Estimating field body temperature and the Constant Temperature Equivalent with NCEP and NicheMapR

Michael Kearney

2019-04-02

Contents

Overview

In DEB parameter estimation it is critical to get a good estimate of the body temperature at which growth, reproduction, development and aging rates are observed. A measure of mean 'ambient' temperature will often be inaccurate, leading to poor parameter estimates. For example, often the observations of these rates are made for organisms experiencing fluctuating body temperatures, as typically occurs in nature or if animals are cultured under cycling temperature regimes. Moreover, the temperatures experienced may be above ambient air temperature if the organism is in the boundary layer near the ground, exposed to sunlight or high levels of infrared radiation. It may also be below air temperature if it is cooling by evaporation (e.g. a frog) or to cold radiative environemnts (e.g. clear night skies).

This document shows how to estimate the body temperatures of animals in nature by two methods of increasing sophistication. First, it shows how to extract 6-hourly air temperature data for a location of interest (e.g. from where field growth rates have been observed) from a global database of historical daily weather data via the RNCP package (Kemp et al. 2012). Second it shows how to use the NicheMapR package to estimate microclimates (e.g. near-ground air temperatures, soil temperatures) and to simulate heat exchange for an organism of given physical characteristics (e.g. solar reflectance) and behavioural thermoregulatory responses.

In each case the resulting temperature vector is converted to a 'constant temperature equivalent', CTE, which can be used as the temperature estimate for zero-variate observations of e.g. age at birth, age at puberty, longevity and reproduction rate.

For the example we will use the sand lizard, *Lacerta agilis*, which is in the AmP collection and has field growth observations for a number of locations in Russia (Roitberg and Smirina 2006).

Package Installation

First install and load the necessary packages (if you don't have them already).

```r
install.packages("rvest")
install.packages("RNCEP")
install.packages("ncdf4")
install.packages("devtools")
install.packages("stringr")
devtools::install_github("mrke/NicheMapR")
# to install with devtools on Windows you'll need rtools:
# https://cran.r-project.org/bin/windows/Rtools/ you can also
# download a binary from
# https://github.com/mrke/NicheMapR/releases/
```

Load the libraries.

```r
library(rvest)
library(RNCEP)
library(ncdf4)
```

```r
library(stringr)
library(NicheMapR)
```

Extracting NCEP air temperature for a location and time

The RNCEP package interfaces with two gridded (~2.5 degree) global data set on various meteorological variables that go back to as far as 1957 and as recently as the past month. You can read more about these data sets in (Kemp et al. 2012).

Here we are quering the 'air.2m' variable, which is the estimate of air temperature at the typical weather station height. There are a number of other options you could choose.

We need to specify the location as longitude and latitude (decimal degrees), as well as the starting and ending year and the starting and ending months. Note that you ONLY get the months you ask for, so if you're trying to get a continuous period that spans a year, you should do separate queries per year and combine them.

```r
# define query inputs
lonlat <- c(46.8, 43.3)  # the Kostek site of Roitberg and Smirina (2006)
y.start <- 1985
y.finish <- 1985
m.start <- 1
m.finish <- 12
var <- "air.2m"
# you could also choose e.g. 'skt.sfc' (surface temperature)
# or 'tmp.0-10cm' (shallow soil)
lev <- "gaussian"

# run the query
tair <- NCEP.gather(variable = var, level = lev, months.minmax = c(m.start,
    m.finish), years.minmax = c(y.start, y.finish), lat.southnorth = c(lonlat[2],
    lonlat[2]), lon.westeast = c(lonlat[1], lonlat[1]), reanalysis2 = FALSE,
    return.units = TRUE, status.bar = FALSE)
```
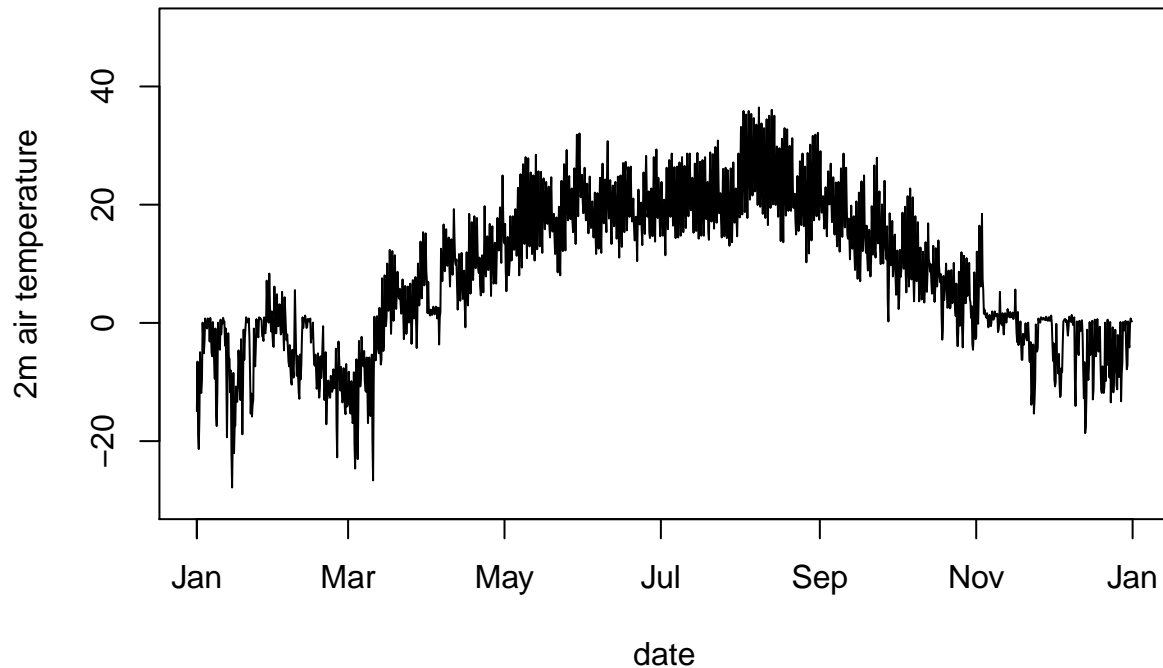
```
## [1] Units of variable 'air.2m' are degK
```

```r
# extract the actual coordinates you got data from (given
# that the grid is coarse)
lonlat.returned <- c(as.numeric(dimnames(tair)[[2]][2]), as.numeric(dimnames(tair)[[1]][1]))

# dates
dates1 <- seq(ISOdate(y.start, m.start, 1, tz = "UTC") - 3600 *
    12, ISOdate(y.finish + 1, 1, 1, tz = "UTC") - 3600 * 13,
    by = "hours")
dates <- dates1[seq(1, length(dates1), 6)]  # cut down to 6-hourly

t.air <- cbind(as.data.frame(dates), tair[1, 1, ])
colnames(t.air) <- c("date", var)
plot(t.air - 273.15, type = "l", ylim = c(-30, 50), ylab = "2m air temperature")  # plot, converting to
```

```
# write to disk if you want write.csv(tair[1, 1, ],
# 'tair.csv')
```

Compute constant temperature equivalent (CTE) based on air temperature

Now we want to compute a 'constant temperature equivalent', CTE, which is an estimate of the mean temperature accounting for the nonlinearity of the thermal response curve.

First, some parameters are defined for the 5-parameter Arrhenius response curve. These are a rough guess for *Lacerta agilis*.

```
T_A <- 8000
T_AL <- 20000
T_AH <- 20000
T_L <- 10 + 273.15
T_H <- 40 + 273.15
T_REF <- 20 + 273.15
```

Now we define the temperature vector of interest, T, as the 2 m air temperature NCEP data, and use the 5 parameter Arrhenius formula to convert it to the temperature correction factor, c_T.

```
T_b <- t.air$air.2m  # temps from simulation to be used to estimate CTE
Tjuly <- mean(T[which(format(dates, "%m") == "07")] - 273.15)  # get mean July air temperature

# temperature correction function
get_c_T <- function(T_A, T_L, T_H, T_AL, T_AH, T_REF, T_b) {
    c_T <- exp(T_A/(T_REF) - T_A/T_b) * (1 + exp(T_AL/T_REF -
        T_AL/T_L) + exp(T_AH/T_H - T_AH/T_REF))/(1 + exp(T_AL/T_b -
```
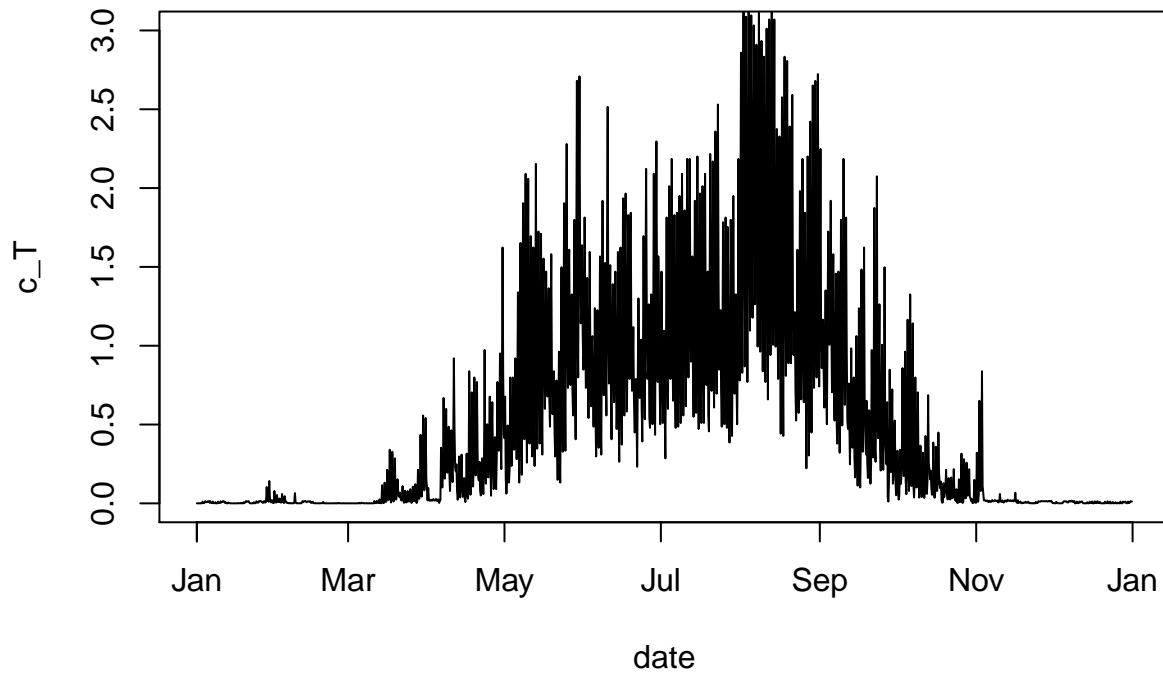
3

```
        T_AL/T_L) + exp(T_AH/T_H - T_AH/T_b))
}

# convert T_b each hour to temperature correction factor
c_T <- get_c_T(T_A, T_L, T_H, T_AL, T_AH, T_REF, T_b)
plot(c_T ~ t.air$date, type = "l", xlab = "date", ylim = c(0,
    3))
```


Now we compute the mean c_T and define a function for use in iteratively guessing values of body temperature that result in this mean c_T value, i.e. we turn the mean temperature correction factor back into a temperature. The 'uniroot' function is used to do the guessing, with bracketing starting T_L and T_H.

```
c_T.mean <- mean(c_T)  # get mean temperature correction factor

getCTE <- function(T_b) {
    # function finding the difference between a temperature
    # correction factor, c_T.mean, for a specified T_b compared
    # to the mean calculated one (aim to make this zero)
    x <- get_c_T(T_A, T_L, T_H, T_AL, T_AH, T_REF, T_b) - c_T.mean
}
# search for a T_b (CTE) that gives the same temperature
# correction factor as the mean of the simulated temperature
# corrections
CTE <- uniroot(f = getCTE, c(T_L, T_H), check.conv = TRUE)$root -
    273.15  # converting from K to deg C
```

We now have the constant temperature equivalent. As you can see from the values below, the naïve

temperature correction factor we'd get from the mean of the raw input air temperatures (8.1 °C) is around 0.1 but transformation of the 6-hourly temperatures into c_T gives a mean of 0.5. This corresponds to a CTE of 14.5 °C, i.e. over 6 °C higher than the mean air temperature. Note that we also get a mean July air temperature of ~21 °C which is a bit lower than the value of 24 °C reported in Roitberg and Smirina (2006).

```r
{
    cat(paste0("mean T_air = ", round(mean(T_b - 273.15), 2)),
        "\n")  # report value to console
    Tjuly <- mean(T_b[which(format(dates, "%m") == "07")] - 273.15)
    cat(paste0("mean T_air, July = ", round(Tjuly, 2)), "\n")  # report value to console
    cat(paste0("mean c_T = ", round(c_T.mean, 2)), "\n")  # report value to console
    cat(paste0("naïve c_T = ", round(as.numeric(exp(T_A/(T_REF) -
        T_A/mean(T_b)) * (1 + exp(T_AL/T_REF - T_AL/T_L) + exp(T_AH/T_H -
        T_AH/T_REF))/(1 + exp(T_AL/mean(T_b) - T_AL/T_L) + exp(T_AH/T_H -
        T_AH/mean(T_b)))), 2)), "\n")
    # report value to console
    cat(paste0("CTE = ", round(CTE, 2)), "\n")  # report value to console
}
```

```
## mean T_air = 8.09
## mean T_air, July = 20.76
## mean c_T = 0.49
## naïve c_T = 0.13
## CTE = 14.52
```

Using NicheMapR to estimate body temperature including thermoregulatory behaviour

The use of fluctuating air temperature is clearly making a difference to the temperature we would associate with field observations for this lizard. The next section shows an example of how to do this using the NicheMapR package (Kearney and Porter, 2017). This package includes a microclimate model, which computes aboveground and belowground conditions relevant to solving heat and water budgets of organisms (radiation, wind, humidity, air temperature). It also includes an ectotherm model of heat and water exchange in the context of behavioural regulation under spatially implicit microclimatic variation (Kearney et al. 2012; Kearney et al. 2018). I.e. extremes of available shade, and the range of possible depths, are provided as well as body temperature thresholds for different behavioural responses, and the model computes an hourly temperature trajectory given the microclimate.

Here we run the microclimate function 'micro_ncep' which forces the microclimate model via the RNCEP package used above, but also extracting information on humidity, wind speed, cloud cover, air pressure and solar radiation from the NCEP data.

```r
dstart <- "01/01/1985"
dfinish <- "31/12/1985"
micro <- micro_ncep(loc = lonlat, dstart = dstart, dfinish = dfinish)
```

```
## Warning: package 'raster' was built under R version 3.5.2

## downloading DEM via package elevatr

## Merging DEMs

## Reprojecting DEM to original projection

## Note: Elevation units are in meters.
## Note: The coordinate reference system is:
##  +init=epsg:3395 +proj=merc +lon_0=0 +k=1 +x_0=0 +y_0=0 +datum=WGS84 +units=m +no_defs +ellps=WGS84

## downloading weather data with package RNCEP via package microclima
## computing radiation and elevation effects with package microclima
```

```
## Warning in microclima::microclimaforNMR(lat = longlat[2], long =
## longlat[1], : Data sequence in UTC/GMT. Some or all dates using system
## timezone are in AEDT

## Extracting rainfall data from NCEP
## Downscaling radiation and wind speed
## Calculating elevation and cold-air drainage effects
## running microclimate model for  365  days from  1985-01-01  to  1985-12-31 23:00:00  at site  long 4(
##    user  system elapsed
##   18.97    0.27   23.38
```
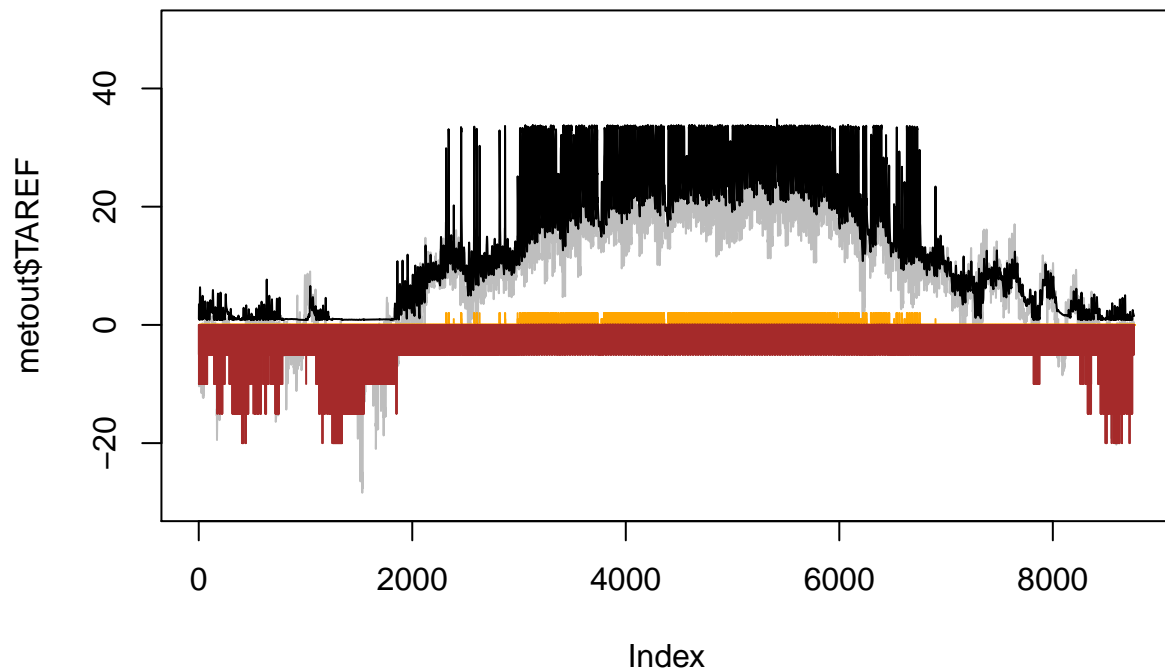
Next we have to specify the parameters determining how the species behaviourally thermoregulates. Brief
explanations are provided next to each one.

```
Ww_g <- 30  # wet weight, grams
CT_max <- 42  # critical thermal maximum
CT_min <- 0.7  # critical thermal minimum
T_F_max <- 40  # maximum T_b at which activity occurs
T_F_min <- 25  # minimum T_b at which activity occurs
T_pref <- 33.8  # preferred T_b (will try and regulate to this)
T_B_min <- 20  # minimum T_b at which leaves retreat
T_RB_min <- 15  # minimum T_b at which they will attempt to leave retreat
shape <- 3  # shape (3 = lizard)
shade_seek <- 1  # shade seeking?
burrow <- 1  # can it move below ground?
minshade <- 0  # minimum available shade?
mindepth <- 3  # minimum depth (node, 1-10) allowed
maxdepth <- 10  # maxmaximum depth (node, 1-10) allowed
nocturn <- 0  # nocturnal activity?
crepus <- 0  # crepuscular activity?
diurn <- 1  # diurnal activity?
```

Finally we run the ectotherm model and plot the results. In the figure you can see in grey the air temperature,
as extracted via RNCEP above, but also in black the predicted temperature incorporating the thermoregulatory
response, which includes moving underground to escape extremes of heat and cold, and choosing shade to
attempt to stay close to the preferred range. In brown is the depth selected (cm) and the orange indicates the
hours that foraging was possible. You can see the lizard is warmer in the winter due to being underground
and under snow, and can also get warmer in the active season due to exposure to solar radiation.
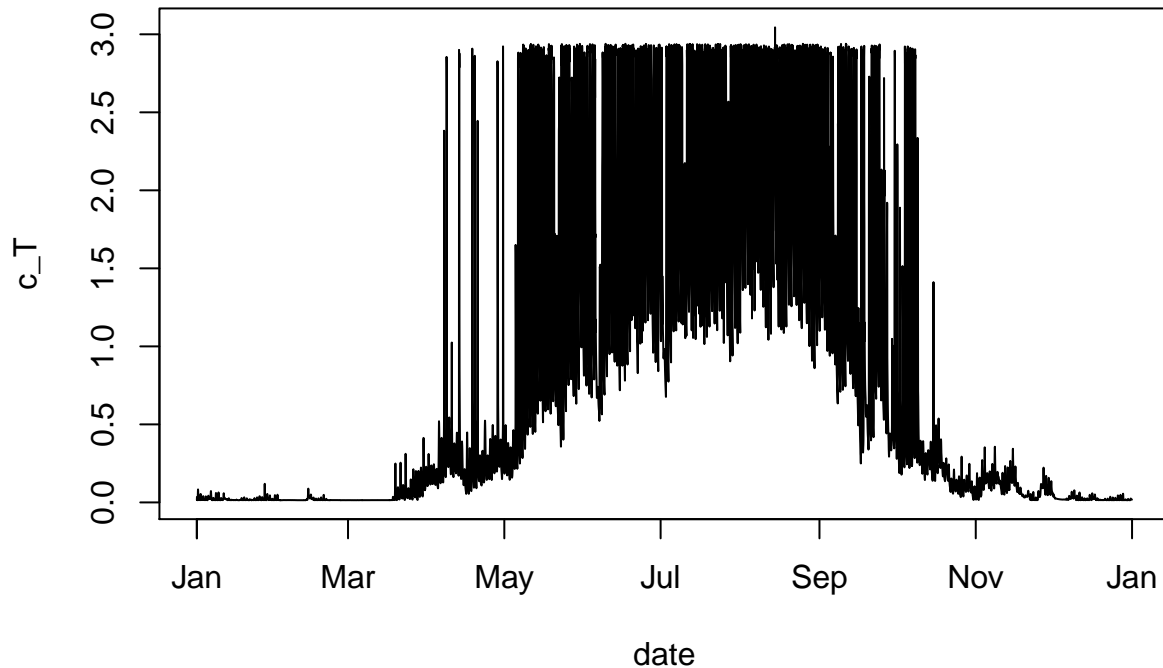
```
ecto <- ectotherm(nocturn = nocturn, diurn = diurn, shape = shape,
    T_F_max = T_F_max, T_F_min = T_F_min, T_pref = T_pref, T_B_min = T_B_min,
    T_RB_min = T_RB_min, CT_max = CT_max, CT_min = CT_min, crepus = crepus,
    burrow = burrow, shade_seek = shade_seek, mindepth = mindepth,
    maxdepth = maxdepth, minshade = minshade)

environ <- as.data.frame(ecto$environ)
metout <- as.data.frame(micro$metout)
# plot some results
plot(metout$TAREF, ylim = c(-30, 50), type = "l", col = "grey")
points(environ$TC, type = "l")
points(environ$ACT, type = "h", col = "orange")
points(environ$DEP, type = "h", col = "brown")
```

6

Now we repeat the procedure above for getting the CTE, using estimated body temperature as our input vector.

```
T_b <- environ$TC + 273.15   # temps from simulation to be used to estimate CTE
# convert T_b each hour to temperature correction factor
c_T <- get_c_T(T_A, T_L, T_H, T_AL, T_AH, T_REF, T_b)
plot(c_T ~ dates1, type = "l", xlab = "date")
```

```r
c_T.mean <- mean(c_T)  # get mean temperature correction factor
# search for a T_b (CTE) that gives the same temperature
# correction factor as the mean of the simulated temperature
# corrections
CTE <- uniroot(f = getCTE, c(T_L, T_H), check.conv = TRUE)$root -
    273.15  # converting from K to deg C
Tjuly <- mean(T_b[which(format(dates1, "%m") == "07")] - 273.15)
{
    # report values to console
    cat(paste0("mean T_b = ", round(mean(T_b) - 273.15, 2)),
        "\n")
    cat(paste0("mean T_b, July = ", round(Tjuly, 2)), "\n")
    cat(paste0("mean c_T = ", round(c_T.mean, 2)), "\n")
    cat(paste0("naïve c_T = ", round(as.numeric(exp(T_A/(T_REF) -
        T_A/mean(T_b)) * (1 + exp(T_AL/T_REF - T_AL/T_L) + exp(T_AH/T_H -
        T_AH/T_REF))/(1 + exp(T_AL/mean(T_b) - T_AL/T_L) + exp(T_AH/T_H -
        T_AH/mean(T_b)))), 2)), "\n")
    cat(paste0("CTE = ", round(CTE, 2)), "\n")  # report value to console
}
```

```
## mean T_b = 13.56
## mean T_b, July = 26.95
## mean c_T = 0.81
## naïve c_T = 0.42
## CTE = 18.15
```

When we incorporate behaviour, we end up with a higher raw input temperature of 13.6 °C and a higher

CTE of 18.2 °C. Roitberg and Smirina (2006) reported a mean July air temperature of 24 °C, which was the value used in the original AmP entry for this species. It would be more appropriate, however, to use a value of 18.2 °C incorporating the nonlinear rate effects and considering the whole year, because the growth curves are over multiple years. This would result in parameters permitting more rapid growth in cold climates.

References

Kearney M. R., Munns S. L., Moore D., Malishev M. & Bull C. M. (2018) Field tests of a general ectotherm niche model show how water can limit lizard activity and distribution. Ecological Monographs 88 , 672–693.

Kearney, M. R., S. J. Simpson, D. Raubenheimer, and S. A. L. M. Kooijman. 2013. Balancing heat, water and nutrients under environmental change: a thermodynamic niche framework. Functional Ecology 27:950–966.

Kearney, M. R., and W. P. Porter. 2017. NicheMapR - an R package for biophysical modelling: the microclimate model. Ecography 40:664–674.

Kemp, M. U., E. Emiel van Loon, J. Shamoun-Baranes, and W. Bouten. 2012. RNCEP: global weather and climate data at your fingertips: RNCEP. Methods in Ecology and Evolution 3:65–70.

Roitberg, E. S., and E. M. Smirina. 2006. Age, body size and growth of *Lacerta agilis boemica* and *Lacerta strigata*: a comparative study of two closely related lizard species based on skeletochronology. Herpetological Journal 16:133–148.