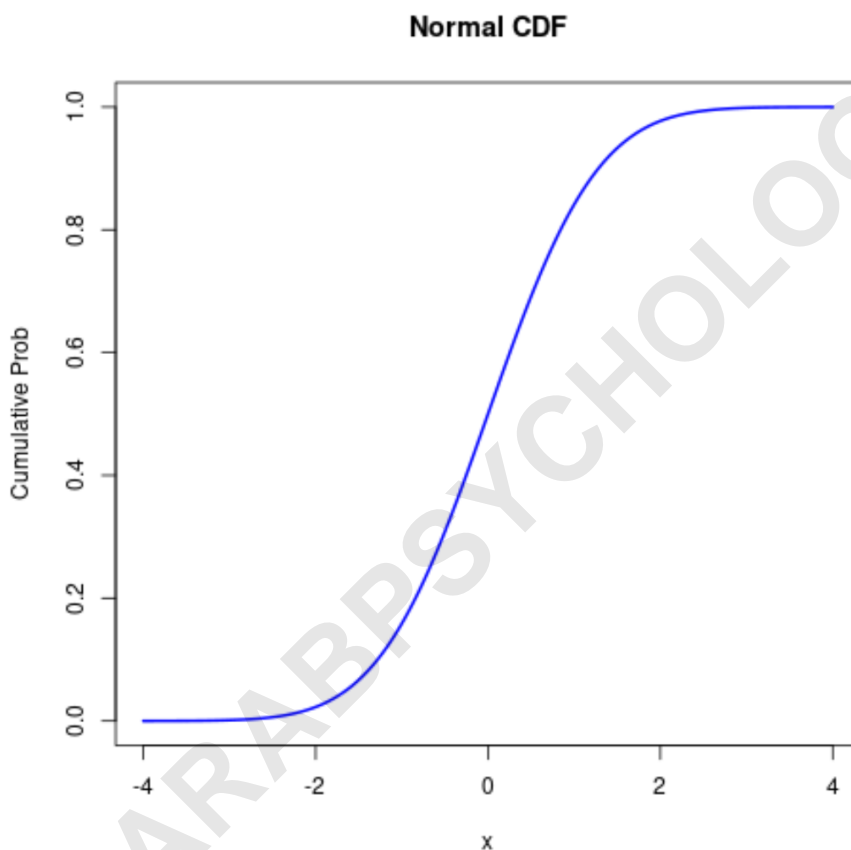
```
x <- seq(-4, 4, .01)
```

```
#calculate normal CDF probabilities
```

```
prob <- pnorm(x)
```

```
#plot normal CDF
```

```
plot(x, prob, type='l', col='blue', lwd=2, main='Normal  
CDF', ylab='Cumulative Prob')
```



Related:

Additional Resources