

## Dados de contagem com deslocamento no preditor linear em R

Os dados deste exemplo, apresentados abaixo, referem-se a contagens de partículas de vírus em um líquido para cinco diluições diferentes. O objetivo consiste em estimar o número médio de partículas de vírus por unidade de volume, verificando se este número varia com a diluição.

O parâmetro de interesse (número médio de partículas de vírus por unidade de volume) é denotado por  $\lambda$ , de modo que a média da variável resposta (contagem de partículas de vírus), denotada por  $\mu$ , é dada  $\mu = \lambda x$ , em que  $x$  representa a variável explicativa (diluição) e  $\lambda$  pode variar com a diluição (temos cinco diferentes diluições).

Adotando a função de ligação logarítmica, o preditor linear é dado por  $\eta = \log(\lambda) + \log(x) = \beta + \log(x)$ . O coeficiente do termo  $\log(x)$  no preditor linear é constante ( $= 1$ ), de modo que  $\log(x)$  representa um deslocamento (*offset*) no preditor linear.

Ajustamos dois modelos. No primeiro especificamos diferentes interceptos  $\beta_1, \dots, \beta_5$ . No segundo especificamos um único intercepto  $\beta$ .

```
## Exemplo 4.6, p. 98
## Demétrio, C. G. B. (2002), "Modelos Lineares Generalizados em
## Experimentação Agronômica", ESALQ

# Dados (Tabela 4, p. 21)
contagens <- c(13, 14, 17, 22, 9, 14, 6, 14, 4, 4, 3, 5, 3, 2, 1, 3,
              2, 1, 3, 2, 2)
dil <- rep(c(0.3162, 0.1778, 0.1, 0.0562, 0.0316), times = c(4, 4, 4, 4, 5))
dilf <- factor(rep(c("d1", "d2", "d3", "d4", "d5"),
                  times = c(4, 4, 4, 4, 5)))
```

O vetor `dilf` corresponde aos valores de diluição em forma de fator para possibilitar o ajuste com diferentes interceptos.

```
# Modelo com diferentes interceptos
m1 <- glm(contagens ~ dilf + offset(log(dil)), family = poisson)

# Modelo com intercepto único
m2 <- glm(contagens ~ offset(log(dil)), family = poisson)
```

O deslocamento também pode ser especificado como um argumento da função `glm`:

```
m1 <- glm(contagens ~ dilf, offset = log(dil), family = poisson)
```

Em seguida comparamos os dois modelos, que são encaixados.

```
# Tabela ANODEV (Tabela 40)
anova(m2, m1, test = "Chi")
```

```
Analysis of Deviance Table
```

```

Model 1: contagens ~ offset(log(dil))
Model 2: contagens ~ dilf + offset(log(dil))
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      20      13.549
2      16      10.381  4   3.1678  0.5302

```

Nota 1. A tabela ANODEV acima justifica (por quê?) adotarmos o modelo mais simples, com intercepto único.

Alguns resultados do ajuste do modelo com intercepto único são mostrados abaixo.

```
Call: glm(formula = contagens ~ offset(log(dil)), family = poisson)
```

```
Deviance Residuals:
```

```

      Min       1Q   Median       3Q      Max
-1.30940 -0.57859 -0.09256  0.26408  1.43976

```

```
Coefficients:
```

```

              Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.95502     0.08333   47.46 <2e-16 ***
---

```

```
Signif. codes:
```

```
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for poisson family taken to be 1)
```

```

Null deviance: 13.549 on 20 degrees of freedom
Residual deviance: 13.549 on 20 degrees of freedom
AIC: 87.194

```

Nota 2. Baseando-se apenas no valor da desviância, você afirmaria que o modelo acima ajusta-se bem aos dados?

Uma estimativa pontual do número médio de partículas de vírus por unidade de volume é dada por

```
cat("Número médio de partículas =", exp(coef(m2)), "\n")
```

```
Número médio de partículas = 52.19661
```

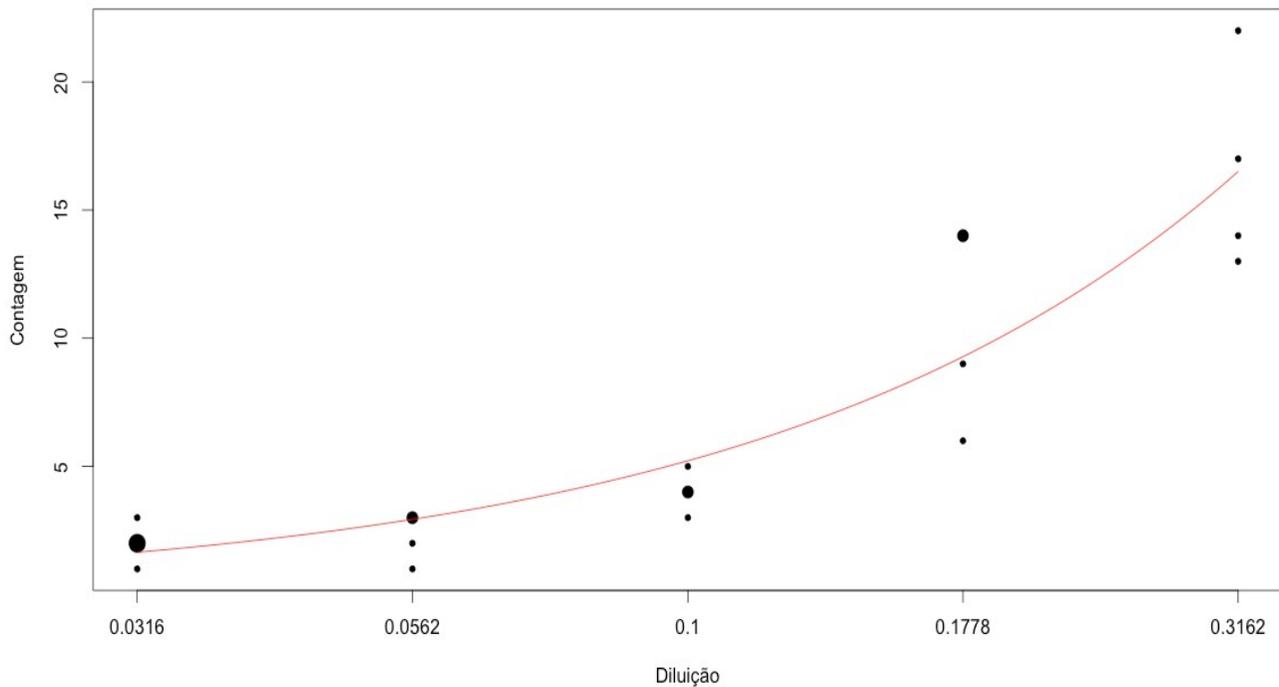
Nota 3. Apresente uma estimativa intervalar para o número médio de partículas de vírus por unidade de volume.

Os dados e o modelo ajustado são representados na figura a seguir, notando que no eixo horizontal a escala é logarítmica.

```

# Dados e modelo ajustado
freq <- as.data.frame(table(contagens, dil))
dilaux <- seq(min(dil), max(dil), length = 200)
plot(as.vector(freq$dil), as.vector(freq$contagens), xlab = "Diluição",
      ylab = "Contagem", pch = 20, cex = freq$Freq, axes = FALSE, log = "x")
axis(1, unique(dil), unique(dil))
axis(2)
lines(dilaux, predict(m2, type = "response", data.frame(dil = dilaux)),
      col = "red")
box()

```



Nota 4. Superponha ao gráfico acima a curva ajustada obtida do modelo com interceptos diferentes (m1).

Nota 5. Procure reproduzir todos os resultados acima utilizando outros pacotes estatísticos.