RegresionPoisson

Cleber Perez

2024-10-29

Regresión Poisson

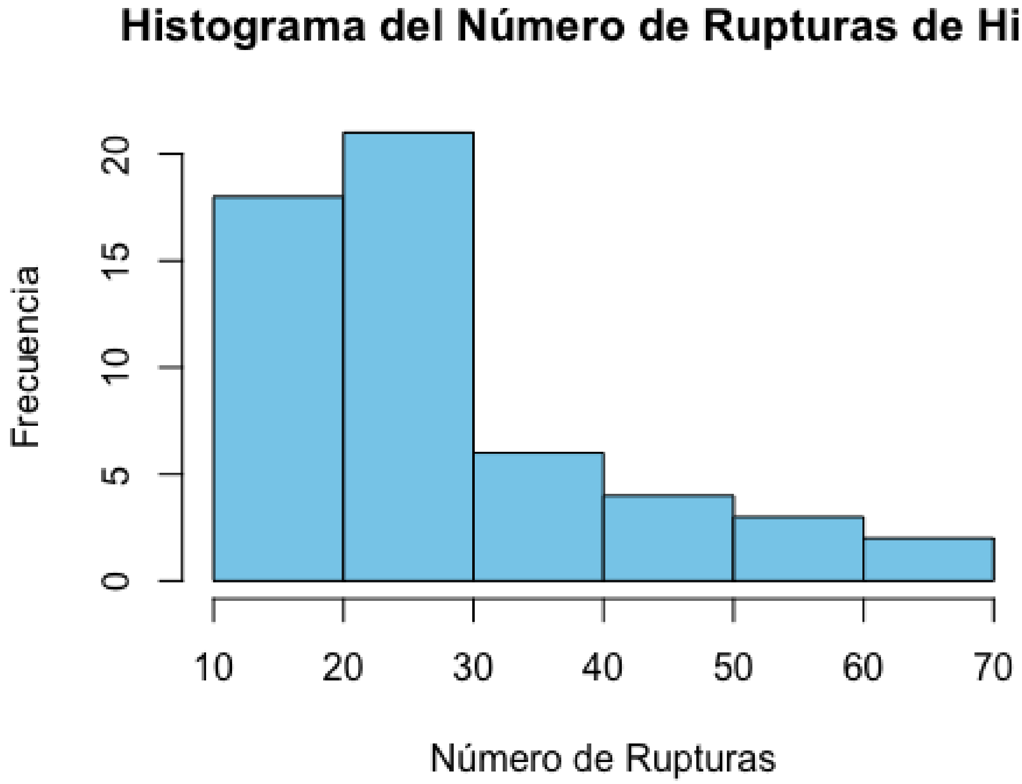
```
data<-warpbreaks
head(data,10)
```

```
##    breaks wool tension
## 1      26    A       L
## 2      30    A       L
## 3      54    A       L
## 4      25    A       L
## 5      70    A       L
## 6      52    A       L
## 7      51    A       L
## 8      26    A       L
## 9      67    A       L
## 10     18    A       M
```

I. Análisis Descriptivo

Histograma del número de rupturas

```
hist(warpbreaks$breaks, main = "Histograma del Número de Rupturas de Hilos",
     xlab = "Número de Rupturas", ylab = "Frecuencia", col = "skyblue",
border = "black")
```



Obtén la media y la varianza de la variable dependiente

```
media_breaks <- mean(warpbreaks$breaks)
varianza_breaks <- var(warpbreaks$breaks)
cat("Media:", media_breaks, "\n")

## Media: 28.14815

cat("Varianza:", varianza_breaks, "\n")

## Varianza: 174.2041
```

Debido a estos numeros de media y varianza puede la que la regresion Poisson no se ajuste bien a lo que necesitamos, por que se necesita que esten muy parecidos o iguales, y se observa que la varianza es muy grande.

II. Ajusta dos modelos de Regresión Poisson

Ajusta el modelo de regresión Poisson sin interacción

```
poisson_model <- glm(breaks ~ wool + tension, data = warpbreaks, family =
poisson(link = "log"))
S <- summary(poisson_model)

print(S)
```

```
##
## Call:
## glm(formula = breaks ~ wool + tension, family = poisson(link = "log"),
##     data = warpbreaks)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.69196    0.04541  81.302  < 2e-16 ***
## woolB       -0.20599    0.05157  -3.994 6.49e-05 ***
## tensionM    -0.32132    0.06027  -5.332 9.73e-08 ***
## tensionH    -0.51849    0.06396  -8.107 5.21e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 297.37  on 53  degrees of freedom
## Residual deviance: 210.39  on 50  degrees of freedom
## AIC: 493.06
##
## Number of Fisher Scoring iterations: 4
```

Ajusta el modelo de regresión Poisson con interacción

```
poisson_model_interaction <- glm(breaks ~ wool * tension, data = warpbreaks,
family = poisson(link = "log"))
C <- summary(poisson_model_interaction)

print(C)
```

```
##
## Call:
## glm(formula = breaks ~ wool * tension, family = poisson(link = "log"),
##     data = warpbreaks)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     3.79674    0.04994  76.030  < 2e-16 ***
## woolB          -0.45663    0.08019  -5.694 1.24e-08 ***
## tensionM       -0.61868    0.08440  -7.330 2.30e-13 ***
## tensionH       -0.59580    0.08378  -7.112 1.15e-12 ***
## woolB:tensionM  0.63818    0.12215   5.224 1.75e-07 ***
## woolB:tensionH  0.18836    0.12990   1.450    0.147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 297.37  on 53  degrees of freedom
## Residual deviance: 182.31  on 48  degrees of freedom
## AIC: 468.97
```

```
##
## Number of Fisher Scoring iterations: 4
```

Podemos observar que el modelo con interaccion es el mas optimo a desarrollar debido a su bajo valor de AIC

III. Selección del modelo

Desviacion Residual de Modelo Con Interaccion

```
glc = C$df.null-C$df.residual
qchisq(0.05,glc)

## [1] 1.145476

drc = C$deviance
cat("Estadístico de prueba =",drc, "\n")

## Estadístico de prueba = 182.3051

vpc = 1-pchisq(drc,glc)
cat("Valor p =",vpc)

## Valor p = 0
```

Desviacion Residual de Modelo Sin Interaccion

```
gls = S$df.null-S$df.residual
qchisq(0.05,gls)

## [1] 0.3518463

drs = S$deviance
cat("Estadístico de prueba =",drs, "\n")

## Estadístico de prueba = 210.3919

vps = 1-pchisq(drs,gls)
cat("Valor p =",vps)

## Valor p = 0
```

Con estas desviaciones podemos ver el estadistico de prueba de cada modelo, aqui el estadistico de prueba es menor en el modelo de con interaccion que significa que tiene mejor ajuste.

Grafica

```
S1 <- glm(breaks ~ wool + tension, data = warpbreaks, family = poisson(link =
"log"))
C1 <- glm(breaks ~ wool * tension, data = warpbreaks, family = poisson(link =
"log"))

coef_s <- summary(S1)$coefficients
coef_c <- summary(C1)$coefficients
```

```
coeficientes_comparacion <- data.frame(
  Termino = union(rownames(coef_s), rownames(coef_c)),
  Coef_Sin_Interaccion = coef_s[match(union(rownames(coef_s),
rownames(coef_c)), rownames(coef_s)), "Estimate"],
  Coef_Con_Interaccion = coef_c[match(union(rownames(coef_s),
rownames(coef_c)), rownames(coef_c)), "Estimate"]
)

errores_estandar_comparacion <- data.frame(
  Termino = union(rownames(coef_s), rownames(coef_c)),
  Error_Sin_Interaccion = coef_s[match(union(rownames(coef_s),
rownames(coef_c)), rownames(coef_s)), "Std. Error"],
  Error_Con_Interaccion = coef_c[match(union(rownames(coef_s),
rownames(coef_c)), rownames(coef_c)), "Std. Error"]
)

print(coeficientes_comparacion)
```

```
##                       Termino Coef_Sin_Interaccion Coef_Con_Interaccion
## (Intercept)       (Intercept)            3.6919631            3.7967368
## woolB                   woolB           -0.2059884           -0.4566272
## tensionM             tensionM           -0.3213204           -0.6186830
## tensionH             tensionH           -0.5184885           -0.5957987
## woolB:tensionM woolB:tensionM                   NA            0.6381768
## woolB:tensionH woolB:tensionH                   NA            0.1883632
```

```
print(errores_estandar_comparacion)
```

```
##                       Termino Error_Sin_Interaccion Error_Con_Interaccion
## (Intercept)       (Intercept)            0.04541069            0.04993753
## woolB                   woolB            0.05157117            0.08019202
## tensionM             tensionM            0.06026580            0.08440012
## tensionH             tensionH            0.06395944            0.08377723
## woolB:tensionM woolB:tensionM                    NA            0.12215312
## woolB:tensionH woolB:tensionH                    NA            0.12989529
```

IV. Evaluación de los supuestos

```
library(epiDisplay)
```

```
## Loading required package: foreign

## Loading required package: survival

## Loading required package: MASS

## Loading required package: nnet
```

```
poisgof(C)
```

```
## $results
## [1] "Goodness-of-fit test for Poisson assumption"
##
## $chisq
## [1] 182.3051
##
## $df
## [1] 48
##
## $p.value
## [1] 1.582538e-17
```

```
library(epiDisplay)
poisgof(S)
```

```
## $results
## [1] "Goodness-of-fit test for Poisson assumption"
##
## $chisq
## [1] 210.3919
##
## $df
## [1] 50
##
## $p.value
## [1] 1.44606e-21
```

```
poisson.model3<-glm(breaks ~ wool * tension, data = data, family =
quasipoisson(link = "log"))
summary(poisson.model3)
```

```
##
## Call:
## glm(formula = breaks ~ wool * tension, family = quasipoisson(link =
"log"),
##     data = data)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.79674    0.09688  39.189  < 2e-16 ***
## woolB           -0.45663    0.15558  -2.935 0.005105 **
## tensionM        -0.61868    0.16374  -3.778 0.000436 ***
## tensionH        -0.59580    0.16253  -3.666 0.000616 ***
## woolB:tensionM  0.63818    0.23699   2.693 0.009727 **
## woolB:tensionH  0.18836    0.25201   0.747 0.458436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 3.76389)
##
##     Null deviance: 297.37  on 53  degrees of freedom
## Residual deviance: 182.31  on 48  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 4
```

El parámetro de dispersión para el modelo quasi-Poisson es 3.76389, que confirma la presencia de sobredispersión, ya que este valor es mayor a 1. Esto justifica el uso del modelo quasi-Poisson en lugar del modelo Poisson estándar.