

# Package: NB.MClust (via r-universe)

August 21, 2024

**Title** Negative Binomial Model-Based Clustering

**Version** 1.1.1

**Description** Model-based clustering of high-dimensional non-negative data that follow Generalized Negative Binomial distribution. All functions in this package applies to either continuous or integer data. Correlation between variables are allowed, while samples are assumed to be independent.

**Depends** R (>= 3.3.3)

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**Imports** MASS, utils

**NeedsCompilation** no

**Author** Qian Li [aut, cre]

**Maintainer** Qian Li <qian.li10000@gmail.com>

**Date/Publication** 2017-06-03 12:08:50 UTC

**Repository** <https://qianli10000.r-universe.dev>

**RemoteUrl** <https://github.com/cran/NB.MClust>

**RemoteRef** HEAD

**RemoteSha** 955e816c52a138121939f2a4acf26ab6a6f82ab1

## Contents

nb.density . . . . .	2
NB.MClust . . . . .	2
Simulated_Count . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

nb.density	<i>dnb, ldnb Functions</i>
------------	----------------------------

---

**Description**

These functions allow you to compute (log-)density of generalized Negative Binomial distribution.

**Usage**

```
ldnb(x, theta, mu)
```

```
dnb(x, theta, mu)
```

**Arguments**

x	A positive numeric scalar or vector. Decimals and integers are both allowed.
theta	Value of dispersion.
mu	Value of mean.

**Value**

dnb	Density of generalized Negative Binomial
ldnb	Log-density of generalized Negative Binomial

**Examples**

```
ldnb(x=10.4, theta=3.2, mu=5)
dnb(x=10.4, theta=3.2, mu=5)
```

---

NB.MClust	<i>NB.MClust Function</i>
-----------	---------------------------

---

**Description**

This function performs model-based clustering on positive integer or continuous data that follow Generalized Negative Binomial distribution.

**Usage**

```
NB.MClust(Count, K, ini.shift.mu = 0.01, ini.shift.theta = 0.01,
  tau0 = 10, rate = 0.9, bic = TRUE, iteration = 100)
```

**Arguments**

Count	Data matrix of discrete counts. This function groups rows of the data matrix.
K	Number of clusters or components specified. It can be a positive integer or a vector of positive integer.
ini.shift.mu	Initial value in EM algorithm for the shift between clusters in mean.
ini.shift.theta	Initial value in EM algorithm for the shift between clusters in dispersion.
tau0	Initial value of annealing rates in EM Algorithm. Default and suggested value is 10.
rate	Stochastic decreasing speed for annealing rate. Default and suggested value is 0.9
bic	Whether Bayesian Information should be computed when K is an integer. BIC is forced to be TRUE when K is a vector.
iteration	Maximum number of iterations in EM Algorithm, default at 50.

**Value**

parameters	Estimated parameters
\$prior	Prior probability that a sample belongs to each cluster
\$mu	Mean of each cluster
\$theta	Dispersion of each cluster
\$posterior	Posterior probability that a sample belongs to each cluster
cluster	Estimated cluster assignment
BIC	Value of Bayesian Information
K	Optional or estimated number of clusters, if input K is a vector

**Examples**

```
# Example:

data("Simulated_Count") # A 50x100 integer data frame.

m1=NB.MClust(Simulated_Count,K=2:5)
cluster=m1$cluster #Estimated cluster assignment
k_hat=m1$K #Estimated optimal K
```

---

Simulated_Count	<i>Data set for illustration: Simulated_Count</i>
-----------------	---

---

**Description**

Data set for illustration: Simulated\_Count

**Usage**

Simulated\_Count

**Format**

A simulated data frame with 50 rows (i.e. samples) and 100 columns (i.e. variables ). It can be viewed as simulated RNA-Seq integer counts of 100 genes for 50 patients.

# Index

\* **NB.MClust**

NB.MClust, [2](#)

\* **datasets**

Simulated\_Count, [4](#)

dnb (nb.density), [2](#)

ldnb (nb.density), [2](#)

nb.density, [2](#)

NB.MClust, [2](#)

Simulated\_Count, [4](#)