Modelos lineales generalizados para conteos

Guillermo Ayala Gallego

Modelos lineales generalizados para conteos

Guillermo Ayala Gallego

2024-03-25

Modelos loglineales de Poisson

Datos

```
pacman::p_load(SummarizedExperiment)
data(PRJNA218851,package="tamidata2")
```

```
table(colData(PRJNA218851)[,"Stage"])
```

```
    Cancer Metastasis     Normal
        18         18         18
```

```
df = data.frame(count = assay(PRJNA218851)[1000,],
                Stage=colData(PRJNA218851)[,"Stage"])
head(df)
```

```
                               count  Stage
SRR975551Aligned.out.sam.bam     539 Cancer
SRR975552Aligned.out.sam.bam     563 Cancer
SRR975553Aligned.out.sam.bam    1018 Cancer
SRR975554Aligned.out.sam.bam     393 Cancer
SRR975555Aligned.out.sam.bam     398 Cancer
SRR975556Aligned.out.sam.bam     672 Cancer
```

Modelo loglineal de Poisson

Ajustamos un modelo loglineal de Poisson.

```
fit = glm(count ~ Stage, family = poisson(link = log),
          data = df)
summary(fit)
```

```
Call:
glm(formula = count ~ Stage, family = poisson(link = log), data = df)

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      6.729957   0.008146  826.14   <2e-16 ***
StageMetastasis -0.306800   0.012512  -24.52   <2e-16 ***
StageNormal      0.429249   0.010467   41.01   <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: 13236.9  on 53  degrees of freedom
Residual deviance:  8723.7  on 51  degrees of freedom
AIC: 9189.2

Number of Fisher Scoring iterations: 4
```

- La desviacion nula es la desviacion para el modelo que tiene solo la constante.
- La desviacion residual es la desviacion del modelo que tiene la constante y las variables binarias que describen `Stage`.
- La diferencia entre los valores tiene una distribucion ji-cuadrado con dos grados de libertad y nos permite contrastar si los coeficientes de `StageMetastasis` y `StageNormal` pueden considerarse simultáneamente nulos.

```
1 fit$null.deviance - fit$deviance
```

```
[1] 4513.206
```

Podemos rechazar confortablemente la hipotesis nula.

```
1 attributes(summary(fit))
```

```
$names
 [1] "call"           "terms"          "family"         "deviance"
 [5] "aic"            "contrasts"      "df.residual"    "null.deviance"
 [9] "df.null"        "iter"           "deviance.resid" "coefficients"
[13] "aliased"        "dispersion"     "df"             "cov.unscaled"
[17] "cov.scaled"

$class
[1] "summary.glm"
```

```
1 summary(fit)$coefficients
```

```
                  Estimate  Std. Error   z value      Pr(>|z|)
(Intercept)      6.7299568 0.008146275 826.13916  0.000000e+00
StageMetastasis -0.3068000 0.012512160 -24.52015 9.006867e-133
StageNormal      0.4292487 0.010467369  41.00827  0.000000e+00
```

```
attributes(fit)
```

```
$names
 [1] "coefficients"      "residuals"         "fitted.values"
 [4] "effects"           "R"                 "rank"
 [7] "qr"                "family"            "linear.predictors"
[10] "deviance"          "aic"               "null.deviance"
[13] "iter"              "weights"           "prior.weights"
[16] "df.residual"       "df.null"           "y"
[19] "converged"         "boundary"          "model"
[22] "call"              "formula"           "terms"
[25] "data"              "offset"            "control"
[28] "method"            "contrasts"         "xlevels"

$class
[1] "glm" "lm"
```

```
head(fit$fitted.values)
```

```
SRR975551Aligned.out.sam.bam SRR975552Aligned.out.sam.bam
                    837.1111                     837.1111
SRR975553Aligned.out.sam.bam SRR975554Aligned.out.sam.bam
                    837.1111                     837.1111
SRR975555Aligned.out.sam.bam SRR975556Aligned.out.sam.bam
                    837.1111                     837.1111
```

Podemos predecir la media de la respuesta para el valor de `Stage` que queramos con `predict`.

```
predict(fit,type = "response",
        newdata = data.frame(Stage =c("Cancer","Metastasis","Normal")))
```

```
        1         2         3
 837.1111  615.9444 1285.8889
```

Sobredispersión

- En una distribución de Poisson, la media y la varianza son iguales.
- Cuando trabajamos con conteos reales no suele ser cierta esta hipótesis.
- Con frecuencia la varianza es mayor que la media.
- A esto se le llama **sobredispersión**.

```
fit1 = glm(count ~ Stage, family = quasipoisson(link = log),
                    data = df)
summary(fit1)$dispersion
```

```
[1] 198.0956
```

GLM binomiales negativos

```r
library(MASS)
fit2 = glm(count ~ Stage,
           family = negative.binomial(theta = 1,link = "log"),
           data = df, start = coef(fit))
summary(fit2)
```

```
Call:
glm(formula = count ~ Stage, family = negative.binomial(theta = 1,
    link = "log"), data = df, start = coef(fit))

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)       6.7300     0.1195  56.334   <2e-16 ***
StageMetastasis  -0.3068     0.1690  -1.816   0.0753 .
StageNormal       0.4292     0.1689   2.541   0.0141 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(1) family taken to be 0.2565866)

    Null deviance: 15.757  on 53  degrees of freedom
Residual deviance: 10.801  on 51  degrees of freedom
AIC: 845.31

Number of Fisher Scoring iterations: 1
```