

Regressão Logística

Bioestatística em R

André M Ribeiro-dos-Santos

13 de Abr, 2017

Regressão Logística

- Modelar a relação entre variáveis a uma variável binomial
- Compreender os significado dos coeficientes.
- Ilustrar uma regressão logística e o efeito dos coeficientes.

Em um estudo caso/controle, pesquisadores desejavam associar a presença de câncer com um polimorfismo do gene CDH1, no entanto fatores como sexo e tabagismo influenciam o risco de desenvolvimento do câncer.

Qual o risco de desenvolver câncer em função da presença do alelo mutante?

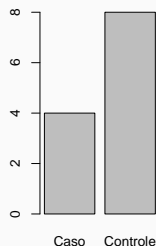
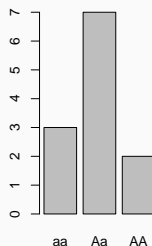
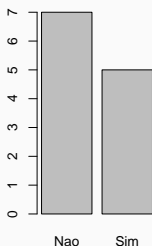
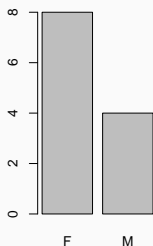
Table 1: Dados de estudo caso/controle para câncer

sexo	tabagismo	CDH1	Cancer	sexo	tabagismo	CDH1	Cancer
M	Nao	Aa	Controle	F	Nao	aa	Controle
F	Sim	AA	Controle	F	Nao	Aa	Caso
F	Nao	Aa	Caso	F	Nao	Aa	Caso
F	Sim	aa	Controle	M	Sim	Aa	Controle
M	Sim	Aa	Controle	M	Nao	Aa	Controle
F	Sim	aa	Controle	F	Nao	AA	Caso

```

> sex <- c("M", "F", "F", "F", "M", "F", "F", "F", "F", "M", "M", "F")
> tab <- c("Nao", "Sim", "Nao", "Sim", "Sim", "Sim",
+         "Nao", "Nao", "Nao", "Sim", "Nao", "Nao")
> cdh1 <- c("Aa", "AA", "Aa", "aa", "Aa", "aa",
+         "aa", "Aa", "Aa", "Aa", "Aa", "AA")
> cancer <- c("Controle", "Controle", "Caso", "Controle", "Controle",
+         "Controle", "Controle", "Caso", "Caso", "Controle",
+         "Controle", "Caso")
> par(mfrow = c(1,4))
> barplot(table(sex)); barplot(table(tab))
> barplot(table(cdh1)); barplot(table(cancer))

```



- Qual o tipo das variáveis sendo relacionadas?
- Qual o objetivo?

- Qual o tipo das variáveis sendo relacionadas? Queremos associar duas variáveis categóricas, sendo uma binomial
- Qual o objetivo? Queremos associar duas variáveis categóricas, medindo a influência de uma sobre a outra

Se quiséssemos somente associar o genótipo a presença ou não da doença usariamos um **qui-quadrado**.

```
> chisq.test(table(cancer, cdh1))
```

```
##
```

```
## Pearson's Chi-squared test
```

```
##
```

```
## data: table(cancer, cdh1)
```

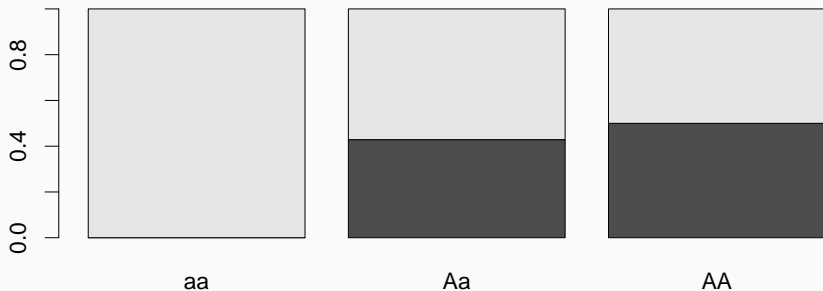
```
## X-squared = 2.0357, df = 2, p-value = 0.3614
```

No entanto, essa análise não indica a intensidade da relação, nem é capaz de considerar **confudidores** (Outros fatores que sabe-se estar relacionando com a resposta, mas não é o alvo principal).

Regressão Logística

Para esses casos, a análise mais indicada é uma **regressão logística**. Esta análise é uma especificação de um regressão linear que no lugar de buscar relacionar medidas a um variável resposta quantitativa, ela relaciona a uma variável resposta binária (TRUE ou FALSE).

```
> barplot(prop.table(table(cancer, cdh1), 2))
```



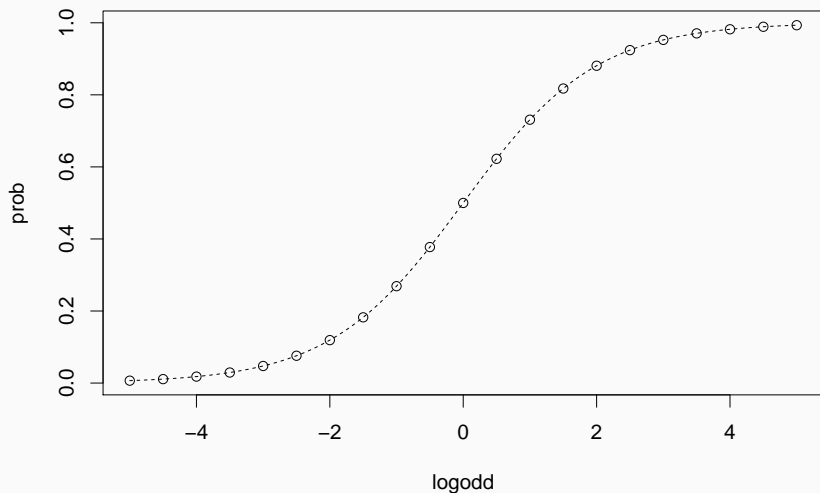
Isso é possível pela conversão das probabilidades na escala linear do **log-odds**.

$$\text{logit}(p_i) = \ln\left(\frac{p_i}{1 - p_i}\right) = \alpha + \beta * x + \epsilon$$

ou

$$p_i = \frac{1}{1 + e^{-(\alpha + \beta * x + \epsilon)}}$$

```
> logodd <- seq(-5, 5, 0.5)
> prob <- 1 / (1 + exp(-logodd)) ## ou plogis(logodd)
> plot(logodd, prob)
> curve(plogis, lty=2, add = TRUE)
```



```
> ?glm
> ## glm                                package:stats                R Documentation
> ## Fitting Generalized Linear Models
> ## Description:
> ##      'glm' is used to fit generalized linear models, specified by
> ##      giving a symbolic description of the linear predictor and a
> ##      description of the error distribution.
> ## Usage:
> ##      glm(formula, family = gaussian, data, weights, subset,
> ##          na.action, start = NULL, etastart, mustart, offset,
> ##          control = list(...), model = TRUE, method = "glm.fit",
> ##          x = FALSE, y = TRUE, contrasts = NULL, ...)
```

```

> has_cancer = (cancer == "Caso")
> (model <- glm(has_cancer~cdh1, family = "binomial"))

##
## Call:  glm(formula = has_cancer ~ cdh1, family = "binomial")
##
## Coefficients:
## (Intercept)      cdh1Aa      cdh1AA
##      -18.57       18.28       18.57
##
## Degrees of Freedom: 11 Total (i.e. Null);  9 Residual
## Null Deviance:      15.28
## Residual Deviance: 12.33      AIC: 18.33

```

```

> summary(model)

##
## Call:
## glm(formula = has_cancer ~ cdh1, family = "binomial")
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.17741  -1.05794  -0.00013   1.20850   1.30177
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -18.57     3765.85  -0.005    0.996
## cdh1Aa         18.28     3765.85   0.005    0.996
## cdh1AA         18.57     3765.85   0.005    0.996
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 15.276  on 11  degrees of freedom
## Residual deviance: 12.333  on  9  degrees of freedom
## AIC: 18.333
##
## Number of Fisher Scoring iterations: 17

```

```
> ## Log-odds effect
> (eff <- cbind(coef(model), confint(model)))
```

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

```
##                2.5 %   97.5 %
## (Intercept) -18.56607      NA 472.844
## cdh1Aa      18.27839 -374.0779      NA
## cdh1AA      18.56607 -757.4514      NA
```

```
> ## Odds-Ratio effect
> exp(eff)
```

```
##                2.5 %           97.5 %
## (Intercept) 8.646869e-09      NA 2.256981e+205
## cdh1Aa      8.673659e+07 3.467758e-163      NA
## cdh1AA      1.156488e+08 0.000000e+00      NA
```

```

> model <- glm(has_cancer~cdh1+sex+tab)
> summary(model)

##
## Call:
## glm(formula = has_cancer ~ cdh1 + sex + tab)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38462  -0.11538   0.00000   0.08654   0.38462
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.1538     0.1720   0.894  0.40080
## cdh1Aa        0.8462     0.2189   3.866  0.00616 **
## cdh1AA        0.4615     0.2156   2.141  0.06958 .
## sexM         -0.8846     0.1959  -4.515  0.00275 **
## tabSim        -0.2308     0.1592  -1.449  0.19057
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.05494505)
##
##      Null deviance: 2.66667  on 11  degrees of freedom

```



```
> or <- exp(cbind(coef(model), confint(model)))
```

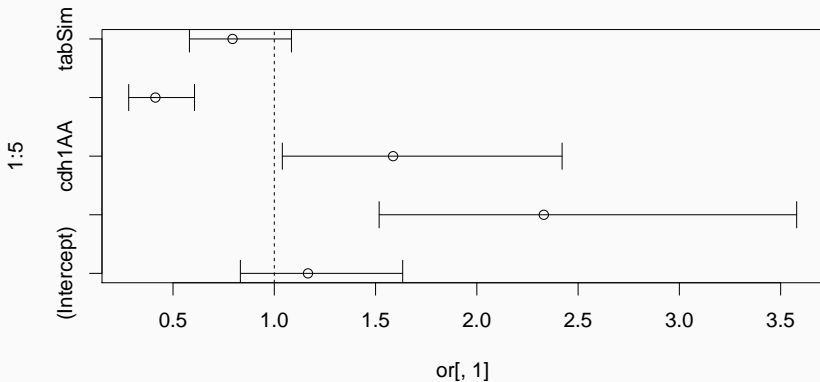
```
## Waiting for profiling to be done...
```

```
> plot(or[,1], 1:5, xlim = range(or), yaxt= "n")
```

```
> arrows(or[,2], 1:5, or[,3], 1:5, angle = 90, length = 0.15, code=3)
```

```
> abline(v = 1, lty = 2)
```

```
> axis(2, 1:5, rownames(or))
```



Ao final...

Até a próxima
