

Package ‘PH1XBAR’

November 14, 2023

Type Package

Title Phase I Shewhart X-Bar Chart

Version 0.11.1

Maintainer Yuhui Yao <yyao17@ua.edu>

Description The purpose of 'PH1XBAR' is to build a Phase I Shewhart control chart for the basic Shewhart, the variance components and the ARMA models in R for subgrouped and individual data. More details can be found: Yao and Chakraborti (2020) <[doi:10.1002/qre.2793](https://doi.org/10.1002/qre.2793)>, and Yao and Chakraborti (2021) <[doi:10.1080/0891](https://doi.org/10.1080/0891)>

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Imports forecast, mvtnorm, pracma

URL <https://github.com/bolus123/PH1XBAR>

NeedsCompilation no

Author Yuhui Yao [aut, cre],
Subha Chakraborti [ctb],
Tyler Thomas [ctb],
Jason Parton [ctb],
Xin Yang [ctb]

Repository CRAN

Date/Publication 2023-11-14 09:43:26 UTC

R topics documented:

PH1XBAR-package	2
bore_diameter_data	3
getCC.ARMA	3
getCC.XBAR	5
grinder_data	6
PH1ARMA	7

PH1XBAR	9
preston_data	10
snowfall_data	11

Index	12
--------------	-----------

PH1XBAR-package	<i>Phase I Shewhart X-bar Control Chart</i>
-----------------	---

Description

The utility of this package is in building a Shewhart-type control chart based on new methods for subgrouped and individual data. The Phase I chart is based on the multivariate normal/t or ARMA process.

Details

Package: PH1XBAR
 Type: Package
 Date: 2023-10-20
 License: GPL (>= 3)

Function 'PH1XBAR' builds a Phase I Shewhart X-bar/individual control chart with a correct charting constant. Function 'getCC.XBAR' gets a charting constant for the data with a balanced one-way random effects model. Function 'PH1ARMA' builds a Phase I individual control chart with an ARMA model using a correct charting constant. Function 'getCC.ARMA' gets a charting constant for the data with an ARMA model.

Author(s)

Yuhui Yao, Subha Chakraborti, Tyler Thomas, Xin Yang, Jason Parton.
 Maintainer: Yuhui Yao <yyao17@crimson.ua.edu>

References

Champ, C.W., and Jones, L.A. (2004) Designing Phase I X-bar charts with small sample sizes. *Quality and Reliability Engineering International*. 20(5), 497-510
 Yao, Y., Hilton, C.W., and Chakraborti, S. (2017) Designing Phase I Shewhart X-bar charts: Extended tables and software. *Quality and Reliability Engineering International*. 33(8), 2667-2672.
 Yao, Y., and Chakraborti, S. (2021). Phase I monitoring of individual normal data: Design and implementation. *Quality Engineering*, 33(3), 443-456.
 Yao, Y., and Chakraborti, S. (2021). Phase I process monitoring: The case of the balanced one-way random effects model. *Quality and Reliability Engineering International*, 37(3), 1244-1265.

Examples

```
# Build a Phase I basic Shewhart control chart
data(grinder_data)
PH1XBAR(grinder_data, nsim=10)

# Build a Phase I individual control chart with an ARMA model
data( Preston_data)
PH1ARMA(Preston_data, nsim.process=10, nsim.coefs=10)
```

bore_diameter_data *Bore Diameter in Manufacturing Automotive Driver Gears (mm)*

Description

Source: Wooluru, Yerriswamy, D. R. Swamy, and P. Nagesh. "THE PROCESS CAPABILITY ANALYSIS-A TOOL FOR PROCESS PERFORMANCE MEASURES AND METRICS-A CASE STUDY." International Journal for Quality Research 8.3 (2014).

A dataset containing the diameter of the bore in manufacturing automotive driver gears. The variables are as follows:

Format

A data frame with 20 rows and 5 variables:

- X1** Diameter measurement at Position 1
- X2** Diameter measurement at Position 2
- X3** Diameter measurement at Position 3
- X4** Diameter measurement at Position 4
- X5** Diameter measurement at Position 5

Examples

```
data(bore_diameter_data)
```

getCC.ARMA *get Phase I corrected charting constant with an ARMA model*

Description

Obtain a corrected charting constant.

Usage

```

getCC.ARMA(
  fap0 = 0.1
  ,interval = c(1, 4)
  ,n = 50
  ,order = c(1, 0, 0)
  ,phi.vec = 0.5
  ,theta.vec = NULL
  ,case = 'U'
  ,method = 'Method 3'
  ,nsim.coefs = 100
  ,nsim.process = 1000
  ,burn.in = 50
  ,sim.type = 'Matrix'
  ,logliktol = 1e-2
  ,verbose = FALSE
)

```

Arguments

fap0	nominal false Alarm Probabilty in Phase 1
interval	searching range of charting constants for the exact method
n	number of observations
order	order for ARMA model
phi.vec	given vectors of autoregressive parameters for ARMA models
theta.vec	given vectors of moving-average parameters for ARMA models
case	known or unknown case. When case = 'U', the parameters are estimated
method	estimation method for the control chart. When method = 'Method 3' is maximum likelihood estimations plus method of moments. Other options are 'Method 1' which is pure MLE and 'Method 2' which is pure CSS.
nsim.coefs	number of simulation for coefficients. It is functional when double.sim = TRUE.
nsim.process	number of simulation for ARMA processes
burn.in	number of burn-ins. When burn.in = 0, the ECM gets involved. When burn.in is large enough, the ACM gets involved.
sim.type	type of simulation. When sim.type = 'Matrix', the simulation is generated using matrix computation. When sim.type = 'Recursive', the simulation is based on a recursion.
logliktol	convergence tolerance for the log likelihood
verbose	print diagnostic information about fap0 and the charting constant during the simulations for the exact method

Value

Object type double. The corrected charting constant.

Examples

```
set.seed(12345)

# Calculate the charting constant using fap0 of 0.05, and 50 observations
getCC.ARMA(fap0=0.05, n=50, nsim.coefs=10, nsim.process=10)
```

getCC.XBAR	<i>get Phase I corrected charting constant</i>
------------	--

Description

Obtain a corrected charting constant.

Usage

```
getCC.XBAR(
  fap0
  ,m
  ,var.est = c('S', 'MR')
  ,ub.cons = 1
  ,method = c('exact', 'BA')
  ,interval = c(1, 5)
  ,nsim = 10000
  ,nu = m - 1
  ,lambda = 1
  ,verbose = FALSE
)
```

```
getCC(
  fap0
  ,m
  ,var.est = c('S', 'MR')
  ,ub.cons = 1
  ,method = c('exact', 'BA')
  ,interval = c(1, 5)
  ,nsim = 10000
  ,nu = m - 1
  ,lambda = 1
  ,verbose = FALSE
)
```

Arguments

fap0	nominal false Alarm Probabilty in Phase 1
m	number of subgroups

var.est	'S' - use mean-square-based estimator, 'MR' - use moving-range-based estimator
ub.cons	unbiasing constant
method	'exact' - calculate results using the exact method, 'BA' - calculate results using the Bonfferoni approximation
interval	searching range of charting constants for the exact method
nsim	number of simulation for the exact method
nu	degrees of freedom for the Bonfferoni approximation
lambda	constant for the Bonfferoni approximation
verbose	print diagnostic information about fap0 and the charting constant during the simulations for the exact method

Value

Object type double. The corrected charting constant.

Examples

```
set.seed(12345)

# Calculate the charting constant using 10 simulations and mean-square-based estimator
getCC.XBAR(fap0=0.05, m=20, nsim=10, var.est='S', verbose = TRUE)

# Calculate the charting constant using 10 simulations and moving-range-based estimator
getCC.XBAR(fap0=0.05, m=20, nsim=10, var.est='MR', verbose = TRUE)
```

grinder_data

Thickness measurement of silicon wafer

Description

A dataset containing the thickness in micrometer at different positions in a grinder. The variables are as follows:

Format

A data frame with 30 rows and 5 variables:

- pos1** Thickness measurement at Position 1
- pos2** Thickness measurement at Position 2
- pos3** Thickness measurement at Position 3
- pos4** Thickness measurement at Position 4
- pos5** Thickness measurement at Position 5

Examples

```
data(grinder_data)
```

PH1ARMA	<i>Build Phase I individual control chart with an ARMA model using a corrected charting constant</i>
---------	--

Description

Build a Phase I individual control chart for the ARMA models. The charting constant is corrected by this approach.

Usage

```
PH1ARMA(
X
,cc = NULL
,fap0 = 0.1
,order = NULL
,plot.option = TRUE
,interval = c(1, 4)
,case = 'U'
,method = 'Method 3'
,nsim.coefs = 100
,nsim.process = 1000
,burn.in = 50
,sim.type = 'Matrix'
,logliktol = 1e-2
,verbose = FALSE
,max.p=1
,max.q=0
,max.d=0
,standardize=TRUE
)
```

Arguments

X	input and it must be a vector
cc	nominal Phase I charting constant. If this is given, the function will not re-compute the charting constant.
fap0	nominal false Alarm Probabilty in Phase I
order	order for ARMA model
plot.option	TRUE - draw a plot for the process; FALSE - Not draw a plot for the process
interval	searching range of charting constants for the exact method
case	known or unknown case. When case = 'U', the parameters are estimated

method	estimation method for the control chart. When method = 'Method 3' is maximum likelihood estimations plus method of moments. Other options are 'Method 1' which is pure MLE and 'Method 2' which is pure CSS.
nsim.coefs	number of simulation for coefficients.
nsim.process	number of simulation for ARMA processes
burn.in	number of burn-ins. When burn.in = 0, the ECM gets involved. When burn.in is large enough, the ACM gets involved.
sim.type	type of simulation. When sim.type = 'Matrix', the simulation is generated using matrix computation. When sim.type = 'Recursive', the simulation is based on a recursion.
logliktol	convergence tolerance for the log likelihood
verbose	print diagnostic information about fap0 and the charting constant during the simulations for the exact method
max.p	max.p for auto.arima
max.q	max.q for auto.arima
max.d	max.d for auto.arima
standardize	Output z-scores instead of raw data

Value

Object of type list. A named list with the following objects:

CL	Object type double - central line
gamma	Object type double - process variance estimate
cc	Object type double - charting constant
order	Object type integer - order for ARMA model
phi.vec	Object type integer - values of autoregressors
theta.vec	Object type integer - values of moving averages
LCL	Object type double - lower charting limit
UCL	Object type double - upper charting limit
CS	Object type double - charting statistic

Examples

```
# load the data in the package as an example
data(preston_data)

# set number of simulations
nsim.process <- 10
nsim.coefs <- 10

# An example using the default setting whose fap0 = 0.1
PH1ARMA(preston_data, nsim.process = nsim.process, nsim.coefs = nsim.coefs)
```

```
# When users get an error message about the size of matrix,
# the function needs to use the alternative simulation type as follows
PH1ARMA(preston_data, fap0 = 0.05,
        nsim.process = nsim.process, nsim.coefs = nsim.coefs, sim.type = 'Recursive')
```

PH1XBAR

*Build Phase I X-bar control chart with a corrected charting constant***Description**

Build a Phase I Shewhart control chart for the variance components model if the data are subgrouped or for the basic Shewhart model if the data are individual. The charting constant is corrected by this approach.

Usage

```
PH1XBAR(
  X
  ,cc = NULL
  ,fap0 = 0.1
  ,var.est = c('S', 'MR')
  ,ub.option = TRUE
  ,method = c('exact', 'BA')
  ,plot.option = TRUE
  ,interval = c(1, 5)
  ,nsim = 10000
  ,verbose = FALSE
)
```

Arguments

X	input and it must be a matrix
cc	nominal Phase I charting constant. If this is given, the function will not recompute the charting constant.
fap0	nominal false Alarm Probability in Phase 1
var.est	'S' - use mean-square-based estimator, 'MR' - use moving-range-based estimator
ub.option	TRUE - the standard deviation estimator corrected by a unbiasing constant. For MS, it is c4 and for MR, it is d2. FALSE - no unbiasing constant
method	'exact' - calculate results using the exact method, 'BA' - calculate results using the Bonfferoni approximation
plot.option	TRUE - draw a plot for the process; FALSE - Not draw a plot for the process
interval	searching range of charting constants for the exact method
nsim	number of simulation for the exact method
verbose	print diagnostic information about fap0 and the charting constant during the simulations for the exact method

Value

Object of type list. A named list with the following objects:

CL	Object type double - central line
var.est	Object type double - variance estimate
ub.cons	Object type double - unbiasing constant
cc	Object type double - charting constant
m	Object type integer - number of observations
nu	Object type integer - degrees of freedom
lambda	Object type integer - chi-squared unbiasing constant
LCL	Object type double - lower charting limit
UCL	Object type double - upper charting limit
CS	Object type double - charting statistic

Examples

```
set.seed(12345)

# load the data in the package as an example
data(grinder_data)

# An example using a false alarm probability of 0.05, and 10 simulations
PH1XBAR(grinder_data, fap0 = 0.05, nsim=10, verbose=TRUE)
```

preston_data

Prescription fentanyl consumption in Preston county, WV

Description

A dataset containing prescription fentanyl consumption in Preston county, WV, measured using MME percapita. This is a subset from Rich et al. <doi: 10.21105/joss.02450>

Examples

```
data(preston_data)
```

`snowfall_data`*Snowfall in inches in Minneapolis/St. Paul, MN*

Description

A dataset containing snowfalls measured in inches in Minneapolis/St. Paul, MN. This is a subset of that on the website of Minnesota Department of Natural Resources(https://www.dnr.state.mn.us/climate/twin_cities/snowfall).

Examples

```
data(snowfall_data)
```

Index

* datasets

- bore_diameter_data, 3
- grinder_data, 6

bore_diameter_data, 3

getCC (getCC.XBAR), 5

getCC.ARMA, 3

getCC.XBAR, 5

grinder_data, 6

PH1ARMA, 7

PH1XBAR, 9

PH1XBAR-package, 2

preston_data, 10

snowfall_data, 11