

The `germinationmetrics` Package: A Brief Introduction

Aravind, J., Vimala Devi, S., Radhamani, J., Jacob, S. R., and Kalyani Srinivasan

2020-06-30

ICAR-National Bureau of Plant Genetic Resources, New Delhi.

Contents

Overview	1
Installation	1
Version History	2
Germination count data	2
Single-value germination indices	3
Non-linear regression analysis	28
Four-parameter hill function	28
Wrapper functions	39
Citing <code>germinationmetrics</code>	51
Session Info	51
References	52

Overview

The package `germinationmetrics` is a collection of functions which implements various methods for describing the time-course of germination in terms of single-value germination indices as well as fitted curves.

The goal of this vignette is to introduce the users to these functions and get started in describing sequentially recorded germination count data. This document assumes a basic knowledge of R programming language.



Installation

The package can be installed using the following functions:

```
# Install from CRAN
install.packages('germinationmetrics', dependencies=TRUE)

# Install development version from Github
devtools::install_github("aravind-j/germinationmetrics")
```

Then the package can be loaded using the function

```
library(germinationmetrics)
```

Version History

The current version of the package is 0.1.4. The previous versions are as follows.

Table 1. Version history of `germinationmetrics` R package.

Version	Date
0.1.0	2018-04-17
0.1.1	2018-07-26
0.1.1.1	2018-10-16
0.1.2	2018-10-31
0.1.3	2019-01-19

To know detailed history of changes use `news(package='germinationmetrics')`.

Germination count data

Typically in a germination test, the germination count data of a fixed number of seeds is recorded at regular intervals for a definite period of time or until all the seeds have germinated. These germination count data can be either partial or cumulative (Table 2).

Table 2 : A typical germination count data.

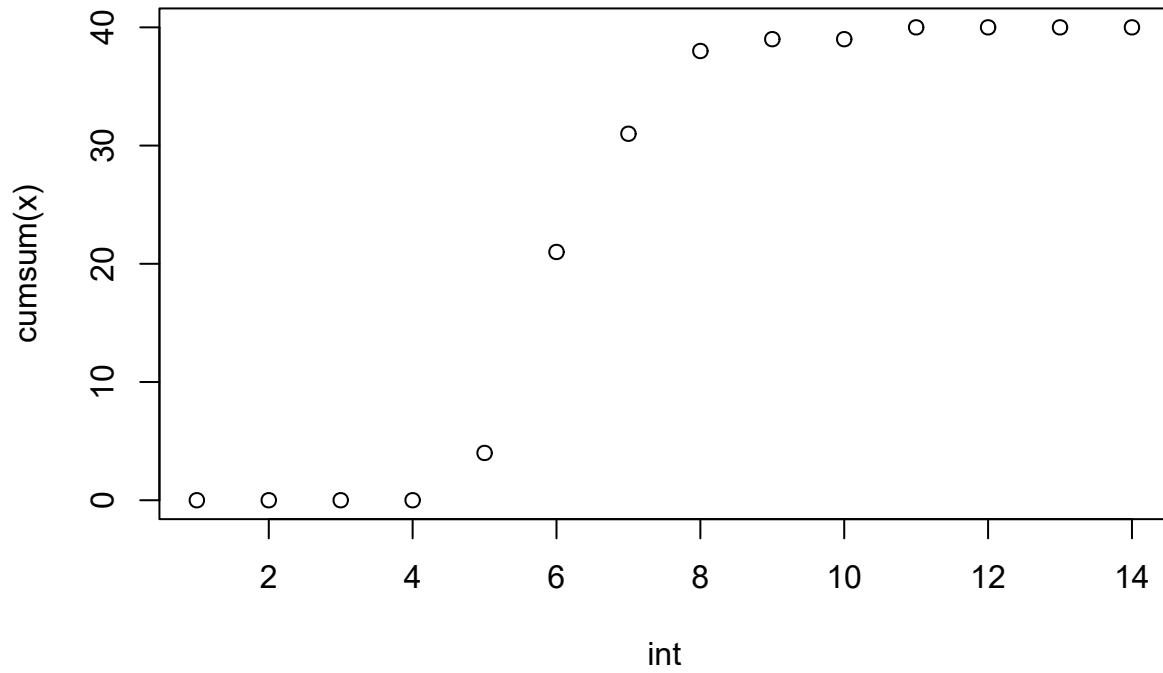
intervals	counts	cumulative.counts
1	0	0
2	0	0
3	0	0
4	0	0
5	4	4
6	17	21
7	10	31
8	7	38
9	1	39
10	0	39
11	1	40
12	0	40
13	0	40
14	0	40

The time-course of germination can be plotted as follows.

```
data <- data.frame(intervals = 1:14,
                  counts = c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0))

# Partial germination counts
x <- data$counts
# Cumulative germination counts
y <- cumsum(x)
# Time intervals of observations
int <- data$intervals

plot(int, cumsum(x))
```



Single-value germination indices

The details about the single-value germination indices implemented in `germinationmetrics` are described in Table 3.

Table 3 : Single-value germination indices implemented in `germinationmetrics`.

Germination index	Function	Details	Unit	Measures	Reference
Germination percentage or Final germination percentage or Germinability (<i>GP</i>)	GermPercent	It is computed as follows. $GP = \frac{N_g}{N_t} \times 100$ Where, N_g is the number of germinated seeds and N_t is the total number of seeds.	Percentage (%)	Germination capacity	ISTA (2015)
Time for the first germination or Germination time lag (t_0)	FirstGermTime	It is the time for first germination to occur (e.g. First day of germination)	time	Germination time	Edwards (1932); Czabator (1962); Goloff and Bazzaz (1975); Labouriau (1983a); Ranal (1999); Quintanilla et al. (2000)
Time for the last germination (t_g)	LastGermTime	It is the time for last germination to occur (e.g. Last day of germination)	time	Germination time	Edwards (1932)
Time spread of germination or Germination distribution	TimeSpreadGerm	It is the difference between time for last germination (t_g) and time for first germination (t_0). $\text{Time spread of germination} = t_g - t_0$	time	Germination time	Al-Mudaris (1998); Schrader and Graves (2000); Kader (2005)
Peak period of germination or Modal time of germination	PeakGermTime	It is the time in which highest frequency of germinated seeds are observed and need not be unique.	time	Germination time	Ranal and Santana (2006)
Median germination time (t_{50}) (Coolbear)	t50	It is the time to reach 50% of final/maximum germination. With argument <code>method</code> specified as " <code>coolbear</code> ", it is computed as follows. $t_{50} = T_i + \frac{(\frac{N+1}{2} - N_i)(T_j - T_i)}{N_j - N_i}$ Where, t_{50} is the median germination time, N is the final number of germinated seeds, and N_i and N_j are the total number of seeds germinated in adjacent counts at time T_i and T_j respectively, when $N_i < \frac{N+1}{2} < N_j$.	time	Germination time	Coolbear et al. (1984)
Median germination time (t_{50}) (Farooq)	t50	With argument <code>method</code> specified as " <code>farooq</code> ", it is computed as follows. $t_{50} = T_i + \frac{(\frac{N}{2} - N_i)(T_j - T_i)}{N_j - N_i}$ Where, t_{50} is the median germination time, N is the final number of germinated seeds, and N_i and N_j are the total number of seeds germinated in adjacent counts at time T_i and T_j respectively, when $N_i < \frac{N}{2} < N_j$.	time	Germination time	Farooq et al. (2005)

Germination index	Function	Details	Unit	Measures	Reference
Mean germination time or Mean length of incubation time (\bar{T}) or Germination resistance (GR) or Sprouting index (SI) or Emergence index (EI)	MeanGermTime	<p>It is the average length of time required for maximum germination of a seed lot and is estimated according to the following formula.</p> $\bar{T} = \frac{\sum_{i=1}^k N_i T_i}{\sum_{i=1}^k N_i}$ <p>Where, T_i is the time from the start of the experiment to the ith interval, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p> <p>It is the inverse of mean germination rate (\bar{V}).</p> $\bar{T} = \frac{1}{\bar{V}}$	time	Germination time	Edmond and Drapala (1958); Czabator (1962); Smith and Millet (1964); Gordon (1969); Gordon (1971); Mock and Eberhart (1972); Ellis and Roberts (1980) Labouriau (1983a); Ranal and Santana (2006)
Variance of germination time (s_T^2)	VarGermTime	<p>It is computed according to the following formula.</p> $s_T^2 = \frac{\sum_{i=1}^k N_i (T_i - \bar{T})^2}{\sum_{i=1}^k N_i - 1}$ <p>Where, T_i is the time from the start of the experiment to the ith interval, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p>	time	Germination time	Labouriau (1983a); Ranal and Santana (2006)
Standard error of germination time ($s_{\bar{T}}$)	SEGermTime	<p>It signifies the accuracy of the calculation of the mean germination time.</p> <p>It is estimated according to the following formula:</p> $s_{\bar{T}} = \sqrt{\frac{s_T^2}{\sum_{i=1}^k N_i}}$ <p>Where, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval) and k is the total number of time intervals.</p>	time	Germination time	Labouriau (1983a); Ranal and Santana (2006)

Germination index	Function	Details	Unit	Measures	Reference
Mean germination rate (\bar{V})	MeanGermRate	<p>It is computed according to the following formula:</p> $\bar{V} = \frac{\sum_{i=1}^k N_i}{\sum_{i=1}^k N_i T_i}$ <p>Where, T_i is the time from the start of the experiment to the ith interval, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p> <p>It is the inverse of mean germination time (\bar{T}).</p> $\bar{V} = \frac{1}{\bar{T}}$	time ⁻¹	Germination rate	Labouriau and Valadares (1976); Labouriau (1983b); Ranal and Santana (2006)
Coefficient of velocity of germination (CVG) or Coefficient of rate of germination (CRG) or Kotowski's coefficient of velocity	CVG	<p>It is estimated according to the following formula.</p> $CVG = \frac{\sum_{i=1}^k N_i}{\sum_{i=1}^k N_i T_i} \times 100$ $CVG = \bar{V} \times 100$ <p>Where, T_i is the time from the start of the experiment to the ith interval, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p>	% day ⁻¹	Germination rate	Kotowski (1926), Nichols and Heydecker (1968); Bewley and Black (1994); Labouriau (1983b); Scott et al. (1984)
Variance of germination rate (s_V^2)	VarGermRate	<p>It is calculated according to the following formula.</p> $s_V^2 = \bar{V}^4 \times s_T^2$ <p>Where, s_T^2 is the variance of germination time.</p>	time ⁻²	Germination rate	Labouriau (1983b); Ranal and Santana (2006)
Standard error of germination rate ($s_{\bar{V}}$)	SEGermRate	<p>It is estimated according to the following formula.</p> $s_{\bar{V}} = \sqrt{\frac{s_V^2}{\sum_{i=1}^k N_i}}$ <p>Where, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p>	time ⁻¹	Germination rate	Labouriau (1983b); Ranal and Santana (2006)
Germination rate as the reciprocal of the median time (v_{50})	GermRateRecip	<p>It is the reciprocal of the median germination time (t_{50}).</p> $v_{50} = \frac{1}{t_{50}}$	time ⁻¹	Germination rate	Went (1957); Labouriau (1983b); Ranal and Santana (2006)

Germination index	Function	Details	Unit	Measures	Reference
Speed of germination or Germination rate Index or index of velocity of germination or Emergence rate index (Allan, Vogel and Peterson; Erbach; Hsu and Nelson) or Germination index (AOSA)	GermSpeed	<p>It is the rate of germination in terms of the total number of seeds that germinate in a time interval. It is estimated as follows.</p> $S = \sum_{i=1}^k \frac{N_i}{T_i}$ <p>Where, T_i is the time from the start of the experiment to the ith interval, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals. Instead of germination counts, germination percentages may also be used for computation of speed of germination.</p>	% time ⁻¹	Mixed	Throneberry and Smith (1955); Maguire (1962); Allan et al. (1962); Kendrick and Frankland (1969); Bouton et al. (1976); Erbach (1982); AOSA (1983); Khandakar and Bradbeer (1983); Hsu and Nelson (1986); Bradbeer (1988); Wardle et al. (1991)
Speed of accumulated germination	GermSpeedAccumulated	<p>It is the rate of germination in terms of the accumulated/cumulative total number of seeds that germinate in a time interval. It is estimated as follows.</p> $S_{accumulated} = \sum_{i=1}^k \frac{\sum_{j=1}^i N_j}{T_i}$ <p>Where, T_i is the time from the start of the experiment to the ith interval, $\sum_{j=1}^i N_j$ is the cumulative/accumulated number of seeds germinated in the ith interval, and k is the total number of time intervals. Instead of germination counts, germination percentages may also be used for computation of speed of germination.</p>	% time ⁻¹	Mixed	Bradbeer (1988); Wardle et al. (1991); Haugland and Brandsaeter (1996); Santana and Ranal (2004)
Corrected germination rate index	GermSpeedCorrected	<p>It is computed as follows.</p> $S_{corrected} = \frac{S}{FGP}$ <p>Where, FGP is the final germination percentage or germinability.</p>	time ⁻¹	Mixed	Evetts and Burnside (1972)
Weighted germination percentage (WGP)	WeightGermPercent	<p>It is estimated as follows.</p> $WGP = \frac{\sum_{i=1}^k (k - i + 1)N_i}{k \times N} \times 100$ <p>Where, N_i is the number of seeds that germinated in the time interval i (not cumulative, but partial count), N is the total number of seeds tested, and k is the total number of time intervals.</p>		Mixed	Reddy et al. (1985); Reddy (1978)

Germination index	Function	Details	Unit	Measures	Reference
Mean germination percentage per unit time (\overline{GP})	MeanGermPercent	<p>It is estimated as follows.</p> $\overline{GP} = \frac{GP}{T_k}$ <p>Where, GP is the final germination percentage, T_k is the time at the kth time interval, and k is the total number of time intervals required for final germination.</p>		Mixed	Czabator (1962)
Number of seeds germinated per unit time \overline{N}	MeanGermNumber	<p>It is estimated as follows.</p> $\overline{N} = \frac{N_g}{T_k}$ <p>Where, N_g is the number of germinated seeds at the end of the germination test, T_k is the time at the kth time interval, and k is the total number of time intervals required for final germination.</p>		Mixed	Khamassi et al. (2013)
Timson's index [\sum 10 (Ten summation), \sum 5 or \sum 20] or Germination energy index (GEI)	TimsonsIndex	<p>It is the progressive total of cumulative germination percentage recorded at specific intervals for a set period of time and is estimated in terms of cumulative germination percentage (G_i) as follows.</p> $\Sigma k = \sum_{i=1}^k G_i$ <p>Where, G_i is the cumulative germination percentage in time interval i, and k is the total number of time intervals. It also estimated in terms of partial germination percentage as follows.</p> $\Sigma k = \sum_{i=1}^k g_i(k - j)$ <p>Where, g_i is the germination (not cumulative, but partial germination) in time interval i (i varying from 0 to k), k is the total number of time intervals, and $j = i - 1$.</p>		Mixed	Grose and Zimmer (1958); Timson (1965); Lyon and Coffelt (1966); Chaudhary and Ghildyal (1970); Negm and Smith (1978); Brown and Mayer (1988); Baskin and Baskin (1998); Goodchild and Walker (1971)
Modified Timson's index (Σk_{mod}) (Labouriau)	TimsonsIndex	<p>It is estimated as Timson's index Σk divided by the sum of partial germination percentages.</p> $\Sigma k_{mod} = \frac{\Sigma k}{\sum_{i=1}^k g_i}$		Mixed	Ranal and Santana (2006)

Germination index	Function	Details	Unit	Measures	Reference
Modified Timson's index (Σk_{mod}) (Khan and Unger)	TimsonsIndex	It is estimated as Timson's index (Σk) divided by the total time period of germination (T_k). $\Sigma k_{mod} = \frac{\Sigma k}{T_k}$		Mixed	Khan and Ungar (1984)
George's index (GR)	GermRateGeorge	It is estimated as follows. $GR = \sum_{i=1}^k N_i K_i$ Where N_i is the number of seeds germinated by i th interval and K_i is the number of intervals(eg. days) until the end of the test, and k is the total number of time intervals.		Mixed	George (1961); Tucker and Wright (1965); Nichols and Heydecker (1968)
Germination Index (GI) (Melville)	GermIndex	It is estimated as follows. $GI = \sum_{i=1}^k \frac{ (T_k - T_i) N_i }{N_t}$ Where, T_i is the time from the start of the experiment to the i th interval (day for the example), N_i is the number of seeds germinated in the i th time interval (not the accumulated number, but the number corresponding to the i th interval), N_t is the total number of seeds used in the test, and k is the total number of time intervals.		Mixed	Melville et al. (1980)
Germination Index (GI_{mod}) (Melville; Santana and Ranal)	GermIndex	It is estimated as follows. $GI_{mod} = \sum_{i=1}^k \frac{ (T_k - T_i) N_i }{N_g}$ Where, T_i is the time from the start of the experiment to the i th interval (day for the example), N_i is the number of seeds germinated in the i th time interval (not the accumulated number, but the number corresponding to the i th interval), N_g is the total number of germinated seeds at the end of the test, and k is the total number of time intervals.		Mixed	Melville et al. (1980); Santana and Ranal (2004); Ranal and Santana (2006)

Germination index	Function	Details	Unit	Measures	Reference
Emergence Rate Index (ERI) or Germination Rate Index (Shmueli and Goldberg)	EmergenceRateIndex	<p>It is estimated as follows.</p> $ERI = \sum_{i=i_0}^{k-1} N_i(k-i)$ <p>Where, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), i_0 is the time interval when emergence/germination started, and k is the total number of time intervals.</p>		Mixed	Shmueli and Goldberg (1971)
Modified Emergence Rate Index (ERI_{mod}) or Modified Germination Rate Index (Shmueli and Goldberg; Santana and Ranal)	EmergenceRateIndex	<p>It is estimated by dividing Emergence rate index (ERI) by total number of emerged seedlings (or germinated seeds).</p> $ERI_{mod} = \frac{\sum_{i=i_0}^{k-1} N_i(k-i)}{N_g} = \frac{ERI}{N_g}$ <p>Where, N_g is the total number of germinated seeds at the end of the test, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), i_0 is the time interval when emergence/germination started, and k is the total number of time intervals.</p>		Mixed	Shmueli and Goldberg (1971); Santana and Ranal (2004); Ranal and Santana (2006)
Emergence Rate Index (ERI) or Germination Rate Index (Bilbro & Wanjura)	EmergenceRateIndex	<p>It is the estimated as follows.</p> $ERI = \frac{\sum_{i=1}^k N_i}{\bar{T}} = \frac{N_g}{\bar{T}}$ <p>Where, N_g is the total number of germinated seeds at the end of the test, N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and \bar{T} is the mean germination time or mean emergence time.</p>		Mixed	Bilbro and Wanjura (1982)
Emergence Rate Index (ERI) or Germination Rate Index (Fakorede)	EmergenceRateIndex	<p>It is estimated as follows.</p> $ERI = \frac{\bar{T}}{FGP/100}$ <p>Where, \bar{T} is the Mean germination time and FGP is the final germination time.</p>		Mixed	Fakorede and Ayoola (1980); Fakorede and Ojo (1981); Fakorede and Agbana (1983)

Germination index	Function	Details	Unit	Measures	Reference
Peak value(<i>PV</i>) (Czabator) or Emergence Energy (<i>EE</i>)	PeakValue	<p>It is the accumulated number of seeds germinated at the point on the germination curve at which the rate of germination starts to decrease. It is computed as the maximum quotient obtained by dividing successive cumulative germination values by the relevant incubation time.</p> $PV = \max \left(\frac{G_1}{T_1}, \frac{G_2}{T_2}, \dots, \frac{G_k}{T_k} \right)$ <p>Where, T_i is the time from the start of the experiment to the ith interval, G_i is the cumulative germination percentage in the ith time interval, and k is the total number of time intervals.</p>	% time ⁻¹	Mixed	Czabator (1962); Bonner (1967)
Germination value (<i>GV</i>) (Czabator)	GermValue	<p>It is computed as follows.</p> $GV = PV \times MDG$ <p>Where, PV is the peak value and MDG is the mean daily germination percentage from the onset of germination. It can also be computed for other time intervals of successive germination counts, by replacing MDG with the mean germination percentage per unit time (GP). GV value can be modified (GV_{mod}), to consider the entire duration from the beginning of the test instead of just from the onset of germination.</p>		Mixed	Czabator (1962); Brown and Mayer (1988)
Germination value (<i>GV</i>) (Diavanshir and Pourbiek)	GermValue	<p>It is computed as follows.</p> $GV = \frac{\sum DGS}{N} \times GP \times c$ <p>Where, DGS is the daily germination speed computed by dividing cumulative germination percentage by the number of days since the onset of germination, N is the frequency or number of DGS calculated during the test, GP is the germination percentage expressed over 100, and c is a constant. The value of c is decided on the basis of average daily speed of germination ($\frac{\sum DGS}{N}$). If it is less than 10, then c value of 10 can be used and if it is more than 10, then value of 7 or 8 can be used for c. GV value can be modified (GV_{mod}), to consider the entire duration from the beginning of the test instead of just from the onset of germination.</p>		Mixed	Djavanshir and Pourbeik (1976); Brown and Mayer (1988)

Germination index	Function	Details	Unit	Measures	Reference
Coefficient of uniformity of germination (CUG)	CUGerm	<p>It is computed as follows.</p> $CUG = \frac{\sum_{i=1}^k N_i}{\sum_{i=1}^k (\bar{T} - T_i)^2 N_i}$ <p>Where, \bar{T} is the mean germination time, T_i is the time from the start of the experiment to the ith interval (day for the example), N_i is the number of seeds germinated in the ith time interval (not the accumulated number, but the number corresponding to the ith interval), and k is the total number of time intervals.</p>		Germination uniformity	Heydecker (1972); Bewley and Black (1994)
Coefficient of variation of the germination time (CV_T)	CVGermTime	<p>It is estimated as follows.</p> $CV_T = \sqrt{\frac{s_T^2}{\bar{T}}}$ <p>Where, s_T^2 is the variance of germination time and \bar{T} is the mean germination time.</p>		Germination uniformity	Gomes (1960); Ranal and Santana (2006)
Synchronization index (\bar{E}) or Uncertainty of the germination process (U) or informational entropy (H)	GermUncertainty	<p>It is estimated as follows.</p> $\bar{E} = - \sum_{i=1}^k f_i \log_2 f_i$ <p>Where, f_i is the relative frequency of germination ($f_i = \frac{N_i}{\sum_{i=1}^k N_i}$), N_i is the number of seeds germinated on the ith time interval, and k is the total number of time intervals.</p>	bit	Germination synchrony	Shannon (1948); Labouriau and Valadares (1976); Labouriau (1983b)
Synchrony of germination (Z index)	GermSynchrony	<p>It is computed as follows.</p> $Z = \frac{\sum_{i=1}^k C_{N_i,2}}{C_{\Sigma N_i,2}}$ <p>Where, $C_{N_i,2}$ is the partial combination of the two germinated seeds from among N_i, the number of seeds germinated on the ith time interval (estimated as $C_{N_i,2} = \frac{N_i(N_i-1)}{2}$), and $C_{\Sigma N_i,2}$ is the partial combination of the two germinated seeds from among the total number of seeds germinated at the final count, assuming that all seeds that germinated did so simultaneously.</p>		Germination synchrony	Primack (1985); Ranal and Santana (2006)

Examples

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
```

```
# From partial germination counts
GermPercent(germ.counts = x, total.seeds = 50)
```

```
GermPercent()
```

```
[1] 80
```

```
# From cumulative germination counts
GermPercent(germ.counts = y, total.seeds = 50, partial = FALSE)
```

```
[1] 80
```

```
# From number of germinated seeds
GermPercent(germinated.seeds = 40, total.seeds = 50)
```

```
[1] 80
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
z <- c(0, 0, 0, 0, 11, 11, 9, 7, 1, 0, 1, 0, 0, 0)
int <- 1:length(x)
```

```
# From partial germination counts
#-----
FirstGermTime(germ.counts = x, intervals = int)
```

```
FirstGermTime(), LastGermTime(), PeakGermTime(), TimeSpreadGerm()
```

```
[1] 5
```

```
LastGermTime(germ.counts = x, intervals = int)
```

```
[1] 11
```

```
TimeSpreadGerm(germ.counts = x, intervals = int)
```

```
[1] 6
```

```
PeakGermTime(germ.counts = x, intervals = int)
```

```
[1] 6
```

```
# For multiple peak germination times
PeakGermTime(germ.counts = z, intervals = int)
```

```
Warning in PeakGermTime(germ.counts = z, intervals = int): Multiple peak germination times exist.
```

```
[1] 5 6
```

```
# From cumulative germination counts
#-----
FirstGermTime(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 5
```

```
LastGermTime(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 11
```

```
TimeSpreadGerm(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 6
```

```
PeakGermTime(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 6
```

```
# For multiple peak germination time
```

```
PeakGermTime(germ.counts = cumsum(z), intervals = int, partial = FALSE)
```

Warning in PeakGermTime(germ.counts = cumsum(z), intervals = int, partial = FALSE): Multiple peak germination times exist.

```
[1] 5 6
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
t50(germ.counts = x, intervals = int, method = "coolbear")
```

```
t50()
```

```
[1] 5.970588
```

```
t50(germ.counts = x, intervals = int, method = "farooq")
```

```
[1] 5.941176
```

```
# From cumulative germination counts
```

```
#-----
t50(germ.counts = y, intervals = int, partial = FALSE, method = "coolbear")
```

```
[1] 5.970588
```

```
t50(germ.counts = y, intervals = int, partial = FALSE, method = "farooq")
```

```
[1] 5.941176
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
MeanGermTime(germ.counts = x, intervals = int)
```

```
MeanGermTime(), VarGermTime(), SEGermTime(), CVGermTime()
```

```
[1] 6.7
```

```

VarGermTime(germ.counts = x, intervals = int)

[1] 1.446154
SEGermTime(germ.counts = x, intervals = int)

[1] 0.1901416
CVGermTime(germ.counts = x, intervals = int)

[1] 0.1794868
# From cumulative germination counts
#-----
MeanGermTime(germ.counts = y, intervals = int, partial = FALSE)

[1] 6.7
VarGermTime(germ.counts = y, intervals = int, partial = FALSE)

[1] 19.04012
SEGermTime(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.2394781
CVGermTime(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.6512685

x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)

# From partial germination counts
#-----
MeanGermRate(germ.counts = x, intervals = int)

MeanGermRate(), CVG(), VarGermRate(), SEGermRate(), GermRateRecip()

[1] 0.1492537
CVG(germ.counts = x, intervals = int)

[1] 14.92537
VarGermRate(germ.counts = x, intervals = int)

[1] 0.0007176543
SEGermRate(germ.counts = x, intervals = int)

[1] 0.004235724
GermRateRecip(germ.counts = x, intervals = int, method = "coolbear")

[1] 0.1674877
GermRateRecip(germ.counts = x, intervals = int, method = "farooq")

[1] 0.1683168

```

```

# From cumulative germination counts
#-----
MeanGermRate(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.1492537
CVG(germ.counts = y, intervals = int, partial = FALSE)

[1] 14.92537
VarGermRate(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.009448666
SEGermRate(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.005334776
GermRateRecip(germ.counts = y, intervals = int,
              method = "coolbear", partial = FALSE)

[1] 0.1674877
GermRateRecip(germ.counts = y, intervals = int,
              method = "farooq", partial = FALSE)

[1] 0.1683168

x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)

# From partial germination counts
#-----
GermSpeed(germ.counts = x, intervals = int)

GermSpeed(), GermSpeedAccumulated(), GermSpeedCorrected()

[1] 6.138925
GermSpeedAccumulated(germ.counts = x, intervals = int)

[1] 34.61567
GermSpeedCorrected(germ.counts = x, intervals = int, total.seeds = 50,
                  method = "normal")

[1] 0.07673656
GermSpeedCorrected(germ.counts = x, intervals = int, total.seeds = 50,
                  method = "accumulated")

[1] 0.4326958

# From partial germination counts (with percentages instead of counts)
#-----
GermSpeed(germ.counts = x, intervals = int,
          percent = TRUE, total.seeds = 50)

[1] 12.27785

```



```
GermSpeedAccumulated(germ.counts = x, intervals = int,
                      percent = TRUE, total.seeds = 50)
```

```
[1] 69.23134
```

```
# From cumulative germination counts
```

```
#-----
```

```
GermSpeed(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 6.138925
```

```
GermSpeedAccumulated(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 34.61567
```

```
GermSpeedCorrected(germ.counts = y, intervals = int,
                    partial = FALSE, total.seeds = 50, method = "normal")
```

```
[1] 0.07673656
```

```
GermSpeedCorrected(germ.counts = y, intervals = int,
                    partial = FALSE, total.seeds = 50, method = "accumulated")
```

```
[1] 0.4326958
```

```
# From cumulative germination counts (with percentages instead of counts)
```

```
#-----
```

```
GermSpeed(germ.counts = y, intervals = int, partial = FALSE,
           percent = TRUE, total.seeds = 50)
```

```
[1] 12.27785
```

```
GermSpeedAccumulated(germ.counts = y, intervals = int, partial = FALSE,
                      percent = TRUE, total.seeds = 50)
```

```
[1] 69.23134
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
```

```
GermSpeed(germ.counts = x, intervals = int)
```

```
GermSpeed(), GermSpeedAccumulated(), GermSpeedCorrected()
```

```
[1] 6.138925
```

```
GermSpeedAccumulated(germ.counts = x, intervals = int)
```

```
[1] 34.61567
```

```
GermSpeedCorrected(germ.counts = x, intervals = int, total.seeds = 50,
                    method = "normal")
```

```
[1] 0.07673656
```

```
GermSpeedCorrected(germ.counts = x, intervals = int, total.seeds = 50,
                    method = "accumulated")
```

```
[1] 0.4326958
```

```
# From partial germination counts (with percentages instead of counts)
#-----
GermSpeed(germ.counts = x, intervals = int,
           percent = TRUE, total.seeds = 50)
```

```
[1] 12.27785
```

```
GermSpeedAccumulated(germ.counts = x, intervals = int,
                     percent = TRUE, total.seeds = 50)
```

```
[1] 69.23134
```

```
# From cumulative germination counts
```

```
#-----
GermSpeed(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 6.138925
```

```
GermSpeedAccumulated(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 34.61567
```

```
GermSpeedCorrected(germ.counts = y, intervals = int,
                   partial = FALSE, total.seeds = 50, method = "normal")
```

```
[1] 0.07673656
```

```
GermSpeedCorrected(germ.counts = y, intervals = int,
                   partial = FALSE, total.seeds = 50, method = "accumulated")
```

```
[1] 0.4326958
```

```
# From cumulative germination counts (with percentages instead of counts)
#-----
GermSpeed(germ.counts = y, intervals = int, partial = FALSE,
           percent = TRUE, total.seeds = 50)
```

```
[1] 12.27785
```

```
GermSpeedAccumulated(germ.counts = y, intervals = int, partial = FALSE,
                     percent = TRUE, total.seeds = 50)
```

```
[1] 69.23134
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
WeightGermPercent(germ.counts = x, total.seeds = 50, intervals = int)
```

```
WeightGermPercent()
```

```
[1] 47.42857
```

```
# From cumulative germination counts
```

```
#-----
```

```
WeightGermPercent(germ.counts = y, total.seeds = 50, intervals = int,
                  partial = FALSE)
```

```
[1] 47.42857
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
```

```
MeanGermPercent(germ.counts = x, total.seeds = 50, intervals = int)
```

```
MeanGermPercent(), MeanGermNumber()
```

```
[1] 5.714286
```

```
MeanGermNumber(germ.counts = x, intervals = int)
```

```
[1] 2.857143
```

```
# From cumulative germination counts
```

```
#-----
```

```
MeanGermPercent(germ.counts = y, total.seeds = 50, intervals = int, partial = FALSE)
```

```
[1] 5.714286
```

```
MeanGermNumber(germ.counts = y, intervals = int, partial = FALSE)
```

```
[1] 2.857143
```

```
# From number of germinated seeds
```

```
#-----
```

```
MeanGermPercent(germinated.seeds = 40, total.seeds = 50, intervals = int)
```

```
[1] 5.714286
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
```

```
# Without max specified
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50)
```

```
TimsonsIndex(), GermRateGeorge()
```

```
[1] 664
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              modification = "none")
```

```
[1] 664
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              modification = "labouriau")
```

```
[1] 8.3
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              modification = "khanungar")
```

```
[1] 47.42857
```

```
GermRateGeorge(germ.counts = x, intervals = int)
```

```
[1] 332
```

```
# With max specified
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50, max = 10)
```

```
[1] 344
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              max = 10, modification = "none")
```

```
[1] 344
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              max = 10, modification = "labouriau")
```

```
[1] 4.410256
```

```
TimsonsIndex(germ.counts = x, intervals = int, total.seeds = 50,
              max = 10, modification = "khanungar")
```

```
[1] 24.57143
```

```
GermRateGeorge(germ.counts = x, intervals = int, max = 10)
```

```
[1] 172
```

```
GermRateGeorge(germ.counts = x, intervals = int, max = 14)
```

```
[1] 332
```

```
# From cumulative germination counts
```

```
#-----
```

```
# Without max specified
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50)
```

```
[1] 664
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              modification = "none")
```

```
[1] 664
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              modification = "labouriau")
```

```
[1] 8.3
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              modification = "khanungar")
```

```
[1] 47.42857
```

```
GermRateGeorge(germ.counts = y, intervals = int, partial = FALSE,)
```

```
[1] 332
```

```
# With max specified
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50, max = 10)
```

```
[1] 344
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              max = 10, modification = "none")
```

```
[1] 344
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              max = 10, modification = "labouriau")
```

```
[1] 4.410256
```

```
TimsonsIndex(germ.counts = y, intervals = int, partial = FALSE,
              total.seeds = 50,
              max = 10, modification = "khanungar")
```

```
[1] 24.57143
```

```
GermRateGeorge(germ.counts = y, intervals = int, partial = FALSE,
                max = 10)
```

```
[1] 172
```

```
GermRateGeorge(germ.counts = y, intervals = int, partial = FALSE,
                max = 14)
```

```
[1] 332
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
```

```
#-----
```

```
GermIndex(germ.counts = x, intervals = int, total.seeds = 50)
```

```
GermIndex()
```

```
[1] 5.84
```

```
GermIndex(germ.counts = x, intervals = int, total.seeds = 50,
          modification = "none")
```

```
[1] 5.84
```

```
GermIndex(germ.counts = x, intervals = int, total.seeds = 50,
          modification = "santanaranal")
```

```
[1] 7.3
```

```
# From cumulative germination counts
#-----
GermIndex(germ.counts = y, intervals = int, partial = FALSE,
           total.seeds = 50)
```

```
[1] 5.84
```

```
GermIndex(germ.counts = y, intervals = int, partial = FALSE,
           total.seeds = 50,
           modification = "none")
```

```
[1] 5.84
```

```
GermIndex(germ.counts = y, intervals = int, partial = FALSE,
           total.seeds = 50,
           modification = "santanaranal")
```

```
[1] 7.3
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
```

```
# From partial germination counts
#-----
EmergenceRateIndex(germ.counts = x, intervals = int)
```

```
EmergenceRateIndex()
```

```
[1] 292
```

```
EmergenceRateIndex(germ.counts = x, intervals = int,
                    method = "melville")
```

```
[1] 292
```

```
EmergenceRateIndex(germ.counts = x, intervals = int,
                    method = "melvillessantanaranal")
```

```
[1] 7.3
```

```
EmergenceRateIndex(germ.counts = x, intervals = int,
                    method = "bilbrowanjura")
```

```
[1] 5.970149
```

```
EmergenceRateIndex(germ.counts = x, intervals = int,
                    total.seeds = 50, method = "fakorede")
```

```
[1] 8.375
```

```
# From cumulative germination counts
#-----
EmergenceRateIndex(germ.counts = y, intervals = int, partial = FALSE,)
```

```
[1] 292
```

```
EmergenceRateIndex(germ.counts = y, intervals = int, partial = FALSE,
                    method = "melville")
```

[1] 292

```
EmergenceRateIndex(germ.counts = y, intervals = int, partial = FALSE,
  method = "melvillesantanaranal")
```

[1] 7.3

```
EmergenceRateIndex(germ.counts = y, intervals = int, partial = FALSE,
  method = "bilbrowanjura")
```

[1] 5.970149

```
EmergenceRateIndex(germ.counts = y, intervals = int, partial = FALSE,
  total.seeds = 50, method = "fakorede")
```

[1] 8.375

```
x <- c(0, 0, 34, 40, 21, 10, 4, 5, 3, 5, 8, 7, 7, 6, 6, 4, 0, 2, 0, 2)
y <- c(0, 0, 34, 74, 95, 105, 109, 114, 117, 122, 130, 137, 144, 150,
  156, 160, 160, 162, 162, 164)
int <- 1:length(x)
total.seeds = 200
```

```
# From partial germination counts
```

```
#-----
PeakValue(germ.counts = x, intervals = int, total.seeds = 200)
```

```
PeakValue(), GermValue()
```

[1] 9.5

```
GermValue(germ.counts = x, intervals = int, total.seeds = 200,
  method = "czabator")
```

```
$`Germination Value`
```

[1] 38.95

[[2]]

	germ.counts	intervals	Cumulative.germ.counts	Cumulative.germ.percent	DGS
3	34	3	34	17.0	5.666667
4	40	4	74	37.0	9.250000
5	21	5	95	47.5	9.500000
6	10	6	105	52.5	8.750000
7	4	7	109	54.5	7.785714
8	5	8	114	57.0	7.125000
9	3	9	117	58.5	6.500000
10	5	10	122	61.0	6.100000
11	8	11	130	65.0	5.909091
12	7	12	137	68.5	5.708333
13	7	13	144	72.0	5.538462
14	6	14	150	75.0	5.357143
15	6	15	156	78.0	5.200000
16	4	16	160	80.0	5.000000
17	0	17	160	80.0	4.705882
18	2	18	162	81.0	4.500000
19	0	19	162	81.0	4.263158
20	2	20	164	82.0	4.100000

```
GermValue(germ.counts = x, intervals = int, total.seeds = 200,
           method = "dp", k = 10)
```

```
$`Germination Value`
```

```
[1] 53.36595
```

```
[[2]]
```

	germ.counts	intervals	Cumulative.germ.counts	Cumulative.germ.percent	DGS	SumDGSbyN	GV
3	34	3	34	17.0	5.666667	5.666667	9.633333
4	40	4	74	37.0	9.250000	7.458333	27.595833
5	21	5	95	47.5	9.500000	8.138889	38.659722
6	10	6	105	52.5	8.750000	8.291667	43.531250
7	4	7	109	54.5	7.785714	8.190476	44.638095
8	5	8	114	57.0	7.125000	8.012897	45.673512
9	3	9	117	58.5	6.500000	7.796769	45.611097
10	5	10	122	61.0	6.100000	7.584673	46.266503
11	8	11	130	65.0	5.909091	7.398497	48.090230
12	7	12	137	68.5	5.708333	7.229481	49.521942
13	7	13	144	72.0	5.538462	7.075752	50.945411
14	6	14	150	75.0	5.357143	6.932534	51.994006
15	6	15	156	78.0	5.200000	6.799262	53.034246
16	4	16	160	80.0	5.000000	6.670744	53.365948
17	0	17	160	80.0	4.705882	6.539753	52.318022
18	2	18	162	81.0	4.500000	6.412268	51.939373
19	0	19	162	81.0	4.263158	6.285850	50.915385
20	2	20	164	82.0	4.100000	6.164414	50.548194

```
$testend
```

```
[1] 16
```

```
GermValue(germ.counts = x, intervals = int, total.seeds = 200,
           method = "czabator", from.onset = FALSE)
```

```
$`Germination Value`
```

```
[1] 38.95
```

```
[[2]]
```

	germ.counts	intervals	Cumulative.germ.counts	Cumulative.germ.percent	DGS
1	0	1	0	0.0	0.000000
2	0	2	0	0.0	0.000000
3	34	3	34	17.0	5.666667
4	40	4	74	37.0	9.250000
5	21	5	95	47.5	9.500000
6	10	6	105	52.5	8.750000
7	4	7	109	54.5	7.785714
8	5	8	114	57.0	7.125000
9	3	9	117	58.5	6.500000
10	5	10	122	61.0	6.100000
11	8	11	130	65.0	5.909091
12	7	12	137	68.5	5.708333
13	7	13	144	72.0	5.538462
14	6	14	150	75.0	5.357143
15	6	15	156	78.0	5.200000
16	4	16	160	80.0	5.000000


```

17      0      17      160      80.0 4.705882
18      2      18      162      81.0 4.500000
19      0      19      162      81.0 4.263158
20      2      20      164      82.0 4.100000

```

```

GermValue(germ.counts = x, intervals = int, total.seeds = 200,
           method = "dp", k = 10, from.onset = FALSE)

```

```

$`Germination Value`
[1] 46.6952

```

```

[[2]]
   germ.counts intervals Cumulative.germ.counts Cumulative.germ.percent      DGS SumDGSbyN      GV
1           0          1              0              0.0 0.000000  0.000000  0.000000
2           0          2              0              0.0 0.000000  0.000000  0.000000
3          34          3             34             17.0 5.666667  1.888889  3.211111
4          40          4             74             37.0 9.250000  3.729167 13.797917
5          21          5             95             47.5 9.500000  4.883333 23.195833
6          10          6            105             52.5 8.750000  5.527778 29.020833
7           4          7            109             54.5 7.785714  5.850340 31.884354
8           5          8            114             57.0 7.125000  6.009673 34.255134
9           3          9            117             58.5 6.500000  6.064153 35.475298
10          5         10            122             61.0 6.100000  6.067738 37.013202
11          8         11            130             65.0 5.909091  6.053316 39.346552
12          7         12            137             68.5 5.708333  6.024567 41.268285
13          7         13            144             72.0 5.538462  5.987174 43.107655
14          6         14            150             75.0 5.357143  5.942172 44.566291
15          6         15            156             78.0 5.200000  5.892694 45.963013
16          4         16            160             80.0 5.000000  5.836901 46.695205
17          0         17            160             80.0 4.705882  5.770370 46.162961
18          2         18            162             81.0 4.500000  5.699794 46.168331
19          0         19            162             81.0 4.263158  5.624182 45.555871
20          2         20            164             82.0 4.100000  5.547972 45.493374

```

```

$testend
[1] 16

```

```

# From cumulative germination counts

```

```

#-----
PeakValue(germ.counts = y, interval = int, total.seeds = 200,
           partial = FALSE)

```

```

[1] 9.5

```

```

GermValue(germ.counts = y, intervals = int, total.seeds = 200,
           partial = FALSE, method = "czabator")

```

```

$`Germination Value`
[1] 38.95

```

```

[[2]]
   germ.counts intervals Cumulative.germ.counts Cumulative.germ.percent      DGS
3           34          3              34             17.0 5.666667
4           40          4              74             37.0 9.250000
5           21          5              95             47.5 9.500000
6           10          6            105             52.5 8.750000

```

7	4	7	109	54.5	7.785714
8	5	8	114	57.0	7.125000
9	3	9	117	58.5	6.500000
10	5	10	122	61.0	6.100000
11	8	11	130	65.0	5.909091
12	7	12	137	68.5	5.708333
13	7	13	144	72.0	5.538462
14	6	14	150	75.0	5.357143
15	6	15	156	78.0	5.200000
16	4	16	160	80.0	5.000000
17	0	17	160	80.0	4.705882
18	2	18	162	81.0	4.500000
19	0	19	162	81.0	4.263158
20	2	20	164	82.0	4.100000

```
GermValue(germ.counts = y, intervals = int, total.seeds = 200,
           partial = FALSE, method = "dp", k = 10)
```

```
$`Germination Value`
[1] 53.36595
```

```
[[2]]
   germ.counts intervals Cumulative.germ.counts Cumulative.germ.percent      DGS SumDGSbyN      GV
3           34         3              34             17.0 5.666667  5.666667  9.633333
4           40         4              74             37.0 9.250000  7.458333 27.595833
5           21         5              95             47.5 9.500000  8.138889 38.659722
6           10         6             105             52.5 8.750000  8.291667 43.531250
7            4         7             109             54.5 7.785714  8.190476 44.638095
8            5         8             114             57.0 7.125000  8.012897 45.673512
9            3         9             117             58.5 6.500000  7.796769 45.611097
10           5        10             122             61.0 6.100000  7.584673 46.266503
11           8        11             130             65.0 5.909091  7.398497 48.090230
12           7        12             137             68.5 5.708333  7.229481 49.521942
13           7        13             144             72.0 5.538462  7.075752 50.945411
14           6        14             150             75.0 5.357143  6.932534 51.994006
15           6        15             156             78.0 5.200000  6.799262 53.034246
16           4        16             160             80.0 5.000000  6.670744 53.365948
17           0        17             160             80.0 4.705882  6.539753 52.318022
18           2        18             162             81.0 4.500000  6.412268 51.939373
19           0        19             162             81.0 4.263158  6.285850 50.915385
20           2        20             164             82.0 4.100000  6.164414 50.548194
```

```
$testend
[1] 16
```

```
GermValue(germ.counts = y, intervals = int, total.seeds = 200,
           partial = FALSE, method = "czabator", from.onset = FALSE)
```

```
$`Germination Value`
[1] 38.95
```

```
[[2]]
   germ.counts intervals Cumulative.germ.counts Cumulative.germ.percent      DGS
1            0         1              0             0.0 0.000000
2            0         2              0             0.0 0.000000
```

3	34	3	34	17.0	5.666667
4	40	4	74	37.0	9.250000
5	21	5	95	47.5	9.500000
6	10	6	105	52.5	8.750000
7	4	7	109	54.5	7.785714
8	5	8	114	57.0	7.125000
9	3	9	117	58.5	6.500000
10	5	10	122	61.0	6.100000
11	8	11	130	65.0	5.909091
12	7	12	137	68.5	5.708333
13	7	13	144	72.0	5.538462
14	6	14	150	75.0	5.357143
15	6	15	156	78.0	5.200000
16	4	16	160	80.0	5.000000
17	0	17	160	80.0	4.705882
18	2	18	162	81.0	4.500000
19	0	19	162	81.0	4.263158
20	2	20	164	82.0	4.100000

```
GermValue(germ.counts = y, intervals = int, total.seeds = 200,
           partial = FALSE, method = "dp", k = 10, from.onset = FALSE)
```

```
$`Germination Value`
[1] 46.6952
```

```
[[2]]
   germ.counts intervals Cumulative.germ.counts Cumulative.germ.percent      DGS SumDGSbyN      GV
1            0         1                   0                0.0 0.000000 0.000000 0.000000
2            0         2                   0                0.0 0.000000 0.000000 0.000000
3           34         3                   34                17.0 5.666667 1.888889 3.211111
4           40         4                   74                37.0 9.250000 3.729167 13.797917
5           21         5                   95                47.5 9.500000 4.883333 23.195833
6           10         6                  105                52.5 8.750000 5.527778 29.020833
7            4         7                  109                54.5 7.785714 5.850340 31.884354
8            5         8                  114                57.0 7.125000 6.009673 34.255134
9            3         9                  117                58.5 6.500000 6.064153 35.475298
10           5        10                  122                61.0 6.100000 6.067738 37.013202
11           8        11                  130                65.0 5.909091 6.053316 39.346552
12           7        12                  137                68.5 5.708333 6.024567 41.268285
13           7        13                  144                72.0 5.538462 5.987174 43.107655
14           6        14                  150                75.0 5.357143 5.942172 44.566291
15           6        15                  156                78.0 5.200000 5.892694 45.963013
16           4        16                  160                80.0 5.000000 5.836901 46.695205
17           0        17                  160                80.0 4.705882 5.770370 46.162961
18           2        18                  162                81.0 4.500000 5.699794 46.168331
19           0        19                  162                81.0 4.263158 5.624182 45.555871
20           2        20                  164                82.0 4.100000 5.547972 45.493374
```

```
$testend
[1] 16
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
```

```

int <- 1:length(x)

# From partial germination counts
#-----
CUGerm(germ.counts = x, intervals = int)

CUGerm()

[1] 0.7092199

# From cumulative germination counts
#-----
CUGerm(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.05267935

x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)

# From partial germination counts
#-----
GermSynchrony(germ.counts = x, intervals = int)

GermSynchrony(), GermUncertainty()

[1] 0.2666667

GermUncertainty(germ.counts = x, intervals = int)

[1] 2.062987

# From cumulative germination counts
#-----
GermSynchrony(germ.counts = y, intervals = int, partial = FALSE)

[1] 0.2666667

GermUncertainty(germ.counts = y, intervals = int, partial = FALSE)

[1] 2.062987

```

Non-linear regression analysis

Several mathematical functions have been used to fit the cumulative germination count data and describe the germination process by non-linear regression analysis. They include functions such as Richard's, Weibull, logistic, log-logistic, gaussian, four-parameter hill function etc. Currently `germinationmetrics` implements the four-parameter hill function to fit the count data and computed various associated metrics.

Four-parameter hill function

The four-parameter hill function defined as follows (El-Kassaby et al., 2008).

$$f(x) = y = y_0 + \frac{ax^b}{x^b + c^b}$$

Where, y is the cumulative germination percentage at time x , y_0 is the intercept on the y axis, a is the asymptote, b is a mathematical parameter controlling the shape and steepness of the germination curve and c is the “half-maximal activation level”.

The details of various parameters that are computed from this function are given in Table 4.

Table 4 Germination parameters estimated from the four-parameter hill function.

Germination parameters	Details	Unit	Measures
y intercept (y_0)	The intercept on the y axis.		
Asymptote (a)	It is the maximum cumulative germination percentage, which is equivalent to germination capacity.	%	Germination capacity
Shape and steepness (b)	Mathematical parameter controlling the shape and steepness of the germination curve. The larger the b , the steeper the rise toward the asymptote a , and the shorter the time between germination onset and maximum germination.		Germination rate
Half-maximal activation level (c)	Time required for 50% of viable seeds to germinate.	time	Germination time
lag	It is the time at germination onset and is computed by solving four-parameter hill function after setting y to 0 as follows.	time	Germination time
$lag = b \sqrt{\frac{-y_0 c^b}{a + y_0}}$			
D_{lag-50}	The duration between the time at germination onset (lag) and that at 50% germination (c).	time	Germination time
$t_{50_{total}}$	Time required for 50% of total seeds to germinate.	time	Germination time
$t_{50_{germinated}}$	Time required for 50% of viable/germinated seeds to germinate	time	Germination time
$t_{x_{total}}$	Time required for $x\%$ of total seeds to germinate.	time	Germination time
$t_{x_{germinated}}$	Time required for $x\%$ of viable/germinated seeds to germinate	time	Germination time
Uniformity ($U_{t_{max}-t_{min}}$)	It is the time interval between the percentages of viable seeds specified in the arguments umin and umax to germinate.	time	Germination time

Germination parameters	Details	Unit	Measures
Time at maximum germination rate (<i>TMGR</i>)	<p>The partial derivative of the four-parameter hill function gives the instantaneous rate of germination (<i>s</i>) as follows.</p> $s = \frac{\partial y}{\partial x} = \frac{abc^b x^{b-1}}{(c^b + x^b)^2}$ <p>From this function for instantaneous rate of germination, <i>TMGR</i> can be estimated as follows.</p> $TMGR = b \sqrt{\frac{c^b(b-1)}{b+1}}$ <p>It represents the point in time when the instantaneous rate of germination starts to decline.</p>	time	Germination time
Area under the curve (<i>AUC</i>)	It is obtained by integration of the fitted curve between time 0 and time specified in the argument tmax .		Mixed
<i>MGT</i>	Calculated by integration of the fitted curve and proper normalisation.	time	Germination time
<i>Skewness</i>	It is computed as follows.		
$\frac{MGT}{t_{50_{germinated}}}$			

Examples

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
total.seeds = 50

# From partial germination counts
#-----
FourPHFfit(germ.counts = x, intervals = int, total.seeds = 50, tmax = 20)
```

```
FourPHFfit()
```

```
$data
```

```
  gp csgp intervals
1  0    0         1
2  0    0         2
3  0    0         3
4  0    0         4
5  8    8         5
6 34   42         6
7 20   62         7
8 14   76         8
9  2   78         9
10 0   78        10
```

```

11  2   80      11
12  0   80      12
13  0   80      13
14  0   80      14

```

```
$Parameters
```

	term	estimate	std.error	statistic	p.value
1	a	80.000000	1.24158595	64.43372	1.973240e-14
2	b	9.881947	0.70779379	13.96162	6.952322e-08
3	c	6.034954	0.04952654	121.85294	3.399385e-17
4	y0	0.000000	0.91607007	0.00000	1.000000e+00

```
$Fit
```

	sigma	isConv	finTol	logLik	AIC	BIC	deviance	df.residual
1	1.769385	TRUE	1.490116e-08	-25.49868	60.99736	64.19265	31.30723	10

```
$a
```

```
[1] 80
```

```
$b
```

```
[1] 9.881947
```

```
$c
```

```
[1] 6.034954
```

```
$y0
```

```
[1] 0
```

```
$lag
```

```
[1] 0
```

```
$Dlag50
```

```
[1] 6.034954
```

```
$t50.total
```

```
[1] 6.355122
```

```
$txp.total
```

```

      10      60
4.956266 6.744598

```

```
$t50.Germinated
```

```
[1] 6.034954
```

```
$txp.Germinated
```

```

      10      60
4.831809 6.287724

```

```
$Uniformity
```

```

      90      10 uniformity
7.537688 4.831809 2.705880

```

```
$TMGR
```

```
[1] 5.912195
```

```

$AUC
[1] 1108.975

$MGT
[1] 6.632252

$Skewness
[1] 1.098973

$msg
[1] "#1. Relative error in the sum of squares is at most `ftol'. "

$isConv
[1] TRUE

attr("class")
[1] "FourPHFfit" "list"

# From cumulative germination counts
#-----
FourPHFfit(germ.counts = y, intervals = int, total.seeds = 50, tmax = 20,
partial = FALSE)

$data
  gp csgp intervals
1  0    0         1
2  0    0         2
3  0    0         3
4  0    0         4
5  8    8         5
6 34   42         6
7 20   62         7
8 14   76         8
9  2   78         9
10 0   78        10
11 2   80        11
12 0   80        12
13 0   80        13
14 0   80        14

$Parameters
  term estimate std.error statistic    p.value
1   a 80.000000  1.2415867   64.43368 1.973252e-14
2   b  9.881927  0.7077918   13.96163 6.952270e-08
3   c  6.034953  0.0495266  121.85275 3.399437e-17
4  y0  0.000000  0.9160705    0.00000 1.000000e+00

$Fit
  sigma isConv   finTol   logLik    AIC    BIC deviance df.residual
1 1.769385  TRUE 1.490116e-08 -25.49868 60.99736 64.19265 31.30723         10

$a
[1] 80

```



```

$b
[1] 9.881927

$c
[1] 6.034953

$y0
[1] 0

$lag
[1] 0

$Dlag50
[1] 6.034953

$t50.total
[1] 6.355121

$txp.total
      10      60
4.956263 6.744599

$t50.Germinated
[1] 6.034953

$txp.Germinated
      10      60
4.831806 6.287723

$Uniformity
      90      10 uniformity
7.537691 4.831806 2.705885

$TMGR
[1] 5.912194

$AUC
[1] 1108.976

$MGT
[1] 6.632252

$Skewness
[1] 1.098973

$msg
[1] "#1. Relative error in the sum of squares is at most `ftol'. "

$isConv
[1] TRUE

attr(,"class")
[1] "FourPHFfit" "list"

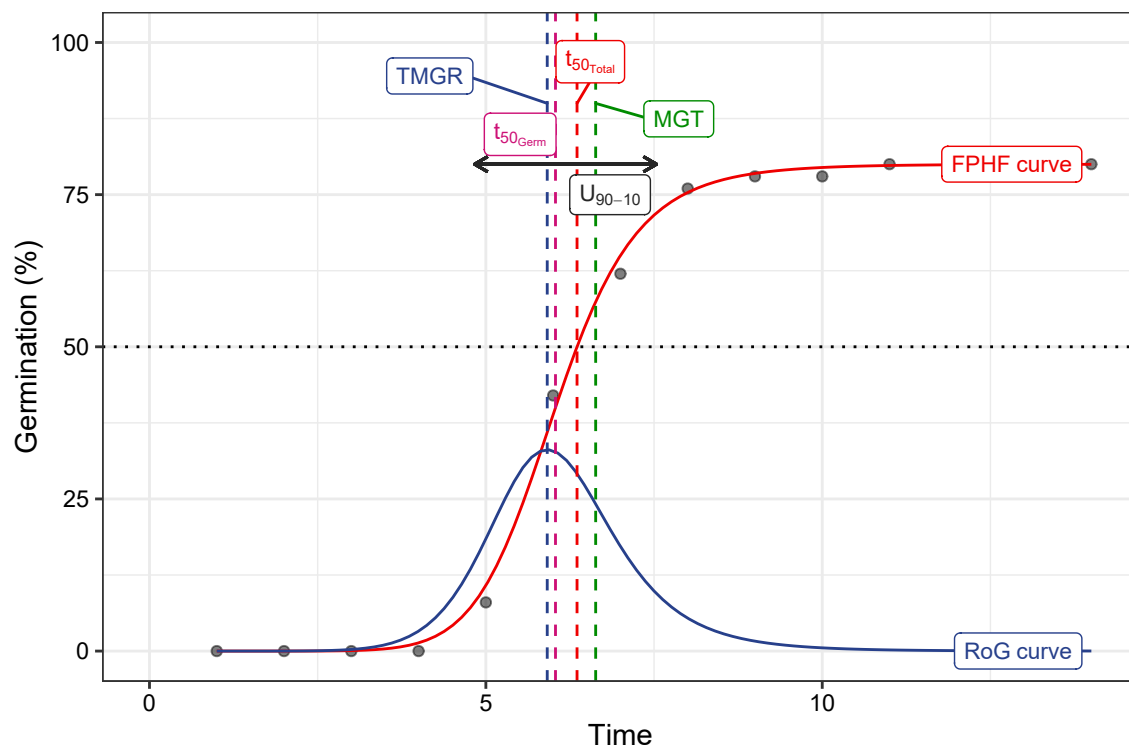
```

```
x <- c(0, 0, 0, 0, 4, 17, 10, 7, 1, 0, 1, 0, 0, 0)
y <- c(0, 0, 0, 0, 4, 21, 31, 38, 39, 39, 40, 40, 40, 40)
int <- 1:length(x)
total.seeds = 50

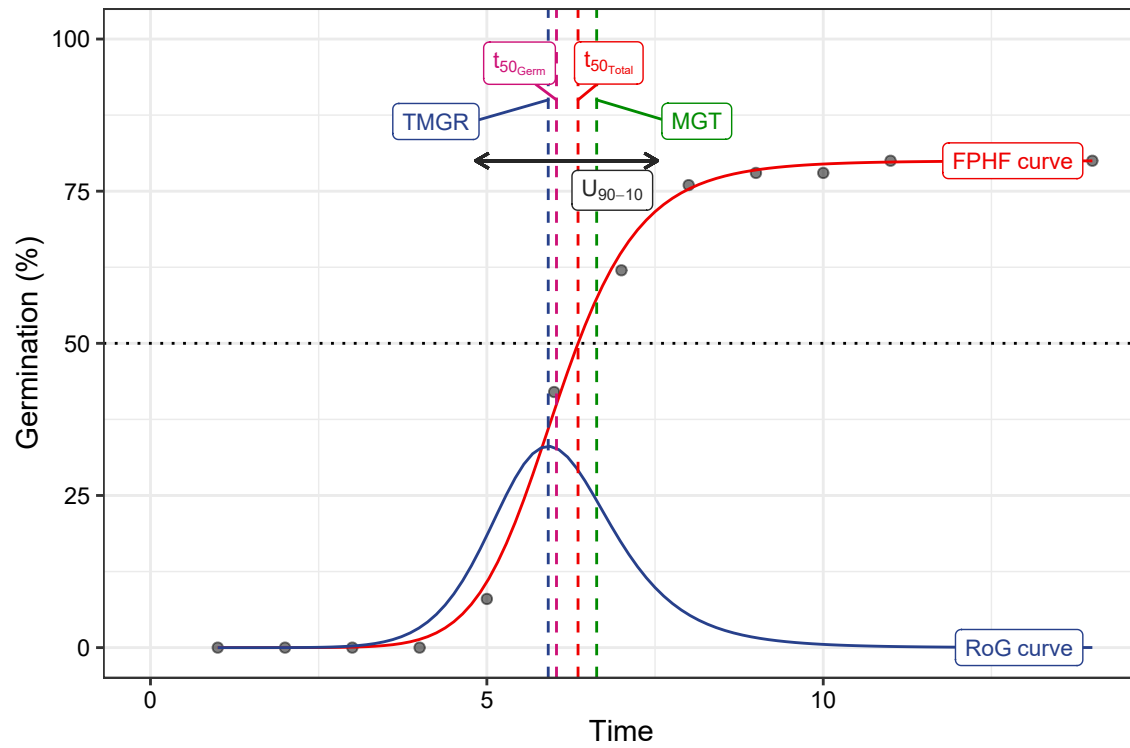
# From partial germination counts
#-----
fit1 <- FourPHFfit(germ.counts = x, intervals = int,
                  total.seeds = 50, tmax = 20)

# From cumulative germination counts
#-----
fit2 <- FourPHFfit(germ.counts = y, intervals = int,
                  total.seeds = 50, tmax = 20, partial = FALSE)

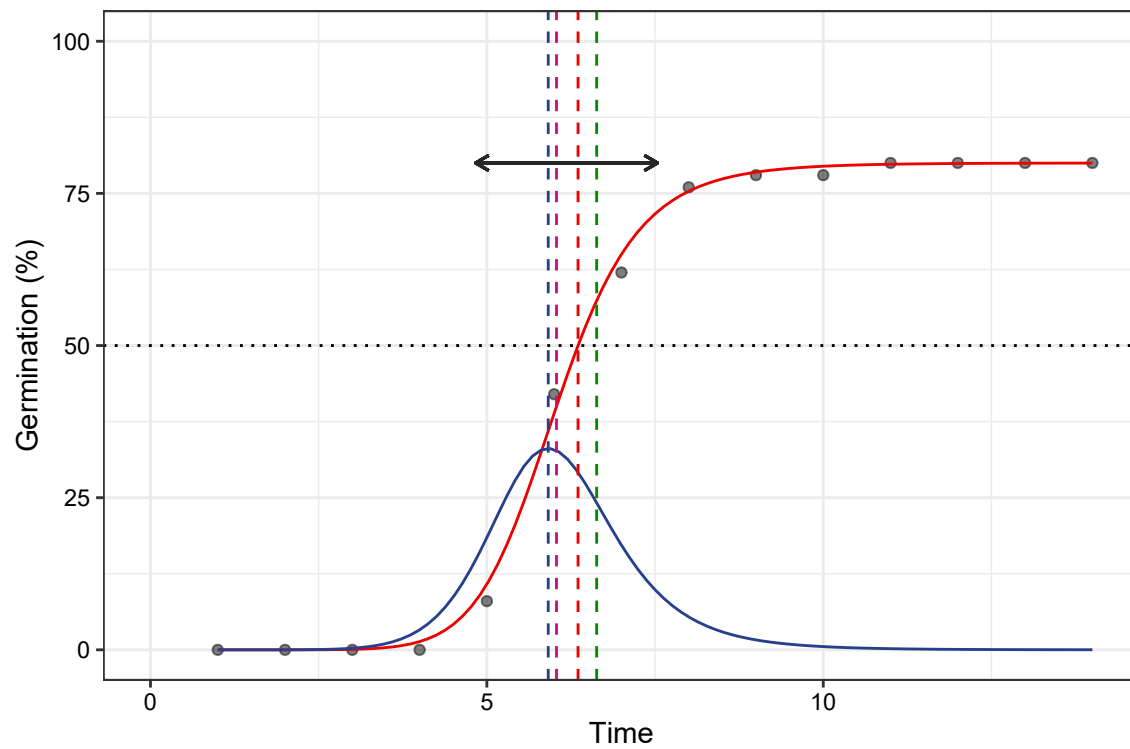
# Default plots
plot(fit1)
```



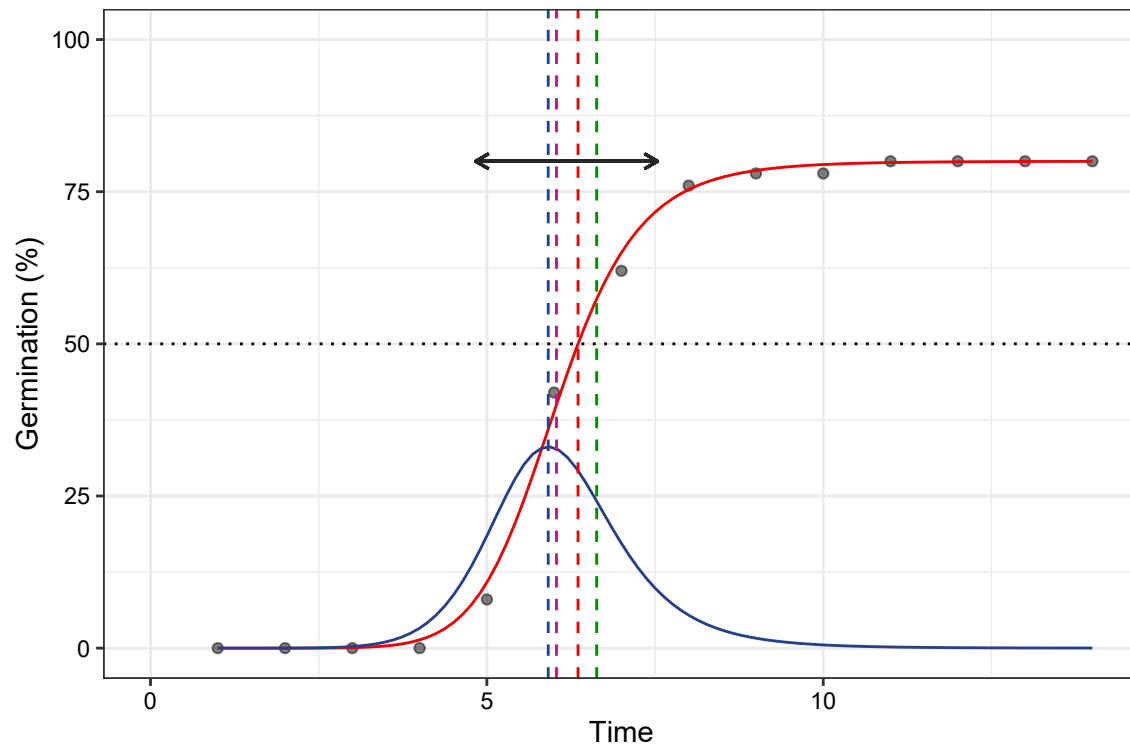
```
plot(fit2)
```



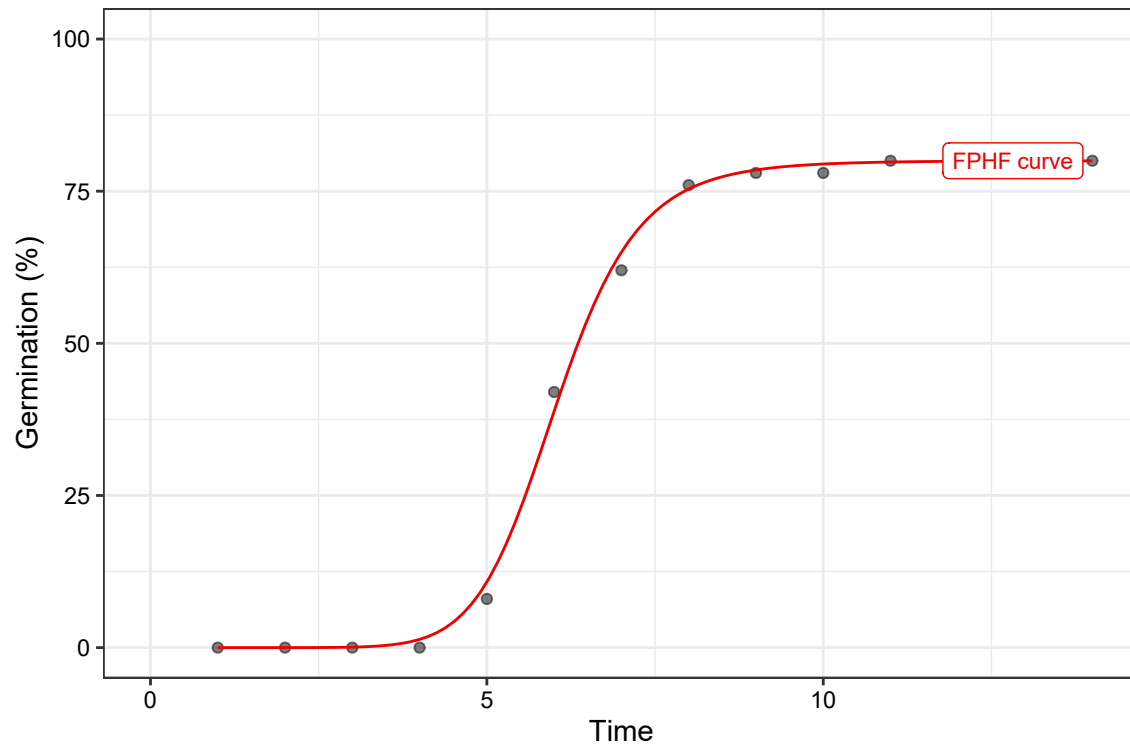
```
# No labels
plot(fit1, plotlabels = FALSE)
```



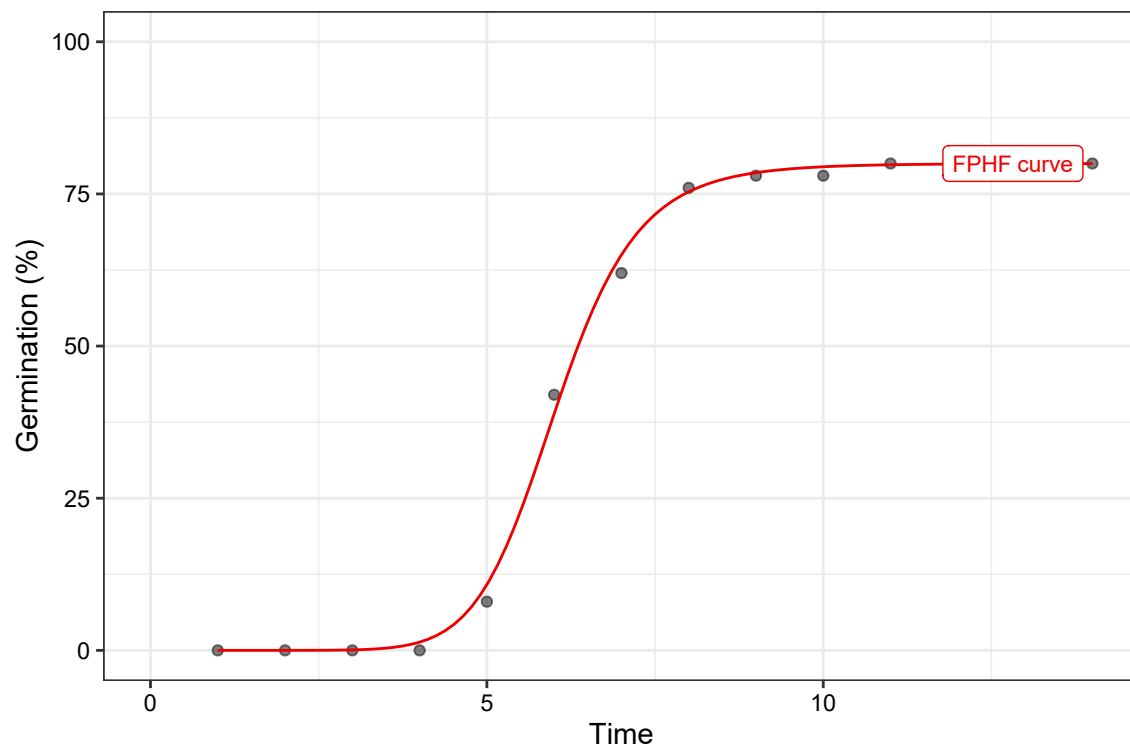
```
plot(fit2, plotlabels = FALSE)
```



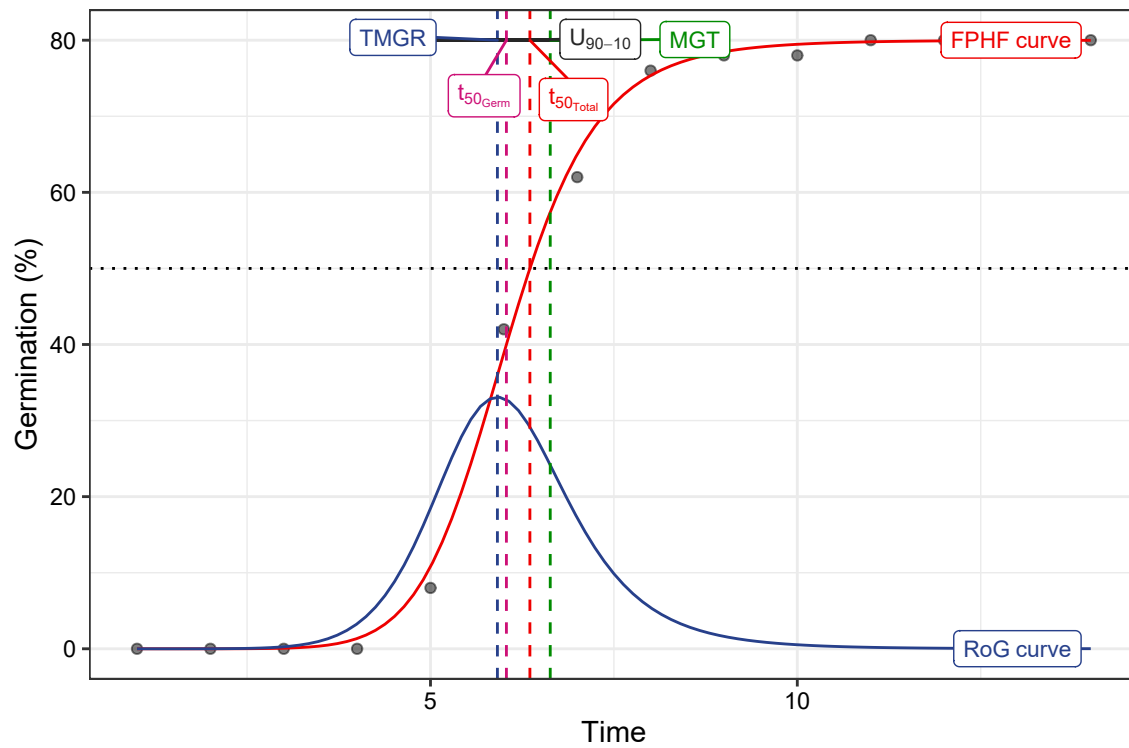
```
# Only the FPHF curve
plot(fit1, rog = FALSE, t50.total = FALSE, t50.germ = FALSE,
     tmgr = FALSE, mgt = FALSE, uniformity = FALSE)
```



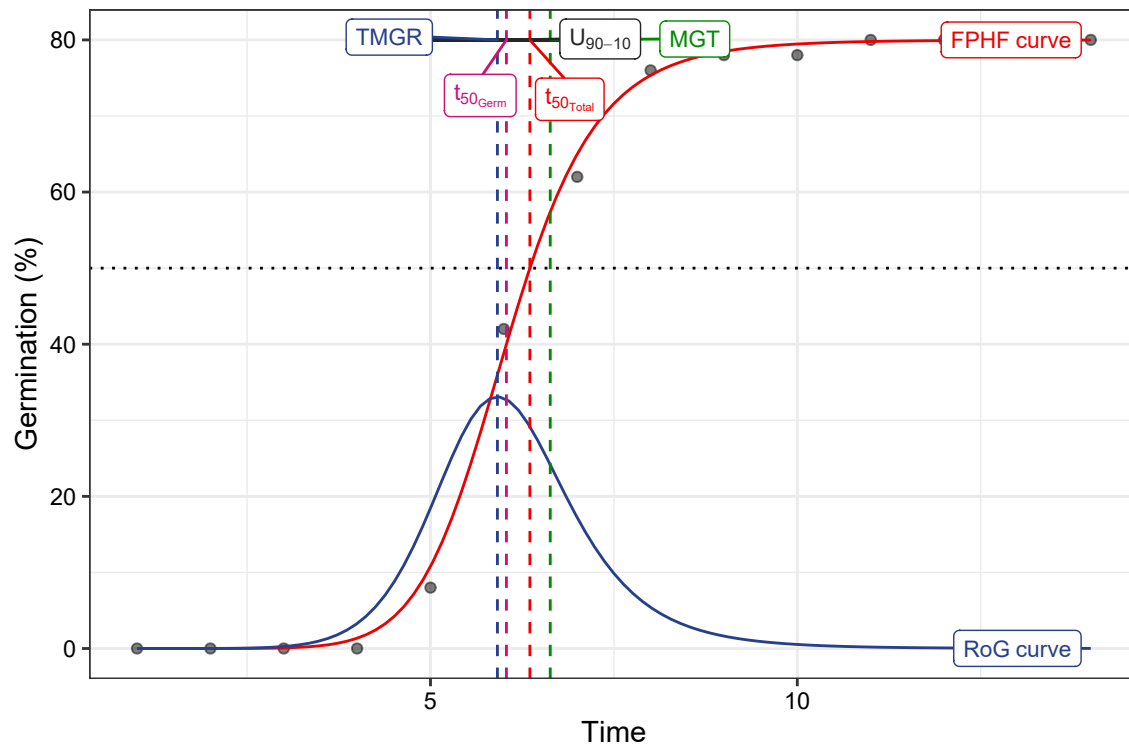
```
plot(fit2, rog = FALSE, t50.total = FALSE, t50.germ = FALSE,
     tmgr = FALSE, mgt = FALSE, uniformity = FALSE)
```



```
# Without y axis limits adjustment
plot(fit1, limits = FALSE)
```



```
plot(fit2, limits = FALSE)
```



Wrapper functions

Wrapper functions `germination.indices()` and `FourPHFfit.bulk()` are available in the package for computing results for multiple samples in batch from a data frame of germination counts recorded at specific time intervals.

`germination.indices()` This wrapper function can be used to compute several germination indices simultaneously for multiple samples in batch.

```
data(gcdata)
```

```
counts.per.intervals <- c("Day01", "Day02", "Day03", "Day04", "Day05",
                          "Day06", "Day07", "Day08", "Day09", "Day10",
                          "Day11", "Day12", "Day13", "Day14")
germination.indices(gcdata, total.seeds.col = "Total Seeds",
                   counts.intervals.cols = counts.per.intervals,
                   intervals = 1:14, partial = TRUE, max.int = 5)
```

	Genotype	Rep	Day01	Day02	Day03	Day04	Day05	Day06	Day07	Day08	Day09	Day10	Day11	Day12	Day13	Day14	Tot
1	G1	1	0	0	0	0	4	17	10	7	1	0	1	0	0	0	
2	G2	1	0	0	0	1	3	15	13	6	2	1	0	1	0	0	
3	G3	1	0	0	0	2	3	18	9	8	2	1	1	1	0	0	
4	G4	1	0	0	0	0	4	19	12	6	2	1	1	1	0	0	
5	G5	1	0	0	0	0	5	20	12	8	1	0	0	1	1	0	
6	G1	2	0	0	0	0	3	21	11	7	1	1	1	1	0	0	
7	G2	2	0	0	0	0	4	18	11	7	1	0	1	0	0	0	
8	G3	2	0	0	0	1	3	14	12	6	2	1	0	1	0	0	
9	G4	2	0	0	0	1	3	19	10	8	1	1	1	1	0	0	
10	G5	2	0	0	0	0	4	18	13	6	2	1	0	1	0	0	
11	G1	3	0	0	0	0	5	21	11	8	1	0	0	1	1	0	
12	G2	3	0	0	0	0	3	20	10	7	1	1	1	1	0	0	
13	G3	3	0	0	0	0	4	19	12	8	1	1	0	1	1	0	
14	G4	3	0	0	0	0	3	21	11	6	1	0	1	1	0	0	
15	G5	3	0	0	0	0	4	17	10	8	1	1	1	0	0	0	
	GermPercent	FirstGermTime	LastGermTime		PeakGermTime		TimeSpreadGerm		t50_Coolbear		t50_Farooq		MeanGermTime				
1	80.00000		5		11		6		6		5.970588		5.941176		6.700000		
2	82.35294		4		12		6		8		6.192308		6.153846		6.857143		
3	93.75000		4		12		6		8		6.000000		5.972222		6.866667		
4	90.19608		5		12		6		7		6.041667		6.000000		6.891765		
5	96.00000		5		13		6		8		5.975000		5.950000		6.812500		
6	93.87755		5		12		6		7		5.976190		5.952381		6.869565		
7	87.50000		5		11		6		6		5.972222		5.944444		6.690476		
8	85.10638		4		12		6		8		6.208333		6.166667		6.875000		
9	86.53846		4		12		6		8		6.000000		5.973684		6.866667		
10	90.00000		5		12		6		7		6.076923		6.038462		6.822581		
11	94.11765		5		13		6		8		5.928571		5.904762		6.791667		
12	86.27451		5		12		6		7		5.975000		5.950000		6.886364		
13	95.91837		5		13		6		8		6.083333		6.041667		6.936364		
14	91.66667		5		12		6		7		5.928571		5.904762		6.772727		
15	87.50000		5		11		6		6		6.050000		6.000000		6.809524		
	SEGermTime	CVGermTime	MeanGermRate		VarGermRate		SEGermRate		CVG		GermRateRecip_Coolbear		GermRateRecip_Farooq				
1	0.1901416	0.1794868	0.1492537		0.0007176543		0.004235724		14.92537				0.1674877				
2	0.2197333	0.2076717	0.1458333		0.0009172090		0.004673148		14.58333				0.1614907				
3	0.2391061	0.2335882	0.1456311		0.0011572039		0.005071059		14.56311				0.1666667				
4	0.2180907	0.2146419	0.1451104		0.0009701218		0.004592342		14.51104				0.1655172				

5	0.2221275	0.2259002	0.1467890	0.0010995627	0.004786184	14.67890	0.1673640
6	0.2122088	0.2095140	0.1455696	0.0009301809	0.004496813	14.55696	0.1673307
7	0.1818989	0.1761967	0.1494662	0.0006935558	0.004063648	14.94662	0.1674419
8	0.2297923	0.2113940	0.1454545	0.0009454531	0.004861721	14.54545	0.1610738
9	0.2260777	0.2208604	0.1456311	0.0010345321	0.004794747	14.56311	0.1666667
10	0.2017321	0.1983606	0.1465798	0.0008453940	0.004334343	14.65798	0.1645570
11	0.2227295	0.2272072	0.1472393	0.0011191581	0.004828643	14.72393	0.1686747
12	0.2210295	0.2129053	0.1452145	0.0009558577	0.004660905	14.52145	0.1673640
13	0.2324392	0.2297410	0.1441718	0.0010970785	0.004831366	14.41718	0.1643836
14	0.2078370	0.2035568	0.1476510	0.0009033254	0.004531018	14.76510	0.1686747
15	0.1994129	0.1897847	0.1468531	0.0007767634	0.004300508	14.68531	0.1652893
GermSpeed_Count GermSpeed_Percent GermSpeedAccumulated_Count GermSpeedAccumulated_Percent GermSpeedC							
1	6.138925	12.27785		34.61567		69.23134	
2	6.362698	12.47588		35.54058		69.68741	
3	6.882179	14.33787		38.29725		79.78594	
4	6.927417	13.58317		38.68453		75.85202	
5	7.318987	14.63797		41.00786		82.01571	
6	6.931782	14.14649		38.77620		79.13509	
7	6.448449	13.43427		36.38546		75.80304	
8	6.053175	12.87909		33.77079		71.85275	
9	6.830592	13.13575		38.11511		73.29829	
10	6.812698	13.62540		38.19527		76.39054	
11	7.342796	14.39764		41.17452		80.73436	
12	6.622258	12.98482		37.00640		72.56158	
13	7.052320	14.39249		39.29399		80.19182	
14	6.706782	13.97246		37.69490		78.53103	
15	6.363925	13.25818		35.69697		74.36868	
GermSpeedCorrected_Accumulated WeightGermPercent MeanGermPercent MeanGermNumber TimsonsIndex TimsonsI							
1		0.4326958	47.42857	5.714286	2.857143	8.000000	
2		0.4315642	47.89916	5.882353	3.000000	9.803922	
3		0.4085040	54.46429	6.696429	3.214286	14.583333	
4		0.4288937	52.24090	6.442577	3.285714	7.843137	
5		0.4271652	56.14286	6.857143	3.428571	10.000000	
6		0.4130508	54.51895	6.705539	3.285714	6.122449	
7		0.4158338	51.93452	6.250000	3.000000	8.333333	
8		0.3968068	49.39210	6.079027	2.857143	10.638298	
9		0.4404413	50.27473	6.181319	3.214286	9.615385	
10		0.4243919	52.57143	6.428571	3.214286	8.000000	
11		0.4374793	55.18207	6.722689	3.428571	9.803922	
12		0.4289379	50.00000	6.162465	3.142857	5.882353	
13		0.4096608	55.24781	6.851312	3.357143	8.163265	
14		0.4112171	53.86905	6.547619	3.142857	6.250000	
15		0.4079653	51.19048	6.250000	3.000000	8.333333	
TimsonsIndex_KhanUngar GermRateGeorge GermIndex GermIndex_mod EmergenceRateIndex_Melville							
1	0.5714286	4	5.840000	7.300000		292	
2	0.7002801	5	5.882353	7.142857		300	
3	1.0416667	7	6.687500	7.133333		321	
4	0.5602241	4	6.411765	7.108696		327	
5	0.7142857	5	6.900000	7.187500		345	
6	0.4373178	3	6.693878	7.130435		328	
7	0.5952381	4	6.395833	7.309524		307	
8	0.7598784	5	6.063830	7.125000		285	
9	0.6868132	5	6.173077	7.133333		321	
10	0.5714286	4	6.460000	7.177778		323	

11	0.7002801	5	6.784314	7.208333	346		
12	0.4201681	3	6.137255	7.113636	313		
13	0.5830904	4	6.775510	7.063830	332		
14	0.4464286	3	6.625000	7.227273	318		
15	0.5952381	4	6.291667	7.190476	302		
EmergenceRateIndex_Melville_mod EmergenceRateIndex_BilbroWanjura EmergenceRateIndex_Fakorede PeakVal							
1	7.300000	5.970149	8.375000	9.500000			
2	7.142857	6.125000	8.326531	9.313750			
3	7.133333	6.553398	7.324444	10.416667			
4	7.108696	6.675079	7.640359	10.049012			
5	7.187500	7.045872	7.096354	11.250000			
6	7.130435	6.696203	7.317580	10.714286			
7	7.309524	6.277580	7.646259	10.416667			
8	7.125000	5.818182	8.078125	9.574464			
9	7.133333	6.553398	7.934815	9.855769			
10	7.177778	6.596091	7.580247	10.250000			
11	7.208333	7.067485	7.216146	11.029412			
12	7.113636	6.389439	7.981921	9.803922			
13	7.063830	6.776074	7.231326	10.969388			
14	7.227273	6.496644	7.388430	10.677083			
15	7.190476	6.167832	7.782313	10.156250			
GermValue_Czabator GermValue_DP GermValue_Czabator_mod GermValue_DP_mod CUGerm GermSynchrony Germ							
1	54.28571	57.93890	54.28571	39.56076	0.7092199	0.2666667	
2	54.78662	52.58713	54.78662	40.99260	0.5051546	0.2346109	
3	69.75446	68.62289	69.75446	53.42809	0.3975265	0.2242424	
4	64.74158	70.43331	64.74158	48.86825	0.4672113	0.2502415	
5	77.14286	80.16914	77.14286	56.23935	0.4312184	0.2606383	
6	71.84506	76.51983	71.84506	53.06435	0.4934701	0.2792271	
7	65.10417	69.41325	65.10417	47.37690	0.7371500	0.2729384	
8	58.20345	56.00669	58.20345	43.67948	0.4855842	0.2256410	
9	60.92165	58.13477	60.92165	45.30801	0.4446640	0.2494949	
10	65.89286	70.91875	65.89286	49.10820	0.5584666	0.2555556	
11	74.14731	77.39782	74.14731	54.27520	0.4288905	0.2686170	
12	60.41632	64.44988	60.41632	44.71582	0.4760266	0.2737844	
13	75.15470	78.16335	75.15470	54.94192	0.4023679	0.2506938	
14	69.90947	74.40140	69.90947	51.41913	0.5383760	0.2991543	
15	63.47656	67.62031	63.47656	46.48043	0.6133519	0.2497096	

FourPHFfit.bulk() This wrapper function can be used to fit the four-parameter hill function for multiple samples in batch.

```
data(gcdata)

counts.per.intervals <- c("Day01", "Day02", "Day03", "Day04", "Day05",
                          "Day06", "Day07", "Day08", "Day09", "Day10",
                          "Day11", "Day12", "Day13", "Day14")

FourPHFfit.bulk(gcdata, total.seeds.col = "Total Seeds",
                counts.intervals.cols = counts.per.intervals,
                intervals = 1:14, partial = TRUE,
                fix.y0 = TRUE, fix.a = TRUE, xp = c(10, 60),
                tmax = 20, tries = 3, umax = 90, umin = 10)
```

	Genotype	Rep	Day01	Day02	Day03	Day04	Day05	Day06	Day07	Day08	Day09	Day10	Day11	Day12	Day13	Day14	Tot
1	G1	1	0	0	0	0	4	17	10	7	1	0	1	0	0	0	

2	G2	1	0	0	0	1	3	15	13	6	2	1	0	1	0	0
3	G3	1	0	0	0	2	3	18	9	8	2	1	1	1	0	0
4	G4	1	0	0	0	0	4	19	12	6	2	1	1	1	0	0
5	G5	1	0	0	0	0	5	20	12	8	1	0	0	1	1	0
6	G1	2	0	0	0	0	3	21	11	7	1	1	1	1	0	0
7	G2	2	0	0	0	0	4	18	11	7	1	0	1	0	0	0
8	G3	2	0	0	0	1	3	14	12	6	2	1	0	1	0	0
9	G4	2	0	0	0	1	3	19	10	8	1	1	1	1	0	0
10	G5	2	0	0	0	0	4	18	13	6	2	1	0	1	0	0
11	G1	3	0	0	0	0	5	21	11	8	1	0	0	1	1	0
12	G2	3	0	0	0	0	3	20	10	7	1	1	1	1	0	0
13	G3	3	0	0	0	0	4	19	12	8	1	1	0	1	1	0
14	G4	3	0	0	0	0	3	21	11	6	1	0	1	1	0	0
15	G5	3	0	0	0	0	4	17	10	8	1	1	1	0	0	0
b c y0 lag Dlag50 t50.total t50.Germinated TMGR AUC MGT Skewness																
1	9.881947	6.034954	0	0	6.034954	6.355122	6.034954	5.912195	1108.975	6.632252	1.098973					
2	9.227667	6.175193	0	0	6.175193	6.473490	6.175193	6.031282	1128.559	6.784407	1.098655					
3	7.793055	6.138110	0	0	6.138110	6.244190	6.138110	5.938179	1283.693	6.772742	1.103392					
4	8.925668	6.125172	0	0	6.125172	6.276793	6.125172	5.972686	1239.887	6.739665	1.100323					
5	9.419194	6.049641	0	0	6.049641	6.103433	6.049641	5.914289	1328.328	6.654980	1.100062					
6	9.450187	6.097412	0	0	6.097412	6.182276	6.097412	5.961877	1294.463	6.702470	1.099232					
7	10.172466	6.029851	0	0	6.029851	6.202812	6.029851	5.914057	1213.908	6.622417	1.098272					
8	8.940702	6.189774	0	0	6.189774	6.439510	6.189774	6.036193	1164.346	6.804000	1.099232					
9	8.617395	6.125121	0	0	6.125121	6.352172	6.125121	5.961631	1188.793	6.745241	1.101242					
10	9.608849	6.109503	0	0	6.109503	6.253042	6.109503	5.978115	1240.227	6.711899	1.098600					
11	9.400248	6.018759	0	0	6.018759	6.099434	6.018759	5.883558	1305.200	6.624247	1.100600					
12	9.162558	6.108449	0	0	6.108449	6.326181	6.108449	5.964079	1188.021	6.718636	1.099892					
13	8.995233	6.149011	0	0	6.149011	6.207500	6.149011	5.998270	1316.407	6.762272	1.099733					
14	10.391898	6.015907	0	0	6.015907	6.122385	6.015907	5.905179	1273.386	6.604963	1.097916					
15	9.136762	6.121580	0	0	6.121580	6.317392	6.121580	5.976088	1203.664	6.732267	1.099760					
msg isConv txp.total_10 txp.total_60 Uniformity																
1	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.956266	6.744598	7	
2	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.983236	6.872603	7	
3	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.673022	6.608437	8	
4	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.850876	6.614967	7	
5	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.814126	6.386788	7	
6	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.868635	6.477594	7	
7	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.930423	6.510495	7	
8	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.940058	6.823299	7	
9	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.836659	6.733275	7	
10	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.920629	6.566505	7	
11	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.798630	6.391288	7	
12	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.893597	6.684521	7	
13	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.841310	6.509952	7	
14	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.915143	6.397486	7	
15	#1. Relative error in the sum of squares is at most `ftol'.											TRUE	4.892505	6.667247	7	
Uniformity_10 Uniformity																
1	4.831809											2.705880				
2	4.866755											2.968652				
3	4.630062											3.507277				
4	4.788598											3.046208				
5	4.790947											2.848078				
6	4.832474											2.860984				
7	4.858477											2.625165				

8	4.841106	3.073056
9	4.746574	3.157466
10	4.860681	2.818494
11	4.764249	2.839354
12	4.806015	2.957830
13	4.816395	3.033943
14	4.869401	2.562960
15	4.813086	2.972718

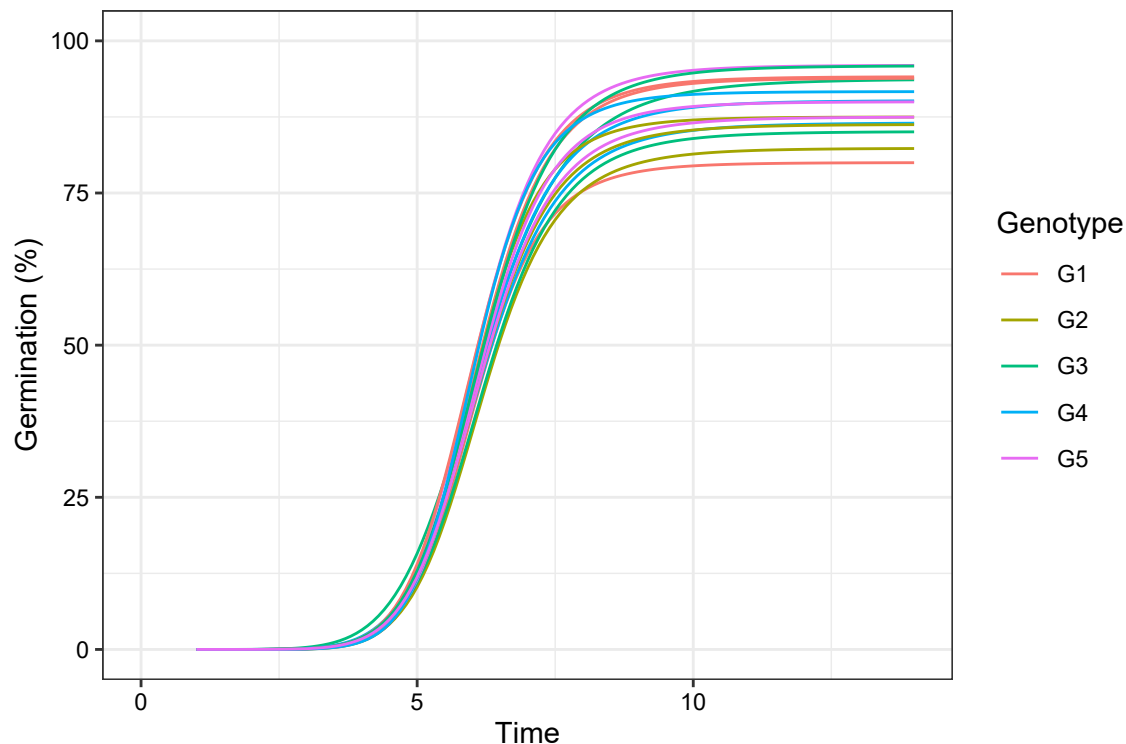
Multiple fitted curves generated in batch can also be plotted.

```
data(gcdata)

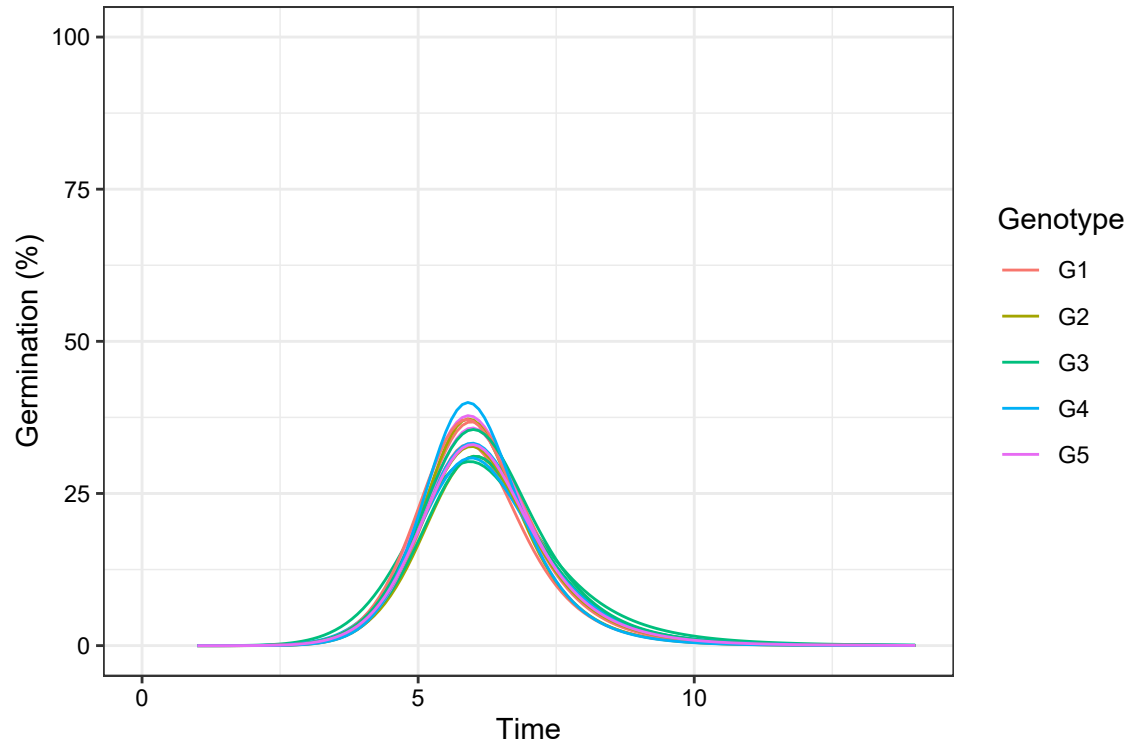
counts.per.intervals <- c("Day01", "Day02", "Day03", "Day04", "Day05",
                          "Day06", "Day07", "Day08", "Day09", "Day10",
                          "Day11", "Day12", "Day13", "Day14")

fits <- FourPHFfit.bulk(gcdata, total.seeds.col = "Total Seeds",
                       counts.intervals.cols = counts.per.intervals,
                       intervals = 1:14, partial = TRUE,
                       fix.y0 = TRUE, fix.a = TRUE, xp = c(10, 60),
                       tmax = 20, tries = 3, umax = 90, umin = 10)

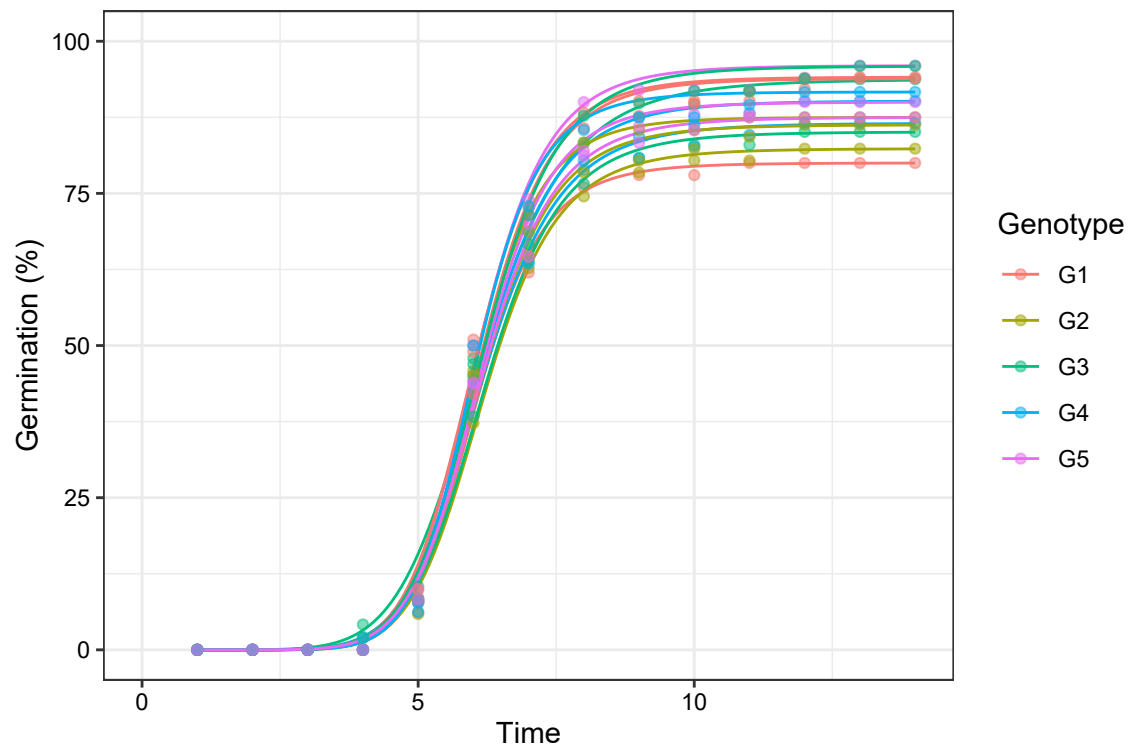
# Plot FPHF curves
plot(fits, group.col = "Genotype")
```



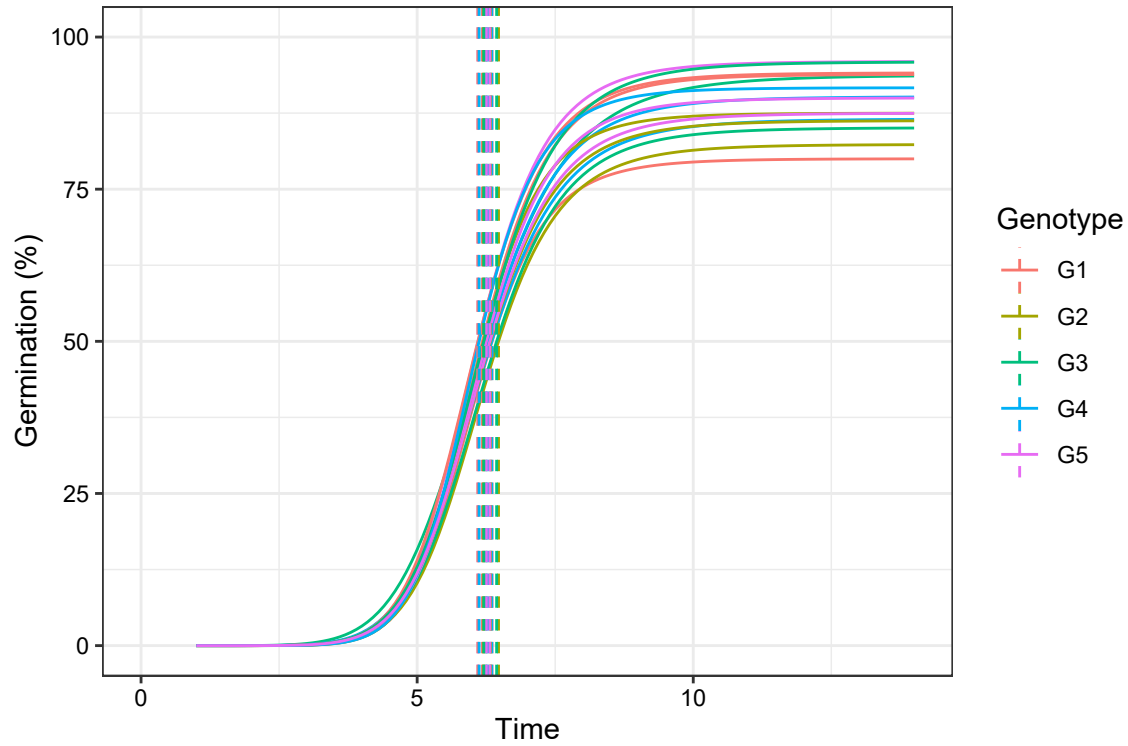
```
# Plot ROG curves
plot(fits, rog = TRUE, group.col = "Genotype")
```



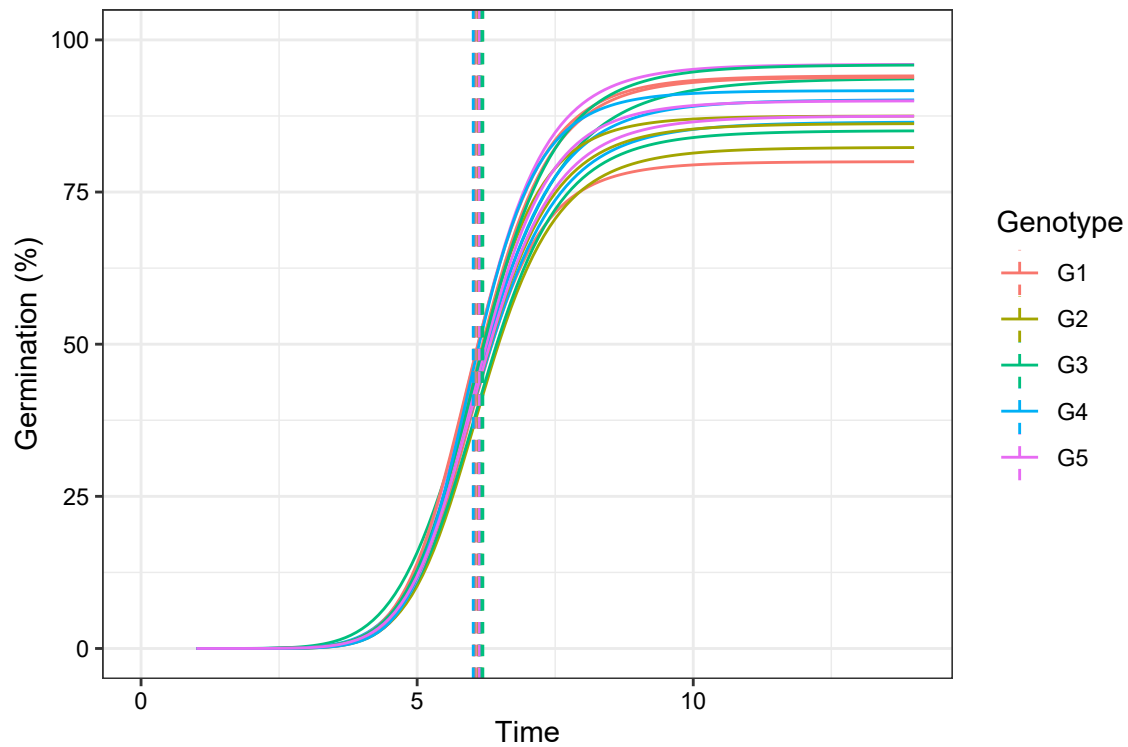
```
# Plot FPHF curves with points
plot(fits, group.col = "Genotype", show.points = TRUE)
```



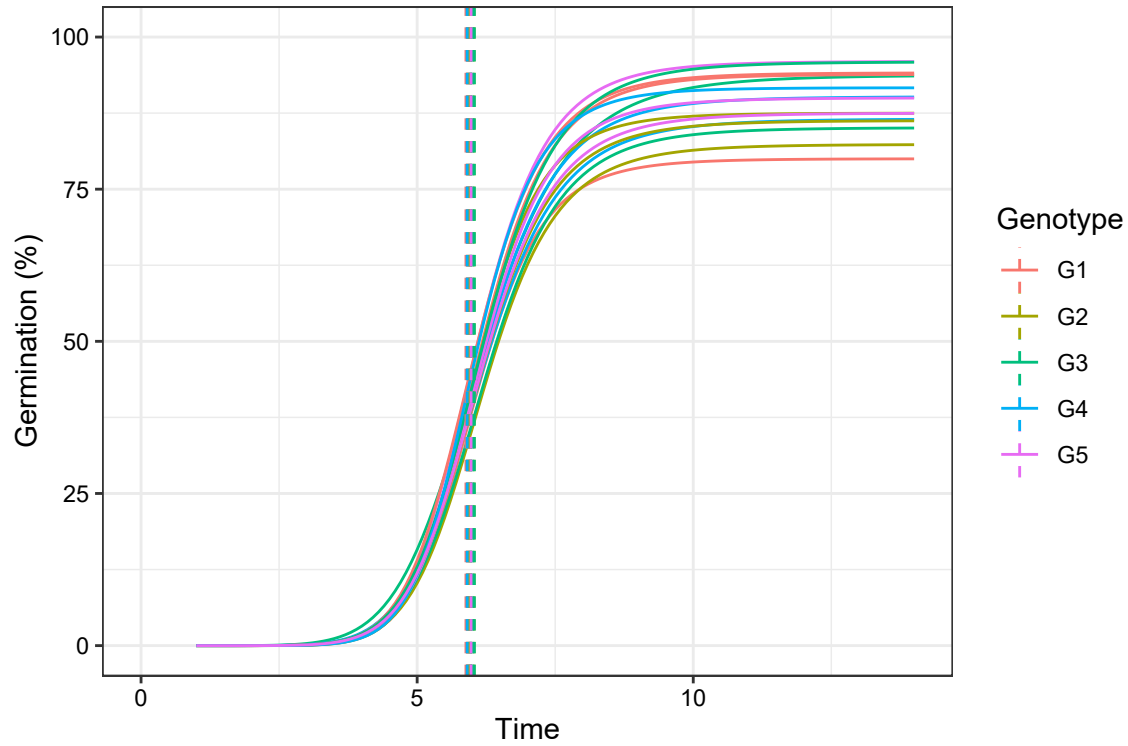
```
# Plot FPHF curves with annotations
plot(fits, group.col = "Genotype", annotate = "t50.total")
```



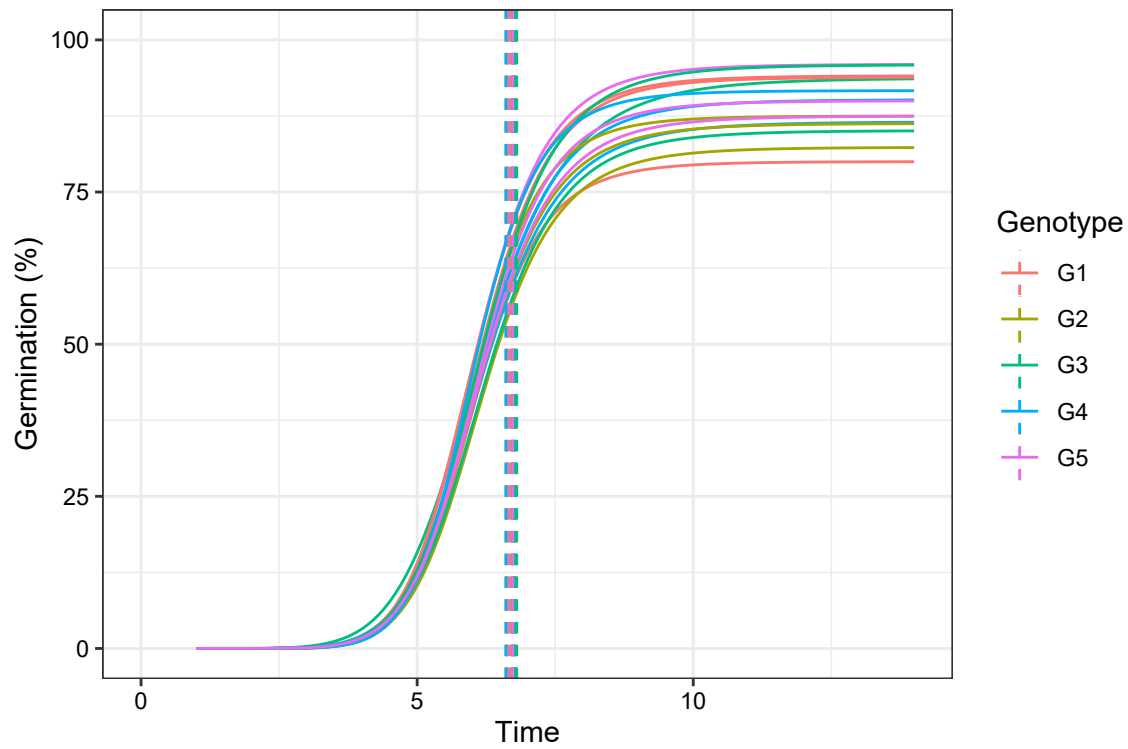
```
plot(fits, group.col = "Genotype", annotate = "t50.germ")
```



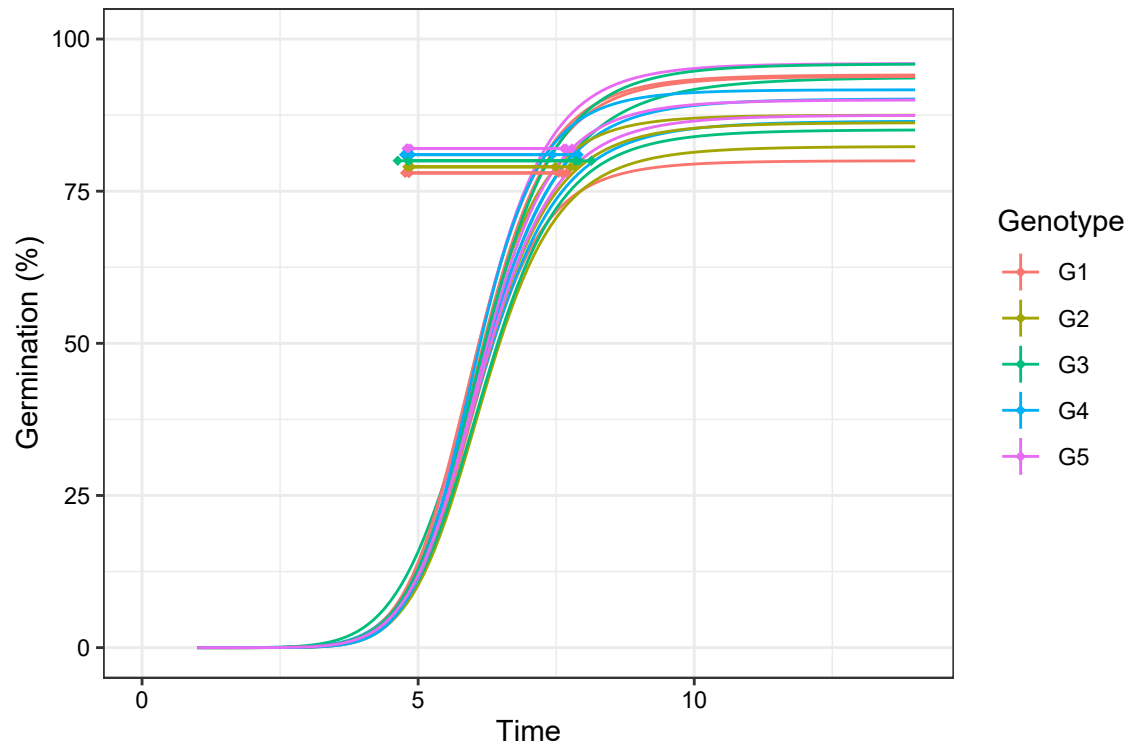
```
plot(fits, group.col = "Genotype", annotate = "tmgr")
```



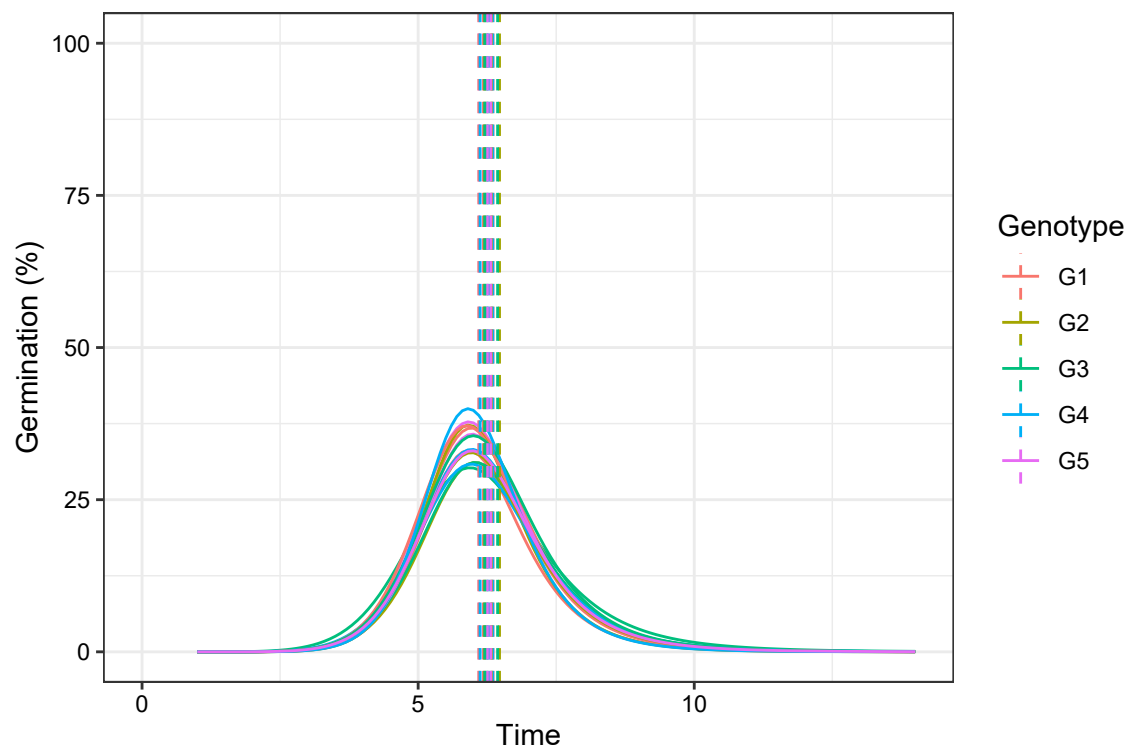
```
plot(fits, group.col = "Genotype", annotate = "mgt")
```



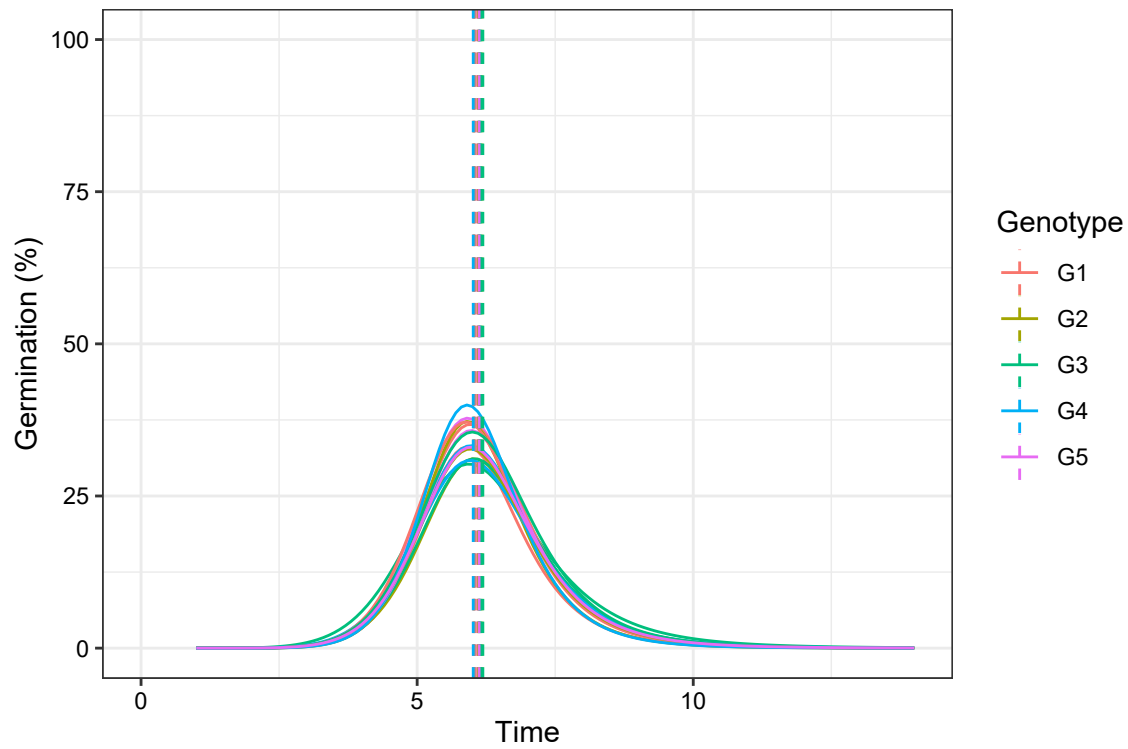
```
plot(fits, group.col = "Genotype", annotate = "uniformity")
```



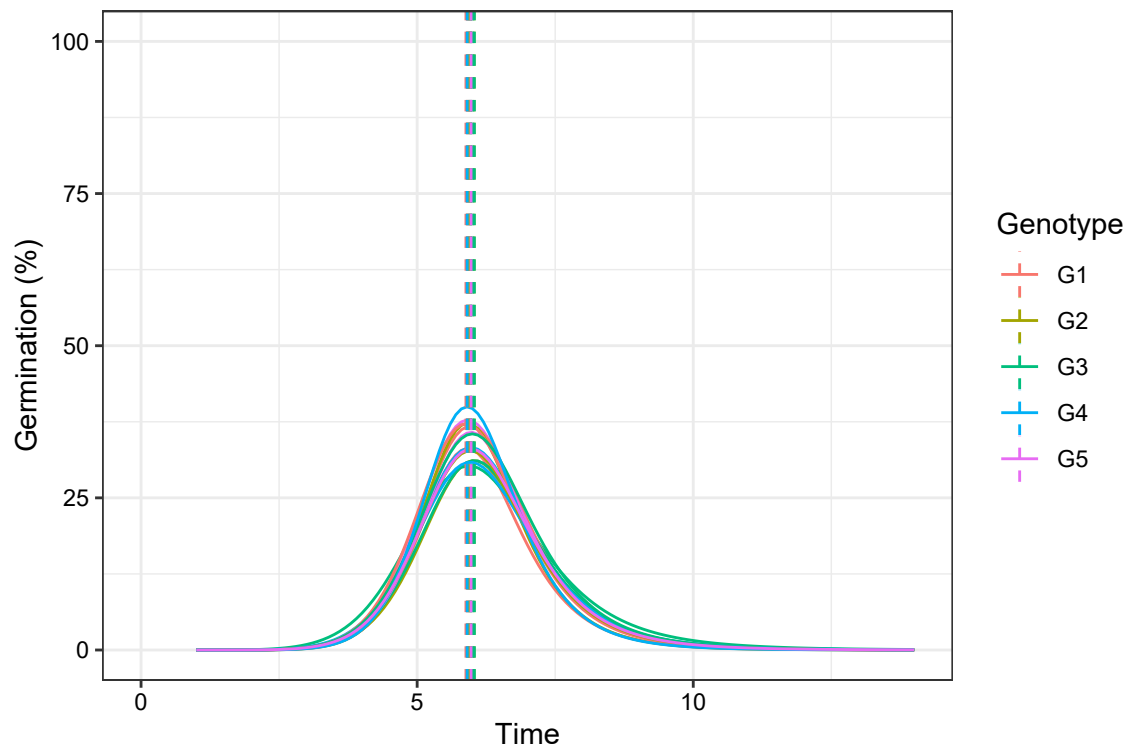
```
# Plot ROG curves with annotations
plot(fits, rog = TRUE, group.col = "Genotype", annotate = "t50.total")
```



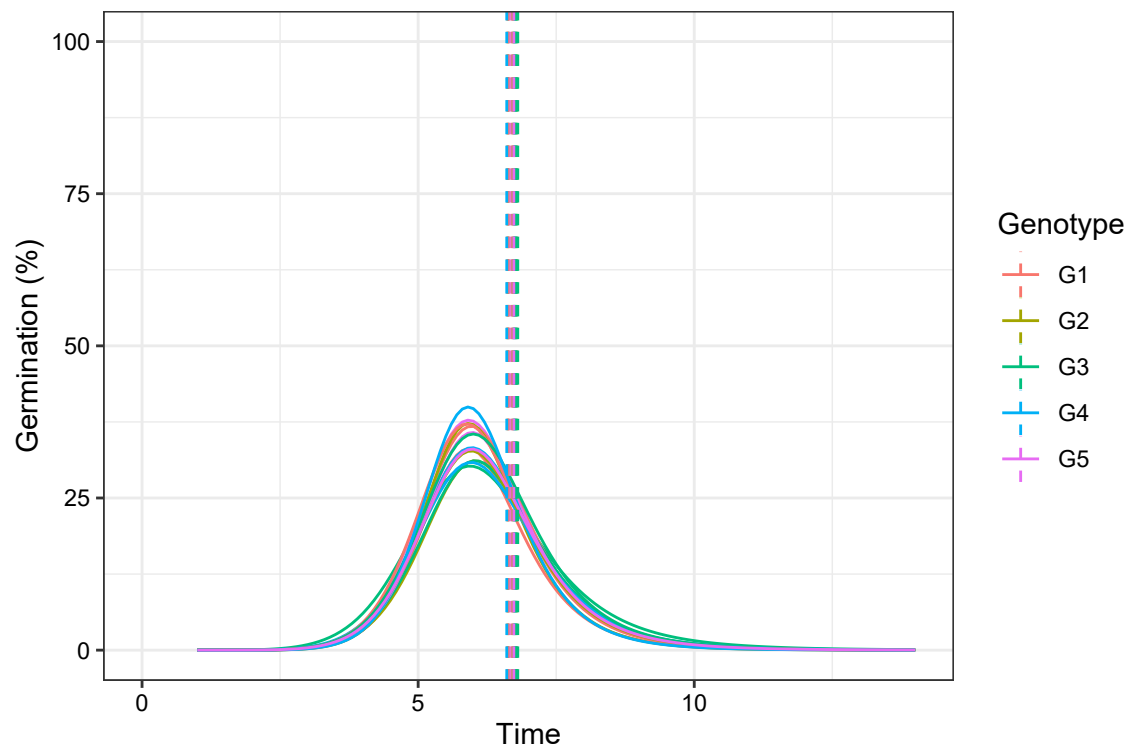
```
plot(fits, rog = TRUE, group.col = "Genotype", annotate = "t50.germ")
```



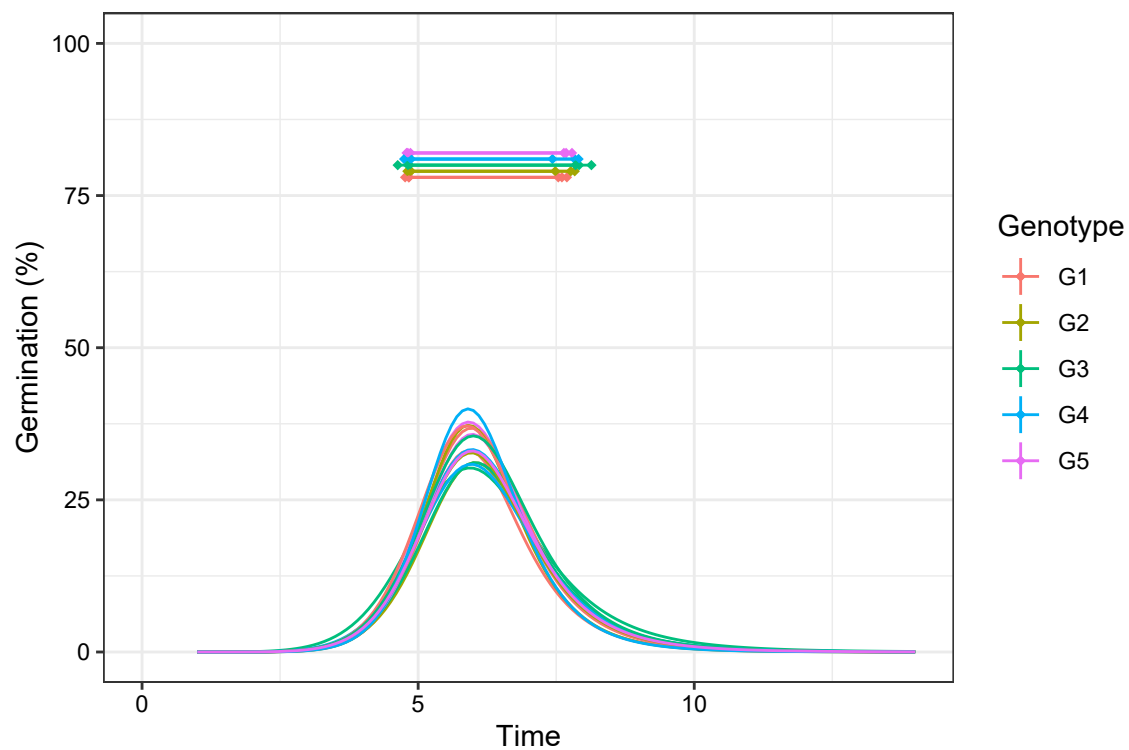
```
plot(fits, rog = TRUE, group.col = "Genotype", annotate = "tmgr")
```




```
plot(fits, rog = TRUE, group.col = "Genotype", annotate = "mgt")
```

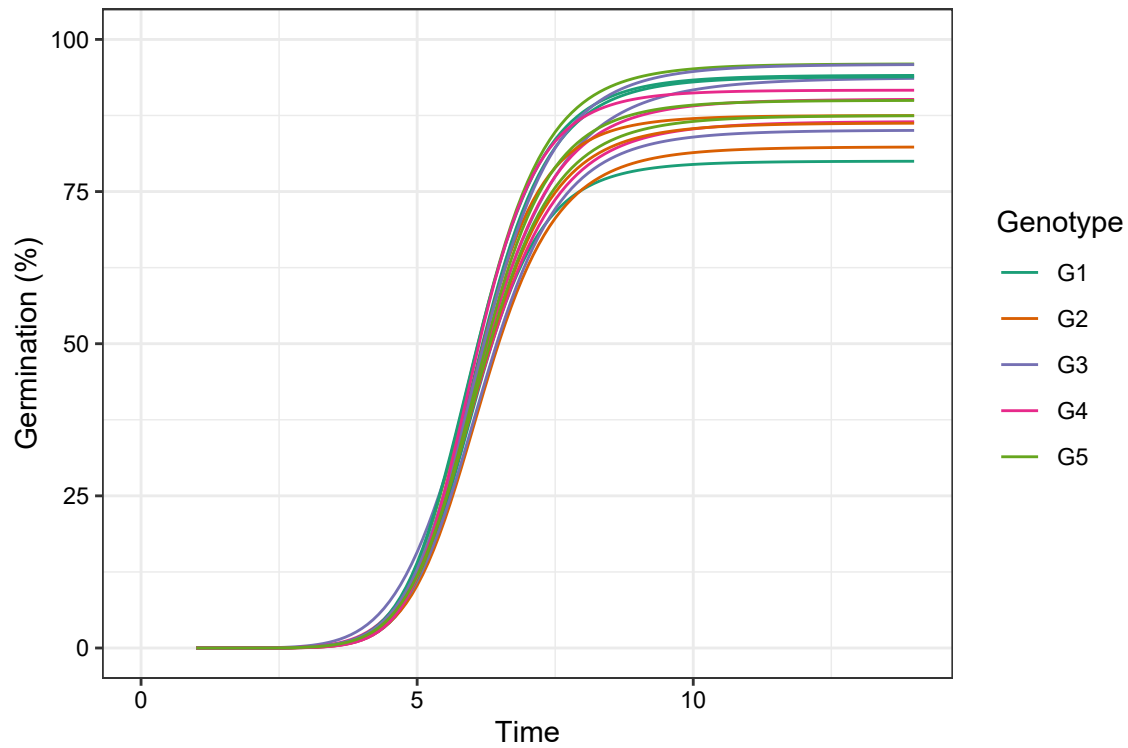


```
plot(fits, rog = TRUE, group.col = "Genotype", annotate = "uniformity")
```

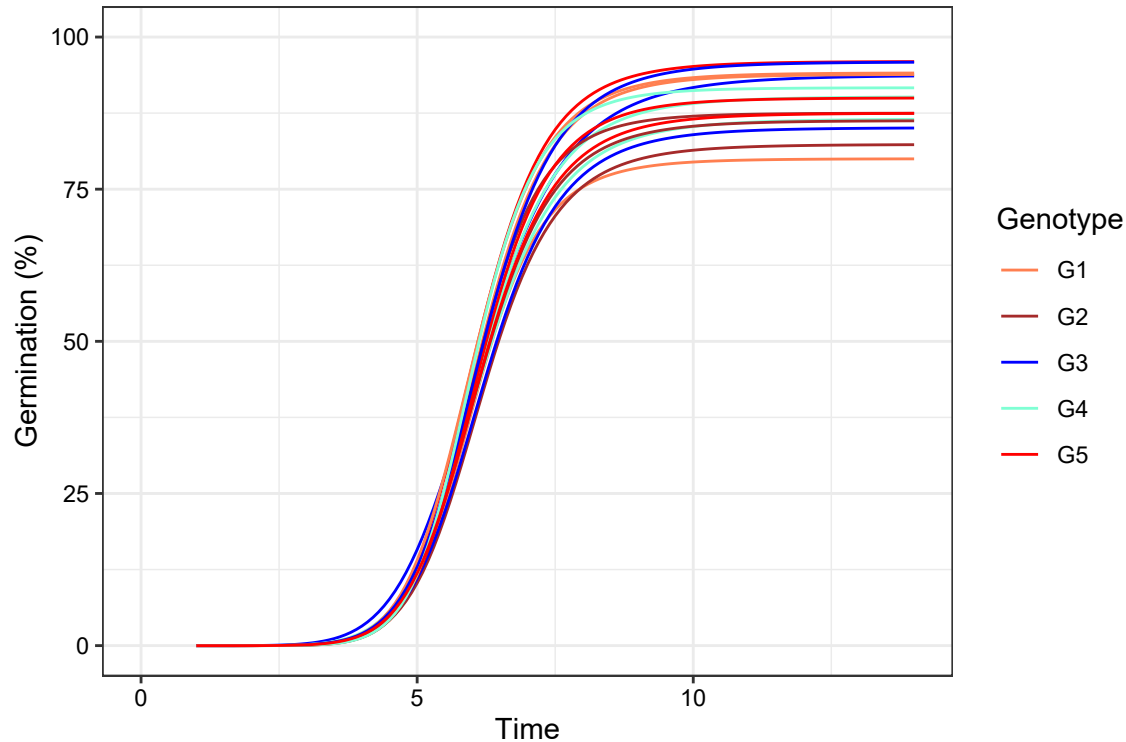


```
# Change colour of curves using ggplot2 options
library(ggplot2)
curvesplot <- plot(fits, group.col = "Genotype")

# 'Dark2' palette from RColorBrewer
curvesplot + scale_colour_brewer(palette = "Dark2")
```



```
# Manual colours
curvesplot +
  scale_colour_manual(values = c("Coral", "Brown", "Blue",
                                "Aquamarine", "Red"))
```



Citing `germinationmetrics`

To cite the R package '`germinationmetrics`' in publications use:

Aravind, J., Vimala Devi, S., Radhamani, J., Jacob, S. R., and Kalyani Srinivasan (2020).
`germinationmetrics`: Seed Germination Indices and Curve Fitting. R package version 0.1.4.9000,
<https://github.com/aravind-j/germinationmetrics><https://cran.r-project.org/package=germinationmetrics>.

A BibTeX entry for LaTeX users is

```
@Manual{,
  title = {germinationmetrics: Seed Germination Indices and Curve Fitting},
  author = {J. Aravind and S. {Vimala Devi} and J. Radhamani and Sherry Rachel Jacob and {Kalyani Sri},
  year = {2020},
  note = {R package version 0.1.4.9000},
  note = {https://github.com/aravind-j/germinationmetrics},
  note = {https://cran.r-project.org/package=germinationmetrics},
}
```

This free and open-source software implements academic research by the authors and co-workers. If you use it, please support the project by citing the package.

Session Info

```
sessionInfo()
```

```
R Under development (unstable) (2020-06-10 r78678)
Platform: x86_64-w64-mingw32/x64 (64-bit)
```

Running under: Windows 10 x64 (build 19041)

Matrix products: default

locale:

```
[1] LC_COLLATE=English_India.1252  LC_CTYPE=English_India.1252    LC_MONETARY=English_India.1252
[4] LC_NUMERIC=C                    LC_TIME=English_India.1252
```

attached base packages:

```
[1] stats      graphics  grDevices  utils      datasets  methods    base
```

other attached packages:

```
[1] germinationmetrics_0.1.4.9000 ggplot2_3.3.1
```

loaded via a namespace (and not attached):

```
[1] whoami_1.3.0          nlme_3.1-148          bitops_1.0-6          fs_1.4.1
[5] xopen_1.0.0           usethis_1.6.1         devtools_2.3.0        RColorBrewer_1.1-2
[9] covr_3.5.0            httr_1.4.1            rprojroot_1.3-2       hunspell_3.0
[13] tools_4.1.0           backports_1.1.7       R6_2.4.1              lazyeval_0.2.2
[17] colorspace_1.4-1      withr_2.2.0           tidyselect_1.1.0      prettyunits_1.1.1
[21] processx_3.4.2        curl_4.3              compiler_4.1.0        cli_2.0.2
[25] xml2_1.3.2            desc_1.2.0            labeling_0.3          scales_1.1.1
[29] callr_3.4.3           goodpractice_1.0.2.9000 pkgdown_1.5.1         stringr_1.4.0
[33] digest_0.6.25         rmarkdown_2.2         lintr_2.0.1           pkgconfig_2.0.3
[37] htmltools_0.4.0       bibtex_0.4.2.2        sessioninfo_1.1.1     highr_0.8
[41] rlang_0.4.6           rstudioapi_0.11.0-9000 farver_2.0.3          generics_0.0.2
[45] jsonlite_1.6.1        dplyr_1.0.0           RCurl_1.98-1.2        magrittr_1.5
[49] Rcpp_1.0.4.6          munsell_0.5.0         fansi_0.4.1           lifecycle_0.2.0
[53] stringi_1.4.6         yaml_2.2.1            mathjaxr_1.0-0        gbrd_0.4-11
[57] MASS_7.3-51.6         pkgbuild_1.0.8        plyr_1.8.6            grid_4.1.0
[61] ggrepel_0.8.2         crayon_1.3.4          lattice_0.20-41       pander_0.6.3
[65] knitr_1.28            ps_1.3.3              pillar_1.4.4          reshape2_1.4.4
[69] clisymbols_1.2.0      pkgload_1.1.0         XML_3.99-0.3          glue_1.4.1
[73] packrat_0.5.0         praise_1.0.0           evaluate_0.14         rex_1.2.0
[77] data.table_1.12.8     remotes_2.1.1         vctrs_0.3.1          Rdpack_0.11-2
[81] testthat_2.3.2        gtable_0.3.0          purrr_0.3.4           rcmdcheck_1.3.3
[85] tidyr_1.1.0           assertthat_0.2.1      xfun_0.14             broom_0.5.6
[89] roxygen2_7.1.0        cyclocomp_1.1.0       minpack.lm_1.2-1      tibble_3.0.1
[93] memoise_1.1.0         ellipsis_0.3.1        xmlparsedata_1.0.3
```

References

- Allan, R. E., Vogel, O. A., and Peterson, C. J. (1962). Seedling emergence rate of fall-sown wheat and its association with plant height and coleoptile length. *Agronomy Journal* 54, 347. doi:[10/cm7jct](https://doi.org/10.1007/jct).
- Al-Mudaris, M. A. (1998). Notes on various parameters recording the speed of seed germination. *Der Tropenlandwirt-Journal of Agriculture in the Tropics and Subtropics* 99, 147–154.
- AOSA (1983). *Seed Vigor Testing Handbook*. Ithaca, NY, USA: Association of Official Seed Analysts.
- Baskin, C. C., and Baskin, J. M. (1998). *Seeds: Ecology, Biogeography, and Evolution of Dormancy and Germination*. San Diego: Academic Press.
- Bewley, J. D., and Black, M. (1994). *Seeds: Physiology of Development and Germination*. New York, USA: Plenum Publishing Corporation Available at: <https://www.cabdirect.org/cabdirect/abstract/19950315483>.

- Bilbro, J. D., and Wanjura, D. F. (1982). Soil crusts and cotton emergence relationships. *Transactions of the ASAE* 25, 1484–1487. doi:[10/ggjc3x](https://doi.org/10.2307/3023333).
- Bonner, F. T. (1967). Ideal sowing depth for sweetgum seed. *Tree Planters' Notes* 18, 1–1. Available at: <https://www.fs.usda.gov/treesearch/pubs/download/42583.pdf>.
- Bouton, J. H., Dudeck, A. E., and Smith, R. L. (1976). Germination in freshly harvested seed of centipedegrass. *Agronomy Journal* 68, 991. doi:[10/cskpqh](https://doi.org/10.2307/2709444).
- Bradbeer, J. W. (1988). *Seed Dormancy and Germination*. Glasgow; London: Blackie Available at: www.springer.com/in/book/9780216916364 [Accessed January 15, 2018].
- Brown, R. F., and Mayer, D. G. (1988). Representing cumulative germination. 1. A critical analysis of single-value germination indices. *Annals of Botany* 61, 117–125. doi:[10.1093/oxfordjournals.aob.a087534](https://doi.org/10.1093/oxfordjournals.aob.a087534).
- Chaudhary, T. N., and Ghildyal, B. P. (1970). Effect of temperature associated with levels of bulk density on rice seedling emergence. *Plant and Soil* 33, 87–90. doi:[10/bzdwmnt](https://doi.org/10.1007/BF02631001).
- Coolbear, P., Francis, A., and Grierson, D. (1984). The effect of low temperature pre-sowing treatment on the germination performance and membrane integrity of artificially aged tomato seeds. *Journal of Experimental Botany* 35, 1609–1617. doi:[10.1093/jxb/35.11.1609](https://doi.org/10.1093/jxb/35.11.1609).
- Czabator, F. J. (1962). Germination value: An index combining speed and completeness of pine seed germination. *Forest Science* 8, 386–396. doi:[10.1093/forestscience/8.4.386](https://doi.org/10.1093/forestscience/8.4.386).
- Djavanshir, K., and Pourbeik, H. (1976). Germination value-A new formula. *Silvae genetica* 25, 79–83.
- Edmond, J. B., and Drapala, W. J. (1958). The effects of temperature, sand and soil, and acetone on germination of okra seed. *Proceedings of the American Society for Horticultural Science* 71, 428–434.
- Edwards, T. I. (1932). Temperature relations of seed germination. *The Quarterly Review of Biology* 7, 428–443.
- El-Kassaby, Y. A., Moss, I., Kolotelo, D., and Stoehr, M. (2008). Seed germination: Mathematical representation and parameters extraction. *Forest Science* 54, 220–227. doi:[10.1093/forestscience/54.2.220](https://doi.org/10.1093/forestscience/54.2.220).
- Ellis, R. H., and Roberts, E. H. (1980). Improved equations for the prediction of seed longevity. *Annals of Botany* 45, 13–30. doi:[10.1093/oxfordjournals.aob.a085797](https://doi.org/10.1093/oxfordjournals.aob.a085797).
- Erbach, D. C. (1982). Tillage for continuous corn and corn-soybean rotation. *Transactions of the ASAE* 25, 906–9911. doi:[10/ggjc4g](https://doi.org/10.2307/3023333).
- Evett, L. L., and Burnside, O. C. (1972). Germination and seedling development of common milkweed and other species. *Weed Science* 20, 371–378. doi:[10.1017/S004317450003589x](https://doi.org/10.1017/S004317450003589x).
- Fakorede, M. A. B., and Agbana, S. B. (1983). Heterotic effects and association of seedling vigour with mature characteristics and grain yield in some tropical maize cultivars. *Maydica* 28, 327–338.
- Fakorede, M. A. B., and Ayoola, A. O. (1980). Relation between seedling vigor and selection for yield improvement in maize. *Maydica* 25, 135–147.
- Fakorede, M. A. B., and Ojo, D. K. (1981). Variability for seedling vigour in maize. *Experimental Agriculture* 17, 195–201. doi:[10/cxr9cn](https://doi.org/10.1017/cxr9cn).
- Farooq, M., Basra, S. M. A., Ahmad, N., and Hafeez, K. (2005). Thermal hardening: A new seed vigor enhancement tool in rice. *Journal of Integrative Plant Biology* 47, 187–193. doi:[10.1111/J.1744-7909.2005.00031.x](https://doi.org/10.1111/J.1744-7909.2005.00031.x).
- George, D. W. (1961). Influence of germination temperature on the expression of post-harvest dormancy in wheat. *Crop Science Abstracts; Western Society of Crop Science Annual Meeting, 1961*, 15.
- Goloff, A. A., and Bazzaz, F. A. (1975). A germination model for natural seed populations. *Journal of Theoretical Biology* 52, 259–283. doi:[10.1016/0022-5193\(75\)90001-6](https://doi.org/10.1016/0022-5193(75)90001-6).

- Gomes, F. P. (1960). *Curso De Estatística Experimental*. Escola Superior de Agricultura Luiz de Queiroz, Universidade de São Paulo Available at: <https://books.google.de/books?id=ZckqGwAACAAJ>.
- Goodchild, N. A., and Walker, M. G. (1971). A method of measuring seed germination in physiological studies. *Annals of Botany* 35, 615–621. Available at: <https://www.jstor.org/stable/42908843> [Accessed January 15, 2018].
- Gordon, A. G. (1969). Some observations on the germination energy tests for cereals. *Proceedings of the Association of Official Seed Analysts* 59, 58–72. Available at: <https://www.jstor.org/stable/23432357> [Accessed December 11, 2018].
- Gordon, A. G. (1971). The germination resistance test - A new test for measuring germination quality of cereals. *Canadian Journal of Plant Science* 51, 181–183. doi:[10/fh6586](https://doi.org/10.1007/bf06586).
- Grose, R. J., and Zimmer, W. J. (1958). Some laboratory germination responses of the seeds of river red gum, *Eucalyptus camaldulensis* Dehn. Syn. *Eucalyptus rostrata* Schlecht. *Australian Journal of Botany* 6, 129. doi:[10/bkp42t](https://doi.org/10.1007/bf0642t).
- Haugland, E., and Brandsaeter, L. O. (1996). Experiments on bioassay sensitivity in the study of allelopathy. *Journal of Chemical Ecology* 22, 1845–1859.
- Heydecker, W. (1972). *Seed Ecology. Proceedings of the Nineteenth Easter School in Agricultural Science, University of Nottingham, 1972*. University Park, USA: Pennsylvania State University Press.
- Hsu, F. H., and Nelson, C. J. (1986). Planting date effects on seedling development of perennial warm-season forage grasses. I. Field emergence. *Agronomy Journal* 78, 33–38. doi:[10/bh39mm](https://doi.org/10.1007/bf0639mm).
- ISTA (2015). Chapter 5: The germination test. *International Rules for Seed Testing. International Seed Testing Association, Zurich, Switzerland*. 2015, i–5–56. Available at: <https://doi.org/10.15258/istarules.2015.05>.
- Kader, M. A. (2005). A comparison of seed germination calculation formulae and the associated interpretation of resulting data. *Journal and Proceedings of the Royal Society of New South Wales* 138, 65–75.
- Kendrick, R. E., and Frankland, B. (1969). Photocontrol of germination in *Amaranthus caudatus*. *Planta* 85, 326–339. doi:[10.1007/bf00381281](https://doi.org/10.1007/bf00381281).
- Khamassi, K., Harbaoui, K., Jaime, A. T. da S., and Jeddi, F. B. (2013). Optimal germination temperature assessed by indices and models in field bean (*Vicia faba* L. Var. *Minor*). *Agriculturae Conspectus Scientificus* 78, 131–136. Available at: <https://hrcak.srce.hr/104663>.
- Khan, M. A., and Ungar, I. A. (1984). The effect of salinity and temperature on the germination of polymorphic seeds and growth of *Atriplex triangularis* Willd. *American Journal of Botany* 71, 481–489. doi:[10.2307/2443323](https://doi.org/10.2307/2443323).
- Khandakar, A. L., and Bradbeer, J. W. (1983). Jute seed quality. *Bangladesh Agricultural Research Council, Dhaka*.
- Kotowski, F. (1926). Temperature relations to germination of vegetable seeds. *Proceedings of the American Society for Horticultural Science* 23, 176–184.
- Labouriau, L. G. (1983a). *A Germinação Das Sementes*. Organização dos Estados Americanos. Programa Regional de Desenvolvimento Científico e Tecnológico. Série de Biologia. Monografia 24.
- Labouriau, L. G. (1983b). Uma nova linha de pesquisa na fisiologia da germinação das sementes. *Anais do XXXIV Congresso Nacional de Botânica. SBB, Porto Alegre*, 11–50.
- Labouriau, L. G., and Valadares, M. E. B. (1976). On the germination of seeds of *Calotropis procera* (Ait.) Ait. F. *Anais da Academia Brasileira de Ciências* 48.
- Lyon, J. L., and Coffelt, R. J. (1966). Rapid method for determining numerical indexes for time-course curves. *Nature* 211, 330–330. doi:[10/b46knw](https://doi.org/10.1038/b46knw).

- Maguire, J. D. (1962). Speed of germination - Aid in selection and evaluation for seedling emergence and vigor. *Crop Science* 2, 176–177. doi:[10.2135/cropsci1962.0011183x000200020033x](https://doi.org/10.2135/cropsci1962.0011183x000200020033x).
- Melville, A. H., Galletta, G. J., Draper, A. D., and Ng, T. J. (1980). Seed germination and early seedling vigor in progenies of inbred strawberry selections. *HortScience* 15, 749–750.
- Mock, J. J., and Eberhart, S. A. (1972). Cold tolerance in adapted maize populations. *Crop Science* 12, 466–469. doi:[10/cvtgtd](https://doi.org/10/cvtgtd).
- Negm, F. B., and Smith, O. E. (1978). Effects of Ethylene and Carbon Dioxide on the Germination of Osmotically Inhibited Lettuce Seed. *Plant Physiology* 62, 473–476. doi:[10/bd98fn](https://doi.org/10/bd98fn).
- Nichols, M. A., and Heydecker, W. (1968). Two approaches to the study of germination data. *Proceedings of the International Seed Testing Association* 33, 531–540.
- Primack, R. B. (1985). Longevity of individual flowers. *Annual Review of Ecology and Systematics* 16, 15–37. doi:[10.1146/annurev.es.16.110185.000311](https://doi.org/10.1146/annurev.es.16.110185.000311).
- Quintanilla, L. G., Pajarón, S., Pangua, E., and Amigo, J. (2000). Effect of temperature on germination in northernmost populations of *Culcita macrocarpa* and *Woodwardia radicans*. *Plant Biology* 2, 612–617. doi:[10.1055/s-2000-16638](https://doi.org/10.1055/s-2000-16638).
- Ranal, M. A. (1999). Effects of temperature on spore germination in some fern species from semideciduous mesophytic forest. *American Fern Journal* 89, 149. doi:[10.2307/1547349](https://doi.org/10.2307/1547349).
- Ranal, M. A., and Santana, D. G. de (2006). How and why to measure the germination process? *Brazilian Journal of Botany* 29, 1–11. doi:[10.1590/s0100-84042006000100002](https://doi.org/10.1590/s0100-84042006000100002).
- Reddy, L. V. (1978). Effect of temperature on seed dormancy and alpha-amylase activity during kernel maturation and germination in wheat (*Triticum aestivum* L.) Cultivars. Available at: https://ir.library.oregonstate.edu/concern/graduate_thesis_or_dissertations/1j92gb854.
- Reddy, L. V., Metzger, R. J., and Ching, T. M. (1985). Effect of temperature on seed dormancy of wheat. *Crop Science* 25, 455. doi:[10.2135/cropsci1985.0011183X002500030007x](https://doi.org/10.2135/cropsci1985.0011183X002500030007x).
- Santana, D. G. de, and Ranal, M. A. (2004). *Análise Da Germinação: Um Enfoque Estatístico*. Brasília: Universidade de Brasília.
- Schrader, J. A., and Graves, W. R. (2000). Seed germination and seedling growth of *Alnus maritima* from its three disjunct populations. *Journal of the American Society for Horticultural Science* 125, 128–134. Available at: <http://journal.ashspublications.org/content/125/1/128> [Accessed December 12, 2018].
- Scott, S. J., Jones, R. A., and Williams, W. A. (1984). Review of data analysis methods for seed germination. *Crop Science* 24, 1192–1199. doi:[10.2135/cropsci1984.0011183x002400060043x](https://doi.org/10.2135/cropsci1984.0011183x002400060043x).
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal* 27, 379–423. doi:[10.1002/j.1538-7305.1948.tb01338.x](https://doi.org/10.1002/j.1538-7305.1948.tb01338.x).
- Shmueli, M., and Goldberg, D. (1971). Emergence, early growth, and salinity of five vegetable crops germinated by sprinkle and trickle irrigation in an arid zone. *HortScience* 6, 563–565.
- Smith, P. G., and Millet, A. H. (1964). Germinating and sprouting responses of the tomato at low temperatures. *Journal of the American Society for Horticultural Science* 84, 480–484.
- Throneberry, G. O., and Smith, F. G. (1955). Relation of respiratory and enzymatic activity to corn seed viability. *Plant Physiology* 30, 337–343.
- Timson, J. (1965). New Method of Recording Germination Data. *Nature* 207, 216. doi:[10.1038/207216a0](https://doi.org/10.1038/207216a0).
- Tucker, H., and Wright, L. N. (1965). Estimating rapidity of germination. *Crop Science* 5, 398–399. doi:[10.2135/cropsci1965.0011183X000500050006x](https://doi.org/10.2135/cropsci1965.0011183X000500050006x).

Wardle, D. A., Ahmed, M., and Nicholson, K. S. (1991). Allelopathic influence of nodding thistle (*Carduus nutans* L.) Seeds on germination and radicle growth of pasture plants. *New Zealand Journal of Agricultural Research* 34, 185–191. doi:[10.1080/00288233.1991.10423358](https://doi.org/10.1080/00288233.1991.10423358).

Went, F. W. (1957). *The experimental control of plant growth*. Chronica Botanica Co., Waltham, Mass., USA; The Ronald Press Co., New York.