

# Package ‘nimbleCarbon’

October 13, 2022

**Title** Bayesian Analyses of Radiocarbon Dates with NIMBLE

**Version** 0.2.1

**Description** Provides utility functions and custom probability distribution for Bayesian analyses of radiocarbon dates within the ‘nimble’ modelling framework. It includes various population growth models, nimbleFunction objects, as well as a suite of functions for prior and posterior predictive checks for demographic inference (Crema and Shoda (2021) <[doi:10.1371/journal.pone.0251695](https://doi.org/10.1371/journal.pone.0251695)>) and other analyses.

**Depends** R (>= 3.5.0), nimble (>= 0.12.0)

**Imports** rcarbon,graphics,grDevices,utils,snow,doSNOW,foreach,coda

**Suggests** knitr, rmarkdown

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**VignetteBuilder** knitr

**RoxygenNote** 7.1.1

**Language** en-GB

**NeedsCompilation** no

**Author** Enrico Crema [aut, cre] (<<https://orcid.org/0000-0001-6727-5138>>),  
Robert Di Napoli [ctb] (<<https://orcid.org/0000-0003-2180-2195>>)

**Maintainer** Enrico Crema <[enrico.crema@gmail.com](mailto:enrico.crema@gmail.com)>

**Repository** CRAN

**Date/Publication** 2022-05-09 15:00:02 UTC

## R topics documented:

agreementIndex . . . . .	2
compare.models . . . . .	3
dAsymLaplace . . . . .	3
dDoubleExponentialGrowth . . . . .	4
dExponentialGrowth . . . . .	5

dExponentialLogisticGrowth . . . . .	6
dLogisticExponentialGrowth . . . . .	7
dLogisticGrowth . . . . .	8
dTrapezoidal . . . . .	9
intcal20 . . . . .	10
interpLin . . . . .	10
marine20 . . . . .	11
modelPlot . . . . .	12
plot.spdppc . . . . .	13
postHPDplot . . . . .	14
postPredCor . . . . .	15
postPredSPD . . . . .	16
shcal20 . . . . .	17

<b>Index</b>	<b>18</b>
--------------	-----------

---

agreementIndex	<i>Calculate Agreement Indices.</i>
----------------	-------------------------------------

---

### Description

Computes OxCal-style (Bronk-Ramsey 1995) individual and overall agreement index for evaluating model consistency.

### Usage

```
agreementIndex(CRA, CRAError, calCurve = "intcal20", theta, verbose = TRUE)
```

### Arguments

CRA	vector of C14 ages.
CRAError	vector of C14 errors associated with CRA.
calCurve	character string naming a calibration curve, one between 'intcal20', 'intcal13', 'shcal20', 'shcal13', 'marine13' and 'marine20'.
theta	a Matrix containing the posterior samples of each date.
verbose	a logical variable indicating whether extra information on progress should be reported. Default is TRUE.

### Value

a list containing the individual and overall agreement indices.

### References

Bronk-Ramsey, C. (1995). Radiocarbon Calibration and Analysis of Stratigraphy: The OxCal Program. *Radiocarbon*, 37(2), 425–430.

---

compare.models	<i>WAIC-based model comparison</i>
----------------	------------------------------------

---

**Description**

Compute delta WAIC and WAIC weights for model comparison.

**Usage**

```
compare.models(...)
```

**Arguments**

... MCMC output from either [nimbleMCMC](#) or [runMCMC](#) functions in the nimble R package. Note that in argument WAIC should be set to TRUE.

**Value**

A table containing WAIC, delta WAIC, and WAIC weights.

---

dAsymLaplace	<i>Asymmetric Laplace Distribution</i>
--------------	--

---

**Description**

Density, distribution function, quantile function, and random generation for a Asymmetric Laplace Distribution.

**Usage**

```
dAsymLaplace(x, mu, sigma, tau, log)
```

```
rAsymLaplace(n, mu, sigma, tau)
```

```
pAsymLaplace(q, mu, sigma, tau, lower.tail = 1, log.p = 0)
```

```
qAsymLaplace(p, mu, sigma, tau, lower.tail = 1, log.p = 0)
```

**Arguments**

x value to be computed.

mu location parameter.

sigma scale parameter.

tau asymmetry parameter.

log, log.p TRUE or 1 to return log probability. FALSE or 0 to return probability.

n	number of random draws. Currently only $n = 1$ is supported, but the argument exists for standardization of "r" functions.
q	quantile to be computed.
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ otherwise, $P[X > x]$ .
p	probability to be computed.

**Author(s)**

Enrico Crema

---

dDoubleExponentialGrowth

*Double Exponential Growth Model*

---

**Description**

Density and random generation of an exponential growth model distribution.

**Usage**

`dDoubleExponentialGrowth(x, a, b, r1, r2, mu, log)`

`rDoubleExponentialGrowth(n, a, b, r1, r2, mu)`

**Arguments**

x	vector of calendar years (in BP).
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
r1	growth rate before change point mu.
r2	growth rate after change point mu.
mu	change point (in BP).
log	TRUE or 1 to return log probability. FALSE or 0 to return probability.
n	number of random draws. Currently only $n = 1$ is supported, but the argument exists for standardization of "r" functions.

**Value**

For `dDoubleExponentialGrowth`: the probability (or likelihood) or log probability of an observed date x (in Cal BP). For `rDoubleExponentialGrowth` a simulated date in Cal BP.

**Author(s)**

Enrico Crema

**Examples**

```
p = list(r1=0.003,r2=-0.001,mu=5200)
modelPlot(model = dDoubleExponentialGrowth,a=6000,b=4000,params=p,alpha = 1)
```

---

dExponentialGrowth      *Exponential Growth Model*

---

**Description**

Density and random generation of an exponential growth model distribution.

**Usage**

```
dExponentialGrowth(x, a, b, r, log)
rExponentialGrowth(n, a, b, r)
```

**Arguments**

x	vector of calendar years (in BP).
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
r	intrinsic growth rate.
log	TRUE or 1 to return log probability. FALSE or 0 to return probability.
n	number of random draws. Currently only n = 1 is supported, but the argument exists for standardization of "r" functions.

**Value**

For dExponentialGrowth: the probability (or likelihood) or log probability of an observed date x (in Cal BP). For rExponentialGrowth a simulated date in Cal BP.

**Author(s)**

Enrico Crema

**Examples**

```
p = list(r=0.002)
modelPlot(model = dExponentialGrowth,a=6000,b=4000,params=p,alpha = 1)
```

---

dExponentialLogisticGrowth

*Exponential-Logistic Growth Model*


---

### Description

Density and random generation of a exponential-logistic growth model distribution.

### Usage

```
dExponentialLogisticGrowth(x, a, b, k, r1, r2, mu, log)
```

```
rExponentialLogisticGrowth(n, a, b, k, r1, r2, mu)
```

### Arguments

x	vector of calendar years (in BP).
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
k	initial proportion of the carrying capacity (must be between 0 and 1).
r1	growth rate of the exponential phase.
r2	growth rate of logistic phase.
mu	change point (in BP).
log	TRUE or 1 to return log probability. FALSE or 0 to return probability.
n	number of random draws. Currently only n = 1 is supported, but the argument exists for standardization of "r" functions.

### Value

For dExponentialLogisticGrowth: the probability (or likelihood) or log probability of an observed date x (in Cal BP). For rExponentialLogisticGrowth a simulated date in Cal BP.

### Author(s)

Enrico Crema

### Examples

```
p = list(r1=-0.001,r2=0.01,mu=5200,k=0.2)
modelPlot(model = dExponentialLogisticGrowth,a=6000,b=4000,params=p,alpha = 1)
```

---

`dLogisticExponentialGrowth`*Logistic-Exponential Growth Model*

---

**Description**

Density and random generation of a logistic-exponential growth model distribution.

**Usage**

```
dLogisticExponentialGrowth(x, a, b, r1, r2, k, mu, log)
```

```
rLogisticExponentialGrowth(n, a, b, r1, r2, k, mu)
```

**Arguments**

<code>x</code>	vector of calendar years (in BP).
<code>a</code>	lower (earliest) limit of the distribution (in BP).
<code>b</code>	upper (latest) limit of the distribution (in BP).
<code>r1</code>	growth rate of the logistic phase.
<code>r2</code>	growth rate of exponential phase.
<code>k</code>	initial proportion of the carrying capacity (must be between 0 and 1).
<code>mu</code>	change point (in BP).
<code>log</code>	TRUE or 1 to return log probability. FALSE or 0 to return probability.
<code>n</code>	number of random draws. Currently only <code>n = 1</code> is supported, but the argument exists for standardization of "r" functions.

**Value**

For `dLogisticExponentialGrowth`: the probability (or likelihood) or log probability of an observed date `x` (in Cal BP). For `rLogisticExponentialGrowth` a simulated date in Cal BP.

**Author(s)**

Robert DiNapoli & Enrico Crema

**Examples**

```
p = list(r1=0.01,r2=-0.001,k=0.001,mu=4500)
modelPlot(model = dLogisticExponentialGrowth,a=6000,b=4000,params=p,alpha = 1)
```

---

dLogisticGrowth      *Logistic Growth Model*

---

### Description

Density and random generation of a logistic growth model distribution.

### Usage

```
dLogisticGrowth(x, a, b, k, r, log)
```

```
rLogisticGrowth(n, a, b, k, r)
```

### Arguments

x	vector of calendar years (in BP).
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
k	initial proportion of the carrying capacity (must be between 0 and 1).
r	intrinsic growth rate.
log	TRUE or 1 to return log probability. FALSE or 0 to return probability.
n	number of random draws. Currently only n = 1 is supported, but the argument exists for standardization of "r" functions.

### Value

For dLogisticGrowth: the probability (or likelihood) or log probability of an observed date x (in Cal BP). For rLogisticGrowth a simulated date in Cal BP.

### Author(s)

Enrico Crema

### Examples

```
p = list(k=0.01,r=0.007)
modelPlot(model = dLogisticGrowth,a=6000,b=4000,params=p,alpha = 1)
```



---

dTrapezoidal	<i>Trapezoidal Distribution</i>
--------------	---------------------------------

---

**Description**

Density and random generation of an Trapezoidal distribution.

**Usage**

```
dTrapezoidal(x, a, m1, m2, b, log)
```

```
rTrapezoidal(n, a, m1, m2, b)
```

**Arguments**

x	A calendar year (in BP).
a	lower (earliest) limit of the distribution (in BP).
m1	lower mode (in BP)
m2	upper mode (in BP).
b	upper (latest) limit of the distribution (in BP).
log	TRUE or 1 to return log probability. FALSE or 0 to return probability.
n	number of random draws. Currently only n = 1 is supported, but the argument exists for standardization of "r" functions.

**Author(s)**

Enrico Crema

**Examples**

```
a=7000  
b=6700  
c=4000  
d=3000  
x=5400  
modelPlot(dTrapezoidal,a=7000,b=5000,params=c(m1=6000,m2=5300),alpha=1,col=1)
```

intcal20                      *IntCal20 radiocarbon age calibration curve for the Northern hemisphere.*

---

**Description**

IntCal20 radiocarbon age calibration curve for the Northern hemisphere.

**Usage**

intcal20

**Format**

A data.frame with the following fields:

CalBP ID of each radiocarbon date

C14Age Radiocarbon age in <sup>14</sup>C years BP

C14Age.sigma Radiocarbon age error

Delta14C Labcode of the radiocarbon date

Delta14C.sigma Material of the dated sample

**Source**

<https://intcal.org/curves/intcal20.14c>

**References**

Reimer, P. J., Austin, W. E. N., Bard, E., Bayliss, A., Blackwell, P. G., Ramsey, C. B., et al. (2020). The IntCal20 Northern Hemisphere Radiocarbon Age Calibration Curve (0–55 cal kBP). *Radiocarbon*, 62(4), 725–757. <https://doi.org/10.1017/RDC.2020.41>

---

interpLin                      *Linear interpolation function*

---

**Description**

A nimbleFunction emulating BUGS/JAGS's interp.lin.

**Usage**

interpLin(z, x, y)

**Arguments**

z	value where the interpolation take place
x	numeric vector giving the coordinates of the points to be interpolated.
y	numeric vector giving the coordinates of the points to be interpolated.

**Value**

interpolated value

**Examples**

```
data(intcal20)
interpLin(4500,intcal20$CalBP,intcal20$C14Age)
# equivalent to:
approx(x=intcal20$CalBP,y=intcal20$C14Age,xout=4500)$y
```

---

marine20	<i>Marine20 radiocarbon age calibration curve.</i>
----------	--

---

**Description**

Marine20 radiocarbon age calibration curve.

**Usage**

```
marine20
```

**Format**

A data.frame with the following fields:

CalBP ID of each radiocarbon date  
C14Age Radiocarbon age in 14C years BP  
C14Age.sigma Radiocarbon age error  
Delta14C Labcode of the radiocarbon date  
Delta14C.sigma Material of the dated sample

**Source**

<https://intcal.org/curves/marine20.14c>

**References**

Heaton, T. J., Köhler, P., Butzin, M., Bard, E., Reimer, R. W., Austin, W. E. N., et al. (2020). Marine20—The Marine Radiocarbon Age Calibration Curve (0–55,000 cal BP). *Radiocarbon*, 62(4), 779–820. <https://doi.org/10.1017/RDC.2020.68>

**Description**

Plots growth models based on user provided parameters for prior and posterior predictive checks.

**Usage**

```
modelPlot(
  model,
  a,
  b,
  params,
  type = c("spaghetti"),
  nsample = NULL,
  interval = 0.9,
  calendar = "BP",
  col = "lightgrey",
  alpha = 0.1,
  ylim = NULL,
  xlim = NULL,
  add = FALSE,
  ...
)
```

**Arguments**

model	growth model.
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
params	a <a href="#">list</a> of vectors containing model parameters. The names attribute of each vector should match growth model parameters.
type	either a 'spaghetti' plot or a quantile based 'envelope' plot. Default is 'spaghetti'.
nsample	number of samples to be used. Default is the length of the parameter vectors supplied in the argument params.
interval	quantile interval used for the envelope plot. Ignored when type is set to 'spaghetti'.
calendar	either 'BP' or 'BCAD'. Indicate whether the calibrated date should be displayed in BP or BC/AD. Default is 'BP'.
col	fill color for the quantile envelope (when type=='envelope') or line colour (when type=='spaghetti').
alpha	transparency value for each line in the spaghetti plot. Ignored when type is set to 'envelope'. Default is 0.1.
ylim	the y limits of the plot.

xlim	the x limits of the plot (in Cal BP).
add	whether or not the new graphic should be added to an existing plot.
...	additional arguments affecting the plot

**Value**

None.

**Examples**

```
params = list(k=runif(100,0.01,0.02),r=runif(100,0.003,0.004))
modelPlot(model=dLogisticGrowth,a=5000,b=2000,params=params,type=c('spaghetti'),alpha=0.5)
```

---

plot.spdppc	<i>Plot SPD-based Posterior Predictive Check</i>
-------------	--

---

**Description**

Plots spdppc class object for SPD-based Posterior Predictive Check.

**Usage**

```
## S3 method for class 'spdppc'
plot(
  x,
  type = "envelope",
  nsample = NULL,
  interval = 0.9,
  obs.lwd = 1.5,
  obs.col = "black",
  sim.col = "lightgrey",
  alpha = 1,
  envelope.col = "lightgrey",
  positive.col = "red",
  negative.col = "blue",
  calendar = "BP",
  ...
)
```

**Arguments**

x	An spdppc class object.
type	Either a 'spaghetti' plot or a quantile based envelope plot. Default is 'envelope'.

nsample	Number of samples to be displayed in the 'spaghetti' plot. Default is the total number of simulations supplied in the 'spdppc' class object, ignored when type is set to 'envelope'.
interval	Quantile interval used for the envelope plot. Ignored when type is set to 'spaghetti'. Default is 0.90.
obs.lwd	Line width of the observed SPD. Default is 1.5.
obs.col	Line colour of the observed SPD. Default is 'black'.
sim.col	Line colour of simulated SPDs. Default is 'lightgrey', ignored when type is set to 'envelope'.
alpha	Transparency value for each line in the spaghetti plot. Default is 1, ignored when type is set to 'envelope'.
envelope.col	Fill colour of the simulation envelope. Default is 'lightgrey', ignored when type is set to 'envelope.'spaghetti'.
positive.col	Fill colour for the area with positive deviation from the simulation envelope. Default is 'red', ignored when type is set to 'spaghetti'.
negative.col	Fill colour for the area with positive deviation from the simulation envelope. Default is 'blue', ignored when type is set to 'spaghetti'.
calendar	Either 'BP' or 'BCAD'. Indicate whether the calibrated date should be displayed in BP or BC/AD. Default is 'BP'.
...	Additional arguments affecting the plot

**Value**

None.

---

postHPDplot

---

*Plot Marginal Posterior Distribution*


---

**Description**

Plot marginal posterior distribution highlighting user-defined higher posterior density interval.

**Usage**

```
postHPDplot(
  x,
  prob = 0.9,
  bw = "SJ",
  hpd.col = "lightblue",
  line.col = "darkgrey",
  rnd = 3,
  HPD = TRUE,
  show.hpd.val = TRUE,
  ...
)
```

**Arguments**

x	Posterior samples
prob	Highest posterior density interval. Default is 0.9.
bw	The smoothing bandwidth to be used. See <a href="#">density</a> for details. Default is "SJ".
hpd.col	Fill colour for the highest density interval. Default is 'lightblue'. Ignored when HPD is set to FALSE.
line.col	Line color for the density plot. Default is 'darkgrey'.
rnd	Integer indicating the number of decimal places to be used in the reporting of the highest posterior density interval.
HPD	Whether the highest posterior density interval is highlighted or not. Default is TRUE.
show.hpd.val	Whether the highest posterior density interval is displayed as subtitle. Default is TRUE.
...	other graphical parameters.

**Value**

None.

---

postPredCor	<i>Calculates correlation between observed and posterior generated SPD.</i>
-------------	---

---

**Description**

Computes the correlation between observed SPDs and posterior generated SPD from the output of postPredSPD() function as an heuristic of the goodness-of-fit of the model.

**Usage**

```
postPredCor(x, method = "pearson")
```

**Arguments**

x	An object of class spdppc.
method	a character string indicating which correlation coefficient is to be computed. One of "pearson" (default), "kendall", or "spearman": can be abbreviated.

**Value**

A vector of correlation values.

---

 postPredSPD

*SPD-based Posterior Predictive Check*


---

### Description

Generates SPDs from posterior samples.

### Usage

```
postPredSPD(
  x,
  errors,
  calCurve,
  model,
  a,
  b,
  params,
  nsim,
  method = NULL,
  spdnormalised = TRUE,
  datenormalised = TRUE,
  ncores = 1,
  verbose = TRUE
)
```

### Arguments

x	a vector of observed uncalibrated radiocarbon ages.
errors	a vector of standard deviations corresponding to each estimated radiocarbon age.
calCurve	character string naming a calibration curve already provided with the rcarbon package (currently 'intcal20', 'intcal13', 'intcal13nhpine16', 'shcal20', 'shcal13', 'shcal13shkauri16', 'marin
model	growth model
a	lower (earliest) limit of the distribution (in BP).
b	upper (latest) limit of the distribution (in BP).
params	list of vectors containing model parameters. The names attribute of each vector should match growth model parameters.
nsim	number of SPDs to be generated. Default is the length of the parameter vectors supplied in the argument params.
method	method for the creation of random dates from the fitted model. Either 'uncal-sample' or 'calsample'.
spdnormalised	a logical variable indicating whether the total probability mass of the SPD is normalised to sum to unity for both observed and simulated data. Default is TRUE.



datenormalised	a logical variable indicating whether dates should be normalised to sum to unity or not. Default is TRUE.
ncores	number of cores used for for parallel execution. Default is 1.
verbose	a logical variable indicating whether extra information on progress should be reported. Default is TRUE.

### Value

An object of class `spdppc` with the following elements

- `obs` A `data.frame` containing the years (in Cal BP) and the corresponding summed probability in the observed data.
- `spdmat` A matrix containing the summed probability distribution of the simulated data.

---

shcal20	<i>IntCal20 radiocarbon age calibration curve for the Southern hemisphere.</i>
---------	--

---

### Description

IntCal20 radiocarbon age calibration curve for the Southern hemisphere.

### Usage

```
shcal20
```

### Format

A `data.frame` with the following fields:

`CalBP` ID of each radiocarbon date

`C14Age` Radiocarbon age in <sup>14</sup>C years BP

`C14Age.sigma` Radiocarbon age error

`Delta14C` Labcode of the radiocarbon date

`Delta14C.sigma` Material of the dated sample

### Source

<https://intcal.org/curves/shcal20.14c>

### References

Reimer, P. J., Austin, W. E. N., Bard, E., Bayliss, A., Blackwell, P. G., Ramsey, C. B., et al. (2020). The IntCal20 Northern Hemisphere Radiocarbon Age Calibration Curve (0–55 cal kBP). *Radiocarbon*, 62(4), 725–757. <https://doi.org/10.1017/RDC.2020.41>

# Index

- \* **datasets**
  - intcal20, [10](#)
  - marine20, [11](#)
  - shcal20, [17](#)
- agreementIndex, [2](#)
- compare.models, [3](#)
- dAsymLaplace, [3](#)
- dAsymLaplace, (dAsymLaplace), [3](#)
- dDoubleExponentialGrowth, [4](#)
- density, [15](#)
- dExponentialGrowth, [5](#)
- dExponentialLogisticGrowth, [6](#)
- dLogisticExponentialGrowth, [7](#)
- dLogisticGrowth, [8](#)
- dTrapezoidal, [9](#)
  
- intcal20, [10](#)
- interpLin, [10](#)
  
- list, [12](#)
  
- marine20, [11](#)
- modelPlot, [12](#)
  
- nimbleMCMC, [3](#)
  
- pAsymLaplace (dAsymLaplace), [3](#)
- plot.spdppc, [13](#)
- postHPDplot, [14](#)
- postPredCor, [15](#)
- postPredSPD, [16](#)
  
- qAsymLaplace (dAsymLaplace), [3](#)
- qAsymLaplace, (dAsymLaplace), [3](#)
  
- rAsymLaplace (dAsymLaplace), [3](#)
- rAsymLaplace, (dAsymLaplace), [3](#)
- rDoubleExponentialGrowth
  - (dDoubleExponentialGrowth), [4](#)
  
- rExponentialGrowth
  - (dExponentialGrowth), [5](#)
- rExponentialLogisticGrowth
  - (dExponentialLogisticGrowth), [6](#)
- rLogisticExponentialGrowth
  - (dLogisticExponentialGrowth), [7](#)
- rLogisticGrowth (dLogisticGrowth), [8](#)
- rTrapezoidal (dTrapezoidal), [9](#)
- runMCMC, [3](#)
  
- shcal20, [17](#)