

# Modelos de ocorrência: uma espécie, uma estação

## Contents

Preparação . . . . .	1
Ajuste dos modelos . . . . .	2
Seleção de modelos . . . . .	5
Cálculo do previsto . . . . .	5

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

## Preparação

Abra o R e carregue os pacotes necessários

```
library(unmarked)
library(RMark)
library(stringr)
library(plyr)
```

Usaremos dados de registros do esquilo *Spermophilus tereticaudus chlorus* em 1917 plots no deserto americano. [Aqui](#) há mais informações sobre este caso de estudo.

Os dados estão no formato nativo do MARK (.inp). Use os comandos abaixo para convertê-los para um objeto da classe *unmarkedFrame*, do pacote *unmarked*:

```
## Link dos dados na página da disciplina
url <- "http://ecologia.ib.usp.br/bie5703/lib/exe/fetch.php?media=roteiros:esquilos.inp"
## Importa arquivo inp
tmp <- convert.inp(url,
                    group.df=data.frame(habitat=c("Mesquite", "Creosote", "Shrub", "Other")),
                    covariates="distance")
## Seleciona historico de capturas e converte em data frame
y <- str_split(tmp$ch, pattern="")
y <- ldply(y, as.numeric)[,2:4]
## Cria objeto para o modelo de ocupação do unmarked
## (Veja vinhetas para os outros tipos de modelos e seus objetos)
esq <- unmarkedFrameOccu(y = y, siteCovs = tmp[,c("habitat","distance")])

## Verifica objeto
summary(esq)
```

```
## unmarkedFrame Object
##
## 1917 sites
## Maximum number of observations per site: 3
## Mean number of observations per site: 1.66
## Sites with at least one detection: 50
##
## Tabulation of y observations:
##      0      1 <NA>
## 3133    58 2560
##
## Site-level covariates:
##      habitat      distance
## Creosote: 206   Min.      : 0
## Mesquite:  17   1st Qu.: 1896
## Other    :1533   Median : 3036
## Shrub    : 161   Mean    : 3560
##                      3rd Qu.: 4478
##                      Max.    :14979
```

## Ajuste dos modelos

O pacote *unmarked* usa a [sintaxe de modelos lineares](#) do R e tem funções para diferentes tipos de modelos de ocupação. Consulte as vinhetas do pacote para mais informações

```
## Lista da vinhetas
vignette(package="unmarked")
## Abre pdf da vinheta de introdução
vignette(topic="unmarked", package="unmarked")
```

Para os modelos de ocupação com covariáveis usamos função *occu*. Seu primeiro argumento é uma fórmula com o formato

~covariáveis de detecção ~covariáveis de ocupação

Um modelo com probabilidade de ocupação e detecção constantes:

```
## Ajuste.
## ~1 indica constante
esq.m1 <- occu(~1 ~1, data=esq)
## Resumo do modelo
summary(esq.m1)
```

```
##
## Call:
## occu(formula = ~1 ~ 1, data = esq)
##
## Occupancy (logit-scale):
##      Estimate      SE      z P(>|z|)
##      -2.09 0.355 -5.89 3.75e-09
##
## Detection (logit-scale):
```

```
## Estimate SE z P(>|z|)
## -1.68 0.369 -4.56 5.11e-06
##
## AIC: 560.9127
## Number of sites: 1917
## optim convergence code: 0
## optim iterations: 23
## Bootstrap iterations: 0
```

```
## Coeficientes na escala logito
coef(esq.m1)
```

```
## psi(Int) p(Int)
## -2.094588 -1.681867
```

```
## Intervalos de confiança dos coeficientes
confint(esq.m1, type='det') #p
```

```
## 0.025 0.975
## p(Int) -2.404709 -0.9590243
```

```
confint(esq.m1, type='state') #psi
```

```
## 0.025 0.975
## psi(Int) -2.791033 -1.398142
```

Um modelo com probabilidade de detecção variável entre as ocasiões:

```
## ~obsNum indica uma detectabilidade por categoria de observação (ocasiões)
esq.m2 <- occu(~obsNum ~1, data=esq)
## Resumo do modelo
summary(esq.m2)
```

```
##
## Call:
## occu(formula = ~obsNum ~ 1, data = esq)
##
## Occupancy (logit-scale):
## Estimate SE z P(>|z|)
## -2.11 0.367 -5.77 8.07e-09
##
## Detection (logit-scale):
## Estimate SE z P(>|z|)
## (Intercept) -1.6237 0.427 -3.802 0.000144
## obsNum2 -0.2177 0.368 -0.591 0.554534
## obsNum3 0.0696 0.403 0.173 0.862909
##
## AIC: 564.4065
## Number of sites: 1917
## optim convergence code: 0
## optim iterations: 35
## Bootstrap iterations: 0
```

Modelo em que a detecção varia entre ocasiões e a ocupação depende do tipo de habitat:

```
esq.m3 <- occu(~obsNum ~habitat, data=esq)
## Resumo do modelo
summary(esq.m3)

##
## Call:
## occu(formula = ~obsNum ~ habitat, data = esq)
##
## Occupancy (logit-scale):
##           Estimate      SE      z  P(>|z|)
## (Intercept)   -0.764   0.398 -1.919 5.49e-02
## habitatMesquite 10.556 66.625  0.158 8.74e-01
## habitatOther   -2.909   0.410 -7.099 1.26e-12
## habitatShrub   -1.724   0.609 -2.831 4.65e-03
##
## Detection (logit-scale):
##           Estimate      SE      z  P(>|z|)
## (Intercept)   -1.067 0.335 -3.180 0.00147
## obsNum2       -0.467 0.386 -1.210 0.22612
## obsNum3       -0.211 0.421 -0.501 0.61637
##
## AIC: 470.8923
## Number of sites: 1917
## optim convergence code: 0
## optim iterations: 50
## Bootstrap iterations: 0
```

Como o modelo acima, mas com a ocupação dependendo também da distância a sítios do habitat *mesquite*:

```
esq.m4 <- occu(~obsNum ~habitat+distance, data=esq)
## Resumo do modelo
summary(esq.m4)

##
## Call:
## occu(formula = ~obsNum ~ habitat + distance, data = esq)
##
## Occupancy (logit-scale):
##           Estimate      SE      z  P(>|z|)
## (Intercept)    2.17905 0.719038  3.031 2.44e-03
## habitatMesquite 2.72058 4.666027  0.583 5.60e-01
## habitatOther   -3.34345 0.637737 -5.243 1.58e-07
## habitatShrub   -1.81245 0.872187 -2.078 3.77e-02
## distance       -0.00124 0.000154 -8.055 7.93e-16
##
## Detection (logit-scale):
##           Estimate      SE      z  P(>|z|)
## (Intercept)   -1.429 0.342 -4.178 2.94e-05
## obsNum2       -0.552 0.378 -1.461 1.44e-01
## obsNum3       -0.282 0.404 -0.699 4.84e-01
##
```

```
## AIC: 460.1755
## Number of sites: 1917
## optim convergence code: 0
## optim iterations: 81
## Bootstrap iterations: 0
```

## Seleção de modelos

O *unmarked* tem funções para criar uma lista de modelos e então realizar sua seleção por diversos critérios

```
modelos <- fitList("p(.)psi(.)"=esq.m1,
                  "p(data)psi(.)"=esq.m2,
                  "p(data)psi(habitat)"=esq.m3,
                  "p(data)psi(habitat+dist)"=esq.m4)
modSel(modelos)
```

##		nPars	AIC	delta	AICwt	cumltvWt
##	p(data)psi(habitat+dist)	8	460.18	0.00	1.0e+00	1.00
##	p(data)psi(habitat)	7	470.89	10.72	4.7e-03	1.00
##	p(.)psi(.)	2	560.91	100.74	1.3e-22	1.00
##	p(data)psi(.)	4	564.41	104.23	2.3e-23	1.00

O modelo de menor AIC ( e portanto  $\Delta AIC = 0$ ) é o mais plausível. Convenciona-se que modelos com  $\Delta AIC \leq 2$  são tão plausíveis quanto o selecionado.

## Cálculo do previsto

O padrão dos modelos de ocupação é usar a função `logito` para as probabilidades de detecção e ocupação:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$

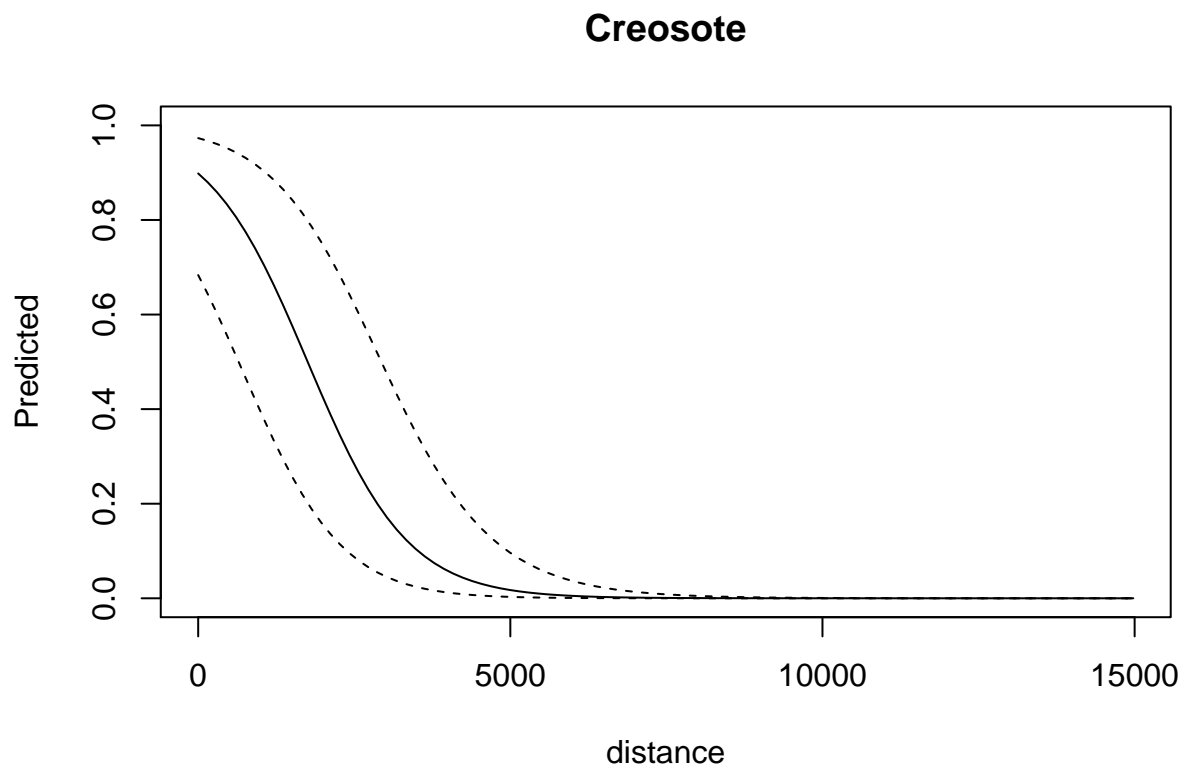
Portanto os coeficientes retornados pelas funções `summary` e `coef` estão nesta escala. Para obter as probabilidades estimadas pelo modelo na escala original use a função `predict`.

Abaixo um exemplo deste cálculo para as probabilidades de ocupação previstas pelo modelo selecionado, que prevê efeito de habitat e de distância:

```
## primeiro criamos um dataframe com os valores das covariaveis em que faremos as previsões
## Objeto com as covariaveis
cv1 <- siteCovs(esq)
## Dataframe com as combinacoes dos 4 habitats e
## 100 distancias de zero ao maximo
df1 <- expand.grid(habitat=levels(cv1$habitat),
                  distance=seq(0, max(cv1$distance), length=100))
esq.m4.pred <- predict(esq.m4, type='state', newdata = df1)
```

E um exemplo de gráfico dos previstos e seus intervalos de confiança para os plots no habitat *Creosote*:

```
## Juntando os previstos as covariaveis
esq.m4.pred <- cbind(df1, esq.m4.pred)
## Plot de psi x distância para o habitat Creosote
plot(Predicted ~ distance, data=esq.m4.pred,
     subset=habitat=="Creosote",
     ylim=c(0,1), type="l",
     main="Creosote")
lines(upper ~ distance, data=esq.m4.pred,
      subset=habitat=="Creosote", lty=2)
lines(lower ~ distance, data=esq.m4.pred,
      subset=habitat=="Creosote", lty=2)
```



Repita os gráficos dos previstos para os outros habitats.