

# Package ‘npdsim’

May 9, 2026

**Type** Package

**Title** Simulate Demand and Attributes for New Products

**Version** 1.0.0

**Description** Simulate demand and attributes for ready to launch new products during their life cycle, or during their introduction and growth phases. You provide the number of products, attributes, time periods and/or other parameters and 'npdsim' can simulate for you the demand for each product during the considered time periods, and the attributes of each product. The simulation for the demand is based on the idea that each product has a shape and a level, where the level is the cumulative demand over the considered time periods, and the shape is the normalized demand across those time periods.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://github.com/mohammedhichame/npdsim>

**BugReports** <https://github.com/mohammedhichame/npdsim/issues>

**RoxygenNote** 7.3.2

**Imports** dplyr (>= 1.1.4), tidyr (>= 1.3.1)

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), ggplot2

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Mohammed Hichame BENBITOUR [aut, cre, cph]

**Maintainer** Mohammed Hichame BENBITOUR <res.mhichame@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-04-09 10:20:02 UTC

## Contents

attribute_sim_dep . . . . .	2
attribute_sim_ind . . . . .	3
demand_sim . . . . .	4
npdsim_bass . . . . .	5
npd_data_sim . . . . .	5
shape_sim . . . . .	7
shape_sim_bass . . . . .	8
shape_sim_ig . . . . .	8
shape_sim_random . . . . .	9
shape_sim_trapezoid . . . . .	9
shape_sim_triangle . . . . .	10
<b>Index</b>	<b>11</b>

---

attribute_sim_dep	<i>Simulate the Attributes with the Assumption of Dependent Attributes</i>
-------------------	--

---

### Description

Simulate the attributes for each product with the assumption that some of the attributes related to shapes are also related to some of the attributes of levels. We mean by dependence the fact that some attributes of a product are related at the same time to its shape and level.

### Usage

```
attribute_sim_dep(
  product_shapes_and_levels,
  attributes_number,
  shape_attributes_number,
  level_attributes_number
)
```

### Arguments

product_shapes_and_levels	A numeric dataframe of three columns: product_id, assigned_shape and assigned_level
attributes_number	The number of attributes
shape_attributes_number	The number of attributes assigned to shape
level_attributes_number	The number of attributes assigned to level

**Value**

A numeric dataframe of the following columns: product\_id, assigned\_shape, assigned\_level and attributes (as columns)

**Examples**

```
attribute_sim_dep(product_shapes_and_levels=
data.frame(product_id=1:4,assigned_shape=c(1,1,2,2),
assigned_level=c(5,3,3,3)),
attributes_number=15,
shape_attributes_number=7,
level_attributes_number=4)
```

---

attribute_sim_ind	<i>Simulate the Attributes with the Assumption of Independent Attributes</i>
-------------------	--

---

**Description**

Simulate the attributes for each product with the assumption that the attributes of shapes are independent of the attributes of levels. We mean by independence the fact that each attribute is related to one of the following: shape, level or nothing.

**Usage**

```
attribute_sim_ind(
  product_shapes_and_levels,
  attributes_number,
  shape_attributes_number,
  level_attributes_number
)
```

**Arguments**

```
product_shapes_and_levels
  A numeric dataframe of three columns: product_id, assigned_shape and assigned_level
attributes_number
  The number of attributes
shape_attributes_number
  The number of attributes assigned to shape
level_attributes_number
  The number of attributes assigned to level
```

**Value**

A numeric dataframe of the following columns: product\_id, assigned\_shape, assigned\_level and attributes (as columns)

**Examples**

```
attribute_sim_ind(product_shapes_and_levels=
data.frame(product_id=1:4,assigned_shape=c(1,1,2,2),
assigned_level=c(5,3,3,3)),
attributes_number=15,
shape_attributes_number=7,
level_attributes_number=4)
```

demand\_sim

*Simulate the demand for new products***Description**

Simulate the demand for new products over their life cycle by specifying their shape type.

**Usage**

```
demand_sim(
  products_number,
  periods_number,
  shape_number,
  shape_type = "random",
  level_number,
  level_range = 1000:10000,
  noise_cv = 0.05
)
```

**Arguments**

products_number	Number of products
periods_number	Number of periods of the introduction and growth phases
shape_number	Number of generic shapes
shape_type	Type of shape to generate. It can take the values: "triangle", "trapezoid", "bass", "random" and "intro & growth". The type "random" picks one of the types "triangle", "trapezoid", "bass" randomly for each product. The type "intro & growth" is used for the shapes of the introduction and growth phases.
level_number	Number of generic levels
level_range	Range of values from which the level is sampled
noise_cv	The coefficient of variation of the noise added to the simulated sales

**Value**

A data frame that contains the following columns: product\_id, shape and assigned\_shape, level and assigned\_level, demand\_wn (demand without noise, not rounded), noise and demand. demand is the rounded value of the Max between (demand\_wn+noise) and 0

**Examples**

```
demand_sim(products_number=100,periods_number=20,shape_number=5, level_number=20)
```

```
demand_sim(products_number=100,periods_number=20,shape_number=5, shape_type="bass", level_number=20,  
level_range=1000:10000,noise_cv=0.05)
```

---

npdsim_bass	<i>Calculate the Bass probability</i>
-------------	---------------------------------------

---

**Description**

Calculate the Bass probability density function of purchase  $f(t)$

**Usage**

```
npdsim_bass(p_param, q_param, t)
```

**Arguments**

p_param	Coefficient of innovation
q_param	Coefficient of imitation
t	A numeric vector of time periods

**Value**

A numeric vector of the probability density function of purchase at time  $t$ ,  $f(t)$

**Examples**

```
npdsim_bass(p_param=0.01,q_param=0.2, t=1:20)
```

---

npd_data_sim	<i>Simulate the demand and attributes for new products</i>
--------------	--

---

**Description**

Simulate the demand and attributes for new products during their life cycle by specifying their life cycle type of shape and providing information about their attributes.

**Usage**

```
npd_data_sim(
  products_number,
  periods_number,
  shape_number,
  shape_type = "random",
  level_number,
  level_range = 1000:10000,
  noise_cv = 0.05,
  attribute_type = "ind",
  attributes_number = 10,
  shape_attributes_number = 5,
  level_attributes_number = 3
)
```

**Arguments**

products_number	Number of products
periods_number	Number of periods of the introduction and growth phases
shape_number	Number of generic shapes
shape_type	Type of shape to generate. It can take the values: "triangle", "trapezoid", "bass", "random" and "intro & growth". The type "random" picks one of the types "triangle", "trapezoid", "bass" randomly for each product. The type "intro & growth" is used for the shapes of the introduction and growth phases.
level_number	Number of generic levels
level_range	Range of values from which the level is sampled
noise_cv	The coefficient of variation of the noise added to the simulated sales
attribute_type	Type of relationship between attributes and shape and level. There can be independent attributes or dependent attributes. attribute_type takes one of the two values: "dep" and "ind". Check 'attribute_sim_dep' and 'attribute_sim_dep'.
attributes_number	The number of attributes
shape_attributes_number	The number of attributes assigned to shape
level_attributes_number	The number of attributes assigned to level

**Value**

A date frame that contains the following columns: product\_id, demand and attributes.

**Examples**

```
npd_data_sim(products_number=100,
             periods_number=30,
```

```
shape_number=5,
level_number=20)

npd_data_sim(products_number=100,
periods_number=20,
shape_number=5,
shape_type="bass",
level_number=20,
level_range=1000:10000,
noise_cv=0.05,
attribute_type="ind",
attributes_number=15,
shape_attributes_number=7,
level_attributes_number=5)
```

---

shape_sim	<i>Generate the shape of demand</i>
-----------	-------------------------------------

---

### Description

Generate the shape of demand for new products by specifying their life cycle shape and the length of their life cycle

### Usage

```
shape_sim(periods_number, shape_number, shape_type = "random")
```

### Arguments

periods_number	Number of time periods of the products life cycle
shape_number	Number of generic shapes
shape_type	Type of shape to generate. It can take the values: "triangle", "trapezoid", "bass", "random" and "intro & growth". The type "random" picks one of the types "triangle", "trapezoid", "bass" randomly for each product. The type "intro & growth" is used for the shapes of the introduction and growth phases.

### Value

A numeric dataframe of three columns: time, shape and assigned\_shape

### Examples

```
shape_sim(periods_number=20, shape_number=5)
shape_sim(periods_number=20, shape_number=5, shape_type="trapezoid")
```

---

shape\_sim\_bass      *Generate generic Bass shapes*

---

**Description**

Generate generic Bass shapes for the demand of new products during their life cycle

**Usage**

```
shape_sim_bass( periods_number, shape_number )
```

**Arguments**

periods\_number    Number of time periods of the products life cycle  
 shape\_number      Number of generic shapes

**Value**

A numeric dataframe of three columns: time, shape and assigned\_shape

**Examples**

```
shape_sim_bass( periods_number=20, shape_number=5 )
```

---

shape\_sim\_ig      *Generate generic shapes for the introduction and growth phases*

---

**Description**

Generate piece-wise linear (4 segments) generic shapes for the introduction and growth phases

**Usage**

```
shape_sim_ig( periods_number, shape_number )
```

**Arguments**

periods\_number    Number of periods of the introduction and growth phases  
 shape\_number      Number of generic shapes

**Value**

A numeric dataframe of three columns: time, shape and assigned\_shape

**Examples**

```
shape_sim_ig( periods_number=20, shape_number=5 )
```

---

shape_sim_random	<i>Generate random (Bass, Trapezoidal or Triangular) shapes</i>
------------------	---

---

**Description**

Generate random (Bass, Trapezoidal or Triangular) shapes for the demand of new products during their life cycle

**Usage**

```
shape_sim_random( periods_number, shape_number )
```

**Arguments**

periods\_number    Number of time periods of the products life cycle  
shape\_number      Number of generic shapes

**Value**

A numeric dataframe of three columns: time, shape and assigned\_shape

**Examples**

```
shape_sim_random( periods_number=20, shape_number=5 )
```

---

shape_sim_trapezoid	<i>Generate trapezoidal shapes</i>
---------------------	------------------------------------

---

**Description**

Generate trapezoidal shapes for the demand of new products during their life cycle

**Usage**

```
shape_sim_trapezoid( periods_number, shape_number )
```

**Arguments**

periods\_number    Number of time periods of the products life cycle  
shape\_number      Number of generic shapes

**Value**

A numeric dataframe of three columns: time, shape and assigned\_shape

**Examples**

```
shape_sim_trapezoid( periods_number=20, shape_number=5 )
```

---

shape\_sim\_triangle     *Generate triangular shapes*

---

**Description**

Generate triangular shapes for the demand of new products during their life cycle

**Usage**

```
shape_sim_triangle( periods_number, shape_number )
```

**Arguments**

periods\_number    Number of time periods of the products life cycle  
shape\_number      Number of generic shapes

**Value**

A numeric dataframe of three columns: time, shape and assigned\_shape

**Examples**

```
shape_sim_triangle( periods_number=20, shape_number=5 )
```

# Index

`attribute_sim_dep`, [2](#)

`attribute_sim_ind`, [3](#)

`demand_sim`, [4](#)

`npd_data_sim`, [5](#)

`npdsim_bass`, [5](#)

`shape_sim`, [7](#)

`shape_sim_bass`, [8](#)

`shape_sim_ig`, [8](#)

`shape_sim_random`, [9](#)

`shape_sim_trapezoid`, [9](#)

`shape_sim_triangle`, [10](#)