

# Package: CTxCC (via r-universe)

October 26, 2024

**Type** Package

**Title** Multivariate Normal Mean Monitoring Through Critical-to-X Control Chart

**Version** 0.1.0

**Maintainer** Dr. Diana Barraza-Barraza <diana.barraza@ujed.mx>

**Description** A comprehensive set of functions designed for multivariate mean monitoring using the Critical-to-X Control Chart. These functions enable the determination of optimal control limits based on a specified in-control Average Run Length (ARL), the calculation of out-of-control ARL for a given control limit, and post-signal analysis to identify the specific variable responsible for a detected shift in the mean. This suite of tools provides robust support for precise and effective process monitoring and analysis.

**License** GPL (>= 2)

**Encoding** UTF-8

**Imports** combinat, matrixcalc, stats, mvtnorm, expm, CompQuadForm

**NeedsCompilation** no

**Author** Dr. Burcu Aytacıoğlu [aut] (burcuaytacoglu@gmail.com), Dr. Diana Barraza-Barraza [aut, cre] (diana.barraza@ujed.mx), Dr. Víctor G. Tercero-Gómez [aut] (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco [aut] (lalo.cordero@gmail.com)

**Repository** CRAN

**Date/Publication** 2024-10-25 07:40:11 UTC

## Contents

C2.allPerms . . . . .	2
C2.Contribution . . . . .	3
C2.DecisionLimit . . . . .	4
decomposeA . . . . .	5
get.R . . . . .	6

SimulatedDistributionC2 . . . . .	7
wChisq.ar1 . . . . .	8
wChisq.CLim . . . . .	9
zConditionalParameters . . . . .	10

## Index 12

C2.allPerms *Contribution to C<sup>2</sup> for all variables*

### Description

Returns a matrix with values for C<sup>2</sup><sub>1</sub> and C<sup>2</sup><sub>k|C<sup>2</sup><sub>k-1</sub>,C<sup>2</sup><sub>k-2</sub>, ..., C<sup>2</sup><sub>1</sub>, k=2,3, 4... for all possible permutations among k variables</sub>

### Usage

```
C2.allPerms(z, W, R)
```

### Arguments

z	observation vector, kx1
W	matrix of variables weigths, kxk
R	correlation matrix, kxk

### Value

Data frame where, the first k columns correspond to variable that entred the model first, second... k-th. The following (k+1) to 2\*k columns contain the conditional contribution of the variable. The last column contains the sum of all contributions, meaning C<sup>2</sup><sub>k</sub>

### Author(s)

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

### References

Paper

### Examples

```
k<-3
sigma0 = matrix(diag(rep(1,k)),ncol = k)
mu0 = matrix(c(0,0,0), ncol = 1)
Weights = diag(c(0.5, 0.25,0.25))

library(mvtnorm)
set.seed(1000)
```

```

X = matrix(ncol= 1, data = rmvnorm(n = 1, mean = mu0, sigma = sigma0))
Z = (X - mu0)/sqrt(as.numeric(diag(sigma0)))
Corr<-get.R(Sigma0 = sigma0)

C2.allPerms(z = Z, W = Weights, R = Corr)

```

---

C2.Contribution

*Contribution of variable z.var to C^2*


---

### Description

Returns contribution of variable z.var to  $C^2$ , even if there are no previous variables in the model

### Usage

```
C2.Contribution(z, mean0, W, R, x.var, z.var = NULL)
```

### Arguments

z	observation vector, $k \times 1$ , where $z[x.var, ]$ correspond to variables already in the model
mean0	Mean vector for multivariate random vector under the null hypothesis. Dimensions: $k \times 1$
W	matrix of variables weights, $k \times k$
R	correlation matrix, $k \times k$
x.var	vector indicating variables already present in the model. length: $k-1$
z.var	scalar indicating variables to be included. Defaults to NULL, indicating there are no previous variables in the model

### Value

$C2.k.extra$ , scalar containing the contribution of variable z to  $C_k$

### Author(s)

Dr. Burcu Aytaçoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
 Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

### References

Paper

**Examples**

```

k<-3
sigma0 = matrix(diag(rep(1,k)),ncol = k)
mu0 = matrix(c(0,0,0), ncol = 1)
Weights = diag(c(0.5, 0.25,0.25))

library(mvtnorm)
set.seed(1000)
X = matrix(ncol= 1, data = rmvnorm(n = 1, mean = mu0, sigma = sigma0))
Z = (X - mu0)/sqrt(as.numeric(diag(sigma0)))
Corr<-get.R(Sigma0 = sigma0)

C2.Contribution(z = Z, W = Weights, R = Corr, x.var = 1:2, z.var = 3)

```

---

C2.DecisionLimit

*Conditional decision limit for z, given x already in model*


---

**Description**

Calculates the conditional decision limit for z, given x already in model, using the exact distribution for the conditional contribution of z to C\_k

**Usage**

```
C2.DecisionLimit(z, mu.C, R.C, A, x.var, alpha)
```

**Arguments**

z	observation vector, kx1, where z[x.var, ] correspond to variables already in the model
mu.C	scalar, conditional mean for z given x
R.C	scalar, conditional covariance for z given x
A	list containing matrix decomposition of A, preferably, obtained from function decomposeA
x.var	vector indicating variables already present in the model. length: k-1
alpha	confidence level for decision limit

**Details**

Proposition Distribution of a C<sup>2</sup> contribution from Paper Criticality Assessment for Enhanced Multivariate Process Monitoring

**Value**

conditionalCL, conditional decision limit for z's contribution to C\_k

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
 Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
k<-3
sigma0 = matrix(diag(rep(1,k)),ncol = k)
mu0 = matrix(c(0,0,0), ncol = 1)
Weights = diag(c(0.5, 0.25,0.25))

library(mvtnorm)
set.seed(1000)
X = matrix(ncol= 1, data = rmvnorm(n = 1, mean = mu0, sigma = sigma0))
Z = (X - mu0)/sqrt(as.numeric(diag(sigma0)))
Corr<-get.R(Sigma0 = sigma0)

A<-decomposeA(W = Weights, R = Corr, x.var = 1:2, z.var = 3)

Par<-zConditionalParameters(mean0 = mu0, R0 = Corr, z = Z, x.var = 1:2, z.var = 3)
C2.DecisionLimit(z = Z, mu.C = Par$muC, R.C = Par$RC, A = A, x.var = 1:2, alpha = 0.95)
```

---

 decomposeA

---

*Calculation and decomposition of matrix A*


---

**Description**

Decomposition of matrix A, required in Proposition 4.3. Decomposition given by equation 41

**Usage**

```
decomposeA(W, R, x.var, z.var)
```

**Arguments**

W	diagonal matrix containing the corresponding weight for each monitored variable. Dimensions $k \times k$
R	correlation matrix for monitored variables, $k \times k$
x.var	vector indicating variables already present in the model. length: $k-1$ .
z.var	scalar indicating variables to be included.

**Details**

Note that  $\text{length}(z.\text{var}) + \text{length}(x.\text{var}) = k$

**Value**

Returns decomposition of matrix A according to Equation 41 in paper.

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
k<-6 # variables
B<-matrix(runif(n = k*k),ncol= k)### creating random matrix for sigma0
sigma0 <- B%*%t(B)
R<-get.R(sigma0)
Weights = diag(rep(1/k,k))
decomposeA(W = Weights, R = R, x.var = 1:5, z.var = 6)
```

---

get.R

*Get Correlation matrix from a Covariance matrix*

---

**Description**

Returns a correlation matrix from a variance-covariance matrix

**Usage**

```
get.R(Sigma0)
```

**Arguments**

Sigma0                    variance-covariance matrix of dimensions kxk

**Value**

R                            correlation matrix correspondig to Sigma0

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
k<-6 # variables
B<-matrix(runif(n = k*k),ncol= k)### creating random matrix for sigma
sigma = B%*%t(B)
get.R(Sigma0=sigma)
```

---

SimulatedDistributionC2

*Distribution for C2, through simulation of its values*

---

**Description**

Simulates  $s$  instances of  $C^2_k$  given 1 to  $k-1$  variables are already in the model. Obtains the quantile indicated by  $\alpha$

**Usage**

```
SimulatedDistributionC2(z, R.C, mu.C, W, R, A, x.var, z.var, alpha, s)
```

**Arguments**

<code>z</code>	observation vector, $k \times 1$
<code>R.C</code>	scalar, conditional covariance for $z$ given $x$
<code>mu.C</code>	scalar, conditional mean for $z$ given $x$
<code>W</code>	matrix of variables weights, $k \times k$
<code>R</code>	correlation matrix, $k \times k$
<code>A</code>	list containing matrix decomposition of $A$ , preferably, obtained from function <code>decomposeA</code>
<code>x.var</code>	vector indicating variables already present in the model. length: $k-1$
<code>z.var</code>	scalar indicating variable to be included
<code>alpha</code>	quantile(s) of the distribution
<code>s</code>	scalar indicating amount of simulations

**Value**

Quantile(s) of the simulated distribution

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

## References

Paper

## Examples

```

k<-3
sigma0 = matrix(diag(rep(1,k)),ncol = k)
mu0 = matrix(c(0,0,0), ncol = 1)
Weights = diag(c(0.5, 0.25,0.25))

library(mvtnorm)
set.seed(1000)
X = matrix(ncol= 1, data = rmvnorm(n = 1, mean = mu0, sigma = sigma0))
Z = (X - mu0)/sqrt(as.numeric(diag(sigma0)))
Corr<-get.R(Sigma0 = sigma0)

A<-decomposeA(W = Weights, R = Corr, x.var = 1:2, z.var = 3)

Par<-zConditionalParameters(mean0 = mu0, R0 = Corr, z = Z, x.var = 1:2, z.var = 3)
SimulatedDistributionC2(z = Z, R.C = Par$RC, mu.C = Par$muC, W = Weights, R = Corr,
  A = A, x.var = 1:2, z.var = Z, alpha = 0.95, s = 1000 )

```

---

wChisq.arl

*Compute ARLs of Weighted Chi-Squared control charts for monitoring multivariate normal mean.*

---

## Description

Computation of the Average Run Length (ARL) for a Weighted Chi-Squared control chart for a given mean vector, delta, correlation matrix, R, control limit, h, and the vector of weights, w. The mean vector, delta, is defined in Proposition 4.2 from Paper Criticality Assessment for Enhanced Multivariate Process Monitoring.

## Usage

```
wChisq.arl(delta, R, h, w)
```

## Arguments

delta	Vector of values representing the change in the mean for each variable, 1xk
R	correlation matrix, kxk
h	Control limit of Weighted Chi-Squared Control chart
w	vector of weights, 1xk

## Value

arl	Average Run Length (ARL) for a Weighted Chi-Squared control chart for a given mean vector
-----	---

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
 Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
#Table 1 in the Paper Criticality Assessment for Enhanced Multivariate Process Monitoring.
delta <- c(0.5, 0.5) # mean vector (change vector)
R <- diag(2) # correlation matrix
h <- 2.649506 # Control limit
w <- c(0.50153, 0.49847) # vector of weights
wChisq.arl(delta, R, h, w)
```

---

wChisq.CLim	<i>Compute control limit of Weighted Chi-Squared control charts for monitoring multivariate normal mean.</i>
-------------	--

---

**Description**

Computation of a control limit of the Weighted Chi-Squared control chart for a given vector of weights,  $w$ , correlation matrix,  $R$ , and the false alarm rate,  $\alpha$ .

**Usage**

```
wChisq.CLim(w,R,alpha)
```

**Arguments**

w	vector of weights, 1xk
R	correlation matrix, kxk
alpha	false alarm rate

**Value**

ContLim	control limit of the Weighted Chi-Squared control chart
---------	---

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
 Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
# Table 1 in the Paper Criticality Assessment for Enhanced Multivariate Process Monitoring.
```

```
w <- c(0.29836,0.70164) #vector of weights
R <- diag(2)
alpha <- 0.005
wChisq.CLim(w,R,alpha)
```

```
w <- c(0.23912,0.76088) #vector of weights
R <- diag(2)
R[1,2] <- R[2,1] <- 0.25
alpha <- 0.005
wChisq.CLim(w,R,alpha)
```

---

zConditionalParameters

*Conditional parameters for z, given x*

---

**Description**

This function calculates and returns conditional parameters for z, given x are being already considered in the model

**Usage**

```
zConditionalParameters(mean0, R0, z, x.var, z.var)
```

**Arguments**

mean0	Mean vector for multivariate random vector under the null hypothesis. Dimensions: kx1
R0	Correlations matrix for multivariate random vector under the null hypothesis. Dimensions kxk
z	vector of random observation. Dimensions kx1
x.var	Elements of z that are already considered in the model
z.var	element of z whose contribution to $C_{-k C_{-k-1},C_{-k-2},\dots,C_{-1}}$ is going to be calculated

**Value**

A list containing

muC	conditional mean for z
RC	Conditional variance for z

**Author(s)**

Dr. Burcu Aytacıoğlu (burcuaytacoglu@gmail.com) Dr. Diana Barraza-Barraza (diana.barraza@ujed.mx),  
Dr. Víctor G. Tercero-Gómez (victor.tercero@tec.mx), Dr. A. Eduardo Cordero-Franco (lalo.cordero@gmail.com),

**References**

Paper

**Examples**

```
k<-3
sigma0 = matrix(diag(rep(1,k)),ncol = k)
mu0 = matrix(c(0,0,0), ncol = 1)
Weights = diag(c(0.5, 0.25,0.25))

library(mvtnorm)
set.seed(1000)
X = matrix(ncol= 1, data = rmvnorm(n = 1, mean = mu0, sigma = sigma0))
Z = (X - mu0)/sqrt(as.numeric(diag(sigma0)))
Corr<-get.R(Sigma0 = sigma0)

zConditionalParameters(mean0 = mu0, R0 = Corr, z = Z, x.var = 1:2, z.var = 3)
```

# Index

C2.allPerms, [2](#)  
C2.Contribution, [3](#)  
C2.DecisionLimit, [4](#)  
  
decomposeA, [5](#)  
  
get.R, [6](#)  
  
SimulatedDistributionC2, [7](#)  
  
wChisq.ar1, [8](#)  
wChisq.CLim, [9](#)  
  
zConditionalParameters, [10](#)