Package ‘Dodge’

February 19, 2015

Type       Package
Title      Functions for Acceptance Sampling Ideas originated by H.F. Dodge
Version    0.8
Date       2013-09-18
Author     Jonathan Godfrey and Raj Govindaraju
Maintainer  Jonathan Godfrey <a.j.godfrey@massey.ac.nz>
Description Various sampling plans are able to be compared using evaluations of their OC, AOQ, ATI etc.
License    GPL
LazyLoad   yes
Depends    R (>= 2.14.0)
NeedsCompilation no
Repository  CRAN
Date/Publication 2013-09-18 08:35:16

R topics documented:

  Dodge-package .................................................. 2
  ChainSamplingPlans ........................................... 2
  CurtASN ....................................................... 3
  DoubleSamplingPlans .......................................... 4
  LSP ............................................................ 5
  plot.AccSampPlan .............................................. 5
  print.AccSampPlan ............................................ 6
  SeqDesign ..................................................... 7
  Sequential ..................................................... 8
  SingleSamplingPlanDesign .................................... 9
  SingleSamplingPlans .......................................... 10
  VariableSamplingPlans ....................................... 11
  VSPDesign .................................................... 11

Index 13
Description

A number of sampling plans can be compared for their operating characteristics and other commonly used functions.

Details

<table>
<thead>
<tr>
<th>Package</th>
<th>Dodge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>0.8</td>
</tr>
<tr>
<td>Date:</td>
<td>2013-09-18</td>
</tr>
<tr>
<td>License:</td>
<td>GPL</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

Author(s)

Raj Govindaraju and Jonathan Godfrey
Maintainer: Jonathan Godfrey <a.j.godfrey@massey.ac.nz>

References

Dodge

Usage

ChainBinomial(N, n, i, p = seq(0, 0.2, 0.001), Plots = TRUE)

ChainPoisson(N, n, i, p = seq(0, 0.3, 0.001), Plots = TRUE)
Arguments

- \( N \) the lot size
- \( n \) the sample size
- \( i \) the number of preceding lots that are free from nonconforming units for the lot to be accepted
- \( p \) a vector of values for the possible fraction of product that is nonconforming
- \( \text{Plots} \) logical to request generation of the four plots

Value

A matrix containing the argument \( p \) as supplied and the calculated OC, ATI and ???

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

References


Examples

```r
require(Dodge)
ChainBinomial(1000, 20, 3)
ChainPoisson(1000, 20, 3)
```

Description

Computes the average sample number for a curtailed inspection plan for single sampling plans. Functionality is currently available for only the binomial distribution.

Usage

```r
CurtBinomial(n, Ac, p = seq(0, 0.5, 0.01), Plots = TRUE)
```

Arguments

- \( n \) the sample size (potential)
- \( Ac \) the acceptance number
- \( p \) a vector of values for the possible fraction of product that is nonconforming
- \( \text{Plots} \) logical to request generation of the four plots
Author(s)
Raj Govindaraju with minor editing by Jonathan Godfrey

Examples
CurtBinomial(20,1)

Double Sampling Plans for the binomial and Poisson distributions.

Usage
DSPlanBinomial(N, n1, n2, Ac1, Re1, Ac2, p = seq(0, 0.25, 0.005), Plots = TRUE)
DSPlanPoisson(N, n1, n2, Ac1, Re1, Ac2, p = seq(0, 0.25, 0.005), Plots = TRUE)

Arguments
N the lot size
n1 the sample size in the first stage of the plan
n2 the sample size in the second stage of the plan
Ac1 the first stage acceptance number
Re1 the first stage rejection number
Ac2 the second stage acceptance number
p a vector of values for the possible fraction of product that is nonconforming
Plots logical to request generation of the four plots

Author(s)
Raj Govindaraju with minor editing by Jonathan Godfrey

References

Examples
DSPlanBinomial(1000, 10, 10, 0, 2, 1)
DSPlanPoisson(1000, 10, 10, 0, 2, 1)
Description

The lot sensitive compliance sampling plans for given parameters.

Usage

LSP(N, LTPD, beta, p = seq(0, 0.3, 0.001), Plots = TRUE)

Arguments

N the lot size
LTPD the lot tolerance percent defective, also known as the limiting quality
beta consumer risk
p fraction nonconforming
Plots logical indicating if the four plots are required

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

References


Examples

LSP(1000, 0.04, 0.05)

plot.AccSampPlan plot methods for the Dodge package

Description

Creates plots for analysing the design of an acceptance sampling procedure.
Usage

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

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

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

Arguments

- `x`: an object of class AccSampPlan, CurtSampPlan, or SeqSampPlan
- `y`: ignored
- `...`: further arguments passed to or from other methods.

Details

At this stage the `plot.AccSampPlan` method only plots the Operating Characteristic (OC) curve, the Average (AOQ) and (ATI) against the proportion (p) of product that is nonconforming. It also plots the curtailed sample size or the average sample number (ASN) against p. Further development is still required.

Author(s)

Jonathan Godfrey with some assistance from Raj Govindaraju

Examples

```r
Plan1 = SSPlanBinomial(1000, 20, 1, Plots=FALSE)
plot(Plan1)
```

Description

`print.AccSampPlan` adds to the base R functionality for the `print()` command. The accompanying plot methods are more sophisticated.
Usage

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

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

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

Arguments

- `x`: an object of class AccSampPlan, CurtSampPlan, or SeqSampPlan
- `...`: further arguments passed to or from other methods.

Details

These methods print the most necessary elements of the corresponding objects.

Author(s)

Jonathan Godfrey

See Also

The corresponding plot method is far more interesting. See `plot.AccSampPlan` for example.

---

**SeqDesign**

*Create a sequential sampling plan*

Description

Selects the appropriate sequential sampling plan from the given inputs. The only distribution that has been used in functions thus far is the binomial, but further development is expected.

Usage

```r
SeqDesignBinomial(N, AQL, alpha, LQL, beta, Plots = TRUE)
```

Arguments

- `N`: the lot size, ignored for the design of the plan unless the underlying distribution is hypergeometric
- `AQL`: Acceptable quality level
- `alpha`: producer's risk
- `LQL`: Limiting quality level
- `beta`: consumers' risk
- `Plots`: logical stating if the sequential chart should be plotted
Author(s)

Raj Govindaraju and Jonathan Godfrey

Examples

#--- Should be DIRECTLY executable !! ----
#-- == Define data, use random,
#-- or do help(data=index) for the standard data sets.

Sequential

Attribute Sequential Sampling Plans

Description

Designs an attribute sequential sampling plan for given AQL, alpha, LQL, and beta. The user can request plots describing the performance of the plan.

Usage

SequentialBinomial(x, Plots = TRUE)

Arguments

x

an object of class SeqSampPlan, or at least having the same elements as one.

Plots

logical indicating if the four plots should be returned

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

Examples

PlanDesign=SeqDesignBinomial(AQL=0.01, alpha=0.05, LQL=0.04, beta=0.05, Plots=FALSE)
SequentialBinomial(PlanDesign)
Description

Design a single sampling plan for given AQL, alpha, LQL, and beta. Currently there are functions for the binomial and Poisson distributions.

Usage

SSPDesignBinomial(AQL, alpha, LQL, beta)
SSPDesignPoisson(AQL, alpha, LQL, beta)

Arguments

AQL  Acceptable quality level
alpha  producer’s risk
LQL  Limiting quality level
beta  consumers’ risk

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

References


Examples

SSPDesignBinomial(0.01, 0.05, 0.04, 0.05)
SSPDesignPoisson(0.01, 0.05, 0.04, 0.05)
Single Sampling Plans

Description

Single sampling plans for the binomial, hypergeometric and Poisson distributions.

Usage

SSPlanBinomial(N, n, Ac, p = seq(0, 0.3, 0.001), Plots = TRUE)
SSPlanHyper(N, n, Ac, p = seq(0, 0.3, 0.001), Plots = TRUE)
SSPlanPoisson(N, n, Ac, p = seq(0, 0.3, 0.001), Plots = TRUE)

Arguments

N            the lot size
n            the sample size
Ac           the acceptance number, being the maximum allowable number of nonconforming units or nonconformities
p            a vector of values for the possible fraction of product that is nonconforming
Plots        logical to request generation of the four plots

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

References


Examples

SSPlanBinomial(1000, 20,1)
SSPlanHyper(5000, 200,3)
SSPlanPoisson(1000, 20,1)
Variablesamplingplans

**Description**

Variable sampling plans for known and unknown sigma, evaluated for given parameters.

**Usage**

VSPKnown(N, n, k, Pa = seq(0, 1, 0.001), Plots = TRUE)

VSPUnknown(N, n, k, Pa = seq(0, 1, 0.001), Plots = TRUE)

**Arguments**

- **N** the lot size
- **n** the sample size
- **k** the acceptability constant
- **Pa** fraction nonconforming
- **Plots** logical indicating whether the four plots are required

**Author(s)**

Raj Govindaraju with minor editing by Jonathan Godfrey

**Examples**

VSPKnown(1000, 20, 1)
VSPUnknown(1000, 20, 1)

---

VariableSamplingPlan Design

**Description**

Design the variable sampling plan for given AQL, alpha, LQL, and beta.

**Usage**

VSPDesign(AQL, alpha, LQL, beta)
Arguments

- **AQL**: Acceptable quality level
- **alpha**: producer’s risk
- **LQL**: Limiting quality level
- **beta**: consumers’ risk

Author(s)

Raj Govindaraju with minor editing by Jonathan Godfrey

Examples

\[
\text{VSPDesign}(\text{AQL}=0.01, \text{alpha}=0.05, \text{LQL}=0.04, \text{beta}=0.05)
\]
Index

*Topic `textasciitildekwd1`
  ChainsamplingPlans, 2
  CurtASN, 3
  DoubleSamplingPlans, 4
  LSP, 5
  plot.AccSampPlan, 5
  print.AccSampPlan, 6
  SeqDesign, 7
  Sequential, 8
  SingleSamplingPlanDesign, 9
  SingleSamplingPlans, 10
  VariableSamplingPlans, 11
  VSPDesign, 11

*Topic `textasciitildekwd2`
  ChainsamplingPlans, 2
  CurtASN, 3
  DoubleSamplingPlans, 4
  LSP, 5
  plot.AccSampPlan, 5
  print.AccSampPlan, 6
  SeqDesign, 7
  Sequential, 8
  SingleSamplingPlanDesign, 9
  SingleSamplingPlans, 10
  VariableSamplingPlans, 11
  VSPDesign, 11

*Topic package
  Dodge-package, 2

  ChainBinomial (ChainsamplingPlans), 2
  ChainPoisson (ChainsamplingPlans), 2
  ChainsamplingPlans, 2
  CurtASN, 3
  CurtBinomial (CurtASN), 3

  Dodge (Dodge-package), 2
  Dodge-package, 2
  DoubleSamplingPlans, 4
  DSPPlanBinomial (DoubleSamplingPlans), 4
  DSPPlanPoisson (DoubleSamplingPlans), 4
  LSP, 5
  plot.AccSampPlan, 5, 7
  plot.CurtSampPlan (plot.AccSampPlan), 5
  plot.SeqSampPlan (plot.AccSampPlan), 5
  print.AccSampPlan, 6
  print.CurtSampPlan (print.AccSampPlan), 6
  print.SeqSampPlan (print.AccSampPlan), 6
  SeqDesign, 7
  SeqDesignBinomial (SeqDesign), 7
  Sequential, 8
  SequentialBinomial (Sequential), 8
  SingleSamplingPlanDesign, 9
  SingleSamplingPlans, 10
  SSPDesign (SingleSamplingPlanDesign), 9
  SSPDesignBinomial (SingleSamplingPlanDesign), 9
  SSPDesignPoisson (SingleSamplingPlanDesign), 9
  SSPlanBinomial (SingleSamplingPlans), 10
  SSPlanHyper (SingleSamplingPlans), 10
  SSPlanPoisson (SingleSamplingPlans), 10
  VariableSamplingPlans, 11
  VSPDesign, 11
  VSPKnown (VariableSamplingPlans), 11
  VSPUnknown (VariableSamplingPlans), 11