Package: allomr (via r-universe)

October 31, 2024

Type Package

Title Removing Allometric Effects of Body Size in Morphological Analysis

Version 0.3.0

Description Implementation of the technique of Lleonart et al. (2000) <doi:10.1006/jtbi.2000.2043> to scale body measurements that exhibit an allometric growth. This procedure is a theoretical generalization of the technique used by Thorpe (1975) <doi:10.1111/j.1095-8312.1975.tb00732.x> and Thorpe (1976) <doi:10.1111/j.1469-185X.1976.tb01063.x>.

License GPL (>= 3)

Encoding UTF-8

NeedsCompilation no

Author Sämi Schär [aut, cre, cph] (<`https://orcid.org/0000-0001-6108-4832`>)

Maintainer Sämi Schär <`saemi.schaer@gmail.com`>

Date/Publication 2023-11-26 11:00:02 UTC

Repository https://ziegenknauff.r-universe.dev

RemoteUrl https://github.com/cran/allomr

RemoteRef HEAD

RemoteSha 1fe3335d07903477a3f3206ead212485b88dd941

Contents

`allomr` *Removing Allometric Effects of Body Size in Morphological Analysis*

Description

Implementation of the technique of Lleonart et al. (2000) <doi:10.1006/jtbi.2000.2043> to scale body measurements that exhibit an allometric growth. This procedure is a theoretical generalization of the technique used by Thorpe (1975) <doi:10.1111/j.1095-8312.1975.tb00732.x> and Thorpe (1976) <doi:10.1111/j.1469-185X.1976.tb01063.x>.

Usage

```
allomr(X, Y, X0, a, b)
```

Arguments

`X`	Required. A numerical vector, containing the first set of body measurements.
`Y`	Required. Either a numerical vector or an entire data set, containing the second set of body measurements. These are the measurements from which allometric effects will be removed. This is achieved by calculating Y*, the normalized value of Y this subject would reach when X=X0.
`X0`	Optional. The value of X for which Y should be normalized. Default is the mean of X.
`a`	Optional. The regression parameter a. Derived from data if not specified or if Y is a data set.
`b`	Optional. The regression parameter b. Derived from data if not specified or if Y is a data set.

Details

Calculates X_0 as the mean of X and the regression parameters a and b (if not specified) according to equation (A.5). Y^* is calculated according to equation (13) and the particular shape factor $exp(\varepsilon_i)$ according to equation (9) in Lleonart et al. (2000) <doi:10.1006/jtbi.2000.2043>.

Value

A list containing sample size for X (n), the regression parameters a (a) and b (b), X_0 (X0), a vector containing the normalized values Y^* (Yx) and a vector containing the values for the particular shape factor $exp(\varepsilon_i)$ (exp_e).

Author(s)

Sämi Schär

References

Lleonart et al. (2000). J. theor. Biol. 205, 85-93. <doi:10.1006/jtbi.2000.2043>

Thorpe (1975). Biol. J. Linn. Soc. 7, 27-43. <doi:10.1111/j.1095-8312.1975.tb00732.x>

Thorpe (1976), Biol. Rev. 51, 407-452. <doi:10.1111/j.1469-185X.1976.tb01063.x>

Examples

```
## examples from Lleonart et al. (2000), Appendix B
## note: small differences due to rounding in paper

## Table B1
gr1<-NULL
gr1$X<- c(10.223, 11.184,12.251,11.922,11.485,11.625,11.303,11.662)
gr1$Y<- c(1.184,1.371,1.676,1.662,1.509,1.539,1.481,1.417)
gr1$Group<-c(rep(1, 8))
gr1d <- as.data.frame(gr1)

gr2<-NULL
gr2$X<- c(11.415,11.684,11.668, 11.322,12.553,12.213, 10.814, 10.493)
gr2$Y<-c(1.364,1.508,1.535,1.387,1.522,1.502,1.256,1.230)
gr2$Group<-c(rep(2, 8))
gr2d <- as.data.frame(gr2)

B1 <- as.data.frame(rbind(gr1d, gr2d))

allomr(B1$X, B1$Y)


## Table B2
gr1<-NULL
gr1$X<- c(3.050,2.783,2.492,3.543,2.495)
gr1$Y<- c(2.349,2.129,1.936,2.813,1.908)
gr1$Group<-c(rep(1, 5))
gr1d <- as.data.frame(gr1)

gr2<-NULL
gr2$X<- c(4.088,4.264,3.200,4.038,3.855)
gr2$Y<- c(3.307,3.405,2.528,3.217,3.102)
gr2$Group<-c(rep(2, 5))
gr2d <- as.data.frame(gr2)

B2 <- as.data.frame(rbind(gr1d, gr2d))

allomr(X=B2$X, Y=B2$Y, X0=3.4, a=0.705, b=1.092)


## The function is currently defined as
allomr <- function(X, Y, X0, a, b){
  if (is.vector(Y)==TRUE){
    m1 <- lm(log(Y)~log(X))
    n <- length(X[!is.na(X)])
```

```
    if (missing(a)){
      a <- exp(m1$coefficients[1])
    }
    if (missing(b)){
      b <- m1$coefficients[2]
    }
    if (missing(X0)){
      X0 <- mean(X, na.rm=TRUE)
    }
    Yx <- Y*(X0/X)^b
    exp_e <- Y/(a*X^b)
    obj <-list(n, a, b, X0, Yx, exp_e)
    names(obj) <-c("n","a","b","X0","Yx","exp_e")
    class(obj) <- "allomr"
    return(obj)
  }
  else {
    if (!missing(a) | !missing(b)){
      message("NOTE: Y is a data set. Regression parameters will be derived from data.")
    }
    a<-NULL
    b<-NULL
    Yx<-NULL
    exp_e<-NULL
    for (i in 1:ncol(Y)){
      m1<- lm(log(Y[[i]])~log(X))
      n<-length(X[!is.na(X)])
      a<- cbind(a, exp(m1$coefficients[1]))
      b<- cbind(b, m1$coefficients[2])
      if (missing(X0)){
        X0<-mean(X, na.rm=TRUE)
      }
      Yx<- cbind(Yx, Y[[i]]*(X0/X)^m1$coefficients[2])
      exp_e<- cbind(exp_e, Y[[i]]/(exp(m1$coefficients[1])*X^m1$coefficients[2]))
    }
    obj <-list(n, a, b, X0, Yx, exp_e)
    names(obj) <-c("n","a","b","X0","Yx","exp_e")
    class(obj) <- "allomr"
    return(obj)
  }
}
```

Index