

Modelos de marcação e recaptura: múltiplos estados

Leonardo Wedekin e Paulo Inácio Prado (<http://ecologia.ib.usp.br/bie5703>)

Contents

Preparação	1
Ajuste dos modelos	1
Seleção de modelos	3
Valores das estimativas	3
Transições proibidas e manipulação da matriz de delineamento	6
Para saber mais	8

-
- [Arquivo em pdf](#)
 - [Arquivo em markdown](#) (para executar os comandos no R studio)
-

Preparação

Vamos usar o *RMark*, que é um pacote do R para usar o programa [MARK](#). Siga as instruções do [site do RMark](#) para instalação.

Com o RMark instalado, abra o R e carregue-o:

```
library(RMark)
```

Usaremos dados simulados de marcação e recaptura com quatro ocasiões de captura em que cada indivíduo pode estar em 3 estados.

Os dados estão no formato nativo do MARK (*.inp*). Use os comandos abaixo para importá-lo para o R:

```
## Link dos dados na página da disciplina
url <- "http://ecologia.ib.usp.br/bie5703/lib/exe/fetch.php?media=roteiros:multi.inp"
## Importa arquivo inp
## use.comments=TRUE usa os rótulos das linhas (id dos indivíduos)
multi.raw <- convert.inp(url)
```

Ajuste dos modelos

Processamento dos dados

O primeiro passo é usar a função `process.data` para criar um objeto com as informações que o Mark precisa para ajustar o modelo. Uma delas é o tipo de modelo, que é indicado no argumento `model`.

Vamos usar o modelo multiestados básico, cuja a sigla no RMark é “Multistrata” ¹.

¹A lista de modelos implementados no RMark está no diretório onde o R instalou o pacote. Você pode consultá-lo lá ou no repositório de desenvolvimento do RMark: (<https://github.com/jlaake/RMark/blob/master/RMark/inst/MarkModels.pdf>).

```
## Processa os dados
multi <- process.data(data=multi.raw, model="Multistrata")
```

O objeto resultante é uma lista, na qual podemos conferir o número de ocasiões de estratos ou o nome dos estratos, por exemplo:

```
## Conteúdo da lista
names(multi)
```

```
## [1] "data"          "model"          "mixtures"
## [4] "freq"          "nocc"           "nocc.secondary"
## [7] "time.intervals" "begin.time"     "age.unit"
## [10] "initial.ages"  "group.covariates" "nstrata"
## [13] "strata.labels" "counts"         "reverse"
```

```
## N de ocasiões primárias
multi$nocc
```

```
## [1] 4
```

```
## N de estratos
multi$nstrata
```

```
## [1] 3
```

```
## Rótulos dos estratos
multi$strata.labels
```

```
## [1] "A" "B" "C"
```

Ajuste dos modelos

Para ajustar os modelos, crie listas que especificam a fórmula de cada termo. No modelo `Multistrata` os nomes parâmetros são:

- S : probabilidade de sobrevivência entre capturas (S),
- p : probabilidade de recaptura (p),
- Ψ : probabilidade de transição entre estados (ψ)

O RMark já cria algumas covariáveis para cada tipo de modelo ². Neste roteiro vamos usar três delas:

- covariável `time`, com um nível para cada intervalo entre seções primárias,
- `stratum`, para definir valores dos parâmetros para cada estrato
- `tostratum`, para definir transições entre estratos.

Podemos então criar objetos com as fórmulas para diferentes combinações dessas covariáveis. Algumas possibilidades:

²Para entender completamente isso estude o comando `make.design.data` e o objeto que ele cria, que é a uma lista de matrizes de delineamento do modelo. Se você entender este objeto saberá quais covariáveis estão disponíveis e como manipulá-las. Veja também o [apêndice sobre o RMark](#) no guia *online* do MARK.

```
## Fórmulas estatísticas para cada parâmetro do modelo
## formula para expressar um parâmetro constante
f.dot <- list(formula=~1)
## formula para expressar um parametro que varia entre estratos
f.stratum <- list(formula=~1+stratum)
## formula para expressar um parametro que varia entre ocasiões
f.time <- list(formula=~1+time)
## formula para expressar parametro diferente para cada combinacao de estrato e ocasiao
f.stratum.time <- list(formula=~1+stratum*time)
## formula para expressar transicoes diferentes entre estratos
f.tostratum <- list(formula=~1+stratum:tostratum)
```

E usamos a função `mark` para fazer os ajustes, usando os objetos com as fórmulas ³:

```
multi0 <- mark(multi,
               model.parameters=
                 list(S=f.dot, p=f.dot, Psi=f.dot),
                 adjust=TRUE, delete=TRUE)
multi1 <- mark(multi, model.parameters=
               list(S=f.stratum, p=f.time, Psi=f.dot),
               adjust=TRUE, delete=TRUE)
multi2 <- mark(multi, model.parameters=
               list(S=f.stratum, p=f.time, Psi=f.stratum),
               adjust=TRUE, delete=TRUE)
multi3 <- mark(multi, model.parameters=
               list(S=f.stratum, p=f.stratum.time, Psi=f.tostratum),
               adjust=TRUE, delete=TRUE)
```

Seleção de modelos

A função abaixo retorna a tabela de seleção de modelos:

```
collect.models(lx=c("multi0", "multi1", "multi2", "multi3"))
```

```
##                                     model npar
## 1                                     S(~1)p(~1)Psi(~1)    3
## 2                                     S(~-1 + stratum)p(~-1 + time)Psi(~1)    7
## 3                                     S(~-1 + stratum)p(~-1 + time)Psi(~-1 + stratum)    9
## 4 S(~-1 + stratum)p(~-1 + stratum * time)Psi(~-1 + stratum:tostratum)   18
##      AICc DeltaAICc      weight Deviance
## 1 30449.77  0.000000 9.495636e-01 42.74341
## 2 30455.90  6.126414 4.438030e-02 40.86196
## 3 30459.88 10.110145 6.055278e-03 40.84061
## 4 30477.78 28.011652 7.850027e-07 40.69884
```

Valores das estimativas

Os objetos dos modelos ajustados são uma lista com todo o output do Mark. Esta lista tem um elemento `results`, com os valores dos coeficientes e muito mais ⁴:

³Use sempre os argumentos `delete=TRUE` para remover os arquivos temporários do Mark que o ajuste cria e `adjust=TRUE` para que o número de parâmetros seja verificado e ajustado, para o cálculo do AIC.

⁴Consulte o [apêndice sobre o RMark](#) no guia *online* do MARK.

```
names(multi1$results)
```

```
## [1] "lnl"           "deviance"      "deviance.df"
## [4] "npar"          "n"             "AICc"
## [7] "beta"          "real"          "beta.vcv"
## [10] "derived"       "derived.vcv"   "covariate.values"
## [13] "singular"      "real.vcv"
```

Os coeficientes na escala da função de ligação estão no dataframe `beta` desta lista. Verifique estes valores para o modelo selecionado:

```
multi1$results$beta
```

```
##           estimate      se      lcl      ucl
## S:stratumA 0.8160815 0.0616512 0.6952452 0.9369178
## S:stratumB 0.8160821 0.0616512 0.6952457 0.9369185
## S:stratumC 0.8027114 0.0613533 0.6824589 0.9229638
## p:time2    0.0262115 0.0505845 -0.0729341 0.1253572
## p:time3    0.0375559 0.0471180 -0.0547954 0.1299071
## p:time4   -0.0295653 0.0538095 -0.1350319 0.0759012
## Psi:(Intercept) -1.1033394 0.0298652 -1.1618752 -1.0448036
```

Os coeficientes na escala original de probabilidades estão no dataframe `real`

```
multi1$results$real
```

```
##           estimate      se      lcl      ucl fixed
## S sA g1 c1 c1 a0 o1 t1 0.6934039 0.0131067 0.6671327 0.7184766
## S sB g1 c1 c1 a0 o1 t1 0.6934040 0.0131067 0.6671328 0.7184768
## S sC g1 c1 c1 a0 o1 t1 0.6905542 0.0131105 0.6642873 0.7156456
## p sA g1 c1 c1 a1 o1 t2 0.5065525 0.0126440 0.4817745 0.5312983
## p sA g1 c1 c1 a2 o2 t3 0.5093879 0.0117753 0.4863046 0.5324312
## p sA g1 c1 c1 a3 o3 t4 0.4926092 0.0134494 0.4662932 0.5189662
## Psi sA toB g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toB g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toB g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toB g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toB g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toB g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sA toC g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toA g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toC g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
```

```

## Psi sB toC g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toC g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toC g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toC g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sB toC g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toA g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c1 c1 a0 o1 t1 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c1 c1 a1 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c1 c1 a2 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c2 c2 a0 o2 t2 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c2 c2 a1 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
## Psi sC toB g1 c3 c3 a0 o3 t3 0.1994330 0.0035804 0.1925080 0.2065434
##
##                                     note
## S sA g1 c1 c1 a0 o1 t1
## S sB g1 c1 c1 a0 o1 t1
## S sC g1 c1 c1 a0 o1 t1
## p sA g1 c1 c1 a1 o1 t2
## p sA g1 c1 c1 a2 o2 t3
## p sA g1 c1 c1 a3 o3 t4
## Psi sA toB g1 c1 c1 a0 o1 t1
## Psi sA toB g1 c1 c1 a1 o2 t2
## Psi sA toB g1 c1 c1 a2 o3 t3
## Psi sA toB g1 c2 c2 a0 o2 t2
## Psi sA toB g1 c2 c2 a1 o3 t3
## Psi sA toB g1 c3 c3 a0 o3 t3
## Psi sA toC g1 c1 c1 a0 o1 t1
## Psi sA toC g1 c1 c1 a1 o2 t2
## Psi sA toC g1 c1 c1 a2 o3 t3
## Psi sA toC g1 c2 c2 a0 o2 t2
## Psi sA toC g1 c2 c2 a1 o3 t3
## Psi sA toC g1 c3 c3 a0 o3 t3
## Psi sB toA g1 c1 c1 a0 o1 t1
## Psi sB toA g1 c1 c1 a1 o2 t2
## Psi sB toA g1 c1 c1 a2 o3 t3
## Psi sB toA g1 c2 c2 a0 o2 t2
## Psi sB toA g1 c2 c2 a1 o3 t3
## Psi sB toA g1 c3 c3 a0 o3 t3
## Psi sB toC g1 c1 c1 a0 o1 t1
## Psi sB toC g1 c1 c1 a1 o2 t2
## Psi sB toC g1 c1 c1 a2 o3 t3
## Psi sB toC g1 c2 c2 a0 o2 t2
## Psi sB toC g1 c2 c2 a1 o3 t3
## Psi sB toC g1 c3 c3 a0 o3 t3
## Psi sC toA g1 c1 c1 a0 o1 t1
## Psi sC toA g1 c1 c1 a1 o2 t2
## Psi sC toA g1 c1 c1 a2 o3 t3
## Psi sC toA g1 c2 c2 a0 o2 t2
## Psi sC toA g1 c2 c2 a1 o3 t3
## Psi sC toA g1 c3 c3 a0 o3 t3

```

```
## Psi sC toB g1 c1 c1 a0 o1 t1
## Psi sC toB g1 c1 c1 a1 o2 t2
## Psi sC toB g1 c1 c1 a2 o3 t3
## Psi sC toB g1 c2 c2 a0 o2 t2
## Psi sC toB g1 c2 c2 a1 o3 t3
## Psi sC toB g1 c3 c3 a0 o3 t3
```

Matrizes de transição

O RMark tem uma função para exibir as estimativas de ψ na forma de uma matriz de transição entre estados:

```
## Crie uma lista com todas as estimativas e sua matriz de covariancia
Psilist <- get.real(model = multi0, parameter = "Psi", vcv=TRUE)
## Conveniência: objeto apenas com as estimativas
Psivalues=Psilist$estimates
## Matrise com probabilidades de transição, seus erros e intervalo de confiança
TransitionMatrix(Psivalues[Psivalues$time==1,], vcv.real=Psilist$vcv.real)
```

```
## $TransitionMat
##           A           B           C
## A 0.6011332 0.1994334 0.1994334
## B 0.1994334 0.6011332 0.1994334
## C 0.1994334 0.1994334 0.6011332
##
## $se.TransitionMat
##           A           B           C
## A 0.007160855 0.003580427 0.003580427
## B 0.003580427 0.007160855 0.003580427
## C 0.003580427 0.003580427 0.007160855
##
## $lcl.TransitionMat
##           A           B           C
## A 0.5870184 0.1925084 0.1925084
## B 0.1925084 0.5870184 0.1925084
## C 0.1925084 0.1925084 0.5870184
##
## $ucl.TransitionMat
##           A           B           C
## A 0.6150819 0.2065438 0.2065438
## B 0.2065438 0.6150819 0.2065438
## C 0.2065438 0.2065438 0.6150819
```

Transições proibidas e manipulação da matriz de delineamento

Algumas transições podem ser impossíveis, como por exemplo a passagem do estado de larva para adulto sem passar por pupa. Os parâmetros ψ correspondentes a estas transições devem então ser fixados em zero. No Rmark isso é feito manipulando-se a matriz de delineamento (*design data*). Quando usamos o comando `mark` sobre o objeto de dados esta matriz é criada internamente mas não é exibida. Vamos criar esta matriz para trabalhar com ela:

```
multi.ddl <- make.design.data(multi)
```

E o modelo com probabilidades de transição entre estados diferentes pode ser ajustado com

```
multi4 <- mark(data=multi, ddl=multi.ddl, model.parameters=
  list(S=f.stratum, p=f.dot, Psi=f.tostratum),
  adjust=TRUE, delete=TRUE)
```

Que neste caso dá na mesma que omitir a matriz de delineamento

```
multi4b <- mark(data=multi, model.parameters=
  list(S=f.stratum, p=f.dot, Psi=f.tostratum),
  adjust=TRUE, delete=TRUE)
```

```
all.equal(multi4$results$beta, multi4b$results$beta, tolerance=1e-5)
```

Explicitar a matriz de delineamento tem a vantagem de que podemos manipulá-la para obter vários outros tipos de modelos que a especificação das fórmulas não permite. Isto oferece uma enorme flexibilidade e é uma das principais vantagens do RMark.

Suponha que as transições entre os estados A e C não possam acontecer, ou que você quer avaliar se impor isto resulta em um modelo melhor. Primeiro temos que criar uma coluna que indica valores fixos na matriz de delineamento do parâmetro ψ . Esta coluna deve ter o nome `fix` e valores `NA` para os parâmetros que devem ser estimados

```
multi.ddl$Psi$fix <- NA
```

Agora indicamos o valor zero para as transições que queremos anular:

```
multi.ddl$Psi$fix[multi.ddl$Psi$stratum=="A" & multi.ddl$Psi$tostratum=="C"] <- 0
multi.ddl$Psi$fix[multi.ddl$Psi$stratum=="C" & multi.ddl$Psi$tostratum=="A"] <- 0
```

E agora ajuste o modelo com esta nova matriz de delineamento:

```
multi4c <- mark(data=multi, ddl=multi.ddl, model.parameters=
  list(S=f.stratum, p=f.dot, Psi=f.tostratum),
  adjust=TRUE, delete=TRUE)
```

Verificando as matrizes de transição

```
## modelo sem transicoes fixas
Psilist <- get.real(model = multi4, parameter = "Psi", vcv=TRUE)
Psivalues=Psilist$estimates
TransitionMatrix(Psivalues[Psivalues$time==1,])
```

```
##           A           B           C
## A 0.6018088 0.1992482 0.1989430
## B 0.1992483 0.6018087 0.1989430
## C 0.2001097 0.2001095 0.5997808
```

```
## Com trasicoes A->C e C->A fixas em zero
Psilist <- get.real(model = multi4c, parameter = "Psi", vcv=TRUE)
Psivalues=Psilist$estimates
TransitionMatrix(Psivalues[Psivalues$time==1,])
```

```
##           A           B           C
## A 0.7028205 0.2971795 0.0000000
## B 0.2238422 0.5532695 0.2228883
## C 0.0000000 0.2986074 0.7013926
```

No caso não foi uma boa ideia anular algumas transições:

```
collect.models(lx=c("multi4", "multi4c"))
```

```
##                                     model npar      AICc
## 1 S(~-1 + stratum)p(~1)Psi(~-1 + stratum:tostratum)  10 30463.72
## 2 S(~-1 + stratum)p(~1)Psi(~-1 + stratum:tostratum)   8 769485.93
##   DeltaAICc weight    Deviance
## 1       0.0      1    42.67842
## 2  739022.2      0  739068.89000
```

Este resultado faz sentido, já que as transições anuladas são observadas nos dados !

Para saber mais

- [Capítulo sobre modelos multiestado](#) no guia *online* do Mark
- Para outro exemplo de ajuste do modelo robusto e outras funções úteis do RMark digite `?mstrata`
- Veja também o pacote Rcapture