

# Package ‘cccm’

May 30, 2022

**Type** Package

**Title** Crossed Classification Credibility Model

**Version** 0.1.0

**Date** 2022-05-05

**Maintainer** Muhlis Ozdemir <muhlisozdemir@gazi.edu.tr>

**Description** Calculates the credit debt for the next period based on the available data using the cross-classification credibility model.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Imports** dplyr, rlang

**RoxygenNote** 7.1.2

**NeedsCompilation** no

**Author** Muhlis Ozdemir [aut, cre] (<<https://orcid.org/0000-0002-4921-8209>>),  
Seda Tugce Altan [aut, ctb] (<<https://orcid.org/0000-0003-0431-5256>>),  
Meral Ebegil [aut, ctb, ths] (<<https://orcid.org/0000-0003-4798-3422>>)

**Depends** R (>= 3.5.0)

**Repository** CRAN

**Date/Publication** 2022-05-30 09:00:02 UTC

## R topics documented:

calculate_generalMean . . . . .	2
calculate_group_averages_matrix . . . . .	3
calculate_obs_and_group_weights . . . . .	4
calculate_varianceComponents . . . . .	5
calculate_variance_and_std . . . . .	6
calculate_weights_of_obs_matrix . . . . .	7
cccm . . . . .	8
col_diff_matrix_with_vector . . . . .	8
control_data . . . . .	9

debt . . . . .	9
div_matrix_cols_with_vector . . . . .	10
div_matrix_rows_with_vector . . . . .	10
estimate_credibility . . . . .	11
mult_matrix_cols_with_vector . . . . .	12
row_diff_matrix_with_vector . . . . .	12
save_names . . . . .	13
set_data . . . . .	13

## Index 15

---

calculate\_generalMean *General Mean*

---

### Description

General Mean

### Usage

```
calculate_generalMean(
  raw_data,
  categorical_columns,
  weights_column,
  debt_column
)
```

### Arguments

`raw_data` a data set of credibility.  
`categorical_columns` categorical column of data set.  
`weights_column` weights column of data set.  
`debt_column` credit dept column of data set.

### Value

general mean

### Examples

```
raw_data <- debt
categorical_columns <- c(1,2)
weights_column <- 3
debt_column <- 4
calculate_generalMean(raw_data, categorical_columns, weights_column, debt_column)
```

---

calculate\_group\_averages\_matrix  
*Group Averages Matrix*

---

## Description

Group Averages Matrix

## Usage

```
calculate_group_averages_matrix(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

## Arguments

raw\_data            a data set of credibility.  
categorical\_columns            categorical column of data set.  
weights\_column    weights column of data set.  
debt\_column        credit dept column of data set.

## Value

group averages matrix

## Examples

```
raw_data <- debt  
categorical_columns <- c(1,2)  
weights_column <- 3  
debt_column <- 4  
calculate_group_averages_matrix(raw_data, categorical_columns, weights_column, debt_column)
```

calculate\_obs\_and\_group\_weights  
*Repeats of observations*

---

### **Description**

Repeats of observations

### **Usage**

```
calculate_obs_and_group_weights(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

### **Arguments**

`raw_data` a data set of credibility.  
`categorical_columns` categorical column of data set.  
`weights_column` weights column of data set.  
`debt_column` credit dept column of data set.

### **Value**

This function returns categorical group sizes.

### **Examples**

```
raw_data <- debt  
categorical_columns <- c(1,2)  
weights_column <- 3  
debt_column <- 4  
calculate_obs_and_group_weights(raw_data, categorical_columns, weights_column, debt_column)
```

---

calculate\_varianceComponents  
*Variance Components*

---

## Description

Variance Components

## Usage

```
calculate_varianceComponents(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

## Arguments

`raw_data` a data set of credibility.  
`categorical_columns` categorical column of data set.  
`weights_column` weights column of data set.  
`debt_column` credit dept column of data set.

## Value

variance components

## Examples

```
raw_data <- debt  
categorical_columns <- c(1,2)  
weights_column <- 3  
debt_column <- 4  
calculate_varianceComponents(raw_data, categorical_columns, weights_column, debt_column)
```

---

`calculate_variance_and_std`*Variance and Standard Deviation*

---

**Description**

Variance and Standard Deviation

**Usage**

```
calculate_variance_and_std(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

**Arguments**

`raw_data` a data set of credibility.  
`categorical_columns` categorical column of data set.  
`weights_column` weights column of data set.  
`debt_column` credit dept column of data set.

**Value**

variance and sd.

**Examples**

```
raw_data <- debt  
categorical_columns <- c(1,2)  
weights_column <- 3  
debt_column <- 4  
calculate_variance_and_std(raw_data, categorical_columns, weights_column, debt_column)
```

---

calculate\_weights\_of\_obs\_matrix  
*Weights of observation matrix*

---

## Description

Weights of observation matrix

## Usage

```
calculate_weights_of_obs_matrix(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

## Arguments

raw\_data            a data set of credibility.  
categorical\_columns            categorical column of data set.  
weights\_column    weights column of data set.  
debt\_column        credit dept column of data set.

## Value

Weights of observation matrix.

## Examples

```
raw_data <- debt  
categorical_columns <- c(1,2)  
weights_column <- 3  
debt_column <- 4  
calculate_weights_of_obs_matrix(raw_data, categorical_columns, weights_column, debt_column)
```

---

`cccm`*Crossed Classification Credibility Model.*

---

**Description**

Estimation of premium credibility for Crossed Classification Credibility Model. In this model an insurance portfolio is subdivided by two qualitative risk factors, modeled in symmetrical way. Especially this model presents an alternative way when data is not classifiable in a hierarchical manner and to determine main effects of both risk factors. Also this model more useful to calculate co-effect both risk factors. Dannenburg et al., (1995, ISBN:90-802117-3-7)

**Author(s)**

Muhlis Ozdemir <muhlisozdemir@gazi.edu.tr> Seda Tugce Altan <stugce.altan@gazi.edu.tr>  
Meral Ebegil <mdemirel@gazi.edu.tr>

**Examples**

```
raw_data <- debt
categorical_columns = c(1,2)
weights_column = 3
debt_column = 4
calculate_generalMean(raw_data, categorical_columns, weights_column, debt_column)
calculate_variance_and_std(raw_data, categorical_columns, weights_column, debt_column)
calculate_group_averages_matrix(raw_data, categorical_columns, weights_column, debt_column)
calculate_weights_of_obs_matrix(raw_data, categorical_columns, weights_column, debt_column)
calculate_varianceComponents(raw_data, categorical_columns, weights_column, debt_column)
estimate_credibility(raw_data, categorical_columns, weights_column, debt_column)
```

---

`col_diff_matrix_with_vector`*Column Wise Matrix Diff*

---

**Description**

This function returns of the column wise difference between the m matrix and the vector v



**Usage**

```
col_diff_matrix_with_vector(m, vec)
```

**Arguments**

m                    is a matrix  
vec                   is a vector

**Value**

This function returns a num matrix.

---

control_data	<i>Data checker</i>
--------------	---------------------

---

**Description**

Throws an error message if at least 2 features is not in categorical format.

**Usage**

```
control_data(x)
```

**Arguments**

x                    a dataset.

**Value**

This function checks whether dataset has at least 2 features in categorical format.

---

debt	<i>Debt Data</i>
------	------------------

---

**Description**

A real data which published by Turkey Banking Regulation and Supervisory Board <<https://www.bddk.org.tr/BultenAylik/en>>

**Usage**

```
debt
```

**Format**

A data frame of 106 rows and 4 columns

**bank** categorical data of bank type. Bank type includes four subcategory such as State Banks, Deposit Banks, Foreign Banks and Privately Owned Deposit Banks

**loan** categorical data of dept type. Loan type includes three subcategory such as non-performing vehicle, home, and consumer loan.

**weights** Numeric values of weights

**debt** Numeric values of debt

---

div\_matrix\_cols\_with\_vector

*Column Wise Matrix Division*

---

**Description**

This function returns of the column wise division of the m matrix and the vector v.

**Usage**

```
div_matrix_cols_with_vector(m, vec)
```

**Arguments**

m	is a matrix
vec	is a vector

**Value**

This function returns a num matrix.

---

div\_matrix\_rows\_with\_vector

*Row Wise Matrix Division*

---

**Description**

This function returns of the row wise division of the m matrix and the vector v.

**Usage**

```
div_matrix_rows_with_vector(m, vec)
```

**Arguments**

m                    is a matrix  
vec                  is a vector

**Value**

This function returns a num matrix.

---

estimate\_credibility    *The Credibility Premium Estimates*

---

**Description**

The Credibility Premium Estimates

**Usage**

```
estimate_credibility(  
  raw_data,  
  categorical_columns,  
  weights_column,  
  debt_column  
)
```

**Arguments**

raw\_data            a data set of credibility.  
categorical\_columns            categorical column of data set.  
weights\_column    weights column of data set.  
debt\_column        credit dept column of data set.

**Value**

returns premium estimation of credibility.

**Examples**

```
raw_data <- debt  
  
categorical_columns <- c(1,2)  
  
weights_column <- 3  
  
debt_column <- 4  
  
estimate_credibility(raw_data, categorical_columns, weights_column, debt_column)
```

---

`mult_matrix_cols_with_vector`*Column Wise Matrix Multiplication*

---

**Description**

This function returns of the column wise multiplication of the m matrix and the vector v.

**Usage**

```
mult_matrix_cols_with_vector(m, vec)
```

**Arguments**

m	is a matrix
vec	is a vector

**Value**

This function returns a num matrix.

---

`row_diff_matrix_with_vector`*Row Wise Matrix Diff*

---

**Description**

This function returns of the row wise difference between the m matrix and the vector v

**Usage**

```
row_diff_matrix_with_vector(m, vec)
```

**Arguments**

m	is a matrix
vec	is a vector

**Value**

This function returns a num matrix.

---

save_names	<i>Get names</i>
------------	------------------

---

**Description**

Get names

**Usage**

```
save_names(raw_data, categorical_columns)
```

**Arguments**

raw\_data            a data set of credibility.  
categorical\_columns            categorical column of data set.

**Value**

returns categorical variables' unique values and column names of data set.

**Examples**

```
raw_data <- debt  
  
categorical_columns <- c(1,2)  
  
save_names(raw_data, categorical_columns)
```

---

set_data	<i>Data prep</i>
----------	------------------

---

**Description**

Data prep

**Usage**

```
set_data(raw_data, categorical_columns, weights_column, debt_column)
```

**Arguments**

raw\_data            a data set of credibility.  
categorical\_columns            categorical column of data set.  
weights\_column    weights column of data set.  
debt\_column        credit debt column of data set.

**Value**

This function returns a tibble as `prepared_data` by using `raw_data`. Adds new columns to raw data as `weighted_obs`, `group_average_weights`, `variance_column`.

**Examples**

```
raw_data <- debt  
  
categorical_columns <- c(1,2)  
  
weights_column <- 3  
  
debt_column <- 4  
  
prepared_data <- set_data(raw_data, categorical_columns, weights_column, debt_column)
```

# Index

## \* datasets

- debt, [9](#)
  
- calculate\_generalMean, [2](#)
- calculate\_group\_averages\_matrix, [3](#)
- calculate\_obs\_and\_group\_weights, [4](#)
- calculate\_variance\_and\_std, [6](#)
- calculate\_varianceComponents, [5](#)
- calculate\_weights\_of\_obs\_matrix, [7](#)
- cccm, [8](#)
- cccm-package (cccm), [8](#)
- col\_diff\_matrix\_with\_vector, [8](#)
- control\_data, [9](#)
  
- debt, [9](#)
- div\_matrix\_cols\_with\_vector, [10](#)
- div\_matrix\_rows\_with\_vector, [10](#)
  
- estimate\_credibility, [11](#)
  
- mult\_matrix\_cols\_with\_vector, [12](#)
  
- row\_diff\_matrix\_with\_vector, [12](#)
  
- save\_names, [13](#)
- set\_data, [13](#)