Package ‘dilutionrisk’

July 24, 2022

Type Package

Title Modelling and assessment of risk based on aerobic plate count (APC) on diluted testing.

Version 0.0.1

Authors Mayooran Thevaraja [aut, cre], Kondaswamy Govindaraju [aut], Mark Bebbington [aut]

URL https://github.com/Mayooran1987/dilutionrisk

BugReports https://github.com/Mayooran1987/dilutionrisk/issues

Description This package aims to develop for getting probability estimations and graphical displays in the study associated with Modelling and assessment of risk based on aerobic plate count (APC) on diluted testing.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Imports extraDistr, ggplot2, ggthemes, reshape2, stats

Suggests spelling, testthat, covr

RoxygenNote 7.2.1

Depends R (>= 4.0)

Maintainer Mayooran Thevaraja <mayooran@eng.jfn.ac.lk>

Language en-US

Config/testthat/edition 3

R topics documented:

`dilution_OC_curve_homogeneous`

Comparison based on OC curves for different dilution schemes when the diluted sample has homogeneous contaminants.

Description

`dilution_OC_curve_homogeneous` provides the operating characteristic(OC) curves when diluted sample has homogeneous contaminants.

Usage

```
dilution_OC_curve_homogeneous(c, lambda_low, lambda_high, a, b, pf, USL, n, n_sim)
```

Arguments

`c`	acceptance number
`lambda_low`	the lower value of the expected cell count (λ) for use in the graphical display's x-axis.
`lambda_high`	the upper value of the expected cell count (λ) for use in the graphical display's x-axis.
`a`	lower domain of the number of cell counts.
`b`	upper domain of the number of cell counts.
`pf`	plating factor (pf = 1/final dilution factor).
`USL`	upper specification limit.
`n`	number of samples which are used for inspection.
`n_sim`	number of simulations (large simulations provide more precise estimation).

Details

`dilution_OC_curve_homogeneous` provides OC curves for different dilution schemes when the diluted sample has homogeneous contaminants (this section will be updated later on).

Value

OC curves when the diluted sample has homogeneous contaminants.

Examples

```
c <- 0
lambda_low <- 0
lambda_high <- 5
a <- 0
b <- 300
pf <- 1000
USL <- c(1000, 2000)
n <- 5
n_sim <- 10000
dilution_OC_curve_homogeneous(c, lambda_low, lambda_high, a, b, pf, USL, n, n_sim)
```

`dilution_pd_curve_homogeneous`
comparison based on probability of detection curves for different dilution schemes when the diluted sample has homogeneous contaminants.

Description

`dilution_pd_curve_homogeneous` provides the probability of detection curves when the diluted sample has homogeneous contaminants.

Usage

```
dilution_pd_curve_homogeneous(lambda_low, lambda_high, a, b, pf, USL, n_sim)
```

Arguments

`lambda_low`	the lower value of the expected cell count (λ) for use in the graphical display's x-axis.
`lambda_high`	the upper value of the expected cell count (λ) for use in the graphical display's x-axis.
`a`	lower domain of the number of cell counts.
`b`	upper domain of the number of cell counts.
`pf`	plating factor (pf = 1/final dilution factor).
`USL`	upper specification limit.
`n_sim`	number of simulations (large simulations provide more precise estimation).

Details

`dilution_pd_curve_homogeneous` provides probability of detection curves for different dilution schemes when the diluted sample has homogeneous contaminants (this section will be updated later on).

Value

Probability of detection curves when the diluted sample has homogeneous contaminants.

Examples

```
lambda_low <- 0
lambda_high <- 10
a <- 0
b <- 300
pf <- 1000
USL <- c(1000, 2000)
n_sim <- 10000
dilution_pd_curve_homogeneous(lambda_low, lambda_high, a, b, pf, USL, n_sim)
```

---

prob_acceptance_homogeneous
Probability of acceptance estimation when diluted sample has homogeneous contaminations.

Description

prob_acceptance_homogeneous provides a probability of acceptance in the original sample when homogeneous contamination.

Usage

```
prob_acceptance_homogeneous(c, lambda, a, b, pf, USL, n, n_sim)
```

Arguments

c	acceptance number
lambda	the expected cell count (λ).
a	lower domain of the number of cell counts.
b	upper domain of the number of cell counts.
pf	plating factor (pf = 1/final dilution factor).
USL	upper specification limit.
n	number of samples which are used for inspection.
n_sim	number of simulations (large simulations provide more precise estimation).

Details

prob_detection_homogeneous provides a probability of acceptance when the diluted sample has homogeneous contaminants (this section will be updated later on).

Value

Probability of acceptance when the diluted sample has homogeneous contaminants.

Examples

```
c <- 0
lambda <- 10
a <- 0
b <- 300
pf <- 1000
USL <- 1000
n <- 5
n_sim <- 10000
prob_acceptance_homogeneous(c, lambda, a, b, pf, USL, n, n_sim)
```

prob_detection_homogeneous *Probability of detection estimation when diluted sample has homogeneous contaminations.*

Description

prob_detection_homogeneous provides a probability of detection in the original sample when homogeneous contamination.

Usage

```
prob_detection_homogeneous(lambda, a, b, pf, USL, n_sim)
```

Arguments

lambda	the expected cell count (λ).
a	lower domain of the number of cell counts.
b	upper domain of the number of cell counts.
pf	plating factor (pf = 1/final dilution factor).
USL	upper specification limit.
n_sim	number of simulations (large simulations provide more precise estimation).

Details

prob_detection_homogeneous provides a probability of detection when the diluted sample has homogeneous contaminants. We define the random variable X_i is the number of colonies on the i^{th} plate. In practice, the acceptance for countable numbers of colonies on a plate must be between 30 and 300. Therefore, we can utilise bounded distributions to model the number of colonies on a plate. Homogeneous case, we employed truncated Poisson distribution to model (this section will be updated later on).

Value

Probability of detection when the diluted sample has homogeneous contaminants.

Examples

```
lambda <- 20
a <- 0
b <- 300
pf <- 1000
USL <- 1000
n_sim <- 10000
prob_detection_homogeneous(lambda, a, b, pf, USL, n_sim)
```

Index