

# Package: StatDeriveR (via r-universe)

May 11, 2026

**Title** Step-by-Step Derivation and Simulation Verification in  
Mathematical Statistics

**Version** 0.1.0

**Description** Provides step-by-step derivations of key results in mathematical statistics, including transformations of random variables, order statistics, and sampling distributions. The package combines analytical derivation with Monte Carlo simulation to compare theoretical and empirical results, facilitating deeper understanding of statistical theory and its computational implementation. The methods are motivated by standard treatments in mathematical statistics (Hogg, McKean, and Craig, 2019, ISBN: 9780134686991).

**License** MIT + file LICENSE

**Encoding** UTF-8

**Language** en-US

**Depends** R (>= 4.0.0)

**Imports** grDevices, graphics, stats

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**RoxygenNote** 7.3.3

**URL** <https://github.com/Jokhrof042/StatDeriveR>

**BugReports** <https://github.com/Jokhrof042/StatDeriveR/issues>

**NeedsCompilation** no

**Author** Jokhrof Ahmed Doha [aut, cre]

**Maintainer** Jokhrof Ahmed Doha <s-2021711817@isrt.du.ac.bd>

**Repository** <https://jokhrof042.r-universe.dev>

**Date/Publication** 2026-04-10 15:10:33 UTC

**RemoteUrl** <https://github.com/cran/StatDeriveR>

**RemoteRef** HEAD

**RemoteSha** aafc9a340785d1d86de25962cf9f1f8bfef9fcbf

## Contents

derive_orderstat . . . . .	2
derive_transform . . . . .	3
imports . . . . .	3
plot.statderive . . . . .	4
print.statderive . . . . .	4
simulate_check . . . . .	5
<b>Index</b>	<b>6</b>

---

derive_orderstat	<i>Derive the distribution of an order statistic</i>
------------------	--

---

### Description

Derive the distribution of an order statistic

### Usage

```
derive_orderstat(dist, params, n, r)
```

### Arguments

dist	Character: base distribution name
params	List: distribution parameters
n	Integer: sample size
r	Integer: order (1 = min, n = max)

### Value

Object of class "statderive" with order statistic derivation

### Examples

```
# Minimum of 5 exponential(rate=2) samples
obj <- derive_orderstat(
  dist = "exponential",
  params = list(rate = 2),
  n = 5,
  r = 1
)
```

---

derive_transform	<i>Derive the distribution of a transformed random variable</i>
------------------	---

---

**Description**

Derive the distribution of a transformed random variable

**Usage**

```
derive_transform(dist, params, transform, support = NULL)
```

**Arguments**

dist	Character: base distribution name
params	List: distribution parameters
transform	Character: transformation (e.g., "x^2", "sqrt(x)", "1/x")
support	Optional vector c(lower, upper) for original support

**Value**

Object of class "statderive" with derivation steps

**Examples**

```
# Transform Y = X^2 where X ~ Uniform(0,1)
obj <- derive_transform(
  dist = "uniform",
  params = list(min = 0, max = 1),
  transform = "x^2"
)
```

---

imports	<i>Imported functions</i>
---------	---------------------------

---

**Description**

Internal imports required for package checks.

plot.statderive      *Plot method for statderive objects*

---

**Description**

Plot method for statderive objects

**Usage**

```
## S3 method for class 'statderive'  
plot(x, y = NULL, ...)
```

**Arguments**

x	Object of class "statderive"
y	Unused (required for S3 plot method)
...	Additional arguments

**Value**

Invisibly returns the input object x (of class statderive). This function is called primarily for its side effect of producing plots that compare simulated data with theoretical probability density functions.

---

print.statderive      *Print method for statderive objects*

---

**Description**

Print method for statderive objects

**Usage**

```
## S3 method for class 'statderive'  
print(x, ...)
```

**Arguments**

x	Object of class "statderive"
...	Additional arguments

**Value**

Invisibly returns the input object x (of class statderive). This function is called primarily for its side effect of printing a formatted summary of the statistical derivation and optional simulation results.

---

simulate_check	<i>Run Monte Carlo simulation and compare with theory</i>
----------------	---

---

**Description**

Run Monte Carlo simulation and compare with theory

**Usage**

```
simulate_check(object, n_sim = 10000, seed = 123)
```

**Arguments**

object	Object of class "statderive" from <code>derive_transform</code> or <code>derive_orderstat</code>
n_sim	Integer: number of simulations (default 10000)
seed	Integer: random seed for reproducibility

**Value**

Updated object with simulation results added

**Examples**

```
obj <- derive_transform("uniform", list(min = 0, max = 1), "x^2")  
obj_sim <- simulate_check(obj, n_sim = 5000)
```

# Index

`derive_orderstat`, 2

`derive_transform`, 3

`imports`, 3

`plot.statderive`, 4

`print.statderive`, 4

`simulate_check`, 5