Package ‘SixSigma’

August 20, 2021

Type Package
Title Six Sigma Tools for Quality Control and Improvement
Version 0.10.3
Encoding UTF-8
Maintainer Emilio L. Cano <emilio@lcano.com>
Description Functions and utilities to perform Statistical Analyses in the Six Sigma way. Through the DMAIC cycle (Define, Measure, Analyze, Improve, Control), you can manage several Quality Management studies: Gage R&R, Capability Analysis, Control Charts, Loss Function Analysis, etc. Data frames used in the books “Six Sigma with R” [ISBN 978-1-4614-3652-2] and “Quality Control with R” [ISBN 978-3-319-24046-6], are also included in the package.

URL https://www.sixsigmawithr.com
License GPL (>= 2)
Depends R (>= 3.5.0)
Imports grDevices, stats, graphics, lattice, ggplot2, reshape2, nortest, e1071, scales, testthat, xtable
LazyLoad yes
LazyData yes
RoxygenNote 7.1.1
Config/testthat/edition 3
NeedsCompilation no
Author Emilio L. Cano [aut, cre] (<https://orcid.org/0000-0002-6101-9755>), Javier M. Moguerza [aut], Mariano Prieto [aut], Andrés Redchuk [aut], Manuel Alfaro [ctb]
Repository CRAN
Date/Publication 2021-08-20 16:10:02 UTC
**R topics documented:**

climProfiles ......................................................... 3
outProfiles .......................................................... 4
plotControlProfiles .................................................... 5
plotProfiles .......................................................... 6
SixSigma ............................................................. 7
smoothProfiles ........................................................ 8
ss.ca.yield ........................................................... 9
ss.ca.z ............................................................... 10
ss.cc ................................................................. 11
ss.cc.constants ....................................................... 13
ss.ceeDiag ........................................................... 14
ss.ci ................................................................. 15
ss.data.batteries ....................................................... 17
ss.data.bills .......................................................... 18
ss.data.bolts .......................................................... 19
ss.data.ca ............................................................ 20
ss.data.density ......................................................... 21
ss.data.doe1 .......................................................... 22
ss.data.doe2 .......................................................... 23
ss.data.pastries ......................................................... 24
ss.data.pb1 ............................................................ 25
ss.data.pb2 ............................................................ 26
ss.data.pb3 ............................................................ 27
ss.data.pb4 ............................................................ 28
ss.data.pc ............................................................. 29
ss.data.pc.big ......................................................... 30
ss.data.pc.r ........................................................... 31
ss.data.rr ............................................................. 32
ss.data.strings ......................................................... 33
ss.data.thickness ...................................................... 34
ss.data.thickness2 .................................................... 35
ss.data.wbx ........................................................... 36
ss.data.wby ........................................................... 37
ss.heli ............................................................... 38
ss.lfa ............................................................... 39
ss.pMap ............................................................. 40
ss.rr ............................................................... 41
ss.study.ca .......................................................... 43

Index ................................................................. 48
climProfiles

Compute profiles limits

Description

Function to compute prototype profile and confidence bands for a set of profiles (Phase I)

Usage

climProfiles(
  profiles,
  x = 1:nrow(profiles),
  smoothprof = FALSE,
  smoothlim = FALSE,
  alpha = 0.01
)

Arguments

profiles Matrix with profiles in columns
x Vector for the independent variable
smoothprof regularize profiles? [FALSE]
smoothlim regularize confidence bands? [FALSE]
alpha limit for control limits [0.01]

Value

a matrix with three profiles: prototype and confidence bands

Author(s)

Javier M. Moguerza and Emilio L. Cano

References


Examples

wby.phase1 <- ss.data.wby[, 1:35]
wby.limits <- climProfiles(profiles = wby.phase1,
  x = ss.data.wbx,
  smoothprof = FALSE,
  smoothlim = FALSE)
plotProfiles(profiles = wby.phase1,
  x = ss.data.wbx,
  cLimits = wby.limits)
**outProfiles**

*Get out-of-control profiles*

**Description**

Returns a list with information about the out-of-control profiles given a set of profiles and some control limits

**Usage**

```r
outProfiles(profiles, x = 1:nrow(profiles), cLimits, tol = 0.5)
```

**Arguments**

- `profiles` Matrix of profiles
- `x` Vector with the independent variable
- `cLimits` Matrix with the prototype and confidence bands profiles
- `tol` Tolerance (%)

**Value**

a list with the following elements:

- `labOut` labels of the out-of-control profiles
- `idOut` ids of the out-of-control profiles
- `pOut` proportion of times the profile values are out of the limits

**References**


**Examples**

```r
wby.phase1 <- ss.data.wby[, 1:35]
wby.limits <- climProfiles(profiles = wby.phase1, 
                          x = ss.data.wbx, 
                          smoothprof = TRUE, 
                          smoothlim = TRUE)
wby.phase2 <- ss.data.wby[, 36:50]
wby.out.phase2 <- outProfiles(profiles = wby.phase2, 
                               x = ss.data.wbx, 
                               cLimits = wby.limits, 
                               tol = 0.8)
wby.out.phase2
plotProfiles(wby.phase2, 
            x = ss.data.wbx, 
```
plotControlProfiles

climits = wb.limits,
outControl = wb.out.phase2$idOut,
onlyout = TRUE)

plotControlProfiles  Profiles control plot

Description

Plots the proportion of times that each profile remains out of the confidence bands

Usage

plotControlProfiles(pOut, tol = 0.5)

Arguments

pOut  identifiers of profiles out of control
tol  tolerance for the proportion of times the value of the profile is out of control

Value

There is only graphical output

Author(s)

Javier M. Moguerza and Emilio L. Cano

References


Examples

wby.phase1 <- ss.data.wby[, 1:35]
wby.limits <- climProfiles(profiles = wby.phase1,
x = ss.data.wbx,
smoothprof = TRUE,
smoothlim = TRUE)
wby.phase2 <- ss.data.wby[, 36:50]
wby.out.phase2 <- outProfiles(profiles = wby.phase2,
x = ss.data.wbx,
cLimits = wb.limits,
tol = 0.8)
plotControlProfiles(wby.out.phase2$pOut, tol = 0.8)
**plotProfiles**  

*Plot Profiles*

**Description**

Plot profiles and optionally control limits

**Usage**

```r
plotProfiles(
  profiles,
  x = 1:nrow(profiles),
  cLimits = NULL,
  outControl = NULL,
  onlyout = FALSE
)
```

**Arguments**

- `profiles`: matrix with profiles in columns
- `x`: vector with the independent variable
- `cLimits`: matrix with three profiles: prototype and confidence bands (limits)
- `outControl`: identifiers of out-of-control profiles
- `onlyout`: plot only out-of-control profiles? [FALSE]

**Value**

Only graphical output with the profiles

**Author(s)**

Javier M. Moguerza and Emilio L. Cano

**References**


**Examples**

```r
plotProfiles(profiles = ss.data.wby,
             x = ss.data.wbx)
```
Description

Six Sigma Tools for Quality and Process Improvement

Details

This package contains functions and utilities to perform Statistical Analyses in the Six Sigma way. Through the DMAIC cycle (Define, Measure, Analyze, Improve, Control), you can manage several Quality Management studies: Gage R&R, Capability Analysis, Control Charts, Loss Function Analysis, etc. Data frames used in "Six Sigma with R" (Springer, 2012) are also included in the package. Use the package to perform Six Sigma Methodology tasks, throughout its breakthrough strategy: Define, Measure, Analyze, Improve, Control (DMAIC)
Define: Process Map (ss.pMap), Cause and effect Diagram (ss.ceDiag);
Measure: Gage R&R study (ss.rr); Capability Analysis (ss.study.ca); Loss Function Analysis (ss.lfa)
Analyze: Confidence Intervals (ss.ci)
Control: Moving Average Control Chart
Soon: further functions

Note

The current version includes Loss Function Analysis, Gage R&R Study, confidence intervals, Process Map and Cause-and-Effect diagram. We plan to regularly upload updated versions, with new functions and improving those previously deployed. The subsequent versions will cover tools for the whole cycle:

- Define
- Measure
- Analyze
- Improve
- Control

Author(s)

Emilio L. Cano, Javier M. Moguerza, Mariano Prieto Corcoba and Andrés Redchuk;
Maintainer: Emilio L. Cano <emilio.lopez@urjc.es>

References

smoothProfiles


See Also

*ss.pMap, ss.rr, ss.ceDiag, ss.ci, ss.heli, ss.lfa*

---

smoothProfiles is a function that takes a set of profiles and regularises them by means of a SVM.

**Usage**

```r
smoothProfiles(
  profiles,
  x = 1:nrow(profiles),
  svm.c = NULL,
  svm.eps = NULL,
  svm.gamma = NULL,
  parsvm.unique = TRUE
)
```

**Arguments**

- `profiles` Matrix of y values, one column per profile
- `x` Vector of predictive variable values, common to all profiles
- `svm.c` SVM parameter (cost)
- `svm.eps` SVM parameter (epsilon)
- `svm.gamma` SVM parameter (gamma)
- `parsvm.unique` Same parameters for all profiles? (logical [TRUE])

**Value**

Regularized profiles
Note

The package e1071 is needed in order to be able to use this function. SVM Parameters can be vectors of the same length as number of profiles, or a single value for all of them.

Author(s)

Javier M. Moguerza and Emilio L. Cano

References


Examples

```r
wby.smooth <- smoothProfiles(profiles = ss.data.wby, x = ss.data.wbx)
plotProfiles(profiles = wby.smooth, x = ss.data.wbx)
```

---

**ss.ca.yield**

*Main calculations regarding The Voice of the Process in SixSigma: Yield, FTY, RTY, DPMO*

---

Description

Computes the Yield, First Time Yield, Rolled Throughput Yield and Defects per Million Opportunities of a process.

Usage

```r
ss.ca.yield(defects = 0, rework = 0, opportunities = 1)
```

Arguments

- **defects**
  - A vector with the number of defects in each product/batch, ...
- **rework**
  - A vector with the number of items/parts reworked
- **opportunities**
  - A numeric value with the size or length of the product/batch

Details

The three arguments must have the same length.
Value

- **Yield**: Number of good stuff / Total items
- **FTY**: (Total - scrap - rework) / Total
- **RTY**: \( \text{prod}(FTY) \)
- **DPMO**: Defects per Million Opportunities

Author(s)

Emilio L. Cano

References


Examples

```r
ss.ca.yield(c(3,5,12),c(1,2,4),1915)
```

---

**ss.ca.z** *Capability Indices*

Description

Compute the Capability Indices of a process, Z (Sigma Score), \( C_p \) and \( C_{pk} \).

Usage

```r
ss.ca.cp(x, LSL = NA, USL = NA, LT = FALSE, f.na.rm = TRUE, ci = FALSE, alpha = 0.05)
```

```r
ss.ca.cpk(x, LSL = NA, USL = NA, LT = FALSE, f.na.rm = TRUE, ci = FALSE, alpha = 0.05)
```

```r
ss.ca.z(x, LSL = NA, USL = NA, LT = FALSE, f.na.rm = TRUE)
```

Arguments

- **x**: A vector with the data of the process performance
- **LSL**: Lower Specification Limit
- **USL**: Upper Specification Limit
- **LT**: Long Term data (TRUE/FALSE). Not used for the moment
- **f.na.rm**: Remove NA data (TRUE/FALSE)
- **ci**: If TRUE computes a Confidence Interval
- **alpha**: Type I error (\( \alpha \)) for the Confidence Interval
Control Charts

Description
Plot control charts

Usage
```r
ss.cc(type, data, cdata, CTQ = names(data)[1], groups, climits, nsigmas = 3)
```

Arguments
- **type**: Type of chart (see details)
- **data**: data.frame with the process data.
- **cdata**: Vector with the controlled process data to compute limits.
- **CTQ**: Name of the column in the data.frame containing the CTQ.
- **groups**: Name of the column in the data.frame containing the groups.
- **climits**: Limits of the controlled process. It should contain three ordered values: lower limit, center line and upper limit.
- **nsigmas**: Number of sigmas to compute the limits from the center line (default is 3)
Details

If control limits are provided, cdata is dismissed and a message is shown. If there are no control limits nor controlled data, the limits are computed using data. Supported types of control charts:

- mrMoving Range

Value

A plot with the control chart, and a list with the following elements:

- LCL Lower Control Limit
- CL Center Line
- UCL Upper Control Limit
- phase II when cdata or climits are provided. I otherwise.
- out Out of control points

Note

We have created this function since the qAnalyst package has been removed from CRAN, and it was used in the “Six Sigma with R” book to plot moving average control charts.

Author(s)

EL Cano

References


See Also

- **ss.cc.constants**

Examples

```r
ss.cc("mr", ss.data.pb1, CTQ = "pb.humidity")
testout <- ss.data.pb1
testout[31,] <- list(31,17)
ss.cc("mr", testout, CTQ = "pb.humidity")
```
ss.cc.constants

**Functions to find out constants of the relative range distribution.**

**Description**

These functions compute the constants d2, d3 and c4 to get estimators of the standard deviation to set control limits.

**Usage**

```
ss.cc.getd2(n = NA)
ss.cc.getd3(n = NA)
ss.cc.getc4(n = NA)
```

**Arguments**

- `n`  
  Sample size

**Value**

A numeric value for the constant.

**Author(s)**

EL Cano

**References**


**See Also**

ss.cc

**Examples**

```
ss.cc.getd2(20)
ss.cc.getd3(20)
ss.cc.getc4(20)
```
**ss.ceDiag**  
*Cause and Effect Diagram*

**Description**
Represents a Cause and Effect Diagram by cause group.

**Usage**
```
ss.ceDiag(
  effect,
  causes.gr,
  causes,
  main = "Six Sigma Cause-and-effect Diagram",
  sub,
  ss.col = c("#666666", "#BBBBBB", "#CCCCCC", "#DDDDDD", "#EEEEEE")
)
```

**Arguments**
- **effect**  
  A short character string that represents the effect we want to analyse.
- **causes.gr**  
  A vector of characters that represents the causes groups.
- **causes**  
  A vector with lists that represents the individual causes for each
- **main**  
  Main title for the diagram
- **sub**  
  Subtitle for the diagram (recommended the Six Sigma project name)
- **ss.col**  
  A vector of colors for a personalized drawing. At least five colors, sorted by descendant intensity

**Details**
The default value for ss.col is c("#666666", "#BBBBBB", "#CCCCCC", "#DDDDDD", "#EEEEEE"), a grayscale style. You can pass any accepted colour string.

**Value**
A drawing of the causes and effect with "fish-bone" shape

**Note**
The cause and effect diagram is also known as "Ishikawa diagram", and has been widely used in Quality Management. It is one of the Seven Basic Tools of Quality.

**Author(s)**
EL Cano
References


See Also

ss.pMap

Examples

```r
effect <- "Flight Time"
causes <- vector(mode = "list", length = length(causes.gr))
causes[1] <- list(c("operator #1", "operator #2", "operator #3"))
causes[2] <- list(c("height", "cleaning"))
causes[3] <- list(c("scissors", "tape"))
causes[4] <- list(c("rotor.length", "rotor.width2", "paperclip"))
causes[5] <- list(c("thickness", "marks"))
causes[6] <- list(c("calibrate", "model"))
ss.ceDiag(effect, causes.gr, causes, sub = "Paper Helicopter Project")
```

---

**ss.ci**

**Confidence Interval for the mean**

**Description**

Computes a confidence interval for the mean of the variable (parameter or feature of the process), and prints the data, a histogram with a density line, the result of the Shapiro-Wilks normality test and a quantile-quantile plot.

**Usage**

```r
ss.ci(
  x,
  sigma2 = NA,
  alpha = 0.05,
  data = NA,
  xname = "x",
  approx.z = FALSE,
  main = "Confidence Interval for the Mean",
  digits = 3,
  sub = "",
  ss.col = c("#666666", "#BBBBBB", "#CCCCCC", "#DDDDDD", "#EEEEEE")
)
```
Arguments

\begin{itemize}
\item \textbf{x} \hspace{1cm} A numeric vector with the variable data
\item \textbf{sigma2} \hspace{1cm} The population variance, if known
\item \textbf{alpha} \hspace{1cm} The error used to compute the $100 \times (1 - \alpha)\%$ confidence interval
\item \textbf{data} \hspace{1cm} The data frame containing the vector
\item \textbf{xname} \hspace{1cm} The name of the variable to be shown in the graph
\item \textbf{approx.z} \hspace{1cm} If TRUE it uses z statistic instead of t when sigma is unknown and sample size is greater than 30. The default is FALSE, change only if you want to compare with results obtained with the old-fashioned method mentioned in some books.
\item \textbf{main} \hspace{1cm} The main title for the graph
\item \textbf{digits} \hspace{1cm} Significant digits for output
\item \textbf{sub} \hspace{1cm} The subtitle for the graph (recommended: six sigma project name)
\item \textbf{ss.col} \hspace{1cm} A vector with colors
\end{itemize}

Details

When the population variance is known, or the size is greater than 30, it uses z statistic. Otherwise, it is uses t statistic.

If the sample size is lower than 30, a warning is displayed so as to verify normality.

Value

The confidence Interval.

A graph with the figures, the Shapiro-Wilks test, and a histogram.

Note

Thanks to the kind comments and suggestions from the anonymous reviewer of a tentative article.

Author(s)

EL Cano

References


See Also

\texttt{ss.data.rr}
Data for the batteries example

Description

This is a simulated data set of 18 measurements of the voltage of batteries using different voltmeters.

Usage

data(ss.data.batteries)

Format

A data frame with 18 observations on the following 4 variables.

voltmeter a factor with levels 1 2
battery a factor with levels 1 2 3
run a factor with levels 1 2 3
voltage a numeric vector

Note

This data set is used in chapter 5 of the book “Six Sigma with R” (see References).

Source

See references.

References


See Also

ss.rr

Examples

data(ss.data.batteries)
summary(ss.data.batteries)
plot(voltage~voltmeter, data = ss.data.batteries)
ss.data.bills  

Errors in bills data set

Description

This data set contains the number of errors detected in a set of bills and the name of the person in charge of the bill.

Usage

data("ss.data.bills")

Format

A data frame with 32 observations on the following 3 variables.

- `nbill` a numeric vector identifying a given bill
- `clerk` a character vector for the clerk responsible for the bill
- `errors` a character vector with the number of errors in the bill

Details

This data set illustrates concepts in the book “Quality Control with R”.

Source

Table 6.1 in the reference below.

References


Examples

data(ss.data.bills)
str(ss.data.bills)
barplot(table(ss.data.bills$clerk),
main = "number of invoices")
aggregate(errors ~ clerk, ss.data.bills, sum)
Description

A data frame with 50 observations of the diameter of the bolts manufactured in a production line.

Usage

data(ss.data.bolts)

Format

A data frame with 50 observations on the following variable.

diameter  a numeric vector with the diameter of the bolts

Note

This data set is used in chapter 4 of the book “Six Sigma with R” (see References).

Source

See references.

References


See Also

ss.lfa

Examples

data(ss.data.bolts)
summary(ss.data.bolts)
hist(ss.data.bolts$diameter)
ss.data.ca  
Data for a filling process in a winery

Description
The only field of the data is the volume measured in 20 bottles.

Usage
data(ss.data.ca)

Format
A data frame with 20 observations on the following variable.

Volume  a numeric vector (volume in cl

Note
This data set is used in chapter 7 of the book “Six Sigma with R” (see References).

Source
See references.

References

See Also
ss.study.ca

Examples
data(ss.data.ca)
summary(ss.data.ca)
hist(ss.data.ca$Volume)
Description

This data set contains the density for 24 pellets.

Usage

data("ss.data.density")

Format

A vector with 24 items for the density of a set of pellets ($gr/cm^3$).

Details

This data set illustrates concepts in the book “Quality Control with R”. Note that, in the book, the vector named `pdensity` is directly created and then used in the examples.

Source

Table 1.2 in the reference below.

References


Examples

data(ss.data.density)
str(ss.data.density)
summary(ss.data.density)
Pizza dough example data

Description

Experimental data for the scores given to a set of pizza doughs.

Usage

data(ss.data.doe1)

Format

A data frame with 16 observations on the following 6 variables.

- repl: Replication id
- flour: Level of flour in the recipe (- +)
- salt: Level of salt in the recipe (- +)
- bakPow: Level of Baking Powder in the recipe (- +)
- score: Scored assigned to the recipe
- ord: Randomized order

Note

This data set is used in chapter 11 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.doe1)
summary(ss.data.doe1)
lattice::bwplot(score ~ flour | salt + bakPow , data = ss.data.doe1, xlab = "Flour", strip = function(..., style) lattice::strip.default(..., strip.names=c(TRUE,TRUE)))
Data for the pizza dough example (robust design)

Description

Experimental data for the scores given to a set of pizza doughs. Noise factors added for robust design.

Usage

```r
data(ss.data.doe2)
```

Format

A data frame with 64 observations on the following 7 variables.

- `repl`  Replication id
- `flour`  Level of flour in the recipe (- +)
- `salt`  Level of salt in the recipe (- +)
- `bakPow`  Level of Baking Powder in the recipe (- +)
- `temp`  Level of temperature in preparation (- +)
- `time`  Level of time in preparation (- +)
- `score`  Scored assigned to the recipe

Note

This data set is used in chapter 11 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

```r
data(ss.data.doe2)
summary(ss.data.doe2)
lattice::bwplot(score ~ temp | time, data = ss.data.doe2)
```
**ss.data.pastries**  
*Pastries data*

**Description**

A data frame with 18 observations of the amount of the CTQ compound in some pastries from a bakery. There are two runs for each combination of two factors (laboratory and batch).

**Usage**

```r
data(ss.data.pastries)
```

**Format**

A data frame with 18 observations on the following 4 variables.

- `lab` laboratory: a factor with levels 1 2 3
- `batch` batch: a factor with levels 1 2 3
- `run` identifies the run: a factor with levels 1 2
- `comp` proportion of the compound in the pastry: a numeric vector

**Note**

This data set is used in chapter 5 exercises of the book “Six Sigma with R” (see References).

**Source**

See references.

**References**


**Examples**

```r
data(ss.data.pastries)
summary(ss.data.pastries)
lattice::xyplot(comp ~ lab | batch, data = ss.data.pastries)
```
Description

Humidity of 30 raw material used to make particle boards for individual control chart.

Usage

data(ss.data.pb1)

Format

A data frame with 30 observations on the following 2 variables.

- pb.group  Group id (distinct for each observation)
- pb.humidity  Humidity of the particle board

Note

This data set is used in chapter 12 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.pb1)
summary(ss.data.pb1)
ss.data.pb2  

Particle Boards Example - by Groups

Description

Humidity of 20 groups of size 5 of raw materials to make particle boards. For the mean and range control chart.

Usage

data(ss.data.pb2)

Format

A data frame with 100 observations on the following 2 variables.

- pb.group  a numeric vector
- pb.humidity  a numeric vector

Note

This data set is used in chapter 12 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.pb2)
summary(ss.data.pb2)
Description

Counts of raw materials stockouts during 22 weekdays in a month.

Usage

data(ss.data.pb3)

Format

A data frame with 22 observations on the following 3 variables.

day  Day id
stockouts  Number of stockouts
orders  Number of orders

Note

This data set is used in chapter 12 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.pb3)
summary(ss.data.pb3)
Data for Practice Boards Example - number of defects

Description

Number of defects detected in an order of particle boards.

Usage

data(ss.data.pb4)

Format

A data frame with 80 observations on the following 2 variables.

order  Order id
defects Number of defects

Note

This data set is used in chapter 12 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.pb4)
summary(ss.data.pb4)
Data set for the printer cartridge example

Description
This data set contains data from a simulated process of printer cartridge filling.

Usage
data(ss.data.pc)

Format
A data frame with 24 observations on the following 6 variables.

- pc.col: a factor with levels C B for the colour
- pc.filler: a factor with levels 1 2 3
- pc.volume: a numeric vector
- pc.density: a numeric vector
- pc.batch: a numeric vector
- pc.op: a factor with levels A B C D for the operator

Note
This data set is used in chapter 8 of the book “Six Sigma with R” (see References).

Source
See references.

References

Examples
data(ss.data.pc)
summary(ss.data.pc)
Description
This data set contains data from a simulated process of printer cartridges filling with complete replications.

Usage
data(ss.data.pc.big)

Format
A data frame with 72 observations on the following 5 variables,

- filler a factor with levels 1 2 3
- batch a factor with levels 1 2 3 4
- col a factor with levels B C
- operator a factor with levels 1 2 3
- volume a numeric vector

Note
This data set is used in chapter 8 of the book “Six Sigma with R” (see References).

Source
See references.

References

Examples
data(ss.data.pc.big)
summary(ss.data.pc.big)
ss.data.pc.r  

Data set for the printer cartridge example, by region

Description

This data set contains data from a simulated process of printer cartridge filling. The dataframe contains defects by region of each type of cartridge.

Usage

data(ss.data.pc.r)

Format

A data frame with 5 observations on the following 4 variables.

pc.regions a factor with levels region.1 region.2 region.3 region.4 region.5
pc.def.a a numeric vector for defects type A
pc.def.b a numeric vector for defects type B
pc.def a numeric vector for total defects

Note

This data set is used in chapter 8 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.pc.r)
summary(ss.data.pc.r)
Description

Example data for Measure phase of the Six Sigma methodology.

Usage

data(ss.data.rr)

Format

A data frame with 27 observations on the following 5 variables.

- prototype: a factor with levels prot #1 prot #2 prot #3
- operator: a factor with levels op #1 op #2 op #3
- run: a factor with levels run #1 run #2 run #3
- time1: a numeric vector
- time2: a numeric vector

Note

This data set is used in chapter 5 of the book “Six Sigma with R” (see References).

Source

See references.

References


Examples

data(ss.data.rr)
summary(ss.data.rr)
**Data set for the Guitar Strings example**

**Description**

This data set contains data from a simulated process of guitar strings production.

**Usage**

data(ss.data.strings)

**Format**

A data frame with 120 observations on the following 6 variables.

- id  a numeric vector
- type  a factor with levels A5 B2 D4 E1 E6 G3
- res  a numeric vector for resistance
- len  a numeric vector for length
- sound  a numeric vector for
- power  a numeric vector

**Note**

This data set is used in chapter 10 of the book “Six Sigma with R” (see References).

**Source**

See references.

**References**


**Examples**

data(ss.data.strings)
summary(ss.data.strings)
**ss.data.thickness**  

---

### Metal Plates Thickness

#### Description

This data set contains the thickness and additional data for 24 metal plates.

#### Usage

```r
data("ss.data.thickness")
```

#### Format

A data frame with 24 observations on the following 5 variables.

- **thickness**: a numeric vector with the thickness (in)
- **day**: a factor with the day (two days)
- **shift**: a factor with the shift (two shifts)
- **dayshift**: a factor with the day-shift combination
- **position**: a factor with the position of the thickness with respect to the nominal value of 0.75 in

#### Details

This data set illustrates concepts in the book “Quality Control with R”. Note that, in the book, the data set is named `plates` and it is created sequentially throughout the examples.

#### Source

Table 5.1 in the reference below.

#### References


#### Examples

```r
data(ss.data.thickness)
str(ss.data.thickness)
lattice::bwplot(thickness ~ shift | day,
    data = ss.data.thickness)
```
This data set contains the thickness and additional data for 84 metal plates.

Usage

```r
data("ss.data.thickness2")
```

Format

A data frame with 84 observations on the following 5 variables.

- **day** a factor with the day (seven days)
- **shift** a factor with the shift (two shifts)
- **thickness** a numeric vector with the thickness (in)
- **ushift** a factor with the day-shift combination
- **flaws** an integer vector with the number of flaws on the surface of sampled plates

Details

This data set illustrates concepts in the book “Quality Control with R”.

Source

Examples 8.1 and 9.9 in the reference below.

References


Examples

```r
data(ss.data.thickness2)
str(ss.data.thickness2)
lattice::dotplot(thickness ~ shift | day,
data = ss.data.thickness2,
layout = c(7, 1))
```
Description

This data set contains the 500 locations at which the density of a 0.5\textit{in}-thick engineered woodboard is measured, i.e., 0.001 \textit{in} apart.

Usage

data("ss.data.wbx")

Format

A vector with 500 items for the locations (\textit{in}).

Details

This data set illustrates concepts in the book “Quality Control with R”. This data set should be used along with the \texttt{ss.data.wby} data set.

Source

Example 10.1 in the reference below. It is a variation of the one introduced by Walker (2002).

References


See Also

\texttt{ss.data.wby}

Examples

```r
data(ss.data.wbx)
data(ss.data.wby)
plotProfiles(profiles = ss.data.wby,
             x = ss.data.wbx)
```
Description
This data set contains 50 profiles corresponding to the density measurements of 50 0.5in-thick engineered woodboard, measured in 500 locations.

Usage
\texttt{data("ss.data.wby")}

Format
A matrix with 500 rows (locations) and 50 columns (woodboard).

Details
This data set illustrates concepts in the book “Quality Control with R”. This data set should be used along with the \texttt{ss.data.wbx} data set.

Source
Example 10.1 in the reference below. It is a variation of the one introduced by Walker (2002).

References

See Also
\texttt{ss.data.wbx}

Examples
\begin{verbatim}
data(ss.data.wbx)
data(ss.data.wby)
plotProfiles(profiles = ss.data.wby,
            x = ss.data.wbx)
\end{verbatim}
ss.heli

*Description*

The pdf file contains a template with lines and indications to build the paper helicopter described in many SixSigma publications.

*Usage*

```r
ss.heli()
```

*Details*

The pdf file must be printed in A4 paper, without adjusting size to paper.

*Value*

No value is returned. A pdf file is saved in the working directory.

*Note*

See the vignette("HelicopterInstructions") to see assembling instructions.

**Author(s)**

EL Cano

**References**


**Examples**

```r
## Not run:
## ss.heli()
## vignette("HelicopterInstructions")
## End(Not run)
```
ss.lf

Evaluates the Loss Function for a process.

Description
The quality loss function is one of the tools of the Six Sigma methodology. The function assigns a cost to an observed value, that is larger as far as it is from the target.

Usage
ss.lf(lfa.Y1, lfa.Delta, lfa.Y0, lfa.L0)

Arguments
- lfa.Y1: The observed value of the CTQ (critical to quality) characteristic that will be evaluated.
- lfa.Delta: The tolerance for the CTQ.
- lfa.Y0: The target for the CTQ.
- lfa.L0: The cost of poor quality when the characteristic is $Y_0 + \Delta$.

Value
ss.lf: A number with the evaluated function at $Y_1$

Author(s)
EL Cano

References

See Also
ss.lfa

Examples
# Example bolts: evaluate LF at 10.5 if Target=10, Tolerance=0.5, L_0=0.001
ss.lf(10.5, 0.5, 10, 0.001)
Loss Function Analysis

Description

This function performs a Quality Loss Function Analysis, based in the Taguchi Loss Function for "Nominal-the-Best" characteristics.

Usage

```r
ss.lfa(
  lfa.data,  # Data frame with the sample to get the average loss.
  lfa.ctq,   # Name of the field in the data frame containing the data.
  lfa.Delta, # Tolerance of the process.
  lfa.Y0,    # Target of the process (see note).
  lfa.L0,    # Cost of poor quality at tolerance limit.
  lfa.size = NA, # Size of the production, batch, etc. to calculate the total loss in a group (span, batch, period, ...)
  lfa.output = "both", # Type of output (see details).
  lfa.sub = "Six Sigma Project"
)
```

Arguments

- `lfa.data`: Data frame with the sample to get the average loss.
- `lfa.ctq`: Name of the field in the data frame containing the data.
- `lfa.Delta`: Tolerance of the process.
- `lfa.Y0`: Target of the process (see note).
- `lfa.L0`: Cost of poor quality at tolerance limit.
- `lfa.size`: Size of the production, batch, etc. to calculate the total loss in a group (span, batch, period, ...)
- `lfa.output`: Type of output (see details).
- `lfa.sub`: Subtitle for the graphic output.

Details

`lfa.output` can take the values "text", "plot" or "both".

Value

- `lfa.k`: Constant k for the loss function
- `lfa.lf`: Expression with the loss function
- `lfa.MSD`: Mean Squared Differences from the target
- `lfa.avLoss`: Average Loss per unit of the process
- `lfa.Loss`: Total Loss of the process (if a size is provided)
Note

For smaller-the-better characteristics, the target should be zero (lfa.Y0 = 0). For larger-the-better characteristics, the target should be infinity (lfa.Y0 = Inf).

Author(s)

EL Cano

References


See Also

ss.lf, ss.data.bolts.

Examples

ss.lfa(ss.data.bolts, "diameter", 0.5, 10, 0.001, lfa.sub = "10 mm. Bolts Project", lfa.size = 100000, lfa.output = "both")

ss.pMap

Process Map

Description

This function takes information about the process we want to represent and draw the Process Map, with its X’s, x’s, Y’s and y’s in each step of the process

Usage

ss.pMap(
  steps,
  inputs.overall, 
  outputs.overall, 
  input.output,    
  x.parameters,   
  y.features, 
  main = "Six Sigma Process Map", 
  sub, 
  ss.col = c("#666666", "#BBBBBB", "#CCCCCC", "#DDDDDD", "#EEEEEE")
)
Arguments

- **steps**: A vector of characters with the name of the ‘n’ steps
- **inputs.overall**: A vector of characters with the name of the overall inputs
- **outputs.overall**: A vector of characters with the name of the overall outputs
- **input.output**: A vector of lists with the names of the inputs of the \(i - th\) step, that will be the outputs of the \((i - 1) - th\) step
- **x.parameters**: A vector of lists with a list of the x parameters of the process. The parameter is a vector with two values: the name and the type (view details)
- **y.features**: A vector of lists with a list of the y features of the step. The feature is a vector with two values: the name and the type (view details)
- **main**: The main title for the Process Map
- **sub**: Subtitle for the diagram (recommended the Six Sigma project name)
- **ss.col**: A vector of colours for a custom drawing. At least five colours, sorted by descendant intensity (see details)

Details

The type of the x parameters and y features can be: C(controllable), N(noise), Cr(Critical), P(Procedure). The default value for ss.col is c("#666666", "BBBBBB", "CCCCCC", "DDDDDD", "EEEEEE"), a grayscale style. You can pass any accepted color string.

Value

A graphic representation of the Map Process.

Note

The process map is the starting point for a Six Sigma Project, and it is very important to find out who the x’s and y’x are.

Author(s)

EL Cano

References


See Also

ss.ceDiag
Examples

```r
inputs.overall<-c("operators", "tools", "raw material", "facilities")
outputs.overall<-c("helicopter")
steps<-c("INSPECTION", "ASSEMBLY", "TEST", "LABELING")
#Inputs of process "i" are inputs of process "i+1"
input.output<-vector(mode="list",length=length(steps))
input.output[1]<-list(c("sheets", "..."))
input.output[2]<-list(c("sheets"))
input.output[3]<-list(c("helicopter"))
input.output[4]<-list(c("helicopter"))

#Parameters of each process
x.parameters<-vector(mode="list",length=length(steps))
x.parameters[1]<-list(c(list(c("width", "NC")), list(c("operator", "C")), list(c("Measure pattern", "P")), list(c("discard", "P"))))
x.parameters[2]<-list(c(list(c("operator", "C")), list(c("cut", "P")), list(c("fix", "P")), list(c("rotor.width", "C")), list(c("rotor.length", "C")), list(c("paperclip", "C")), list(c("tape", "C"))))
x.parameters[3]<-list(c(list(c("operator", "C")), list(c("throw", "P")), list(c("discard", "P")), list(c("environment", "N"))))
x.parameters[4]<-list(c(list(c("operator", "C")), list(c("label", "P"))))

#Features of each process
y.features<-vector(mode="list",length=length(steps))
y.features[1]<-list(c(list(c("ok", "Cr"))))
y.features[2]<-list(c(list(c("weight", "Cr"))))
y.features[3]<-list(c(list(c("time", "Cr"))))
y.features[4]<-list(c(list(c("label", "Cr"))))

ss.pMap(steps, inputs.overall, outputs.overall, input.output, x.parameters, y.features, sub="Paper Helicopter Project")
```

### Description

Performs Gage R&R analysis for the assessment of the measurement system of a process. Related to the Measure phase of the DMAIC strategy of Six Sigma.

### Usage

```r
ss.rr(
  var,
)```
part,  
appr,  
lsl = NA,  
usl = NA,  
sigma = 6,  
tolerance = usl - lsl,  
data,  
main = "Six Sigma Gage R&R Study",  
sub = "",  
alphaLim = 0.05,  
errorTerm = "interaction",  
digits = 4,  
method = "crossed",  
print_plot = TRUE,  
signifstars = FALSE  
)

Arguments

var             Measured variable
part             Factor for parts
appr             Factor for appraisers (operators, machines, ...)
lsl             Numeric value of lower specification limit used with USL to calculate Study Variation as %Tolerance
usl             Numeric value of upper specification limit used with LSL to calculate Study Variation as %Tolerance
sigma             Numeric value for number of std deviations to use in calculating Study Variation
tolerance             Numeric value for the tolerance
data             Data frame containing the variables
main             Main title for the graphic output
sub             Subtitle for the graphic output (recommended the name of the project)
alphaLim             Limit to take into account interaction
errorTerm             Which term of the model should be used as error term (for the model with interaction)
digits             Number of decimal digits for output
method             Character to specify the type of analysis to perform, "crossed" (default) or "nested"
print_plot             if TRUE (default) the plots are printed. Change to FALSE to avoid printing plots.
signifstars             if FALSE (default) the significance stars are ommitted. Change to TRUE to allow printing stars.
Details
Perform an R&R study for the measured variable, taking into account part and appraiser factors. It outputs the sources of Variability, and six graphs: bar chart with the sources of Variability, plots by appraiser, part and interaction and x-bar and R control charts.

Value
Analysis of Variance Table/s. Variance composition and %Study Var. Graphics.

- **anovaTable**: The ANOVA table of the model
- **anovaRed**: The ANOVA table of the reduced model (without interaction, only if interaction not significant)
- **varComp**: A matrix with the contribution of each component to the total variation
- **studyVar**: A matrix with the contribution to the study variation
- **ncat**: Number of distinct categories

Note
The F test for the main effects in the ANOVA table is usually made taken the operator/appraisal interaction as the error term (repeated measures model), thereby computing F as $\frac{MS_{factor}}{MS_{interaction}}$, e.g. in appendix A of AIAG MSA manual, in Montgomery (2009) and by statistical software such as Minitab. However, in the example provided in page 127 of the AIAG MSA Manual, the F test is performed as $\frac{MS_{factor}}{MS_{equipment}}$, i.e., repeatability. Thus, since version 0.9-3 of the SixSigma package, a new argument `errorTerm` controls which term should be used as error Term, one of "interaction", "repeatability".

Argument `alphaLim` is used as upper limit to use the full model, i.e., with interaction. Above this value for the interaction effect, the ANOVA table without the interaction effect is also obtained, and the variance components are computed pooling the interaction term with the repeatability.

Tolerance can be calculated from USL and LSL values or specified by hand.

The type of analysis to perform can be specified with the parameter `method`, "crossed" or "nested". Be sure to select the correct one and to have the data prepare for such type of analysis. If you don’t know which one is for you check it before. It is really important to perform the correct one. Otherwise results have no sense.

Author(s)
EL Cano with contributions by Kevin C Limburg

References


See Also

 ss.data.rr

Examples

ss.rr(time1, prototype, operator, data = ss.data.rr,
sub = "Six Sigma Paper Helicopter Project",
alphalim = 0.05,
errorTerm = "interaction",
lsl = 0.7,
usl = 1.8,
method = "crossed")

---

**ss.study.ca**  
*Graphs and figures for a Capability Study*

**Description**

Plots a Histogram with density lines about the data of a process. Check normality with qqplot and normality tests. Shows the Specification Limits and the Capability Indices.

**Usage**

```r
ss.study.ca(xST, xLT = NA, LSL = NA, USL = NA, Target = NA,
alpha = 0.05, f.na.rm = TRUE, f.main = "Six Sigma Capability Analysis Study",
f.sub = "", f.colours = c("#4682B4","#d1d1e0","#000000","#00C800","#FF0000"))
```

**Arguments**

- **xST**: Short Term process performance data
- **xLT**: Long Term process performance data
- **LSL**: Lower Specification Limit of the process
- **USL**: Upper Specification Limit of the process
- **Target**: Target of the process
- **alpha**: Type I error for the Confidence Interval
- **f.na.rm**: If TRUE NA data will be removed
- **f.main**: Main Title for the graphic output
- **f.sub**: Subtitle for the graphic output
- **f.colours**: Vector of colours for the graphic output

**Value**

Figures and plot for Capability Analysis
Note

The argument f.colours takes a vector of colours for the graphical outputs. The order of the elements are, first the colour for histogram bars, then Density ST lines, Density LT lines, Target, and Specification limits. It can be partially specified.

Author(s)

Main author: Emilio L. Cano. Contributions by Manu Alfaro.

References


See Also

`ss.ca.cp`

Examples

```r
ss.study.ca(ss.data.ca$Volume, rnorm(40, 753, 3),
LSL = 740, USL = 760, T = 750, alpha = 0.05,
  f.sub = "Winery Project")

ss.study.ca(ss.data.ca$Volume, rnorm(40, 753, 3),
LSL = 740, USL = 760, T = 750, alpha = 0.05,
  f.sub = "Winery Project",
  f.colours = c("#990000", "#007700", "#002299"))
```
Index

* Gauge
  ss.rr, 43
* MSA
  ss.rr, 43
* R&R
  ss.rr, 43
* Taguchi
  ss.lf, 39
  ss.lfa, 40
* capability
  ss.ca.z, 10
  ss.data.ca, 20
* cause-and-effect
  ss.ceDiag, 14
* cc
  ss.data.pb1, 25
  ss.data.pb2, 26
  ss.data.pb3, 27
  ss.data.pb4, 28
* charts
  ss.cc.constants, 13
  ss.data.pc, 29
  ss.data.pc.big, 30
  ss.data.pc.r, 31
* confidence
  ss.ci, 15
* constants
  ss.cc.constants, 13
* control
  ss.cc.constants, 13
* cpk
  ss.ca.z, 10
* cp
  ss.ca.z, 10
* datasets
  ss.data.bills, 18
  ss.data.density, 21
  ss.data.thickness, 34
  ss.data.thickness2, 35
  ss.data.wbx, 36
  ss.data.wby, 37
* data
  ss.data.batteries, 17
  ss.data.bolts, 19
  ss.data.ca, 20
  ss.data.doe1, 22
  ss.data.doe2, 23
  ss.data.pastries, 24
  ss.data.pb1, 25
  ss.data.pb2, 26
  ss.data.pb3, 27
  ss.data.pb4, 28
  ss.data.pc, 29
  ss.data.pc.big, 30
  ss.data.pc.r, 31
  ss.data.rr, 32
  ss.data.strings, 33
* doe
  ss.data.doe1, 22
  ss.data.doe2, 23
* function
  ss.lf, 39
  ss.lfa, 40
* interval
  ss.ci, 15
* lfa
  ss.data.bolts, 19
* loss
  ss.lf, 39
  ss.lfa, 40
* map
  ss.pMap, 41
* mean
  ss.ci, 15
* msa
  ss.data.batteries, 17
  ss.data.pastries, 24
  ss.data.rr, 32
INDEX

ss.data.strings, 33
* normality
  ss.ci, 15
* process
  ss.pMap, 41
* repeatability
  ss.rr, 43
* reproducibility
  ss.rr, 43
* test
  ss.ci, 15

climProfiles, 3
outProfiles, 4
plotControlProfiles, 5
plotProfiles, 6

SixSigma, 7
smoothProfiles, 8
ss.ca.cp, 47
ss.ca.cp (ss.ca.z), 10
ss.ca.cpk (ss.ca.z), 10
ss.ca.yield, 9
ss.ca.z, 10
ss.cc, 11
ss.cc.constants, 12, 13
ss.cc.getc4 (ss.cc.constants), 13
ss.cc.getd2 (ss.cc.constants), 13
ss.cc.getd3 (ss.cc.constants), 13
ss.ceOdiag, 8, 14, 42
ss.ci, 8, 15
ss.data.batteries, 17
ss.data.bills, 18
ss.data.bolts, 19, 41
ss.data.ca, 20
ss.data.density, 21
ss.data.doe1, 22
ss.data.doe2, 23
ss.data.pastries, 24
ss.data.pb1, 25
ss.data.pb2, 26
ss.data.pb3, 27
ss.data.pb4, 28
ss.data.pc, 29
ss.data.pc.big, 30
ss.data.pc.r, 31
ss.data.rr, 16, 32, 46

ss.data.strings, 33
ss.data.thickness, 34
ss.data.thickness2, 35
ss.data.wbx, 36, 37
ss.data.wby, 36, 37
ss.heli, 8, 38
ss.1f, 39, 41
ss.1fa, 8, 19, 39, 40
ss.pMap, 8, 15, 41
ss.rr, 8, 17, 43
ss.study.ca, 11, 20, 46