

# Package ‘CopulaDTA’

November 23, 2017

**Type** Package

**Title** Copula Based Bivariate Beta-Binomial Model for Diagnostic Test Accuracy Studies

**Version** 1.0.0

**Date** 2017-11-23

**Depends** R (>= 3.4.2), rstan (>= 2.16.2)

**Imports** methods, ggplot2 (>= 2.2.1), plyr (>= 1.8.4), stats (>= 3.4.2), reshape2 (>= 1.4.2), grDevices (>= 3.4.2)

**Description** Modelling of sensitivity and specificity on their natural scale using copula based bivariate beta-binomial distribution to yield marginal mean sensitivity and specificity. The intrinsic negative correlation between sensitivity and specificity is modelled using a copula function. A forest plot can be obtained for categorical covariates or for the model with intercept only. Nyaga VN, Arbyn M, Aerts M (2017) <doi:10.18637/jss.v082.c01>.

**License** GPL-2

**LazyData** TRUE

**RoxygenNote** 6.0.1

**Suggests** knitr, loo, Rmisc, httr, bayesplot

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Victoria N Nyaga [aut, cre]

**Maintainer** Victoria N Nyaga <victoria.nyaga@outlook.com>

**Repository** CRAN

**Date/Publication** 2017-11-23 15:25:47 UTC

## R topics documented:

ascus . . . . .	2
cdtafit-class . . . . .	3
cdtamodel . . . . .	3

cdtamodel-class . . . . .	6
fit . . . . .	7
fit.cdtamodel . . . . .	8
forestplot.cdtafit . . . . .	10
plot . . . . .	12
print.cdtafit . . . . .	13
summary.cdtafit . . . . .	14
telomerase . . . . .	16
traceplot . . . . .	16
traceplot.cdtafit . . . . .	17

<b>Index</b>	<b>19</b>
--------------	-----------

---

ascus	<i>ASCUS dataset</i>
-------	----------------------

---

## Description

Arbyn et al. (2013) performed a Cochrane review on the accuracy of human papillomavirus testing (HC2) and repeat cytology (RepC) to triage of women with an equivocal Pap smear to diagnose cervical precancer.

## Usage

data(ascus)

## Format

A data frame of 20 observations and six variables:

**StudyID** Study identifier

**Test** Type of diagnostic test

**TP** Number of true positives

**FN** number of false negatives

**TN** number of true negatives

**FP** number of false positives

## References

Arbyn M, Roelens J, Simoens C, Buntinx F, Paraskevaidis E, Martin-Hirsch PPL, Prendiville W (2013). Human Papillomavirus Testing Versus Repeat Cytology for Triage of Minor Cytological Cervical Lesions." Cochrane Database of Systematic Reviews, pp. 31-201.

Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. Journal of Statistical Software, 82(1), 1-27. doi:10.18637/jss.v082.c01

---

cdtafit-class	<i>Class cdtafit</i>
---------------	----------------------

---

### Description

A cdtafit class in the CopulaDTA package.

### Slots

`data` a data-frame with no missing values containing TP, TN, FP, FN, 'SID' and co-variables(if necessary).

`SID` A string indicating the name of the column with the study identifier.

`copula` copula function, 'fgm', 'gauss', 'c90', '270', or 'frank'.

`modelargs` list containing control parameters for the prior distributions.

`fit` an object of class stanfit returned by the function sampling.

### Author(s)

Victoria N Nyaga <victoria.nyaga@outlook.com>

### See Also

[fit](#)

Other cdt: [cdtamodel-class](#)

---

cdtamodel	<i>Specify the copula based bivariate beta-binomial distribution to fit to the diagnostic data.</i>
-----------	---

---

### Description

Specify the copula based bivariate beta-binomial distribution to fit to the diagnostic data.

### Usage

```
cdtamodel(copula, modelargs = list())
```

**Arguments**

**copula** a description of the copula function used to model the correlation between sensitivity and specificity. This is a string naming the copula function. The choices are "fgm", "frank", "gauss", "c90" and "c270".

**modelargs** a (optional) list of control parameter for the prior distributions. The parameters in the list include:

- **formula.se** An object of class "formula": A symbolic description of a linear model to be fitted to mean  $E(x)$  of sensitivity in the logit scale. the default (when no covariates are included) symbolic description is  $SID \sim 1$  corresponds to the model formula  $E(x) = \mu = \exp(a)/(1 + \exp(a))$  where  $a$  is the intercept. When the covariates are categorical and the relative measures are needed it is important to remove the intercept from the model to obtain meaningful parameters. EG for a covariate 'Test' with two levels(A and B) and relative sensitivity of B versus A is needed, then the correct formula is  $SID \sim Test - 1$  or  $SID \sim Test + 0$ . See [formula](#). For further information on interpretation of parameters in logistic regression see Agresti A(2002) Chapter 5.
- **formula.sp** An object of class "formula": A symbolic description of a linear model to be fitted to specificity data. By default the covariate information for sensitivity is used.
- **formula.omega** An object of class "formula": A symbolic description of a linear model to be fitted to the copula function. By default the covariate information for sensitivity is used.
- **transform.omega** A logical value indicating whether a constrained correlation parameter should be mapped into an non-constrained scale. This applies to all the allowed copula functions except "frank". The default is TRUE.
- **param** indication of the parameterisation used to map the marginal mean and precision/dispersion to the alpha and beta parameters of the beta distribution. There are two choices: param=1 which uses

$$\alpha = \mu * \phi, \beta = (1 - \mu) * \phi$$

where

$$\mu = \alpha / (\alpha + \beta), 0 \leq \mu \leq 1,$$

and

$$\phi = \alpha + \beta, \phi \geq 0.$$

param=2 uses

$$\alpha = ((1 - \phi) / \phi) * \mu$$

$$\beta = ((1 - \phi) / \phi) * (1 - \mu)$$

where

$$\mu = \alpha / (\alpha + \beta); 0 \leq \mu \leq 1,$$

and

$$\phi = 1 / (1 + \alpha + \beta); 0 \leq \phi \leq 1.$$

- `prior.lseA` description of prior distribution of the marginal mean sensitivity in the logit scale. The default is "normal" distribution. For other distributions see stan documentation at <http://mc-stan.org/documentation/>.
- `par.lse1A` numeric value indicating the location of the prior distribution of the marginal mean sensitivity in the logit scale. The default is 0 which implying a distribution centered around 0.5 in the 0-1 scale.
- `par.lse2A` numeric value indicating the spread(standard deviation) of the prior distribution of the marginal mean sensitivity in the logit scale and can be interpreted as the quantity of prior information. vague and non-informative priors are specified by a distribution with large variance. The default is `sd=10` implying that the variance is 100.
- `prior.lspA` description of prior distribution of the marginal mean specificity in the logit scale. The default is "normal" distribution.
- `par.lsp1A` numeric value indicating the location of the prior distribution of the marginal mean specificity in the logit scale. The default is 0 which implying a distribution centered around 0.5 in the 0-1 scale.
- `par.lsp2A` numeric value indicating the spread(standard deviation) of the prior distribution of the marginal mean specificity in the logit scale and can be interpreted as the quantity of prior information. vague and non-informative priors are specified by a distribution with large variance. The default is `sd=10` implying that the variance is 100.
- `prior.omegaA` description of prior distribution of the correlation parameter(s). The default is "normal" distribution since "`transform.omega=TRUE`". When "`transform.omega=FALSE`" the candidate prior distributions are  $U[-1, 1]$  for `fgm` and gaussian copulas, and half-cauchy(0, 2.5),  $\text{gamma}(0.001, 0.001)$  for the `C90` and `C270`.
- `par.omega1A` numeric value indicating the location of the prior distribution of the correlation parameter(s). The default is 0.
- `par.omega2A` numeric value indicating the scale/spread(standard deviation) of the prior distribution of the correlation parameter(s). The default is `sd=10`.

**Value**

An object of `cdtamodel` class.

**Author(s)**

Victoria N Nyaga

**References**

- Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. *Journal of Statistical Software*, 82(1), 1-27. doi:10.18637/jss.v082.c01
- Agresti A (2002). *Categorical Data Analysis*. John Wiley & Sons, Inc.

Clayton DG (1978). A model for Association in Bivariate Life Tables and its Application in Epidemiological Studies of Familial Tendency in Chronic Disease Incidence. *Biometrika*, 65(1), 141-151.

Frank MJ (1979). On The Simultaneous Associativity of  $F(x, y)$  and  $x + y - F(x, y)$ . *Aequationes Mathematicae*, pp. 194-226.

Farlie DGJ (1960). The Performance of Some Correlation Coefficients for a General Bivariate Distribution. *Biometrika*, 47, 307-323.

Gumbel EJ (1960). Bivariate Exponential Distributions. *Journal of the American Statistical Association*, 55, 698-707.

Meyer C (2013). The Bivariate Normal Copula. *Communications in Statistics - Theory and Methods*, 42(13), 2402-2422.

Morgenstern D (1956). Einfache Beispiele Zweidimensionaler Verteilungen. *Mitteilungsblatt für Mathematische Statistik*, 8, 23 - 235.

Sklar A (1959). Fonctions de Repartition a n Dimensions et Leurs Marges. *Publications de l'Institut de Statistique de L'Universite de Paris*, 8, 229-231.

## Examples

```
data(telomerase)
model1 <- cdtamodel(copula = 'fgm')

model2 <- cdtamodel(copula = 'fgm',
                    modelargs=list(param=2,
                                   prior.lse='normal',
                                   par.lse1=0,
                                   par.lse2=5,
                                   prior.lsp='normal',
                                   par.lsp1=0,
                                   par.lsp2=5))

model3 <- cdtamodel(copula = 'fgm',
                    modelargs = list(formula.se = StudyID ~ Test - 1))
```

---

cdtamodel-class

*Class cdtamodel*

---

## Description

A cdtamodel class in the CopulaDTA package.

## Slots

`copula` copula function, 'fgm', 'gauss', 'c90', '270', or 'frank'.

`modelcode` character with the model code as returned by the model function

`modelargs` list containing control parameters for the prior distributions

**Author(s)**

Victoria N Nyaga <victoria.nyaga@outlook.com>

**See Also**

[cdtamodel](#)

Other cdata: [cdtafit-class](#)

---

 fit

*A function to fit the model.*


---

**Description**

A function to fit the model.

**Usage**

```
fit(object, ...)

## S4 method for signature 'cdtamodel'
fit(object, data, SID, cores = 3, chains = 3,
     iter = 6000, warmup = 1000, thin = 10, ...)
```

**Arguments**

object	A cdtamodel object created by <a href="#">cdtamodel</a> function.
...	Other optional parameters as specified in <a href="#">stan</a> .
data	A data-frame with no missing values containing TP, TN, FP, FN, 'SID' and covariables(if necessary).
SID	A string indicating the name of the column with the study identifier.
cores	A positive numeric values specifying the number of cores to use to execute parallel sampling. When the hardware has more at least 4 cores, the default is 3 cores and otherwise 1 core.
chains	A positive numeric value specifying the number of chains, default is 3.
iter	A positive numeric value specifying the number of iterations per chain. The default is 6000.
warmup	A positive numeric value (<iter) specifying the number of iterations to be discarded(burn-in/warm-up). The default is 1000.
thin	A positive numeric value specifying the interval in which the samples are stored. The default is 10.

---

fit.cdtamodel	<i>Fit copula based bivariate beta-binomial distribution to diagnostic data.</i>
---------------	--

---

## Description

Fit copula based bivariate beta-binomial distribution to diagnostic data.

## Usage

```
fit.cdtamodel(cdtamodel, data, SID, cores = 3, chains = 3, iter = 6000,
  warmup = 1000, thin = 10, ...)
```

## Arguments

cdtamodel	An object of cdtamodel class from <a href="#">cdtamodel</a> .
data	A data-frame with no missing values containing TP, TN, FP, FN, 'SID' and covariables(if necessary).
SID	A string indicating the name of the column with the study identifier.
cores	A positive numeric values specifying the number of cores to use to execute parallel sampling. When the hardware has more at least 4 cores, the default is 3 cores and otherwise 1 core.
chains	A positive numeric value specifying the number of chains, default is 3.
iter	A positive numeric value specifying the number of iterations per chain. The default is 6000.
warmup	A positive numeric value (<iter) specifying the number of iterations to be discarded(burn-in/warm-up). The default is 1000.
thin	A positive numeric value specifying the interval in which the samples are stored. The default is 10.
...	Other optional parameters as specified in <a href="#">stan</a> .

## Value

An object of cdtafit class.

## Author(s)

Victoria N Nyaga <[victoria.nyaga@outlook.com](mailto:victoria.nyaga@outlook.com)>



## References

- Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. *Journal of Statistical Software*, 82(1), 1-27. doi:10.18637/jss.v082.c01
- Agresti A (2002). *Categorical Data Analysis*. John Wiley & Sons, Inc.
- Clayton DG (1978). A model for Association in Bivariate Life Tables and its Application in Epidemiological Studies of Familial Tendency in Chronic Disease Incidence. *Biometrika*, 65(1), 141-151.
- Frank MJ (1979). On The Simultaneous Associativity of  $F(x, y)$  and  $x + y - F(x, y)$ . *Aequationes Mathematicae*, pp. 194-226.
- Farlie DGJ (1960). The Performance of Some Correlation Coefficients for a General Bivariate Distribution. *Biometrika*, 47, 307-323.
- Gumbel EJ (1960). Bivariate Exponential Distributions. *Journal of the American Statistical Association*, 55, 698-707.
- Meyer C (2013). The Bivariate Normal Copula. *Communications in Statistics - Theory and Methods*, 42(13), 2402-2422.
- Morgenstern D (1956). Einfache Beispiele Zweidimensionaler Verteilungen. *Mitteilungsblatt für Mathematische Statistik*, 8, 23 - 235.
- Sklar A (1959). Fonctions de Repartition a n Dimensions et Leurs Marges. *Publications de l'Institut de Statistique de L'Universite de Paris*, 8, 229-231.

## Examples

```
data(telomerase)
model1 <- cdtamodel(copula = 'fgm')

model2 <- cdtamodel(copula = 'fgm',
  modelargs=list(param=2,
    prior.lse='normal',
    par.lse1=0,
    par.lse2=5,
    prior.lsp='normal',
    par.lsp1=0,
    par.lsp2=5))

model3 <- cdtamodel(copula = 'fgm',
  modelargs = list(formula.se = StudyID ~ Test - 1))
## Not run:
fit1 <- fit(model1,
  SID='ID',
  data=telomerase,
  iter=2000,
  warmup=1000,
  thin=1,
  seed=3)

fit2 <- fit(model2,
```

```

        SID='StudyID',
        data=ascus,
        iter=2000,
        warmup=1000,
        thin=1,
        seed=3)

## End(Not run)

```

---

forestplot.cdtafit      *Produce forest plots for categorical covariates.*

---

## Description

Produce forest plots for categorical covariates.

## Usage

```

forestplot.cdtafit(x, title.1 = NULL, title.2 = NULL, title.3 = NULL,
  graph = NULL, width = 0.2, shape.1 = 19, size.1 = 2.5, shape.2 = 8,
  size.2 = 2.5, shape.0 = 9, size.0 = 3.5, cols.1 = NULL,
  cols.2 = NULL, digits = 3, ...)

```

## Arguments

x	A cdtafit object from <a href="#">fit</a> .
title.1	An optional string indicating the title of graph 1.
title.2	An optional string indicating the title of graph 2.
title.3	An optional string indicating the title of graph 3.
graph	An optional numeric value indicating which forest to plot(s) to graph. Valid values are: 0 - for no graph, 1 - yielding a forest plot of the sensitivity and specificity with a 95 percent exact confidence intervals, 2 - yielding a forest plot of the posterior study-specific sensitivity and specificity and the marginal mean sensitivity and specificity and their corresponding 95 percent credible intervals, 3 - yielding a combination of 1 and 2 in one plot, and NULL(default) - yielding plots of 1, 2 and 3.
width	An optional numeric value to adjust the dogging position. The default is 0.2.
shape.1	An optional numeric value(0-255) indicating the symbol to plot in graph 1. The default is 19 which is a solid circle. See <a href="#">points</a> for more details.
size.1	An optional positive numeric value indicating the size of symbols in graph 1. The default is 2.5.
shape.2	An optional numeric value(0-255) indicating the symbol to plot in graph 2. The default is 8 which is a star. See <a href="#">points</a> for more details.

size.2	An optional positive numeric value indicating the size of symbols in graph 2. The default is 2.5.
shape.0	An optional numeric value(0-255) indicating the symbol representing the posterior marginal mean in graph 2. The default is 19 which is a solid circle. See <a href="#">points</a> for more details.
size.0	An optional numeric value indicating the size of symbols representing the posterior marginal means in graph 2.
cols.1	An optional string vector specifying colours of shapes in graph 1.
cols.2	An optional string vector specifying colours of shapes in graph 2.
digits	An optional positive value to control the number of digits to print when printing numeric values. The default is 3.
...	other <a href="#">stan</a> options.

**Value**

forestplots by ggplot2.

**Author(s)**

Victoria N Nyaga <victoria.nyaga@outlook.com>

**References**

- Watanabe S (2010). Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research*, 11, 3571-3594.
- Vehtari A, Gelman A (2014). WAIC and Cross-validation in Stan. Unpublished, pp. 1-14.

**Examples**

```
data(telomerase)
model1 = cdtamodel(copula = 'fgm')

model2 = cdtamodel(copula = 'fgm',
  modelargs=list(param=2,
    prior.lse='normal',
    par.lse1=0,
    par.lse2=5,
    prior.lsp='normal',
    par.lsp1=0,
    par.lsp2=5))

model3 = cdtamodel(copula = 'fgm',
  modelargs = list(formula.se = StudyID ~ Test - 1))

## Not run:
fit1 <- fit(model1,
  SID='ID',
  data=telomerase,
  iter=2000,
```

```

warmup=1000,
thin=1,
seed=3)

plot(fit1)

## End(Not run)

```

---

plot

*A function to produce forest plots.*


---

### Description

A function to produce forest plots.

### Usage

```

plot(object, ...)

## S4 method for signature 'cdatafit'
plot(object, title.1 = NULL, title.2 = NULL,
      title.3 = NULL, graph = NULL, width = 0.2, shape.1 = 19,
      size.1 = 2.5, shape.2 = 8, size.2 = 2.5, shape.0 = 9, size.0 = 3.5,
      cols.1 = NULL, cols.2 = NULL, digits = 3, ...)

```

### Arguments

object	A <code>cdatafit</code> object from <a href="#">fit</a> .
...	other <a href="#">stan</a> options.
title.1	An optional string indicating the title of graph 1.
title.2	An optional string indicating the title of graph 2.
title.3	An optional string indicating the title of graph 3.
graph	An optional numeric value indicating which forest to plot(s) to graph. Valid values are: 0 - for no graph, 1 - yielding a forest plot of the sensitivity and specificity with a 95 percent exact confidence intervals, 2 - yielding a forest plot of the posterior study-specific sensitivity and specificity and the marginal mean sensitivity and specificity and their corresponding 95 percent credible intervals, 3 - yielding a combination of 1 and 2 in one plot, and NULL(default) - yielding plots of 1, 2 and 3.
width	An optional numeric value to adjust the dogding position. The default is 0.2.
shape.1	An optional numeric value(0-255) indicating the symbol to plot in graph 1. The default is 19 which is a solid circle. See <a href="#">points</a> for more details.
size.1	An optional positive numeric value indicating the size of symbols in graph 1. The default is 2.5.

shape.2	An optional numeric value(0-255) indicating the symbol to plot in graph 2. The default is 8 which is a star. See <a href="#">points</a> for more details.
size.2	An optional positive numeric value indicating the size of symbols in graph 2. The default is 2.5.
shape.0	An optional numeric value(0-255) indicating the symbol representing the posterior marginal mean in graph 2. The default is 19 which is a solid circle. See <a href="#">points</a> for more details.
size.0	An optional numeric value indicating the size of symbols representing the posterior marginal means in graph 2.
cols.1	An optional string vector specifying colours of shapes in graph 1.
cols.2	An optional string vector specifying colours of shapes in graph 2.
digits	An optional positive value to control the number of digits to print when printing numeric values. The default is 3.

---

print.cdtafit	<i>Print a summary of the fitted model.</i>
---------------	---

---

## Description

Print a summary of the fitted model.

## Usage

```
## S3 method for class 'cdtafit'
print(x, digits = 3, ...)
```

## Arguments

x	An cdtafit object from <a href="#">fit</a> .
digits	An optional positive value to control the number of digits to print when printing numeric values. The default is 3.
...	other <a href="#">stan</a> options.

## Value

The posterior mean and 95 percent credible intervals, n\_eff, Rhat and WAIC.

## Author(s)

Victoria N Nyaga

## References

Watanabe S (2010). Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research*, 11, 3571-3594.

Vehtari A, Gelman A (2014). WAIC and Cross-validation in Stan. Unpublished, pp. 1-14.

**Examples**

```

data(telomerase)
model1 <- cdtamodel(copula = 'fgm')

model2 <- cdtamodel(copula = 'fgm',
                    modelargs=list(param=2,
                                   prior.lse='normal',
                                   par.lse1=0,
                                   par.lse2=5,
                                   prior.lsp='normal',
                                   par.lsp1=0,
                                   par.lsp2=5))

model3 <- cdtamodel(copula = 'fgm',
                    modelargs = list(formula.se = StudyID ~ Test - 1))
## Not run:

fit1 <- fit(model1,
            SID='ID',
            data=telomerase,
            iter=2000,
            warmup=1000,
            thin=1,
            seed=3)

print(fit1)

## End(Not run)

```

---

summary.cdtafit

*Function to generate a summary a cdtafit object.*


---

**Description**

Function to generate a summary a cdtafit object.

**Usage**

```

## S3 method for class 'cdtafit'
summary(object, digits = 3, ...)

```

**Arguments**

object	An object from <a href="#">fit</a> .
digits	An optional positive value to control the number of digits to print when printing numeric values.
...	other <a href="#">stan</a> options.

**Value**

The posterior mean and 95 percent credible intervals,  $n_{\text{eff}}$ ,  $R_{\text{hat}}$  and WAIC.

**Author(s)**

Victoria N Nyaga

**References**

Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. *Journal of Statistical Software*, 82(1), 1-27. doi:10.18637/jss.v082.c01

Watanabe S (2010). Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research*, 11, 3571-3594.

Vehtari A, Gelman A (2014). WAIC and Cross-validation in Stan. Unpublished, pp. 1-14.

**Examples**

```
data(telomerase)
model1 <- cdtamodel(copula = 'fgm')

model2 <- cdtamodel(copula = 'fgm',
  modelargs=list(param=2,
    prior.lse='normal',
    par.lse1=0,
    par.lse2=5,
    prior.lsp='normal',
    par.lsp1=0,
    par.lsp2=5))

model3 <- cdtamodel(copula = 'fgm',
  modelargs = list(formula.se = StudyID ~ Test - 1))
## Not run:

fit1 <- fit(model1,
  SID='ID',
  data=telomerase,
  iter=2000,
  warmup=1000,
  thin=1,
  seed=3)

ss <- summary(fit1)

## End(Not run)
```

---

telomerase	<i>Telomerase dataset</i>
------------	---------------------------

---

**Description**

Glas et al. (2003) systematically reviewed the sensitivity and specificity of cytology and other markers including telomerase for primary diagnosis of bladder cancer.

**Usage**

```
data(telomerase)
```

**Format**

A data frame of ten observations and five variables:

**ID** Study identifier

**TP** Number of true positives

**FN** number of false negatives

**TN** number of true negatives

**FP** number of false positives

**References**

Glas AS, Roos D, Deutekom M, Zwinderman AH, Bossuyt PMM, Kurth KH (2003). Tumor Markers in the Diagnosis of Primary Bladder Cancer. A Systematic Review. *The Journal of Urology*, 169(6), 1975-1982.

Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. *Journal of Statistical Software*, 82(1), 1-27. doi:10.18637/jss.v082.c01

---

traceplot	<i>A function to produce traceplots.</i>
-----------	--

---

**Description**

A function to produce traceplots.

A function to produce traceplots.

**Usage**

```
traceplot(object, ...)
```

```
## S4 method for signature 'cdtafit'
```

```
traceplot(object, ...)
```



**Arguments**

object            A cdtafit object from [fit](#)  
 ...                Extra optional arguments as defined in [stan\\_trace](#).

---

traceplot.cdtafit      *Trace plot using ggplot2.*

---

**Description**

Trace plot using ggplot2.

**Usage**

```
traceplot.cdtafit(x, ...)
```

**Arguments**

x                    An cdtafit object from [fit](#).  
 ...                    additional options. See [stan\\_trace](#) for more details.

**Value**

A ggplot object of the parameters of the models mean structure.

**Author(s)**

Victoria N Nyaga

**References**

Nyaga VN, Arbyn M, Aerts M (2017). CopulaDTA: An R Package for Copula-Based Beta-Binomial Models for Diagnostic Test Accuracy Studies in a Bayesian Framework. *Journal of Statistical Software*, 82(1), 1-27. doi:10.18637/jss.v082.c01

**Examples**

```
data(telomerase)
model1 <- cdtamodel(copula = 'fgm')

model2 <- cdtamodel(copula = 'fgm',
  modelargs=list(param=2,
    prior.lse='normal',
    par.lse1=0,
    par.lse2=5,
    prior.lsp='normal',
    par.lsp1=0,
    par.lsp2=5))
```

```
model3 <- cdtamodel(copula = 'fgm',
                    modelargs = list(formula.se = StudyID ~ Test - 1))
## Not run:
fit1 <- fit(model1,
            SID='ID',
            data=telomerase,
            iter=2000,
            warmup=1000,
            thin=1,
            seed=3)

traceplot(fit1)

traceplot(fit1) +
theme(axis.text.x = element_text(size=10, colour='black'),
      axis.text.y = element_text(size=10, colour='black'),
      axis.title.x = element_text(size=10, colour='black'),
      strip.text = element_text(size = 10, colour='black'),
      axis.title.y= element_text(size=10, angle=0, colour='black'),
      strip.text.y = element_text(size = 10, colour='black'),
      strip.text.x = element_text(size = 10, colour='black'),
      plot.background = element_rect(fill = "white", colour='white'),
      panel.grid.major = element_blank(),
      panel.background = element_blank(),
      strip.background = element_blank(),
      axis.line.x = element_line(color = 'black'),
      axis.line.y = element_line(color = 'black'))

## End(Not run)
```

# Index

`ascus`, [2](#)

`cdtafit-class`, [3](#)

`cdtamodel`, [3](#), [7](#), [8](#)

`cdtamodel-class`, [6](#)

`fit`, [3](#), [7](#), [10](#), [12–14](#), [17](#)

`fit`, `cdtamodel-method (fit)`, [7](#)

`fit.cdtamodel`, [8](#)

`forestplot.cdtafit`, [10](#)

`formula`, [4](#)

`plot`, [12](#)

`plot`, `cdtafit-method (plot)`, [12](#)

`points`, [10–13](#)

`print.cdtafit`, [13](#)

`stan`, [7](#), [8](#), [11–14](#)

`stan_trace`, [17](#)

`summary.cdtafit`, [14](#)

`telomerase`, [16](#)

`traceplot`, [16](#)

`traceplot`, `cdtafit-method (traceplot)`, [16](#)

`traceplot.cdtafit`, [17](#)