

Prática de Computador V: Forrageio

Marco A. R. Mello

Universidade de São Paulo

Instituto de Biociências

Departamento de Ecologia

Tópicos Avançados em Ecologia de Animais (BIE0315)

Profs. José Carlos Motta Jr. & Marco A. R. Mello

Artigo de base: Sand et al. (2016, PLoS One)

Sumário

Preparativos

Teste 1

Teste 2

Teste 3

Teste 4

Para saber mais

Preparativos

Leia o tutorial desta prática, disponível aqui em formato PDF e também no moodle da disciplina.

Defina o diretório de trabalho como sendo a origem deste script.

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

Remova todos os objetos prévios.

```
rm(list= ls())
```

Limpe o console.

```
cat("\014")
```

Carregue os pacotes necessários.

```
library(bbmle)
```

```
## Carregando pacotes exigidos: stats4
```

```
library(ggplot2)
```

```
library(glm2)
```

```
library(lme4)
```

```
## Carregando pacotes exigidos: Matrix
```

```
library(MASS)
```

```
##
```

```
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:glm2':
```

```
##
```

```
##      crabs
```

Importe os dados e inspecione-os.

```
dados = read.delim("dados.txt", na.strings = "NA")
```

```
dim(dados)
```

```
## [1] 365  8
```

```
head(dados)
```

```
##      territory preytype moosedensity roedensity daynumber timesincekill
## 1           1         1          3.93          0         48           NA
## 2           1         1          0.81          0         59          10.3
## 3           1         1          2.77          0         61           2.7
## 4           1         1          0.99          0         62           1.1
## 5           1         1          8.30          0         65           2.9
## 6           1         1          0.14          0         69           4.0
##      nutritionalstatus snowdepthm
## 1                     NA         0.62
## 2                     20         0.57
## 3                     77         0.56
## 4                     85         0.56
## 5                    134         0.66
## 6                     68         0.64
```

```
tail(dados)
```

```
##      territory preytype moosedensity roedensity daynumber timesincekill
## 360          16         0          1.88          0        108          24.3
## 361          16         1          2.10          0         61           NA
## 362          16         1          0.38          0         65           3.8
## 363          16         1          2.30          0         68           2.7
## 364          16         0          2.30          0         81          13.8
## 365          16         0          2.36          0         90           8.7
##      nutritionalstatus snowdepthm
## 360                 -27         0.00
## 361                  NA         0.27
## 362                  55         0.17
## 363                  68         0.00
## 364                 -55         0.00
```

```
## 365          2      0.00
```

Deixe os dados já carregados na memória. Isso facilita a sua vida, possibilitando escrever apenas os nomes das colunas nos comandos, sem o nome do objeto.

```
attach(dados)
```

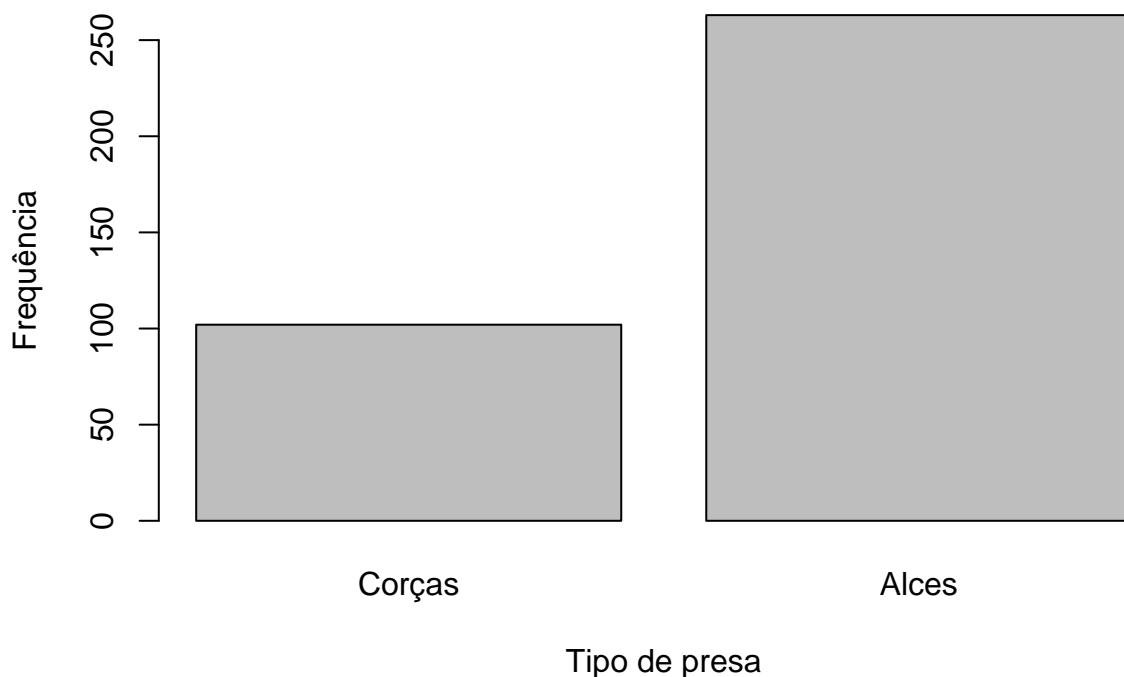
Teste 1

Examine quantas presas de cada tipo foram mortas pelos lobos. Ha várias formas de produzir esse resultado: você conhece outras além desta?

```
counts = table(pretype) # 0 = corças, 1 = alces
counts
```

```
## pretype
##  0   1
## 102 263
```

```
barplot(counts,
        xlab = "Tipo de presa",
        names=c("Corças", "Alces"),
        ylab = "Frequência")
```



Pense com calma sobre quais fatores devem ser mais importantes para determinar a escolha das presas pelos lobos. Não saia pescando resultados a esmo. Lembre-se das correlações espúrias!

Plote as relações entre os fatores escolhidos, adaptando este exemplo aos fatores de sua escolha. Dá para alterar várias coisas nos parâmetros que geram as camadas de desenho do ggplot.

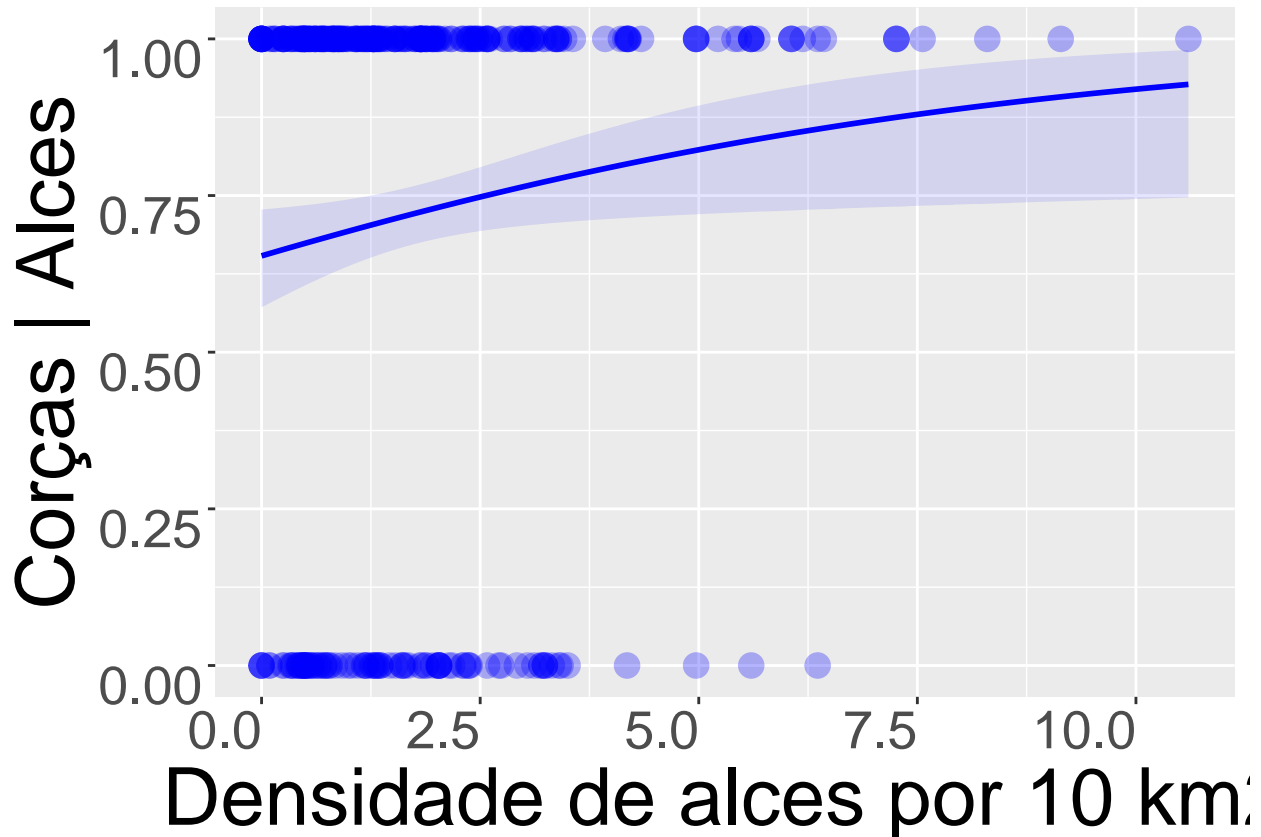
```
p1 = ggplot(dados, aes(x=moosedensity, y=pretype)) +
  geom_point(colour = "blue", size=4, alpha = 0.3) +
  stat_smooth(method="glm", method.args=list(family="binomial"),
             se=T, colour = "blue", fill = "blue", alpha = 0.1) +
  labs(x="Densidade de alces por 10 km2", y = "Corças | Alces") +
  theme(text = element_text(size=20),
```

```
plot.title = element_text(size=40, hjust=0.5),
axis.text.x = element_text(size = 20, angle=0, hjust=1),
axis.text.y = element_text(size = 20, angle=0, vjust=1),
axis.title.x = element_text(size = 30, angle=0),
axis.title.y = element_text(size = 30, angle=90))
```

Dê uma olhada no gráfico produzido.

```
p1 #essa nuvem em torno da reta de tendência é o intervalo de confiança
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Quer exportar o gráfico como uma imagem de alta resolução usando comandos, ao invés de exportá-lo pela janelinha de export? Experimente esta solução:

```
png(filename= "figuras/p1.png", res= 300, height= 3000, width= 3500)
p1
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
dev.off()
```

```
## pdf
```

```
## 2
```

E então? Você acha que o fator que você selecionou explica bem a escolha das presas pelos lobos? Teste isso estatisticamente, usando um modelo linear generalizado com distribuição binomial. Trata-se de uma análise bem parecida com uma regressão logística, que discutimos em outras aulas.

```
fit1 = glm(preystate~moosedensity, family=binomial(link="logit"))
```

Salve o resumo dos resultados como um objeto e depois dê uma olhada. A quais resultados você deve prestar mais atenção?

```
resultados1 = summary(fit1)
resultados1
```

```
##
## Call:
## glm(formula = preystate ~ moosedensity, family = binomial(link = "logit"))
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.9687  -1.4856   0.7746   0.8603   0.9220
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.63556    0.17731   3.584 0.000338 ***
## moosedensity  0.18032    0.08218   2.194 0.028226 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 432.48  on 364  degrees of freedom
## Residual deviance: 427.05  on 363  degrees of freedom
## AIC: 431.05
##
## Number of Fisher Scoring iterations: 4
```

Quer exportar os resultados como um arquivo de texto? Use esta solução, por exemplo.

```
capture.output(resultados1, file = "resultados/res1.txt")
```

Teste 2

Outra forma de testar a significância seria através de uma análise de variância (ANOVA). Experimente essa alternativa.

```
resultados2 = anova(fit1, test="Chisq")
resultados2 #E ai? Mudou alguma coisa?
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: preystate
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL              364      432.48
## moosedensity  1    5.4235      363    427.05 0.01987 *
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Quer exportar os resultados como um arquivo de texto? Use esta solução, por exemplo.

```
capture.output(resultados2, file = "resultados/res2.txt")
```

Você pode também calcular a odds ratio, que te informa o quanto o Y muda a cada mudança de 1 unidade do X.

```
exp(coef(fit1))
```

```
## (Intercept) moosedensity
##      1.888083      1.197598
```

Você pode calcular também o intervalo de confiança da odds ratio.

```
exp(cbind(coef(fit1), confint(fit1)))
```

```
## Waiting for profiling to be done...
```

```
##
##                2.5 %   97.5 %
## (Intercept)  1.888083 1.334561 2.677279
## moosedensity 1.197598 1.027618 1.419975
```

Teste 3

Outra forma de se testar a significância seria através de comparação de modelos: um contendo o fator de interesse (fit1, criado alguns passos atrás) e outro sem ele (nulo). Veja como fazê-lo.

```
nulo = glm(preystate~1, family=binomial(link="logit"))
resultados3 = anova(nulo, fit1, test="Chisq")
resultados3 #E ai? Mudou alguma coisa?
```

```
## Analysis of Deviance Table
```

```
##
```

```
## Model 1: preystate ~ 1
```

```
## Model 2: preystate ~ moosedensity
```

```
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
```

```
## 1         364       432.48
```

```
## 2         363       427.05  1    5.4235  0.01987 *
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Quer exportar os resultados como um arquivo de texto? Use esta solução, por exemplo.

```
capture.output(resultados3, file = "resultados/res3.txt")
```

Não faltam formas de se testar a significância de um GLM. Uma terceira alternativa é por seleção de modelos, usando o AICc: critério de informação de Akaike corrigido. Escolhe-se o modelo que apresenta menor valor de AICc. Considera-se a diferença (delta) entre dois modelos significativa, apenas se $AICc1 - AICc2 > 2$.

```
resultados4 = AICctab(fit1,nulo)
resultados4
```

```
##      dAICc df
## fit1 0.0   2
## nulo 3.4   1
```

Quer exportar os resultados como um arquivo de texto? Use esta solução, por exemplo.

```
capture.output(resultados4, file = "resultados/res4.txt")
```

Teste 4

Caso você decida que mais de um fator importa neste caso, pode fazer um modelo multifatorial, como neste exemplo. Esses modelos podem incluir a interação entre os fatores, mas vamos deixar essa complicação adicional para outra hora.

```
fit2 = glm(pretype~moosedensity+timesincekill, family=binomial(link="logit"))
```

Confira o resultado.

```
resultados5 = summary(fit2)
resultados5
```

```
##
## Call:
## glm(formula = pretype ~ moosedensity + timesincekill, family = binomial(link = "logit"))
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0372  -1.4597   0.7733   0.8467   0.9506
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.56016    0.23143   2.420  0.0155 *
## moosedensity   0.14958    0.08295   1.803  0.0713 .
## timesincekill  0.04052    0.04632   0.875  0.3817
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 406.03  on 341  degrees of freedom
## Residual deviance: 401.86  on 339  degrees of freedom
## (23 observations deleted due to missingness)
## AIC: 407.86
##
## Number of Fisher Scoring iterations: 4
```

Quer exportar os resultados como um arquivo de texto? Use esta solução, por exemplo.

```
capture.output(resultados5, file = "resultados/res5.txt")
```

Para saber mais

Bolker, B. M., Brooks, M. E., Clark, C. J., Geange, S. W., Poulsen, J. R., Stevens, M. H. H., & White, J.-S. S. (2009). Generalized linear mixed models: a practical guide for ecology and evolution. *Trends in Ecology & Evolution*, 24(3), 127–135. <https://doi.org/10.1016/j.tree.2008.10.008>

Chamberlin, T. C. (1890). The method of multiple working hypotheses. *Science*, 15, 92–96.

Dobson, A. J., & Barnett, A. J. (2008). *An introduction to generalized linear models* (3rd ed.). CRC Press.

Ellison, A. M., Gotelli, N. J., Inouye, B. D., & Strong, D. R. (2014). P values, hypothesis testing, and model selection: it's déjà vu all over again. *Ecology*, 95(3), 609–610. <https://doi.org/10.1890/13-1911.1>

Platt, J. R. (1964). Strong Inference: Certain systematic methods of scientific thinking may produce much more rapid progress than others. *Science* (New York, N.Y.), 146(3642), 347–353. <https://doi.org/10.1126/science.146.3642.347>

Zuur, A. F., Ieno, E. N., Walker, N., Saveliev, A. A., & Smith, G. M. (2009). Mixed effects models and extensions in ecology with R (1st ed.). Springer New York. <https://doi.org/10.1007/978-0-387-87458-6>

Zuur, A. F., Ieno, E. N., & Elphick, C. S. (2010). A protocol for data exploration to avoid common statistical problems. *Methods in Ecology and Evolution*, 1(1), 3–14. <https://doi.org/10.1111/j.2041-210X.2009.00001.x>

Regressão logística na Wikipedia: https://en.wikipedia.org/wiki/Logistic_regression

GLM na Wikipedia: https://en.wikipedia.org/wiki/Generalized_linear_model