En el experimento proporcionado por Phelps (1982) se anotó para cada uno de los $i = 24$ grupos, el número de zanahorias dañadas por insectos de entre todas las del grupo. Éstas fueron plantadas en tres bloques, por lo que al ser ésta una covariable de tipo cualitativo, debieron considerarse en el modelo dos covariables indicadoras, `bloque1` y `bloque2`. Además, se fumigó según ocho dosis de un determinado insecticida, considerándose la covariable cuantitativa `log(dosis)` en el modelo.

Se pretende ajustar a estos datos un Modelo de Regresión Binomial clásico y otro robusto.

Los datos del experimento de Phelps (1982) vienen recogidos en el fichero de datos `zanaho`, suministrado entre el Material Didáctico del curso.

El objetivo que se persigue es ajustar un Modelo Lineal Generalizado (en esta sección, clásico) para datos binomiales $B(n_i, p_i)$ (con lo que es $\mu_i = n_i\, p_i$), de la forma

$$\log\left(\frac{\mu_i}{n_i - \mu_i}\right) = \beta_0 + \beta_1\ \log(\texttt{dosis}) + \beta_2\, \text{bloque2} + \beta_3\, \text{bloque1}$$

Como los datos a utilizar deben de estar en forma de *estructura de datos*, ejecutamos `(1)` para incluirlos en R^{mo} con ese formato al utilizar la función `read.table`. A continuación lo comprobamos.

```
> zanahorias<-read.table("a:\\zanaho",header=T)                    (1)

> zanahorias
   dañadas total logdosis bloque bloque1 bloque2
1       10    35     1.52      1       1       0
2       16    42     1.64      1       1       0
.............................................
23       3    22     2.24      3       0       0
24       2    31     2.36      3       0       0
```

Al trabajar con datos binomiales, como dijimos más arriba, la variable de respuesta debe estar formada por una matriz en la que la primera columna sea los *éxitos* y la segunda columna los *fracasos* (=al número de pruebas-éxitos). Los datos de esta variable respuesta (que hemos denominado `respuesta`) la obtenemos en `(2)` utilizando la función de R^{mo} `cbind` que *pega* columnas. A continuación comprobamos que lo ha hecho bien.

```
> respuesta<-cbind(zanahorias[,1],zanahorias[,2]-zanahorias[,1])   (2)
```

```
> respuesta
      [,1] [,2]
 [1,]   10   25
 [2,]   16   26
.................
[23,]    3   19
[24,]    2   29
```

Ahora ya podemos utilizar la función `glm` en `(3)`, apareciendo los resultados en `(4)`, los cuales valoramos ejecutando `(5)`.

```
> resultado<-glm(respuesta~logdosis+bloque2+bloque1,                    (3)
+ family=binomial,data=zanahorias)

> resultado                                                             (4)

Call:  glm(formula = respuesta ~ logdosis + bloque2 + bloque1,
       family = binomial, data = zanahorias)

Coefficients:
(Intercept)     logdosis      bloque2      bloque1
     1.4802      -1.8174       0.8433       0.5424

Degrees of Freedom: 23 Total (i.e. Null);  20 Residual
Null Deviance:     83.34
Residual Deviance: 39.98        AIC: 128.6

> summary(resultado)                                                    (5)

Call:
glm(formula = respuesta ~  logdosis + bloque2 + bloque1,
    family = binomial, data = zanahorias)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9200  -1.0215  -0.3239   1.0602   3.4324

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.4802     0.6554   2.258 0.023918 *
logdosis     -1.8174     0.3434  -5.293 1.20e-07 ***                    (8)
bloque2       0.8433     0.2257   3.736 0.000187 ***
bloque1       0.5424     0.2315   2.343 0.019118 *
```

(6) (7)

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 83.344  on 23  degrees of freedom
Residual deviance: 39.976  on 20  degrees of freedom
                    (9)
AIC: 128.61

Number of Fisher Scoring iterations: 3
```

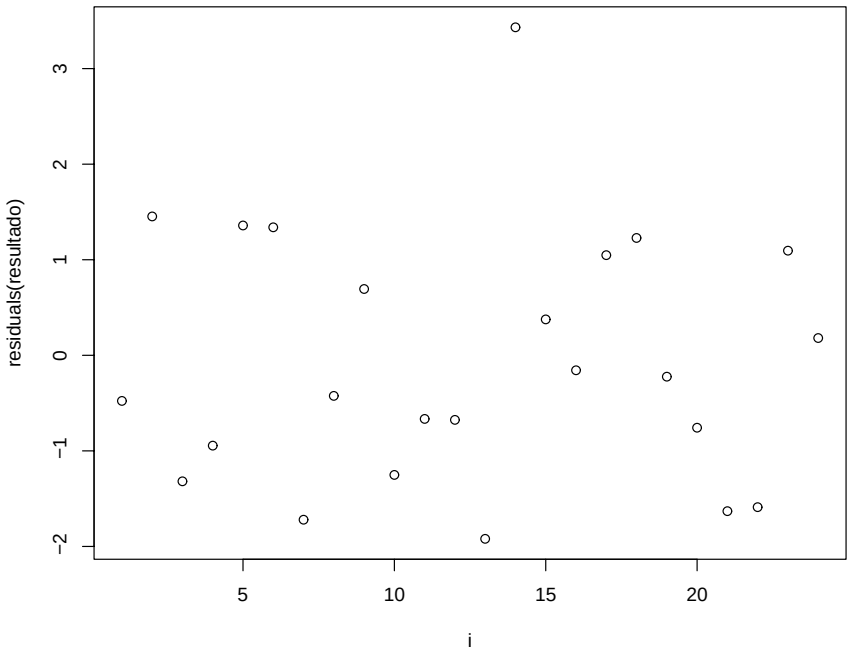

Figura 1: : Gráfico de los Residuos

Los estimadores de los coeficientes aparecen en `(6)`, sus errores estándar en `(7)` (iguales a los que aparecen en la columna izquierda de la Tabla 1 del artículo de Cantoni y Ronchetti, 2001) y los p-valores de los contrastes de la hipótesis nula de ser éstos cero, indican en `(8)` que son significativas las tres covariables independientes consideradas, quedando como modelo ajustado el siguiente,

$$\log\left(\frac{\mu_i}{n_i-\mu_i}\right) = 1'4802 - 1'8174\ \log(\texttt{dosis}) + 0'8433\,\text{bloque2} + 0'5424\,\text{bloque1}$$

El valor del estadístico *deviance* que aparece en `(9)`, igual a $G^2 = 39'976$, se utiliza en el contraste de la hipótesis nula de adecuarse correctamente el modelo anterior a los datos observados y que corresponde a una $\chi^2_{n-(k+1)} = \chi^2_{24-4} = \chi^2_{20}$. El p-valor de este test será, por tanto,

```
> 1-pchisq(39.976,20)
[1] 0.005030426
```

indicando, de forma sorprendente, que debe rechazarse la bondad del ajuste del modelo obtenido cuando los contrastes individuales para los parámetros β_i indicaban que las covariables sí explicaban a la variable respuesta.

Si representamos los residuos del modelo ajustado en la Figura 1.1 mediante la siguiente secuencia,

```
> i<-seq(1,24)
> plot(i,residuals(resultado))
```

observamos que la observación número 14 es un outlier. Es más conveniente, por tanto, utilizar Métodos Robustos.

Primero fijamos el valor de la constante de Huber en `(1)`, ejecutando a continuación la función que nos proporciona las estimaciones robustas. En `(2)` obtenemos éstas y en `(3)` sus errores estimados, iguales a los obtenidos en la columna derecha de la Tabla 1 del trabajo de Cantoni y Ronchetti (2001), con una pequeña diferencia ya que nosotros trabajamos con R^{mo} y ellos con S-Plus.

```
> chuber<-1.2                                                          (1)
> salida.robusta<-glm.rob(as.matrix(zanahorias[,c(3,6,5)]),
+ as.matrix(zanahorias[,1]), choice="binom",ni=as.matrix(zanahorias[,2]))

> salida.robusta$coeff                                                 (2)
[1]  1.9301522 -2.0497142  0.6897909  0.4613198

> salida.robusta$sd.coeff                                              (3)
[1] 0.6984066 0.3689728 0.2366980 0.2413989
```

Si ahora queremos validar el modelo con el que nos quedaremos, podemos hacer contrastes anidados como los que se indicaban más arriba, consistentes en establecer como hipótesis alternativa un modelo con un número determinado de covariables y como hipótesis nula un submodelo de éste. Si rechazamos la hipótesis nula, con un p-valor bajo, podemos concluir que la covariable no incluida en el modelo de

la hipótesis nula (en el submodelo) es relevante a la hora de explicar a la variable dependiente. Todo esto lo haremos con la función anterior `quasi.rob`

Primero platearemos la hipótesis alternativa de un modelo con las tres covariables consideradas, `logdosis`, `bloque1` y `bloque2` frente a la hipótesis nula del submodelo sin la covariable `bloque2`. Para ello ejecutamos la secuencia siguiente en donde destacamos como en la línea marcada con `(4)` incluimos, como primer argumento de la función, un modelo las tres covariables que aparecen en las columnas 3, 5 y 6 de la matriz de datos, y como en la línea `(5)` le decimos, con el argumento `out.col=3`, que como hipótesis nula considere el submodelo sin la que aparece en la columna 3 de las anteriores, es decir, en la columna 6 de la matriz de datos, es decir, sin `bloque2`.

El p-valor de este test lo obtenemos ejecutando `(6)` que claramente indica que rechazamos la hipótesis nula del submodelo, lo que indica cierta significación (i.e., algo explica) la covariable `bloque2`.

```
> resultado<-quasi.rob(as.matrix(zanahorias[,c(3,5,6)]),                    (4)
+ as.matrix(zanahorias[,1]),out.col=3,choice="binom",                       (5)
+ ni=as.matrix(zanahorias[,2]))

> resultado$pvalue                                                          (6)

             [,1]
 [1,] 0.003565751
```

Podemos considerar el siguiente árbol de posibles modelos en una primera tanda de comparaciones

H_0 : `logdosis, bloque1`
H_1 : `logdosis, bloque1, bloque2`

H_0 : `logdosis, bloque2`
H_1 : `logdosis, bloque1, bloque2`

H_0 : `bloque1, bloque2`
H_1 : `logdosis, bloque1, bloque2`

En el primer test obtuvimos el p-valor $0'0036$. Los otros dos p-valores los obtenemos ejecutando

```
> quasi.rob(as.matrix(zanahorias[,c(3,5,6)]),as.matrix(zanahorias[,1]),
+ out.col=2,choice="binom",ni=as.matrix(zanahorias[,2]))$pvalue

             [,1]
  [1,] 0.05600116
```

y

```
> quasi.rob(as.matrix(zanahorias[,c(3,5,6)]),as.matrix(zanahorias[,1]),
+ out.col=1,choice="binom",ni=as.matrix(zanahorias[,2]))$pvalue

              [,1]
  [1,] 2.773081e-08
```

p-valores que llevan a la conclusión de ser muy significativa (muy explicativa) la covariable `logdosis`, algo significativa (como dijimos más arriba) `bloque2` y poco relevante `bloque1`.

Como el único posible modelo sería el que contiene a las covariables `logdosis` y `bloque2` surgen ahora dos posibles tests,

H_0 : `logdosis`
H_1 : `logdosis, bloque2`

H_0 : `bloque2`
H_1 : `logdosis, bloque2`

cuyos p-valores obtenemos ejecutando, respectivamente, las secuencias,

```
> quasi.rob(as.matrix(zanahorias[,c(3,6)]),as.matrix(zanahorias[,1]),
+ out.col=2,choice="binom",ni=as.matrix(zanahorias[,2]))$pvalue

             [,1]
  [1,] 0.01178241
```

y

```
> quasi.rob(as.matrix(zanahorias[,c(3,6)]),as.matrix(zanahorias[,1]),
+ out.col=1,choice="binom",ni=as.matrix(zanahorias[,2]))$pvalue

              [,1]
  [1,] 3.961684e-08
```

los cuales indican, de nuevo, la significación de `bloque2` y, de nuevo, lo significativo que resulta la covariable `logdosis`.

Parece, por tanto, razonable utilizar estas dos covariables, para cuya estimación de parámetros ejecutamos la siguiente secuencia

```
> glm.rob(as.matrix(zanahorias[,c(3,6)]),as.matrix(zanahorias[,1]),
+ choice="binom",ni=as.matrix(zanahorias[,2]))$coeff

[1]  2.1187526 -2.0355601  0.4759153
```

que lleva a quedarnos, finalmente, con el modelo

$$\log\left(\frac{\mu_i}{n_i-\mu_i}\right) = 2'119 - 2'036\ \log(\texttt{dosis}) + 0'476\,\text{bloque2}$$

Observemos que si en `(1)` hacemos la constante de Huber igual a infinito, obtendremos, en lugar de `(2)`, los resultados clásicos obtenidos cuando hicimos este ejemplo con Métodos Clásicos. Veámoslo,

```
> chuber<-Inf

> a<-glm.rob(as.matrix(zanahorias[,c(3,6,5)]),as.matrix(zanahorias[,1]),
+ choice="binom",ni=as.matrix(zanahorias[,2]))
There were 26 warnings (use warnings() to see them)

> a$coeff
[1]  1.4540106 -1.8078152  0.8497862  0.5524021
```