

Onlinebeispiel 5.6 Erfassung von Froschrufen zur Unterstützung des Managements eines Naturschutzgebietes

Kapitel 5.6 aus Henle, K., A. Grimm-Seyfarth & B. Gruber: Erfassung und Analyse von Tierpopulationen. Ulmer Verlag

Annegret Grimm-Seyfarth

2024-04-14

In diesem Beispiel geht es darum, aus Häufigkeitskassen Trends von Populationen zu bestimmen. Dabei kann die Nachweiswahrscheinlichkeit von Kovariablen abhängen, wie wir am Beispiel der Rotbauchunke (*Bombina bombina*) und des Laubfroschs (*Hyla arborea*) zeigen werden.

In dem Naturschutzgebiet der Papitzer Lehmflächen nahe Leipzig werden im Rahmen der wissenschaftlichen Begleitforschung eines Auenrevitalisierungsprojektes jährlich Amphibien an den 35 Gewässern untersucht. Dies erfolgt über Verhören in Anlehnung an das nordamerikanische Programm zur Erfassung von Fröschen anhand ihrer Rufe (Weir & Mossman 2005). Der Häufigkeitsindex für die rufende Frösche nimmt die Werte 0; 1; 2 oder 3 an, die für kein Frosch wurde gehört; diskrete, nicht überlappende Rufe; diskrete, überlappende Rufe und Chor mit andauernd überlappenden Rufen stehen. Dies wird als $N = 0,1,2,3$ state bezeichnet. Das nordamerikanische Programm umfasst drei Durchgänge an zahlreichen Gewässern. Für jeden Durchgang wird der Häufigkeitsindex aufgeschrieben. Diese bezeichnen sich als `time1`, `time2`, `time3`.

Die hier gezeigten Daten stammen aus einer Masterarbeit (Müller 2018) und wurden daher häufiger aufgenommen. Konkret fanden 20 Durchgänge statt. Allerdings wurde im ersten Durchgang versäumt, die Wassertemperatur aufzunehmen, weshalb wir diesen grundsätzlich weglassen. In den übrigen Durchgängen wurden manchmal alle Gewässer, manchmal aber auch nur einige Gewässer aufgenommen. Dadurch reduzieren sich die 19 verbleibenden Durchgänge auf 13 Erfassungen. Diese wurden nun annähernd gleich zusammengefasst, sodass drei Erfassungen (Surveys) für die Analyse dieser Daten mit dem vorgestellten Programm entstehen. Dabei umfasst Survey 1 vier Erfassungen (Anfang April, 05.04. – 10.04.), Survey 2 ebenfalls vier Erfassungen (Ende April bis Anfang Mai, 17.04. – 08.05.), und Survey 3 fünf Erfassungen (Ende Mai, 09.05. – 28.05.). Für die Häufigkeitsklassen (HK) wurden jeweils der Maximalwert über alle Erfassungen angenommen, der innerhalb der jeweiligen Survey verhört wurde. Für die Wassertemperatur wurde ein Mittelwert der Erfassungen pro Survey errechnet.

Einlesen der Daten

Zunächst lesen wir unsere Erfassungsdaten ein.

```
UlmerBuch::beispiel.pfad() #Pfad zu den Beispieldaten
```

```
## Der Pfad zu den Beispiel Daten wurde gesetzt auf:  
## C:/Users/grimm/AppData/Local/Programs/R/R-4.4.3/library/UlmerBuch/extdata
```

```
RBU <- read.csv2("Rotbauchunke.csv")
LF <- read.csv2("Laubfrosch.csv")
WTemp <- read.csv2("Wassertemperatur.csv")
```

Diese Daten wurden als Datenframe eingeladen, wir benötigen aber Matritzen. Daher müssen wir sie umformatieren. Zunächst entfernen wir jedoch ein Gewässer, welches im Mai bereits ausgetrocknet war: Gewässer 15.

```
RBU <- RBU[RBU$Gewaesser!="15",]
LF <- LF[LF$Gewaesser!="15",]
WTemp <- WTemp[WTemp$Gewaesser!="15",]
```

Umwandeln der Daten in Matritzen

```
# dabei wird Zeile 1 (ohne Wassertemperatur) ausgelassen
RBU.data <- as.matrix(RBU[,-1])
colnames(RBU.data) <- c(1:3)
rownames(RBU.data) <- RBU$Gewaesser

LF.data <- as.matrix(LF[,-1])
colnames(LF.data) <- c(1:3)
rownames(LF.data) <- LF$Gewaesser

WTemp.data <- as.matrix(WTemp[,-1])
colnames(WTemp.data) <- c(1:3)
rownames(WTemp.data) <- WTemp$Gewaesser
```

Einladen des Modells von Royle & Link (2005)

Nun müssen wir das Modell von Royle & Link (2005) einladen. Es steht nicht als R-Paket bereit, der Code befindet sich jedoch im Supplement zum Artikel. Er kann hier heruntergeladen werden: https://figshare.com/articles/dataset/Supplement_1_Data_and_computer_programs_for_fitting_multinomial_mixture_models_used_in_the_main_article_/3525107?backTo=/collections/A_GENERAL_CLASS_OF_MULTINOMIAL_MIXTURE_MODELS_FOR_ANURAN_CALLING_SURVEY_DATA/3298763

Dieses Modell berechnet exakt nach dem nordamerikanischen Erfassungsschema, weshalb wir unsere Daten auch an drei Erfassungen (Surveys) angepasst haben und die gleichen Häufigkeitsklassen (HK) genutzt wurden. Möchte man andere Klassen oder Zeitabstände nutzen, kann man den Code entsprechend anpassen. Wir lassen hier die originale Beschriftung von Royle & Link (2005) bestehen.

```
# R code for fitting multinomial mixture models described in "A general class
# of multinomial mixture models for anuran calling survey data" by J.A. Royle
# and W.A. Link.

# this is a function that computes the multinomial cell probabilities of
# the sampling distribution. Note that a multinomial logit transform is used

cp<-function(x){
  c2<-exp(c(0,x[1]))/sum( exp(c(0,x[1])) )
```

```

c3<-exp(c(0,x[2:3]))/sum( exp(c(0,x[2:3])) )
c4<-exp(c(0,x[4:6]))/sum( exp(c(0,x[4:6])) )
P<-matrix(0,4,4)
P[1,1]<-1
P[1:2,2]<-c2
P[1:3,3]<-c3
P[1:4,4]<-c4
P
}

# a utility function, inverse-logit transform
expit<-function(mu)
{
  1/((1/exp(mu)) + 1)
}

# This function evaluates the negative log-likelihood function. A multinomial
# logit transform is used for the latent index distribution

negloglik<-function(x,pim,vars=NULL){
  pim[4:6]<-length(unique(pim[1:3]))+pim[4:6]
  nparm<-length(unique(pim))

  psi<-x[(nparm+1):(nparm+3)]
  psi<-exp(c(0,psi))/sum( exp(c(0,psi)))

  if(length(vars)>0){
    covparms<-x[ 7:length(x)]
    names(covparms)<-vars
    tempparms<-c(0,0,0,0)
    names(tempparms)<-c("time1","time3","temp1","temp2")
    tempparms[vars]<-covparms
    time1<-tempparms[1]
    time3<-tempparms[2]
    temp1<-tempparms[3]
    temp2<-tempparms[4]
  }
  else{
    time1<-time3<-temp1<-temp2<-0
  }

  x<-x[pim]
  beta1<-expit(x[4])
  beta2<-expit(x[5])
  beta3<-expit(x[6])

  cs<-rep(NA,nrow(M))
  for(i in 1:nrow(M)){

    lik<-matrix(NA,nrow=4,ncol=ncol(M))
    for(j in 1:ncol(M)){

      p1<-expit(x[1]+time1*ifelse(j==1,1,0) +time3*ifelse(j==3,1,0) + temp1*temp[i,j]

```

```

      + temp2*temp[i,j]*temp[i,j])
p2<-expit(x[2]+time1*ifelse(j==1,1,0) +time3*ifelse(j==3,1,0) + temp1*temp[i,j]
      + temp2*temp[i,j]*temp[i,j])
p3<-expit(x[3]+time1*ifelse(j==1,1,0) +time3*ifelse(j==3,1,0) + temp1*temp[i,j]
      + temp2*temp[i,j]*temp[i,j])
c2<-c(1-p1,p1)
c3<-c((1-beta1)*(1-p2),beta1*(1-p2),p2)
c4<-c( (1-beta3)*(1-beta2)*(1-p3),(1-beta3)*beta2*(1-p3),beta3*(1-p3),p3)
P<-matrix(0,4,4)
P[1,1]<-1
P[1:2,2]<-c2
P[1:3,3]<-c3
P[1:4,4]<-c4

if(!is.na(M[i,j])){
  # probabilities of observed value for each N=0,1,2,3 state
  this.p<- P[M[i,j]+1, ]
  lik[,j]<- this.p
}
else{
  lik[,j]<-1
}
}
cs[i]<-sum(apply(lik,1,prod)*psi)

}
-2*sum(log(cs))
}

```

Der gegebene R-Code definiert eine Funktion zur Anpassung von multinomialen Mischmodellen. Er besteht aus drei Funktionen:

cp: Diese Funktion berechnet die Wahrscheinlichkeiten der multinomialen Zellen für die Stichprobenverteilung mithilfe einer multinomialen Logit-Transformation. Oder anders ausgedrückt: Mithilfe dieser Funktion kann aus den geschätzten Parametern eine Wahrscheinlichkeit pro HK berechnet werden.

expit: Diese Hilfsfunktion führt die Inverse der logistischen Transformation durch, um Wahrscheinlichkeiten aus Logit-Werten zu berechnen.

negloglik: Diese Funktion bewertet die negative Log-Likelihood der Modelle. Sie verwendet das multinomiale Logit-Modell für die latente Indexverteilung und berechnet die Wahrscheinlichkeit, bestimmte Beobachtungen in den Daten zu erhalten. Sie berücksichtigt verschiedene Einflussfaktoren wie Zeit und Temperatur.

Berechnung der Wahrscheinlichkeiten in den verschiedenen Häufigkeitsklassen

Zunächst müssen wir die Input-Parameter verstehen. Diese unterteilen sich in Basisparameter und Kovariaten. Basisparameter sind die Nachweiswahrscheinlichkeiten zu den Erfassungszeitpunkten $p1$, $p2$ und $p3$. Neben der Wahrscheinlichkeit, die korrekte HK zu beobachten ($p1$ bis $p3$), gibt es natürlich noch die Wahrscheinlichkeit von Fehlklassifizierungen (β). Dabei unterscheiden wir folgendermaßen:

β_{21} - Wahrscheinlichkeit, dass HK1 beobachtet wurde, aber eigentlich HK2 vorliegt

beta32 - Wahrscheinlichkeit, dass HK2 beobachtet wurde, aber eigentlich HK3 vorliegt

beta31 - Wahrscheinlichkeit, dass HK1 beobachtet wurde, aber eigentlich HK3 vorliegt

Diese Parameter werden in einer sogenannten Parameter-Index-Matrix (PIM) definiert, wobei die Reihenfolge $PIM(p1, p2, p3, b21, b32, b31)$ entspricht. Hierbei definieren wir, welche Parameter konstant sind und welche sich unterscheiden können; p und $beta$ können jedoch nie gleich sein, da sie verschiedene Prozesse abbilden. Ein Beispiel für konstante p und $beta$ wäre also $PIM(1,1,1,2,2,2)$, während komplett verschiedene p und $beta$ entsprechend $PIM(1,2,3,4,5,6)$ geschrieben werden. Dazwischen sind alle Kombinationen möglich, z. B. $PIM(1,2,2,3,3,4)$, wobei $p2$ und $p3$ gleich wären, jedoch von $p1$ verschieden, und **beta21** und **beta32** gleich wären, aber von **beta31** verschieden. Es gibt insgesamt 25 solcher Kombinationen. Alle Kombinationen können prinzipiell gerechnet und mittels AIC (Kap. 9.3 des Buches) verglichen werden.

Zusätzlich zu den Basisparametern gibt es die schon angesprochenen Kovariaten, also Faktoren, die beeinflussen, ob die Art überhaupt beobachtet werden konnte. Das Modell berücksichtigt zwei Kovariaten: zeitliche Variation und Abhängigkeit von einem Umweltfaktor, in unserem Fall Temperatur. Für weitere Parameter müsste das Modell oben angepasst werden. Hierbei geht das Modell davon aus, dass für $p2$ immer der Standardwert angenommen wird. Unterscheiden wir also einen zeitlichen Effekt, gibt es zwei Parameter: $b1$ (**time1**), der zeitliche Unterschied von $p1$ zu $p2$, und $b3$ (**time3**), der zeitliche Unterschied von $p3$ zu $p2$. Für den Umweltfaktor, der die Nachweiswahrscheinlichkeit p beeinflusst, gilt jedoch, dass er zu p in einem linearen (**temp1**) oder quadratischen Zusammenhang (**temp2**) stehen kann.

Während die Struktur der Basisparameter über die PIM angegeben wird, wird die Kovariatenstruktur bei **vars** angegeben. Zusammengefasst können folgende Parameter bei **vars** eingegeben werden:

Zeitliche Variabilität: **time1,time3**

linearer Temperatureffekt: **temp1**

quadratischer Temperatureffekt: **temp1,temp2**

Für die Analyse eigener Daten sollten die Daten in Aufbau und Namensgebung den Beispieldaten entsprechen. Dann sollten auch eigenen Daten problemlos berechnet werden können.

Struktur des Modelles festlegen

Wir berechnen zwei Beispiele: konstante p und $beta$ $PIM(1,1,1,2,2,2)$ und das oben beschriebene Beispiel $PIM(1,2,2,3,3,4)$

```
PIM1 <- c(1,1,1,2,2,2)
PIM2 <- c(1,2,2,3,3,4)
```

Für die Kovariatenstruktur nehmen wir an, dass eine zeitliche Variabilität besteht. Außerdem testen wir den Temperatureffekt linear und quadratisch.

```
vars1 <- c("time1","time3","temp1")
vars2 <- c("time1","time3","temp1","temp2")
```

Außerdem benötigt die Funktion noch Startwerte. Da wir diese nicht kennen, nutzen wir die Beispielstartwerte aus Royle & Link (2005).

```
# für p
length(RBU$X1[RBU$X1>0])
```

```
## [1] 19
```

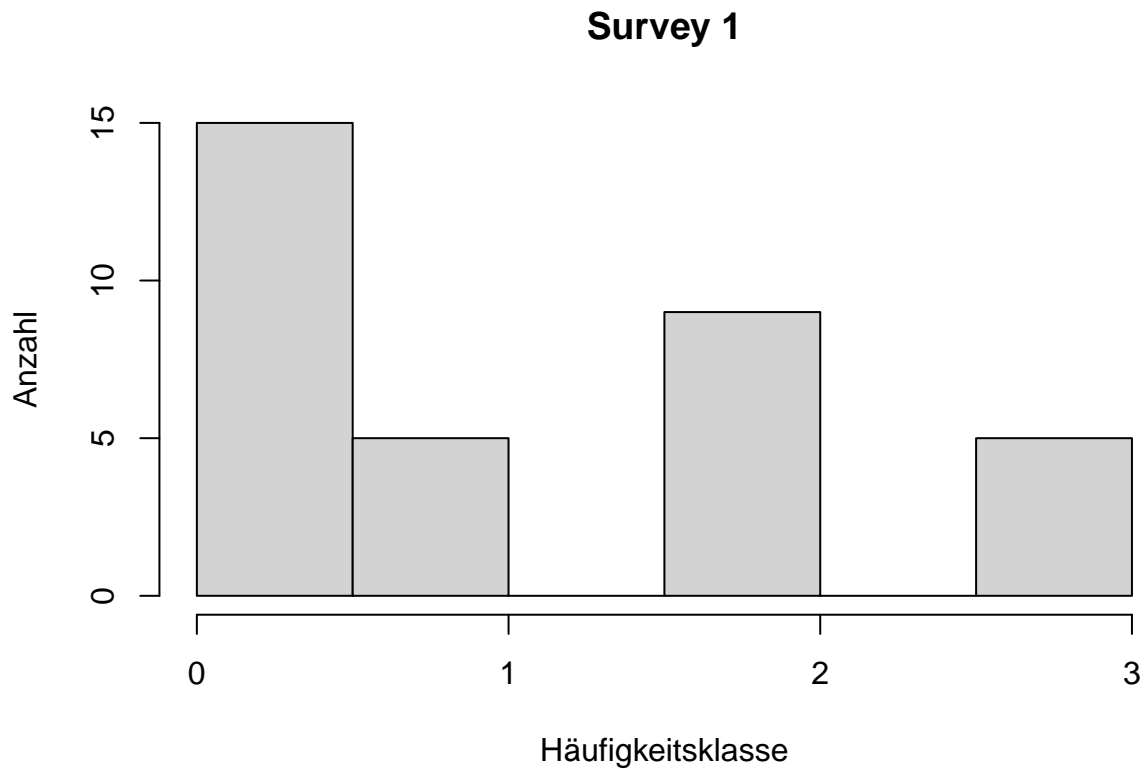
```
length(RBU$X1[RBU$X2>0])
```

```
## [1] 18
```

```
length(RBU$X1[RBU$X3>0])
```

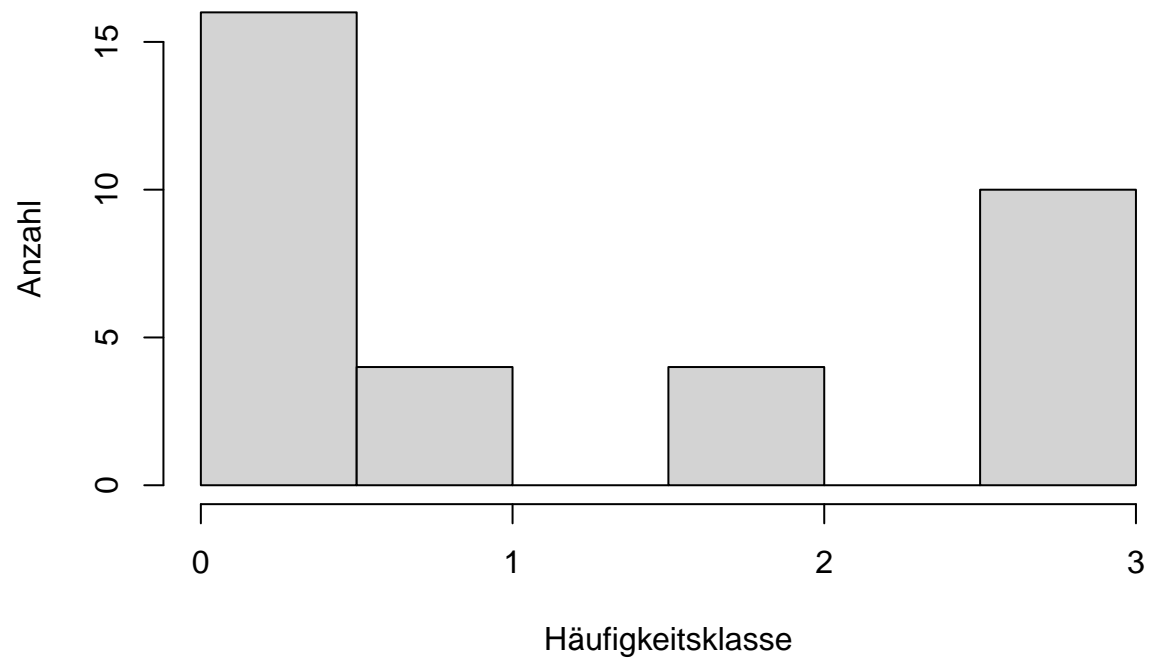
```
## [1] 13
```

```
{hist(RBU$X1, xaxt="n",  
      xlab = "Häufigkeitsklasse",  
      ylab = "Anzahl",  
      main = "Survey 1")  
axis(1, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))  
}
```



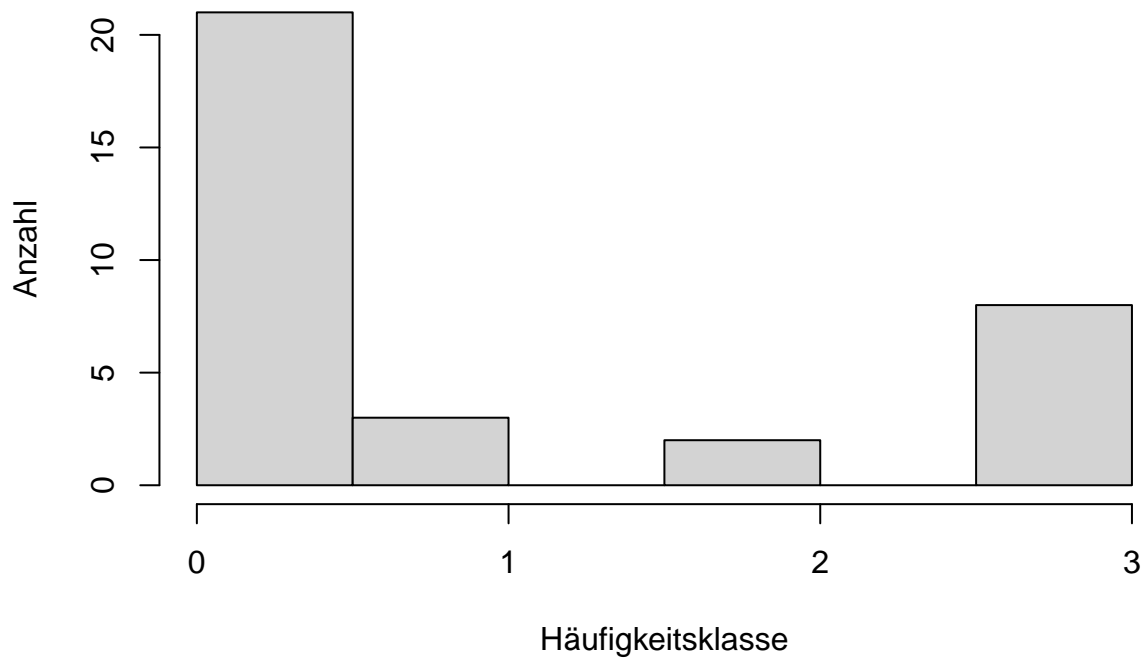
```
{hist(RBU$X2, xaxt="n",  
      xlab = "Häufigkeitsklasse",  
      ylab = "Anzahl",  
      main = "Survey 2")  
axis(1, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))  
}
```

Survey 2



```
{hist(RBU$X3, xaxt="n",  
      xlab = "Häufigkeitsklasse",  
      ylab = "Anzahl",  
      main = "Survey 3")  
axis(1, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))  
}
```

Survey 3



*# wir nehmen an, dass der Beobachter mit der Zeit etwas besser geworden ist.
Als Startwerte setzen wir daher -0,5, 0, 0,5*

für beta

```
summary(RBU$X2-RBU$X1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.0000  0.0000   0.0000  0.1176  0.0000   2.0000
```

```
summary(RBU$X3-RBU$X1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.0000 -1.0000   0.0000 -0.2059  0.0000   3.0000
```

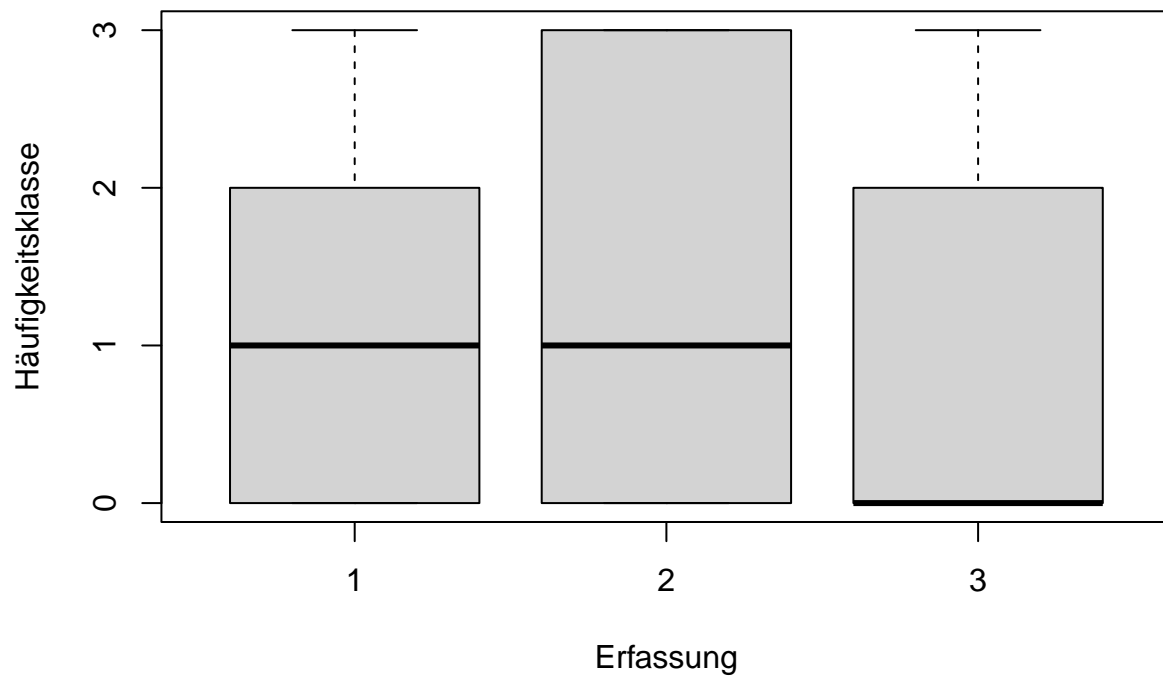
```
summary(RBU$X3-RBU$X2)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.0000  0.0000   0.0000 -0.3235  0.0000   2.0000
```

*# wir nehmen an, dass mit der Zeit weniger Arten übersehen werden.
Als Startwerte setzen wir daher -0,5, 0,5, 0,5*

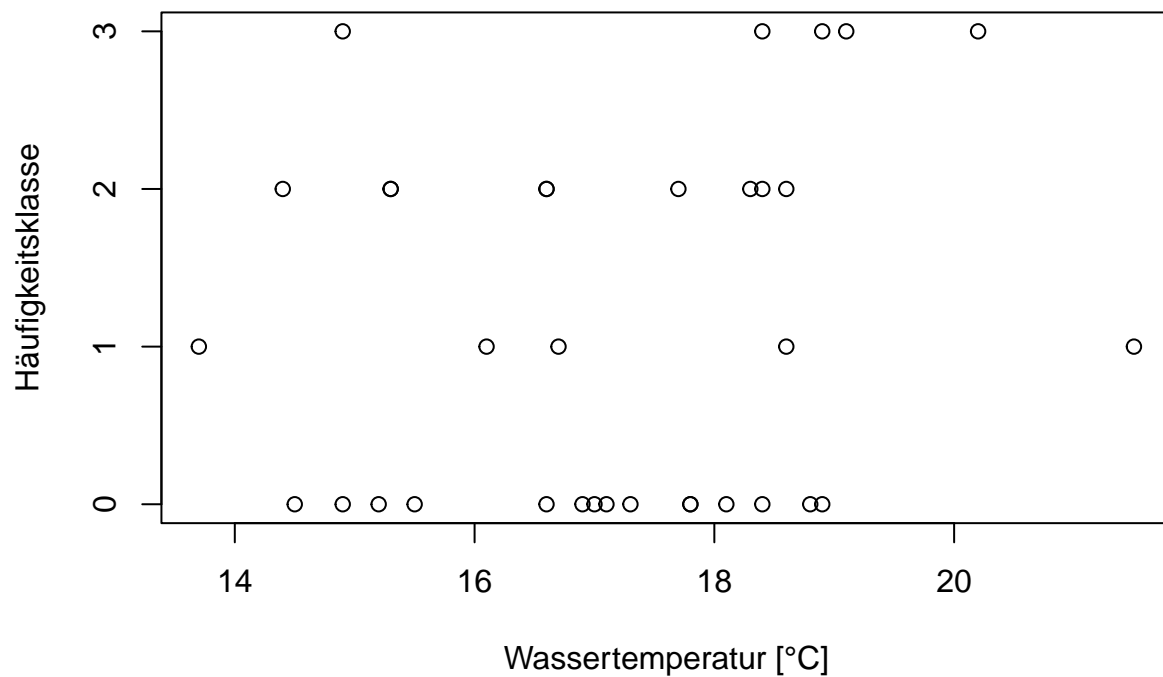
für Zeit b


```
{boxplot(RBU.data, xlab = "Erfassung", ylab = "Häufigkeitsklasse", yaxt="n")
axis(2, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))
}
```



```
# Die Aktivität nimmt wahrscheinlich zum Mai hin ab.
# Als Startwerte setzen wir daher 0, -0,5

# für Temperatur
{plot(RBU$X1~WTemp$X1, yaxt="n",
      xlab = "Wassertemperatur [°C]",
      ylab = "Häufigkeitsklasse")
axis(2, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))
}
```



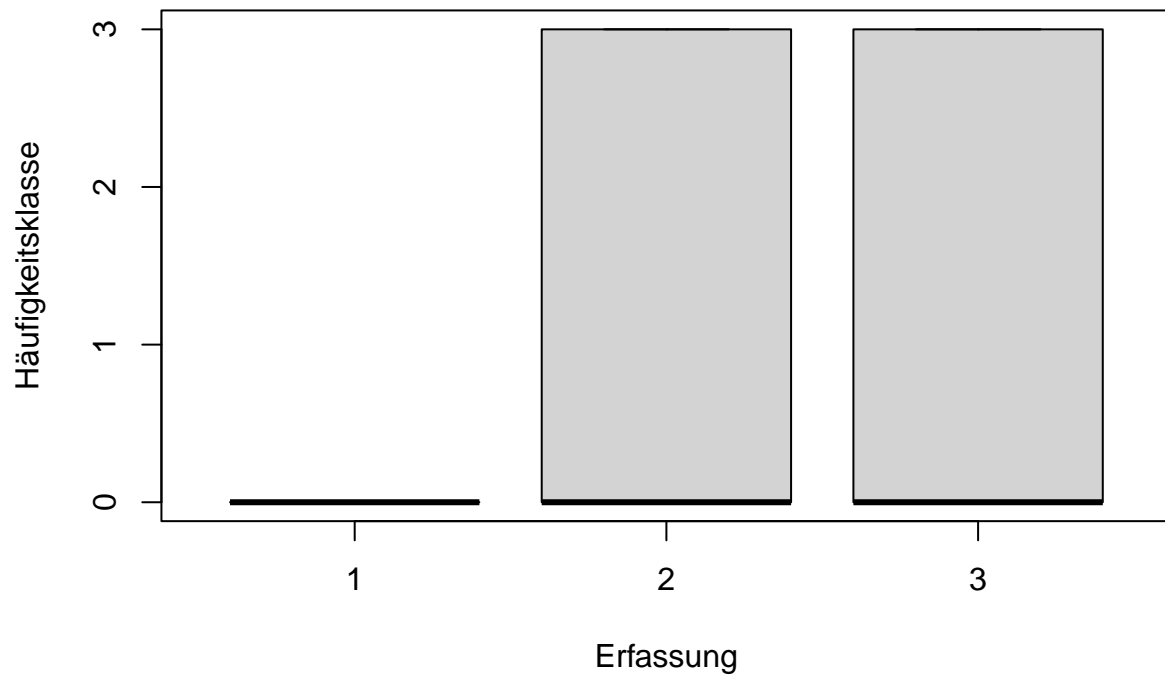
```
# kein Starker Zusammenhang erwartet
# Als Startwert setzen wir -0,5

st.RBU1 <- c(-0.5,0,0.5,-0.5,0.5,0.5,0,-0.5,-0.5)

# für den Fall des quadratischen Temperatureinflusses
# benötigt auch das einen Startwert

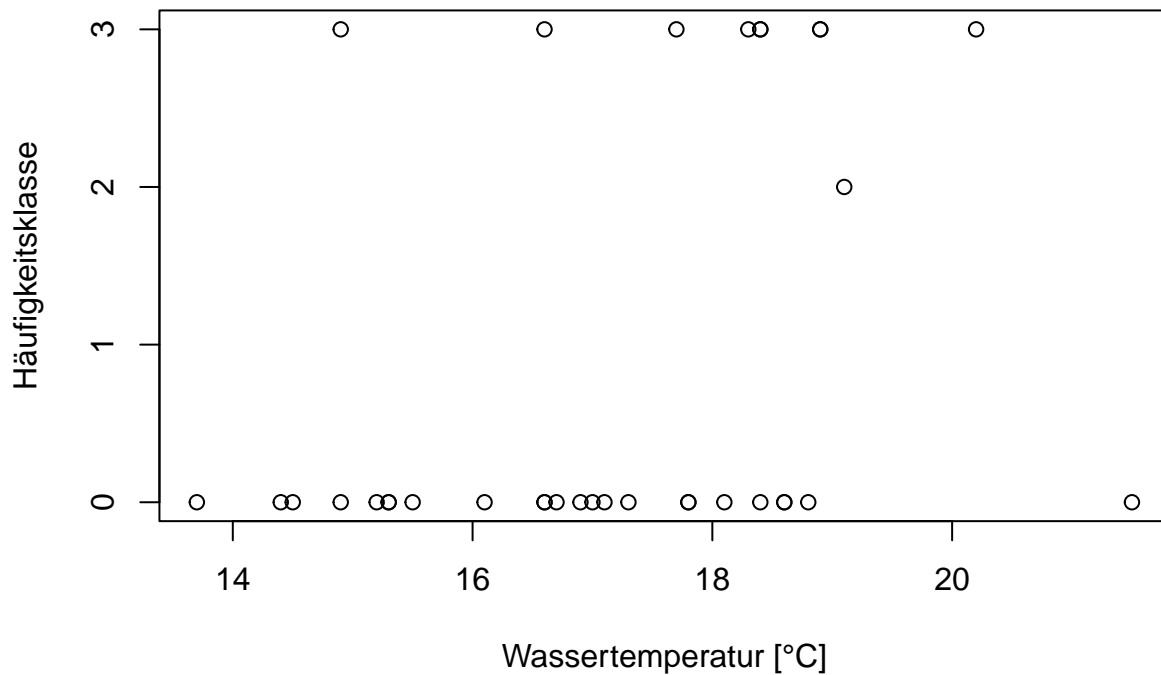
st.RBU2 <- c(-0.5,0,0.5,-0.5,0.5,0.5,0,-0.5,-0.5, -0.5)

# für den Laubfrosch schauen wir uns die Kovariaten noch einmal an
# für Zeit b
{boxplot(LF.data, xlab = "Erfassung", ylab = "Häufigkeitsklasse", yaxt="n")
  axis(2, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))
}
```



```
# Die Aktivität nimmt wahrscheinlich zum Mai hin zu.
# Als Startwerte setzen wir daher -2, 1

# für Temperatur
{plot(LF$X3~WTemp$X1, yaxt="n",
      xlab = "Wassertemperatur [°C]",
      ylab = "Häufigkeitsklasse")
  axis(2, at = c(0, 1, 2, 3), labels = c("0", "1", "2", "3"))
}
```



```
# mit der Temperatur könnte die Aktivität steigen
# Als Startwert setzen wir 1
# als quadratischen Term setzen wir weiterhin -0,5
```

```
st.LF1 <- c(-0.5,0,0.5,-0.5,0.5,0.5,-2,1,1)
st.LF2 <- c(-0.5,0,0.5,-0.5,0.5,0.5,-2,1,1, -0.5)
```

Es empfiehlt sich außerdem, die Temperatur-Kovariate zu standardisieren, sodass sie einen Mittelwert von 0 hat.

```
temp<-WTemp.data-mean(WTemp.data,na.rm=T)
```

Rotbauchunke

Modellanpassung

Bevor wir starten, muss der Datensatz M zugewiesen werden. Im Anschluss wird die Funktion mit ihrer oben definierten Struktur aufgerufen.

```
M <- RBU.data
```

```
RBU.out1 <- nlm(negloglik, st.RBU1, pim=PIM1, vars=vars1, hessian=T)
RBU.out2 <- nlm(negloglik, st.RBU2, pim=PIM1, vars=vars2, hessian=T)
RBU.out3 <- nlm(negloglik, st.RBU1, pim=PIM2, vars=vars1, hessian=T)
RBU.out4 <- nlm(negloglik, st.RBU2, pim=PIM2, vars=vars2, hessian=T)
```

Ergebnisinterpretation

Schauen wir uns exemplarisch ein Ergebnis an:

RBU.out1

```
## $minimum
## [1] 200.0119
##
## $estimate
## [1] 0.2563611 0.0000000 -0.3107747 -0.8284620 0.4438908 0.5000000 0.4957502
## [8] -1.5215499 0.4302958
##
## $gradient
## [1] 2.543743e-05 0.000000e+00 -2.074785e-06 5.425704e-05 -1.477929e-06
## [6] 0.000000e+00 -2.916067e-05 -1.559734e-05 -1.054445e-05
##
## $hessian
##      [,1] [,2]      [,3]      [,4]      [,5] [,6]      [,7]
## [1,] 17.4688296 0 0.8292545 -2.5072353 4.4903089 0 4.1995548
## [2,] 0.0000000 0 0.0000000 0.0000000 0.0000000 0 0.0000000
## [3,] 0.8292545 0 19.0396008 0.3857849 -4.3601318 0 -0.6521219
## [4,] -2.5072353 0 0.3857849 3.5367265 -0.9205593 0 -0.9391869
## [5,] 4.4903089 0 -4.3601318 -0.9205593 13.3891746 0 2.3612557
## [6,] 0.0000000 0 0.0000000 0.0000000 0.0000000 0 0.0000000
## [7,] 4.1995548 0 -0.6521219 -0.9391869 2.3612557 0 6.4929111
## [8,] 6.7800642 0 1.0139274 -0.8030440 0.9144827 0 -1.1610723
## [9,] 14.4335928 0 4.1496321 3.1087978 -8.1861458 0 -9.6633698
##      [,8] [,9]
## [1,] 6.7800642 14.433593
## [2,] 0.0000000 0.000000
## [3,] 1.0139274 4.149632
## [4,] -0.8030440 3.108798
## [5,] 0.9144827 -8.186146
## [6,] 0.0000000 0.000000
## [7,] -1.1610723 -9.663370
## [8,] 8.2922696 20.336196
## [9,] 20.3361964 111.271271
##
## $code
## [1] 1
##
## $iterations
## [1] 32
```

Folgende Werte sind angegeben:

minimum: Minimum der negativen Log-Likelihood-Funktion

estimate: Schätzwerte der Parameter, hier $p1$, $p2$, $p3$, β_{21} , β_{32} , β_{31} , time1 , time3 , temp1

gradient: Der Gradient zeigt die Ableitungen der negativen Log-Likelihood-Funktion bezüglich der Parameter. Nahe null liegende Werte (wie in diesem Fall) deuten darauf hin, dass das Optimum erreicht wurde. Dies ist ein gutes Zeichen für die Konvergenz des Optimierungsprozesses.

hessian: Die Hessian-Matrix enthält Informationen über die Krümmung der negativen Log-Likelihood-Funktion. Sie wird häufig verwendet, um die Unsicherheit der Parameter zu bewerten. Eine positive definit Matrix weist auf ein lokales Minimum hin.

code: Der Wert 1 für den Code bedeutet, dass die Optimierung erfolgreich war.

iterations: Der Wert 32 gibt die Anzahl der Iterationen an, die zur Konvergenz des Algorithmus benötigt wurden. Dies kann Aufschluss über die Komplexität des Modells geben; eine höhere Anzahl kann auf ein schwierigeres Optimierungsproblem hinweisen.

Modellvergleiche

Nutzen wir die Minima der negativen Log-Likelihood und korrigieren diese basierend auf der Anzahl Parameter, um den AIC zu berechnen (siehe Kapitel 9.3 des Buches) und die Modelle miteinander zu vergleichen:

```
RBU.out1$minimum + 2*length(RBU.out1$estimate)
```

```
## [1] 218.0119
```

```
RBU.out2$minimum + 2*length(RBU.out2$estimate)
```

```
## [1] 217.7399
```

```
RBU.out3$minimum + 2*length(RBU.out3$estimate)
```

```
## [1] 216.9218
```

```
RBU.out4$minimum + 2*length(RBU.out4$estimate)
```

```
## [1] 216.4964
```

Wir sehen, dass die Modelle 3 und 4 einen niedrigeren AIC Wert haben als 1 und 2. Die PIM2 ($p2$ und $p3$ verschieden von $p1$) hat also eine etwas bessere Modellgüte gezeigt. Es würde sich sicher lohnen, weitere Kombinationen miteinander zu vergleichen.

Zwischen den Kovariaten gibt es jedoch kaum einen Unterschied. Die quadratische Temperaturanpassung ist minimal besser als die lineare. Der Unterschied ist so gering, dass wir an dieser Stelle lieber die Ergebnisse der linearen Anpassung nutzen (wobei beides natürlich möglich wäre). Dies war Modell 3.

Belegungswahrscheinlichkeit basierend auf Häufigkeitsklassen

Nun können wir die erste Funktion `cp` nutzen, um aus den Schätzwerten (estimates) die tatsächlichen Belegungswahrscheinlichkeiten (psi) pro Häufigkeitsklasse zu berechnen.

```
cp(RBU.out3$estimate)
```

```
##      [,1]      [,2]      [,3]      [,4]
## [1,] 1 0.6599437 0.2204205 0.3165374
## [2,] 0 0.3400563 0.4161676 0.1919896
## [3,] 0 0.0000000 0.3634119 0.2538959
## [4,] 0 0.0000000 0.0000000 0.2375770
```

In der letzten Spalte finden sich die Belegungswahrscheinlichkeiten der HK von $N = 0, 1, 2$, und 3 . Geschätzt haben also 32 % der Gewässer keine Rufer, 19 % einzelne Rufer, 25 % einzelne überlappende Rufer und 24 % Chöre. Vergleichen wir das mit den Beobachtungen, sehen wir, dass wohl einige Rufer überhört worden sein müssen bzw. einer zu niedrigen Klasse zugeteilt worden sind - ganz im Gegensatz zu unserer ursprünglichen Annahme, dass dies nicht zu häufig auftritt. Dies scheint vor allem bei HK 2 der Fall zu sein, weniger bei HK 1.

Die mittlere HK, die zu erwarten ist (= geschätzte mittlere latente HK), berechnet sich als $N = 3 - 3 \times \text{psi0} - 2 \times \text{psi1} + \text{psi2}$.

```
psi <- cp(RBU.out3$estimate)[,4]
N = 3 - 3*psi[1] - 2*psi[2] + psi[3]
N
```

```
## [1] 1.920304
```

Zumeist sollte die RBU also als HK 2 aufgenommen werden.

Laubfrosch

Berechnen wir dasselbe in kurz für den Laubfrosch.

Modellanpassung

Bevor wir starten, muss der Datensatz `M` zugewiesen werden. Im Anschluss wird die Funktion mit ihrer oben definierten Struktur aufgerufen.

```
M <- LF.data

LF.out1 <- nlm(negloglik, st.LF1, pim=PIM1, vars=vars1, hessian=T)
LF.out2 <- nlm(negloglik, st.LF2, pim=PIM1, vars=vars2, hessian=T)
LF.out3 <- nlm(negloglik, st.LF1, pim=PIM2, vars=vars1, hessian=T)
LF.out4 <- nlm(negloglik, st.LF2, pim=PIM2, vars=vars2, hessian=T)
```

Modellvergleiche

Nutzen wir die Minima der negativen Log-Likelihood, korrigieren diese basierend auf der Anzahl Parameter, um den AIC zu berechnen (siehe Kapitel 9.3 des Buches) und die Modelle miteinander zu vergleichen:

```
LF.out1$minimum + 2*length(LF.out1$estimate)
```

```
## [1] 83.24714
```

```
LF.out2$minimum + 2*length(LF.out2$estimate)
```

```
## [1] 84.41436
```

```
LF.out3$minimum + 2*length(LF.out3$estimate)
```

```
## [1] 95.22179
```

```
LF.out4$minimum + 2*length(LF.out4$estimate)
```

```
## [1] 97.04836
```

In diesem Fall ist `vars1` (linearer Temperaturzusammenhang) definitiv besser als `vars2` (quadratischer Temperaturzusammenhang). Zwischen den beiden PIM gibt es jedoch ebenfalls einen Unterschied, wobei das weniger komplexe Modell mit PIM1 (Nachweiswahrscheinlichkeit konstant) besser abschnitt. Das beste Modell war also Modell 1. Auch hier sollten im Realfall weitere Kombinationen berechnet und verglichen werden.

Belegungswahrscheinlichkeit basierend auf Häufigkeitsklassen

Nun können wir die erste Funktion `cp` nutzen, um aus den estimates die tatsächlichen Belegungswahrscheinlichkeiten (*psi*) pro Häufigkeitsklasse zu berechnen.

```
psi <- cp(LF.out1$estimate)[,4]
psi
```

```
## [1] 2.863772e-01 1.104370e-10 2.414667e-01 4.721561e-01
```

Geschätzt haben also 29 % der Gewässer keine Rufer, 0 % einzelne Rufer, 24 % einzelne überlappende Rufer und 47 % Chöre. Auch hier müssen also einige Rufer überhört worden sein.

Die mittlere HK, die zu erwarten ist (= geschätzte mittlere latente HK), berechnet sich als $N = 3 - 3 \times \text{psi0} - 2 \times \text{psi1} + \text{psi2}$.

```
N = 3 - 3*psi[1] - 2*psi[2] + psi[3]
N
```

```
## [1] 2.382335
```

Zumeist sollte der LF also als HK 2 aufgenommen werden, mit Tendenz zur 3.

Literaturverzeichnis

- Müller, F. (2018): Auswirkungen von Habitat-, Witterungs- und Durchflutungsparametern auf das jährliche und räumliche Vorkommen von Amphibienarten in den Papitzer Lehmlachen. — Masterarbeit, Universität Leipzig.
- Royle, J.A., Link, W.A. (2005): A general class of multinomial mixture models for anuran calling survey data. — *Ecology* 86: 2505–2512.
- Weir, L.A., Mossman, M.J. (2005): North American amphibian monitoring program. — S. 307–313 in: Lannoo, M.J. (Hrsg.): — *Declining amphibians: A United States response to the global phenomenon*. University of California Press, Berkeley.