Package ‘TapeR’

January 11, 2022

Type Package
Title Flexible Tree Taper Curves Based on Semiparametric Mixed Models
Version 0.5.0
Date 2022-01-11
Description Implementation of functions for fitting taper curves (a semiparametric linear mixed effects taper model) to diameter measurements along stems. Further functions are provided to estimate the uncertainty around the predicted curves, to calculate timber volume (also by sections) and marginal (e.g., upper) diameters. For cases where tree heights are not measured, methods for estimating additional variance in volume predictions resulting from uncertainties in tree height models (tariffs) are provided. The example data include the taper curve parameters for Norway spruce used in the 3rd German NFI fitted to 380 trees and a subset of section-wise diameter measurements of these trees. The functions implemented here are detailed in Kublin, E., Breidenbach, J., Kaendler, G. (2013) <doi:10.1007/s10342-013-0715-0>.
License GPL (>= 2)
LazyLoad yes
Depends nlme, splines, pracma
Suggests testthat
RoxygenNote 7.1.2
NeedsCompilation no
Author Edgar Kublin [aut], Johannes Breidenbach [ctb], Christian Vonderach [ctb, cre]
Maintainer Christian Vonderach <christian.vonderach@forst.bwl.de>
Repository CRAN
Date/Publication 2022-01-11 22:52:46 UTC

R topics documented:

TapeR-package .................................................................................. 2
TapeR-package

Flexible tree taper curves based on Semiparametric Mixed Models.

Description

This package implements functions for fitting taper curves (a semiparametric linear mixed effects taper model) to diameter measurements along stems. Further functions are provided to estimate the variance/confidence intervals around the predicted curves, to calculate timber volume (also by sections) and marginal (e.g., upper) diameters. For cases where tree heights are not measured, methods for estimating additional variance in volume predictions resulting from uncertainties in tree height models (tariffs) are provided. The example data include the taper curve parameters for Norway spruce used in the 3rd German NFI fitted to 380 trees and a subset of section-wise diameter measurements of these trees.
**Details**

<table>
<thead>
<tr>
<th>Package:</th>
<th>TapeR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>License:</td>
<td>GPL-2</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

Fits taper models using diameter measurements along the stem. Uses fitted models and arbitrary numbers of diameter measurements to estimate diameter at any position along the stem. Estimates timber volume from the taper curve. Provides variances for all estimates.

**Author(s)**

Edgar Kublin

Maintainer: Johannes Breidenbach <job@skogoglandskap.no>

**References**


**See Also**

TapeR_FIT_LME.f, E_DHx_HmDm_HT.f, DxHx.df, SK.par.lme, HT.par

---

**BSplines**  
*builds B-Splines Matrix with appropriate knots for taper fitting*

**Description**

Internal function not usually called by users

**Usage**

BSplines(knots = c(seq(0, 1, 0.1)), ord = 4, der = 0, x = c(seq(0, 1, 0.01)), ...)

**Arguments**

- **knots**: knot positions for spline function
- **ord**: order of the spline function
- **der**: derivatives
- **x**: height measurements
- **...**: not currently used
Details

internally `splineDesign` is called

Value

B-Splines matrix build using `splineDesign`

Author(s)

Edgar Kublin

---

**CdN_DHxHt.f**

percentile for estimated taper curve diameter

---

Description

Internal function not usually called by users

Usage

`CdN_DHxHt.f(Ht, Hx, qD, Hm, Dm, par.lme, Rfn, ...)`

Arguments

- **Ht**: tree height
- **Hx**: Numeric vector of stem heights (m) along which to return the expected diameter.
- **qD**: vector of quantiles, passed to `pnorm`
- **Hm**: measured height of respective diameters Dm
- **Dm**: measured diameter
- **par.lme**: List of taper model parameters obtained by `TapeR_FIT_LME.f`
- **Rfn**: list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- **...**: not currently used

Value

percentile for estimated taper curve diameter at position Hx given Ht, Hm and Dm

Author(s)

Edgar Kublin
dN.f

evaluate Normal distribution

Description
Internal function not usually called by users

Usage
dN.f(x, mw, sd, ...)

Arguments
x vector of quantiles
mw vector of means
sd vector of standard deviations
... not currently used

Value
numeric density of normal distribution

Author(s)
Edgar Kublin

DxHx.df
Example dataset of 10 trees with 10 diameter and height measurements for each tree.

Description
Example dataset of 10 trees with 10 diameter and height measurements for each tree.

Usage
data(DxHx.df)

Format
A data frame with 172 observations on the following 4 variables.

Id Numeric vector of tree IDs.
Dx Numeric vector of diameter measurements.
Hx Numeric vector of height measurements.
Ht Numeric vector of tree height (repeated for each measurement in each tree).
Details

Measured for BWI3.

References


Examples

data(DxHx.df)
head(DxHx.df)

```
EYx_ssp.f
```

*estimate cubic smoothing spline with linear extrapolation*

Description

Internal function not usually called by users

Usage

```
EYx_ssp.f(knt, coe, x, ...)
```

Arguments

- **knt**: knots position of B-Splines
- **coe**: estimated coefficient for B-Splines
- **x**: position at which to evaluate B-Splines model
- **...**: not currently used

Value

expected diameter given knots and coefficients at position (height) x.

Author(s)

Edgar Kublin
Calibrates a taper curve based on at least one diameter measurement and returns the expected diameters and approximate variances.

Usage

```r
E_DHx_HmDm_HT.f(Hx, Hm, Dm, mHt, sHt = 0, par.lme, Rfn = list(fn = "sig2"), ...)
```

Arguments

- **Hx**: Numeric vector of stem heights (m) along which to return the expected diameter.
- **Hm**: Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as Dm.
- **Dm**: Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm.
- **mHt**: Scalar. Tree height (m).
- **sHt**: Scalar. Standard deviation of stem height. Can be 0 if height was measured without error.
- **par.lme**: List of taper model parameters obtained by `TapeR_FIT_LME.f`.
- **Rfn**: List with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions.
- **...**: not currently used

Details

calibrates the tree specific taper curve and calculates approximate confidence intervals, which can be useful for plotting. Uncertainty resulting from tariff height estimates if tree height was not measured is incorporated. Using Rfn the taper curve can be forced through the measured diameters, c.f. `resVar`.
Value

a list holding six elements:

- **DHx**: Numeric vector of diameters (cm) (expected value) along the heights given by **Hx**.
- **Hx**: Numeric vector of heights (m) along which to return the expected diameter.
- **MSE_Mean**: Mean squared error for the expected value of the diameter.
- **CI_Mean**: Confidence interval. Matrix of the 95% conf. int. for the expected value of the diameter (cm). First column: lower limit, second column: mean, third column: upper limit.
- **MSE_Pred**: Mean squared error for the prediction of the diameter.
- **CI_Mean**: Prediction interval. Matrix of the 95% conf. int. for the prediction of the diameter (cm). First column: lower limit, second column: mean, third column: upper limit.
- **Rfn**: Function applied for estimated or assumed residual variance.

Author(s)

Edgar Kublin

References


See Also

*TapeR_FIT_LME.f*

Examples

```r
# example data
data(DxHx.df)
# taper curve parameters based on all measured trees
data(SK.par.lme)

#select data of first tree
Idi <- (DxHx.df['Id'] == unique(DxHx.df$Id)[1])
(tree1 <- DxHx.df[Idi,])

## Predict the taper curve based on the diameter measurement in 2 m
## height and known height
tc.tree1 <- E_DHx_HmDm_HT.f(Hx=1:tree1$Ht[1],
   Hm=tree1$Hx[3],
   Hm=tree1$Dx[3],
   mHt = tree1$Ht[1],
   sHt = 0,
   par.lme = SK.par.lme)

#plot the predicted taper curve
plot(tc.tree1$Hx, tc.tree1$DHx, type="l", las=1)
#lower CI
lines(tc.tree1$Hx, tc.tree1$CI_Mean[,1], lty=2)
```
# upper CI
lines(tc.tree1$Hx, tc.tree1$CI_Mean[,3], lty=2)
# lower prediction interval
lines(tc.tree1$Hx, tc.tree1$CI_Pred[,1], lty=3)
# upper prediction interval
lines(tc.tree1$Hx, tc.tree1$CI_Pred[,3], lty=3)
# add measured diameter used for calibration
points(tree1$Hx[3], tree1$Dx[3], pch=3, col=2)
# add the observations
points(tree1$Hx, tree1$Dx)

## feature of forcing taper curve through measured diameters
i <- c(3, 6)
tc.tree1 <- E_DHx_HmDm_HT.f(Hx=seq(0, tree1$Ht[1], 0.1),
  Hm=tree1$Hx[i],
  Dm=tree1$Dx[i],
  mHt = tree1$Ht[1],
  sHt = 0,
  par.lme = SK.par.lme,
  Rfn=list(fn="sig2"))
tc.tree2 <- E_DHx_HmDm_HT.f(Hx=seq(0, tree1$Ht[1], 0.1),
  Hm=tree1$Hx[i],
  Dm=tree1$Dx[i],
  mHt = tree1$Ht[1],
  sHt = 0,
  par.lme = SK.par.lme,
  Rfn=list(fn="zero"))

# plot the predicted taper curve
plot(tc.tree1$Hx, tc.tree1$DHx, type="l", las=1)
# added taper curve through measurement
points(x=tc.tree2$Hx, y=tc.tree2$DHx, type="l", lty=2)
# closer window
plot(tc.tree1$Hx, tc.tree1$DHx, type="l", las=1, xlim=c(0, 8), ylim=c(24, 30))
# added taper curve through measurement
points(x=tc.tree2$Hx, y=tc.tree2$DHx, type="l", lty=2)
# add measured diameter used for calibration
points(tree1$Hx[i], tree1$Dx[i], pch=3, col=2)
# add the observations
points(tree1$Hx, tree1$Dx)

## apply yet another residual variance function
i <- c(1, 2, 3) # calibrating with 0.5, 1m and 2m, assuming no error in 0.5m
zrv <- tree1$Hx[1] / tree1$Ht[1] # assumed zero residual variance
# assumed residual variance per measurement
TapeR:::resVar(reH = tree1$Hx[i] / tree1$Ht[1], fn = "dlnorm",
  sig2 = SK.par.lme$sig2_eps, par = list(zrv=zrv))
tc.tree3 <- E_DHx_HmDm_HT.f(Hx=seq(0, tree1$Ht[1], 0.1),
  Hm=tree1$Hx[i],
  Dm=tree1$Dx[i],
  mHt = tree1$Ht[1],
  sHt = 0,
  par.lme = SK.par.lme,
  Rfn=list(fn="dlnorm", par=list(zrv=zrv)))
Estimate diameter and exact confidence and prediction intervals

Description

Calibrates a taper curve based on at least one diameter measurement and returns the expected diameters and exact variances

Usage

```r
E_DHx_HmDm_HT_CIdHt.f(
  Hx,
  Hm,
  Dm,
  mHt,
  sHt,
  par.lme,
  Rfn = list(fn = "sig2"),
  ...
)
```

Arguments

- **Hx**: Numeric vector of stem heights (m) along which to return the expected diameter.
- **Hm**: Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as Dm.
- **Dm**: Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm.
- **mHt**: Scalar. Tree height (m).
- **sHt**: Scalar. Standard deviation of stem height. Can be 0 if height was measured without error.
- **par.lme**: List of taper model parameters obtained by `TapeR_FIT_LME.f`.
- **Rfn**: list with function model name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- **...**: not currently used

Details

Calibrates the tree specific taper curve and calculates ’exact’ confidence intervals, which can be useful for plotting. Attention: this function is somewhat time-consuming.
Value

A matrix with six columns:

- **Hx**: Numeric vector of heights (m) along which to return the expected diameter.
- **q\_DHx\_u**: Lower confidence interval (cm). (95% CI except for estimates close to the stem tip.)
- **DHx**: Diameter estimate (cm).
- **q\_DHx\_o**: Upper CI (cm).
- **cP\_DHx\_u**: Probability of observations < q\_DHx\_u.
- **cP\_DHx\_o**: Probability of observations < q\_DHx\_o.

Author(s)

Edgar Kublin

References


See Also

TapeR\_FIT\_LME.f

Examples

```r
# example data
data(DxHx.df)
# taper curve parameters based on all measured trees
data(SK.par.lme)

# select data of first tree
Idi <- (DxHx.df$Id == unique(DxHx.df$Id)[1])
(tree1 <- DxHx.df[Idi,])

## Predict the taper curve based on the diameter measurement in 2 m height and known height
tc.tree1 <- E_DHx_HmDm_HT.f(Hx=1:tree1$Ht[1],
 mHt = tree1$Ht[1],
 par.lme = SK.par.lme)

# plot the predicted taper curve
plot(tc.tree1$Hx, tc.tree1$DHx, type="l", las=1)
# lower CI
lines(tc.tree1$Hx, tc.tree1$CI_Mean[,1], lty=2)
# upper CI
lines(tc.tree1$Hx, tc.tree1$CI_Mean[,3], lty=2)
```
Estimate height of given diameter

Calibrates a taper curve based on at least one diameter measurement and returns the height of a given diameter.

Usage

\[
\text{E\_HDx\_HmDm\_HT.f}(\text{Dx}, \text{Hm}, \text{Dm}, \text{mHt}, \text{sHt} = 0, \text{par.lme}, \text{Rfn} = \text{list(fn} = "\text{sig2}"), \ldots)
\]

Arguments

\begin{itemize}
  \item \textbf{Dx}  
  \hspace{1em} Scalar. Diameter for which to return height.
  \item \textbf{Hm}  
  \hspace{1em} Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as \text{Dm}.
\end{itemize}
**E_DHx_HmDm_HT.f**

**Dm**
Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm.

**mHt**
Scalar. Tree height (m).

**sHt**
Scalar. Standard deviation of stem height. Can be 0 if height was measured without error.

**par.lme**
List of taper model parameters obtained by `TapeR_FIT_LME.f`.

**Rfn**
List with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions

... not currently used

**Details**
returns the height given a certain diameter.

**Value**
A scalar. Estimated height (m) given a diameter.

**Author(s)**
Edgar Kublin

**References**

**See Also**
`TapeR_FIT_LME.f`

**Examples**

```r
# example data
data(DxHx.df)
# taper curve parameters based on all measured trees
data(SK.par.lme)

#select data of first tree
Idi <- (DxHx.df["Id"] == unique(DxHx.df$Id)[1])
(tree1 <- DxHx.df[Idi,])

## Predict the taper curve based on the diameter measurement in 2 m
## height and known height
tc.tree1 <- E_DHx_HmDm_HT.f(Hx=1:tree1$Ht[1],
                           Hm=tree1$Hx[3],
                           Dm=tree1$Dx[3],
                           mHt = tree1$Ht[1],
                           sHt = 0,
                           par.lme = SK.par.lme)
```
#plot the predicted taper curve
plot(tc.tree1$Hx, tc.tree1$DHx, type="l", las=1)

## Calculate the height given a certain diameter threshold, say 8.5 cm
ht.tree1.d8.5 <- E_HDx_HmDm_HT.f (Dx=8.5,
Hm=tree1$Hx[3],
Dm=tree1$Dx[3],
mHt = tree1$Ht[1],
sHt = 1,
par.lme = SK.par.lme,
Rfn=list(fn="sig2"))

# add to plot
points(x=ht.tree1.d8.5, y=8.5, pch=8, col=2, cex=2)

---

### E_VOL_AB_HmDm_HT.f

**Estimate volume for stem and sections**

**Description**

Estimate volume for a complete stem from bottom to tip or for a section defined by lower and upper diameter or height. Variances for estimated volumes are calculated.

**Usage**

```r
E_VOL_AB_HmDm_HT.f(
  Hm,
  Dm,
  mHt,
  sHt = 0,
  A = NULL,
  B = NULL,
  iDH = "D",
  par.lme,
  Rfn = list(fn = "sig2"),
  IA = F,
  nGL = 51,
  ...
)
```

**Arguments**

- **Hm**: Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as Dm.
- **Dm**: Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm.
- **mHt**: Scalar. Tree height (m).
Scalar. Standard deviation of stem height. Can be 0 if height was measured without error.

**A**

Numeric scalar defining the lower threshold of a stem section for volume estimation. Depends on \(iDH\). If \(iDH = "D"\), a diameter (cm), if \(iDH = "H"\), a height (m). If NULL, section starts at lowest point.

**B**

Numeric scalar defining the upper threshold of a stem section for volume estimation. Depends on \(iDH\). If \(iDH = "D"\), a diameter (cm), if \(iDH = "H"\), a height (m). If NULL, section ends at tip.

**iDH**

Character scalar. Either "D" or "H". Type of threshold for section volume estimation. See **A** or **B**.

**par.lme**

List of taper model parameters obtained by TapeR_FIT_LME.f.

**Rfn**

List with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions

**IA**

Logic scalar. If TRUE, variance calculation of height estimate based on 2-point distribution. If FALSE, variance calculation of height estimate based on Normal approximation.

**nGL**

Numeric scalar. Number of support points for numerical integration.

**Details**

Calculates the volume for a complete stem or sections defined by A and B, which might be defined as diameter or height. The parameter Rfn can be used to force the taper curve through the measured points (e.g. by Rfn=list(fn="zero"), cf. resVar).

**Value**

A list holding nine elements:

- **E_VOL**: Estimated volume (m^3).
- **VAR_VOL**: Variance of the volume estimate.
- **Hm**: Height of diameter measurement (m).
- **Dm**: Diameter measurement (cm).
- **Ht**: Tree height (m).
- **Da**: Diameter at lower section threshold (cm).
- **Db**: Diameter at upper section threshold (cm).
- **Ha**: Height at lower section threshold (m).
- **Hb**: Height at upper section threshold (m).
- **Rfn**: Function applied for estimated or assumed residual variance.

**Author(s)**

Edgar Kublin
References


See Also

TapeR_FIT_LME.f

Examples

# example data
data(DxHx.df)
# taper curve parameters based on all measured trees
data(SK.par.lme)

# select data of first tree
Idi <- (DxHx.df[, "Id"] == unique(DxHx.df$Id)[1])
(tree1 <- DxHx.df[Idi, ])

## Calculate the timber volume for the whole stem
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
   Dm=tree1$Dx[3],
   mHt = tree1$Ht[1],
   sHt = 0, # no height variance assumed
   par.lme = SK.par.lme)
VOL$E_VOL # ' expected value
VOL$VAR_VOL # ' corresponding variance
VOL$Rfn

## Calculate the timber volume for the whole stem, using Rfn="zero"
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
   Dm=tree1$Dx[3],
   mHt = tree1$Ht[1],
   sHt = 0, # no height variance assumed
   par.lme = SK.par.lme,
   Rfn = list(fn="zero"))
VOL$E_VOL # ' expected value
VOL$VAR_VOL # ' corresponding variance
VOL$Rfn

## Calculate the timber volume for the whole stem
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
   Dm=tree1$Dx[3],
   mHt = tree1$Ht[1],
   sHt = 1, # no height variance assumed
   par.lme = SK.par.lme)
VOL$E_VOL # ' expected value
VOL$VAR_VOL # ' corresponding variance
VOL$Rfn

## Calculate the timber volume for the whole stem, using Rfn="zero"
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
VOL$E_VOL #' expected value
VOL$VAR_VOL #' corresponding variance
VOL$Rfn

## Calculate the timber volume for a selected section given a height (0.3 - 5 m)
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
  Dm=tree1$Dx[3],
  mHt = tree1$Ht[1],
  sHt = 1,
  par.lme = SK.par.lme,
  A=0.3,
  B=5,
  iDH = "H")
VOL$E_VOL #' expected value
VOL$VAR_VOL #' corresponding variance
VOL$Rfn

## Calculate the timber volume for a selected section given a diameter
## threshold (30cm - 15cm) (negative value if A<B)
VOL <- E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3],
  Dm=tree1$Dx[3],
  mHt = tree1$Ht[1],
  sHt = 1,
  par.lme = SK.par.lme,
  A=30,
  B=15,
  iDH = "D")
VOL$E_VOL #' expected value
VOL$VAR_VOL #' corresponding variance

# Not run:

## The variance estimate resulting from the tree height uncertainty using
## a Normal approximation takes much longer...
```r
ptm <- proc.time()
E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3], Dm=tree1$Dx[3], mHt = tree1$Ht[1],
                   sHt = 1, par.lme = SK.par.lme, IA=FALSE)
proc.time() - ptm

##... than the calculation using a 2-point distribution...
ptm <- proc.time()
E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3], Dm=tree1$Dx[3], mHt = tree1$Ht[1],
                   sHt = 1, par.lme = SK.par.lme, IA=TRUE)
proc.time() - ptm

##...fastest if no height variance is assumed
ptm <- proc.time()
E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3], Dm=tree1$Dx[3], mHt = tree1$Ht[1],
                   sHt = 0, par.lme = SK.par.lme, IA=FALSE)
proc.time() - ptm

## Also the number of supportive points for the numerical integration
## influences the calculation time
ptm <- proc.time()
E_VOL_AB_HmDm_HT.f(Hm=tree1$Hx[3], Dm=tree1$Dx[3], mHt = tree1$Ht[1],
                   sHt = 0, par.lme = SK.par.lme, IA=FALSE, nGL=10)
proc.time() - ptm

##' End(Not run)
```

---

**HT.par**

*Height tariff parameters for estimating tree heights of unmeasured trees in the BWI3.*

---

**Description**

Height is only measured on a subset of the trees on a sample plots. This Height tariff is used to estimate the height of the trees with only a dbh measurement.

**Usage**

```r
data(HT.par)
```

**Format**

The format is: List of 4

- `knt.mw`: num [1:16] 0 0 0 0 19.6 ...
- `coe.mw`: num [1:12] 1.3 7.28 15.1 21.75 24.39 ...
- `knt.sd`: num [1:53] 0 0 0 0 7.52 ...
- `coe.sd`: num [1:49] 0 0.618 1.376 2.142 2.486 ...

**References**

**Hx_root.f**

**Examples**

```
data(HT.par)
## maybe str(HT.par) ; plot(HT.par) ...
```

**Description**

Internal function not usually called by users

**Usage**

```
Hx_root.f(Hx, Dx, Hm, Dm, mHt, sHt, par.lme, Rfn, ...)  
```

**Arguments**

- **Hx**: Numeric vector of stem heights (m) along which to return the expected diameter
- **Dx**: expected diameter
- **Hm**: Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as Dm
- **Dm**: Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm
- **mHt**: Scalar. Tree height (m)
- **sHt**: Scalar. Standard deviation of stem height. Can be 0 if height was measured without error
- **par.lme**: List of taper model parameters obtained by TapeR_FIT_LME.f.
- **Rfn**: list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- **...**: not currently used

**Details**

function is called by unirrot inside E_HDx_HmDm_HT.f

**Value**

deviation between observed diameter Dx and diameter in height Hx.

**Author(s)**

Edgar Kublin
Internal function not usually called by users

Usage

\[ \text{Int}_{-}\text{CdN} \_ \text{DHx} \_ \text{dHt}.f(qD, Hx, Hm, Dm, mHt, sHt, par.lme, Rfn, nGL = 51, \ldots) \]

Arguments

- `qD` vector of quantiles, finally passed to `pnorm`
- `Hx` Numeric vector of stem heights (m) along which to return the expected diameter
- `Hm` Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as `Dm`
- `Dm` Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as `Hm`
- `mHt` Scalar. Tree height (m)
- `sHt` Scalar. Standard deviation of stem height. Can be 0 if height was measured without error
- `par.lme` List of taper model parameters obtained by `TapeR\_FIT\_LME.f`
- `Rfn` list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- `nGL` Numeric scalar. Number of support points for numerical integration
- `\ldots` not currently used

Value

\[ \text{Int}_{-}\text{CdN} \_ \text{dN} \]

Author(s)

Edgar Kublin
Description

Internal function not usually called by users.

Usage

Int_E_VOL_AB_HmDm_HT_dHt.f(
    Hm,
    Dm,
    A = NULL,
    B = NULL,
    iDH = "D",
    mw_HtT,
    sd_HtT,
    par.lme,
    Rfn = list(fn = "sig2"),
    IA = FALSE,
    nGL = 51,
    ...
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hm</td>
<td>Numeric vector of stem heights (m) along which diameter measurements were</td>
</tr>
<tr>
<td></td>
<td>taken for calibration. Can be of length 1. Must be of same length as Dm.</td>
</tr>
<tr>
<td>Dm</td>
<td>Numeric vector of diameter measurements (cm) taken for calibration. Can be</td>
</tr>
<tr>
<td></td>
<td>of length 1. Must be of same length as Hm.</td>
</tr>
<tr>
<td>A</td>
<td>Numeric scalar defining the lower threshold of a stem section for volume</td>
</tr>
<tr>
<td></td>
<td>estimation. Depends on iDH. If iDH = &quot;D&quot;, a diameter (cm), if iDH = &quot;H&quot;,</td>
</tr>
<tr>
<td></td>
<td>a height (m). If NULL, section starts at lowest point.</td>
</tr>
<tr>
<td>B</td>
<td>Numeric scalar defining the upper threshold of a stem section for volume</td>
</tr>
<tr>
<td></td>
<td>estimation. Depends on iDH. If iDH = &quot;D&quot;, a diameter (cm), if iDH = &quot;H&quot;,</td>
</tr>
<tr>
<td></td>
<td>a height (m). If NULL, section ends at tip.</td>
</tr>
<tr>
<td>iDH</td>
<td>Character scalar. Either &quot;D&quot; or &quot;H&quot;. Type of threshold for section volume</td>
</tr>
<tr>
<td></td>
<td>estimation. See A or B.</td>
</tr>
<tr>
<td>mw_HtT</td>
<td>Scalar. Tree height (m)</td>
</tr>
<tr>
<td>sd_HtT</td>
<td>Scalar. Standard deviation of stem height. Can be 0 if height was measured</td>
</tr>
<tr>
<td></td>
<td>without error</td>
</tr>
<tr>
<td>par.lme</td>
<td>List of taper model parameters obtained by TapeR_FIT_LME.f.</td>
</tr>
<tr>
<td>Rfn</td>
<td>List with function name to provide estimated or assumed residual variances</td>
</tr>
<tr>
<td></td>
<td>for the given measurements, optionally parameters for such functions.</td>
</tr>
</tbody>
</table>
IA Logic scalar. If TRUE, variance calculation of height estimate based on 2-point distribution. If FALSE, variance calculation of height estimate based on Normal approximation.

nGL Numeric scalar. Number of support points for numerical integration.

... not currently used

Details
Integrating the taper curve considering uncertainty of height measurement

Value
List with expected volume, variance of volume and squared expected value incorporating the uncertainty of height measurement

Author(s)
Edgar Kublin

Description
Internal function not usually called by users

Usage
qD.rout.f(qD, alpha = 0.975, Hx, Hm, Dm, mHt, sHt, par.lme, Rfn, nGL = 51, ...)

Arguments
qD Vector of quantiles, finally passed to pnorm
alpha Quantile for which root is sought
Hx Numeric vector of stem heights (m) along which to return the expected diameter
Hm Numeric vector of stem heights (m) along which diameter measurements were taken for calibration. Can be of length 1. Must be of same length as Dm
Dm Numeric vector of diameter measurements (cm) taken for calibration. Can be of length 1. Must be of same length as Hm
mHt Scalar. Tree height (m)
sHt Scalar. Standard deviation of stem height. Can be 0 if height was measured without error
par.lme List of taper model parameters obtained by TapeR_fit_lme.f.
Rfn List with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
nGL Numeric scalar. Number of support points for numerical integration.
... not currently used
Value

$q_0$ for given alpha with respect to Int_CdN_DHx_dHt.f

Author(s)

Edgar Kublin

---

**resVar**

*Functions to put different size of uncertainty on given measurements*

Description

When estimating a tree specific taper curve based on given measurements one can modify the assumed measurement uncertainty by these functions.

Usage

```r
resVar(relH, fn, sig2, par = NULL)
```

Arguments

- `relH` relative tree height for which the assumed residual variance should be calculated
- `fn` name of function to be applied as character string
- `sig2` residual variance from fitted model, cf. `TapeR_FIT_LME.f`
- `par` either NULL or a list with parameters to the different functions. See details.

Details

When estimating a tree specific taper curve based on given measurements the residual variance of the model is taken into account to estimate the tree specific random effects. Alternatively, it is possible to make assumptions about the measurement error, eventually at specific relative heights. With that, one can e.g. force the taper curve through the given measurements. Standard behaviour not necessarily leads to passing the measurements, if more than one is given.

Different functions are available. `sig2` applies the model residual variance and hence is the default behaviour. `zero` means assuming no residual variance and forcing the taper curve through the given measurements. Care has to be taken in this case because forcing the taper curve through a lot of measurements might result in implausible results. `linear` interpolates between zero and the given residual variance along the stem, i.e. from bottom to tree top. `bilinear` puts zero variance not at zero but at a predefined location (can be given via `par`). Below and above a linear interpolation is done up to the given residual variance. If zero variance position is not given, it is set at 5% of tree height (approximately height of dbh). `laglinear` assumes zero variance up to a predefined location (defaults to 5% of tree height) and interpolates upwards to the given residual variance of the model. `quadratic` function distributes residual variance according to a quadratic function along the stem. It is build so that zero variance is put at a predefined location (defaults to 5% of tree height) and model residual variance (as a default) at tree top. `dnorm` and `dlnorm` put residual variance in form
of an inverse normal or an inverse log-normal distribution along the stem with a zero-minimum at a predefined location (defaults to 5% of tree height). See examples for a visualisation.

For all functions (except zero and sig2) the point of zero residual variance is defined by par$zrv if given, otherwise set to 0.05. For dnorm one can additionally provide the parameter sd to determine standard deviation. By default it is set to zrv/3; in case of dlnorm one can define lsd (sdlog, cf. dlnorm), which is by default set to 1-sqrt(zrv). It is up to the user to define meaningful parameters and use the functions in appropriate context.

Value

vector of assumed residual variance

Author(s)

Christian Vonderach

Examples

curve(resVar(relH=x, fn = "sig2", 0.5))
curve(resVar(relH=x, fn = "zero", 0.5))
curve(resVar(relH=x, fn = "linear", 0.5))
curve(resVar(relH=x, fn = "bilinear", 0.5))
curve(resVar(relH=x, fn = "laglinear", 0.5))
curve(resVar(relH=x, fn = "quadratic", 0.5))
curve(resVar(relH=x, fn = "dnorm", 0.5))
curve(resVar(relH=x, fn = "dlnorm", 0.5, par=list(zrv=0.2, sd=0.2/3)))
curve(resVar(relH=x, fn = "dlnorm", 0.5))
curve(resVar(relH=x, fn = "dlnorm", 0.5, par=list(zrv=0.2)))
invisible(sapply(seq(0.01, 0.99, length.out=20), function(a){
  curve(resVar(relH=x, fn = "dlnorm", 0.5, par=list(zrv=a, lsd=(1-sqrt(a)))),
  n=1000)
}))

SK.par.lme

Taper model parameters for spruce in Germany based on BW13 data.

Description

Taper model parameters for spruce in Germany based on BW13 data obtained using TapeR_FIT_LME.f.

Usage

data(SK.par.lme)

Format

See Value section of TapeR_FIT_LME.f.
References


Examples

```r
data(SK.par.lme)
```

---

### Description

This is the actual function to estimate diameters according to the fitted mixed B-splines model.

### Usage

```r
SK_EBLUP_LME.f(xm, ym, xp, par.lme, Rfn = list(fn = "sig2"), ...)
```

### Arguments

- `xm`  
  relative heights for which measurements are available
- `ym`  
  corresponding diameter measurements in height `xm`
- `xp`  
  relative heights for which predictions are required
- `par.lme`  
  Fitted model object, return of `TapeR_FIT_LME.f`
- `Rfn`  
  list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- `...`  
  not currently used

### Details

This function is the actual working horse for prediction using the fitted taper model. Based on the model `par.lme` and the measured diameters `ym` and corresponding (relative) heights `xm` of a specific tree (there might be just one measurement), the random effect parameters and subsequently diameters are estimated. Depending on the parameter `Rfn`, the calibrated taper curve is forced through the given diameter `ym` (`Rfn = list(fn="zero")`), or calibrated using the complete residual variance-covariance information (`Rfn = list(fn="sig2")`, the default). Further assumptions are possible, see also `resVar` and Kublin et al. (2013) p. 987 for more details.
Value

a list holding nine elements:

- $b_{\text{fix}}$ fixed effects parameter of taper model
- $b_{\text{rnd}}$ random effects parameter given tree (posterior mean $b_k$)
- $\hat{y}_p$ estimated diameter in height $x_p$
- $\text{KOV}_{\text{Mean}}$ variance-covariance matrix of expected value
- $\text{KOV}_{\text{Pred}}$ variance-covariance matrix of prediction
- $\text{CI}_{\text{Mean}}$ mean and limits of confidence interval
- $\text{MSE}_{\text{Mean}}$ mean mean squared error of expected value
- $\text{MSE}_{\text{Pred}}$ mean squared error of prediction
- $\text{CI}_{\text{Pred}}$ mean and limits of prediction interval

Author(s)

Edgar Kublin

See Also

E\_DHx\_HmDm\_HT.f, E\_VOL\_AB\_HmDm\_HT.f, resVar

Examples

data("SK.par.lme")
TapeR:::SK\_EBLUP\_LME.f(1.3/27, 30, 1.3/27, SK.par.lme)  
## using empirical best linear unbiased estimator: estimate != 30
TapeR:::SK\_EBLUP\_LME.f(1.3/27, 30, 1.3/27, SK.par.lme, Rfn=list(fn="sig2"))$yp  
## interpolate / force through given diameter: estimate == 30
TapeR:::SK\_EBLUP\_LME.f(1.3/27, 30, 1.3/27, SK.par.lme, Rfn=list(fn="zero"))$yp
TapeR:::SK\_EBLUP\_LME.f(1.3/27, 30, c(1.3, 5)/27, SK.par.lme)
par.lme <- SK.par.lme
h <- 12 # tree height
xm <- c(1.3, 3) / h # relative measuring height
ym <- c(8, 7.5) # measured diameter
xp <- c(0.5, 1) / h # relative prediction height
TapeR:::SK\_EBLUP\_LME.f(xm, ym, xp, SK.par.lme)

Description

Internal function not usually called by users
Usage

```r
SK_VOLab_EBLUP_LME.f(
  xm,
  ym,
  a = 0,
  b = 1,
  Ht,
  par.lme,
  Rfn = list(fn = "sig2"),
  IntPolOpt = TRUE,
  ...
)
```

Arguments

- `xm`: relative heights for which measurements are available
- `ym`: corresponding diameter measurements in height `xm`
- `a`: relative height of lower threshold of stem section
- `b`: relative height of upper threshold of stem section
- `Ht`: tree height
- `par.lme`: List of taper model parameters obtained by `TapeR_FIT_LME.f`
- `Rfn`: list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- `IntPolOpt`: option for method of interpolation, if `TRUE` using a natural interpolating spline (`splinefun`), if `FALSE` using a smoothing spline (`smooth.spline`); defaults to `TRUE`
- `...`: not currently used

Details

With `Rfn=list(fn="zero")` one can decide whether the measured diameters are forced to lie exactly on the taper curve; this interferes somewhat with the `IntPolOpt`, which determines the method of taper curve point interpolation for integration. The default `TRUE` (used throughout all function calls) applies natural interpolating splines, hence this does not contradict the optional use of `Rfn=list(fn="zero")`.

Value

List with two elements, the estimated volume and its variance

Author(s)

Edgar Kublin
Description

Fits a taper curve model with random effects on tree-level based on B-Splines to the specified diameter-height data. Number and position of nodes and order of B-Splines can be specified.

Usage

```r
tapeR_FIT_LME.f(
  Id,
  x,
  y,
  knt_x,
  ord_x,
  knt_z,
  ord_z,
  IdKOVb = "pdSymm",
  control = list(),
  ...
)
```

Arguments

- **Id**: Vector of tree identifiers of same length as diameter and height measurements.
- **x**: Numeric vector of height measurements (explanatory variables) along the stem relative to the tree height.
- **y**: Numeric vector of diameter measurements (response) along the stem (in centimeters).
- **knt_x**: Numeric vector of relative knot positions for fixed effects.
- **ord_x**: Numeric scalar. Order of fixed effects Spline (4=cubic).
- **knt_z**: Numeric vector of relative knot positions for random effects.
- **ord_z**: Numeric scalar. Order of random effects Spline (4=cubic).
- **IdKOVb**: Character string. Type of covariance matrix used by `lme`. Only "pdSymm" makes sense. Rather reduce number of knots if function does not converge.
- **control**: a list of control values for the estimation algorithm to replace the default values returned by the function `lmeControl`. Defaults to an empty list.
- **...**: not currently used
Details

If too few trees are given, the linear mixed model (lme) will not converge. See examples for a suggestion of node positions.

The variance parameters $\theta$ are stored in the natural parametrization (Pinheiro and Bates (2004), p. 93). This means log for variances and logit for covariances. $\theta$ is the vectorized triangle of the random effects covariance matrix + the residual variance ($\text{I} \Sigma$). Given there are 2 inner knots for random effects, the structure will be $c(\text{sig}^2_{b1}, \text{sig}_{b1} \text{sig}_{b2}, \text{sig}_{b1} \text{sig}_{b3}, \text{sig}_{b1} \text{sig}_{b4}, \text{sig}^2_{b2},...,\text{sig}^2_{b4}, \text{I} \Sigma)$

Value

List of model properties

- fit.lmeSummary of the fitted lme model.
- par.lmeList of model parameters (e.g., coefficients and variance-covariance matrices) needed for volume estimation and other functions in this package. Components of the par.lme list
  - knt_xRelative positions of the fixed effects Spline knots along the stem.
  - pad_knt_xPadded version of knt_x, as used to define B-Spline design matrix.
  - ord_xOrder of the spline.
  - knt_zRelative positions of the random effects Spline knots along the stem.
  - pad_knt_zPadded version of knt_z, as used to define B-Spline design matrix.
  - ord_zOrder of the spline.
  - b_fixFixed-effects spline coefficients.
  - KOVb_fixCovariance of fixed-effects.
  - sig2_epsResidual variance.
  - dfResResidual degrees of freedom.
  - KOVb_rndCovariance of random effects.
  - thetaVariance parameters in natural parametrization. See Details.
  - KOV_thetaApproximate asymptotic covariance matrix of variance parameters.

Author(s)

Edgar Kublin

References


See Also

E_DHx_HmDm_HT.f, E_DHx_HmDm_HT_CIdHt.f, E_HDx_HmDm_HT.f, E_VOL_AB_HmDm_HT.f
Examples

```r
# load example data
data(DxHx.df)

# prepare the data (could be defined in the function directly)
Id = DxHx.df[, "Id"]
x = DxHx.df[, "Hx"]/DxHx.df[, "Ht"]  # calculate relative heights
y = DxHx.df[, "Dx"]

# define the relative knot positions and order of splines
knt_x = c(0.0, 0.1, 0.75, 1.0); ord_x = 4  # B-Spline knots: fix effects; order (cubic = 4)
knt_z = c(0.0, 0.1, 1.0); ord_z = 4  # B-Spline knots: rnd effects

# fit the model
taper.model <- TapeR_FIT_LME.f(Id, x, y, knt_x, ord_x, knt_z, ord_z,
                               IdKOVb = "pdSymm")

## save model parameters for documentation or dissemination
## parameters can be load()-ed and used to predict the taper
## or volume using one or several measured dbh
#spruce.taper.pars <- taper.model$par.lme
#save(spruce.taper.pars, file="spruce.taper.pars.rdata")
```

---

**TransKnots**

*transform* knot vector

**Description**

Internal function not usually called by users

**Usage**

```r
TransKnots(knots = c(seq(0, 1, 0.1)), ord = 4, ...)
```

**Arguments**

- **knots**: knot positions for spline function
- **ord**: order of the spline function
- **...**: not currently used

**Value**

transformed knots vector, especially with repeated first and last knot given order of spline function

**Author(s)**

Edgar Kublin
**xy0_root.f**

---

**Description**

Internal function not usually called by users

**Usage**

```
xy0_root.f(x, y0, SK, par.lme, ...)
```

**Arguments**

- `x` relative height
- `y0` diameter for which height is required
- `SK` return of `SK_EBLUP_LME.f` containing estimated fixed and the tree specific random effects of the taper model
- `par.lme` List of taper model parameters obtained by `TapeR_FIT_LME.f`
- `...` not currently used

**Details**

used in `xy0_SK_EBLUP_LME.f` to find the root of taper curve (i.e. height x) at given diameter y0

**Value**

difference between actual diameter at height x and given diameter y0

**Author(s)**

Edgar Kublin

---

**xy0_SK_EBLUP_LME.f**

---

**Description**

Internal function not usually called by users

**Usage**

```
xy0_SK_EBLUP_LME.f(xm, ym, y0, par.lme, Rfn = list(fn = "sig2"), ...)
```

---
**Arguments**

- `xm`  
  relative heights for which measurements are available
- `ym`  
  corresponding diameter measurements in height `xm`
- `y0`  
  given diameter for which height is required
- `par.lme`  
  Fitted model object, return of `TapeR_FIT_LME.f`
- `Rfn`  
  list with function name to provide estimated or assumed residual variances for the given measurements, optionally parameters for such functions
- `...`  
  not currently used

**Details**

function used to transform given diameter in volume calculation into height; c.f `E_VOL_AB_HmDm_HT.f`

with `Rfn` one can decide whether the measured diameters are forced to lie exactly on the taper curve `Rfn$fn= "sig2"` or not `Rfn$fn= "zero"`. Other options are possible, see also `SK_EBLUP_LME.f` and `resVar`.

**Value**

relative height of given diameter `y0`

**Author(s)**

Edgar Kublin

---

**XZ_BSPLINE.f**  
*construct B-Splines design matrices*

---

**Description**

Internal function not usually called by users

**Usage**

`XZ_BSPLINE.f(x, knrt, ord, ...)`

**Arguments**

- `x`  
  relative height measurements
- `knt`  
  knot positions for B-Splines, usually taken from a model fit by `TapeR_FIT_LME.f`
- `ord`  
  order of B-Splines, usually taken from a model fit by `TapeR_FIT_LME.f`
- `...`  
  not currently used

**Value**

List with height measurements `(x)`, the fixed effects B-splines matrix and the random effects B-splines matrix.
Author(s)

Edgar Kublin

See Also

TapeR_FIT_LME.f

Description

Internal function not usually called by users

Usage

y2x_isp.f(x, x.grd, y.grd, ...)

Arguments

x relative height
x.grd relative heights for interpolation
y.grd diameter of taper curve at relative heights x.grd for interpolation
... not currently used

Value

squared estimated diameter based on natural interpolating spline (splinefun)

Author(s)

Edgar Kublin
y2x_ssp.f

\textit{squared diameter using smoothing splines}

\section*{Description}

Internal function not usually called by users

\section*{Usage}

\begin{verbatim}
y2x_ssp.f(x, x.grd, y.grd, ...)
\end{verbatim}

\section*{Arguments}

\begin{itemize}
\item \textit{x} \hspace{1cm} relative height
\item \textit{x.grd} \hspace{1cm} relative heights for interpolation
\item \textit{y.grd} \hspace{1cm} diameter of taper curve at relative heights \textit{x.grd} for interpolation
\item \ldots \text{not currently used}
\end{itemize}

\section*{Value}

squared estimated diameter based on smoothing splines (\texttt{smooth.spline})

\section*{Author(s)}

Edgar Kublin

---

yx_isp.f

\textit{diameter using interpolating splines}

\section*{Description}

Internal function not usually called by users

\section*{Usage}

\begin{verbatim}
yx_isp.f(x, x.grd, y.grd, ...)
\end{verbatim}

\section*{Arguments}

\begin{itemize}
\item \textit{x} \hspace{1cm} relative height
\item \textit{x.grd} \hspace{1cm} relative heights for interpolation
\item \textit{y.grd} \hspace{1cm} diameter of taper curve at relative heights \textit{x.grd} for interpolation
\item \ldots \text{not currently used}
\end{itemize}
Value

estimated diameter based on natural interpolating spline (splinefun)

Author(s)

Edgar Kublin

---

**yx_ssp.f**

* diameter using smoothing splines

---

Description

Internal function not usually called by users

Usage

yx_ssp.f(x, x.grd, y.grd, ...)

Arguments

- **x**: relative height
- **x.grd**: relative heights for interpolation
- **y.grd**: diameter of taper curve at relative heights x.grd for interpolation
- **...**: not currently used

Value

estimated diameter based on smoothing splines (smooth spline)

Author(s)

Edgar Kublin
Index

* datasets
  - DxHx.df, 5
  - HT.par, 18
  - SK.par.lme, 24

* package
  - TapeR-package, 2

BSplines, 3
CdN_DHxHt.f, 4
dlnorm, 24
dN.f, 5
DxHx.df, 3, 5

E_DHx_HmDm_HT.f, 3, 7, 26, 29
E_DHx_HmDm_HT_CDHt.f, 10, 29
E_HDx_HmDm_HT.f, 12, 19, 29
E_VOL_AB_HmDm_HT.f, 14, 26, 29, 32
EYx_ssp.f, 6

HT.par, 3, 18
Hx_root.f, 19

Int_CDn_DHx_dHt.f, 20
Int_E_VOL_AB_HmDm_HT_dHt.f, 21
lmeControl, 28
qD.rout.f, 22
resVar, 7, 15, 23, 25, 26, 32

SK.par.lme, 3, 24
SK_EBLUP_LME.f, 25, 31, 32
SK_VOLab_EBLUP_LME.f, 26
smooth.spline, 27, 34, 35
splineDesign, 4
splinefun, 27, 33, 35

TapeR (TapeR-package), 2
TapeR-package, 2

TapeR_FIT_LME.f, 3, 4, 7, 8, 10, 11, 13, 15, 16, 19–25, 27, 28, 31–33
TransKnots, 30
uniroot, 19
xy0_root.f, 31
xy0_SK_EBLUP_LME.f, 31
XZ_BSPLINE.f, 32
y2x_isp.f, 33
y2x_ssp.f, 34
yx_isp.f, 34
yx_ssp.f, 35