

Learning the Geometric Distribution in R: A Tutorial on dgeom, pgeom, qgeom, and rgeom Functions

Authored by
Mohammed looti

November 8, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Learning the Geometric Distribution in R: A Tutorial on dgeom, pgeom, qgeom, and rgeom Functions*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=13083>

The [Geometric Distribution](#) is a cornerstone concept in [probability theory](#). It serves as a powerful model for determining the number of independent [Bernoulli Trials](#) necessary to observe the very first successful outcome. Analyzing and simulating this distribution efficiently within the statistical programming environment [R](#) is achieved through a specialized family of four functions. This comprehensive guide details the effective implementation of these functions:

dgeom: Calculates the value of the geometric [Probability Mass Function](#) (PMF), yielding the exact probability of achieving the first success after a specific number of failures.

pgeom: Computes the geometric [Cumulative Distribution Function](#) (CDF). This determines the cumulative probability of finding the first success within a defined maximum number of trials.

qgeom: Represents the inverse of the CDF, often called the [Quantile Function](#). It identifies the number of preceding failures corresponding to a specified cumulative probability percentile.

rgeom: Facilitates [simulation](#) by generating a vector of random variables that adhere to the parameters of the Geometric Distribution.

A deep understanding of the purpose and practical application of each function is essential for accurate [statistical modeling](#) and data analysis. The following sections provide detailed explanations, syntax definitions, and practical R examples demonstrating their utility in common probability scenarios.

Using `dgeom`: The Probability Mass Function (PMF)

The **dgeom** function is designed to calculate the precise probability of observing a specific, exact number of failures immediately preceding the first success in a sequence of independent Bernoulli trials. As the implementation of the Geometric [Probability Mass Function](#) (PMF), it is the primary tool for calculating point probabilities. If we denote the number of failures before the first success as the random variable X , **dgeom** calculates $P(X=x)$. It requires two core parameters: x , the specific number of preceding failures, and $prob$, the constant probability of success on any single trial.

The standard syntax for utilizing the **dgeom** function is concise and requires the mandatory input of the failure count and the success probability:

dgeom(x, prob)

The necessary input parameters are rigorously defined as follows:

x: Specifies the exact integer count of failures observed immediately prior to the first successful outcome.

prob: Represents the constant likelihood (between 0 and 1) of success occurring during any single independent [trial](#).

Consider a scenario where a political researcher is interviewing voters, and the established probability of finding a supporter (success) is $p = 0.2$. The researcher is interested in the specific probability that the fourth person interviewed is the very first supporter. If the success occurs on the fourth trial, it means there were exactly three failures ($x=3$) preceding it. We use **dgeom** to find this exact point probability:

```
dgeom(x=3, prob=.2)
```

```
#0.1024
```

The resulting calculation shows that the probability of the researcher encountering exactly three non-supporters (failures) before finding the first supporter is **0.1024**. This calculation provides a precise measure of a specific, discrete outcome within the established geometric distribution parameters.

Using **pgeom**: The Cumulative Distribution Function (CDF)

In contrast to **dgeom**, the **pgeom** function calculates the cumulative probability associated with the Geometric Distribution. Instead of determining the probability of a single, exact outcome, **pgeom** sums the probabilities of all outcomes up to and including a specified maximum number of failures (q) before the first success. This corresponds directly to the Geometric [Cumulative Distribution Function](#) (CDF), calculating $P(X \leq q)$. This functionality is essential when determining probabilities for ranges of outcomes, such as scenarios involving "at most" or "no more than" a certain number of trials.

The syntax for **pgeom** is structurally similar to **dgeom**, requiring the upper boundary for failures (q) and the probability of success ($prob$):

```
pgeom(q, prob)
```

The arguments required for this cumulative calculation are defined as:

q: Defines the maximum number of failures allowed before the first success. The function aggregates the probabilities for all outcomes from $x = 0$ up to $x = q$.

prob: Represents the constant probability of success on any single trial.

Revisiting the political survey example (where $p=0.2$), suppose the researcher wants to know the probability that they find the first supporter within the first four people interviewed. Finding the supporter by the fourth person means the number of failures before success must be 0, 1, 2, or 3. Therefore, we set $q=3$ (failures before success) and calculate the cumulative probability $P(X \leq 3)$.

```
pgeom(q=3, prob=.2)
```

```
#0.5904
```

The probability that the researcher experiences 3 or fewer failures before finding the first success (i.e., finding the supporter by the 4th trial) is **0.5904**. The CDF is also powerful for calculating upper-tail probabilities using the complement rule. For instance, to find the probability that the researcher must interview more than 6 people (meaning 6 or more trials, or $X \geq 6$ failures), we calculate $P(X > 5) = 1 - P(X \leq 5)$.

```
1 - pgeom(q=5, prob=.2)
```

```
#0.262144
```

This result shows that the probability of needing to interview more than 6 individuals (experiencing more than 5 failures before success) is **0.262144**. This illustrates the flexibility of **pgeom** in handling both lower-tail and upper-tail probability calculations crucial for comprehensive [statistical analysis](#).

Using qgeom: The Inverse CDF (Quantile Function)

The **qgeom** function performs the inverse operation of the cumulative distribution function calculated by **pgeom**. Instead of accepting a failure count and returning a probability, **qgeom** accepts a specific cumulative probability (or percentile) and returns the corresponding minimum number of failures required to reach or exceed that probability threshold. This functionality is formally known as the [Quantile Function](#). It is an indispensable tool in risk assessment, quality control, and determining confidence intervals, enabling analysts to define the maximum expected number of failures associated with a high cumulative likelihood.

The required syntax for **qgeom** specifies the target cumulative probability (p) and the constant success rate (prob):

```
qgeom(p, prob)
```

The function arguments are explicitly defined as follows:

p: The target cumulative probability (percentile) for which the corresponding quantile (number of failures) is to be determined. This value must strictly lie between 0 and 1.

prob: The constant probability of success on any given trial.

Continuing with our example where the success probability is $p = 0.2$, the researcher might seek

to determine the number of failures that defines the 90th percentile of this geometric distribution. In practical terms, this means finding the maximum number of failures (x) such that $P(X \leq x) = 0.90$. This result establishes a critical threshold: the maximum number of people expected to say "no" (failures) in 90% of all possible survey attempts.

`qgeom(p=.90, prob=0.2)`

#10

The output indicates that the researcher must be prepared to experience up to **10** "failures" before finding the first success to ensure coverage of 90% of all possible scenarios. Consequently, in nine out of ten attempts, the first supporter will be found by the 11th person interviewed (10 failures + 1 success). This quantile provides crucial insight for planning resources based on probabilistic outcomes.

Using `rgeom`: Random Variable Generation for Simulation

The **`rgeom`** function plays a crucial role in simulation and Monte Carlo methods, enabling researchers to generate synthetic data that follows the Geometric Distribution. Specifically, **`rgeom`** produces a specified number of independent [random variables](#) (n), where each generated value represents the simulated number of failures that occurred before the first success, based on the input probability (p). Simulation is an invaluable technique for modeling the expected variability, calculating descriptive statistics, and understanding the long-run behavior of probabilistic processes in complex real-world systems.

The syntax for **`rgeom`** is straightforward, requiring the desired number of observations (n) and the constant success probability:

`rgeom(n, prob)`

The function requires the following two essential parameters:

`n`: The total number of independent random values (or simulation trials) that the function should generate.

`prob`: The consistent probability of success on any single trial, which fundamentally defines the shape and parameters of the resulting geometric distribution.

To demonstrate variability, let us simulate the library survey process 10 times, maintaining the success probability at $p = 0.2$. We are generating 10 independent outcomes, each representing the number of failures encountered until the first supporter is found. By using **`set.seed(0)`**, we ensure that the generated sequence of random numbers is reproducible, allowing others to verify

the exact results.

set.seed(0) #make this example reproducible

```
rgeom(n=10, prob=.2)
```

```
# 1 2 1 10 7 4 1 7 4 1
```

The resulting vector of 10 integer values provides empirical data points reflecting the distribution's inherent randomness. These values are interpreted as the number of failures before the first success in each of the 10 simulated scenarios:

The first simulation required 1 failure (success on the 2nd person).

The second simulation required 2 failures (success on the 3rd person).

The fourth simulation required 10 failures (success on the 11th person).

Other trials showed results of 1, 7, 4, 1, 7, 4, and 1 failures, respectively.

Analyzing the mean and variance of this generated vector can provide valuable insights into the expected behavior of the geometric process over repeated iterations, validating theoretical calculations derived from **dgeom** and **pgeom**.

Conclusion and Further Exploration

The four core functions--**dgeom**, **pgeom**, **qgeom**, and **rgeom**--provide a complete toolkit for comprehensively analyzing and simulating the [Geometric Distribution](#) within the R environment. Mastery of these functions allows data analysts and statisticians to accurately calculate point probabilities, determine cumulative thresholds, define critical quantiles, and perform robust simulations for any process modeled by independent Bernoulli trials.

To further advance your skills in discrete probability distributions using R, we strongly recommend exploring the official documentation and tutorials for related distributions, such as the Binomial, Poisson, and Hypergeometric distributions. These distributions utilize the same powerful and consistent functional naming conventions (d/p/q/r), ensuring a streamlined transition to more complex [statistical modeling](#) tasks.