USC UNIVERSIDADE DE SANTIAGO DE COMPOSTELA
FACULTADE DE MATEMÁTICAS

Traballo Fin de Grao

Regresión Xeralizada Aplicada

Alba Camino Enríquez

Xullo, 2022

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

GRAO DE MATEMÁTICAS

Traballo Fin de Grao

Regresión Xeralizada Aplicada

Alba Camino Enríquez

Xullo, 2022

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

Traballo proposto

Área de Coñecemento: Estatística e Investigación Operativa
Título: Regresión Xeralizada Aplicada
Breve descrición do contido
Os modelos de regresión serven para explicar e modelar a relación que existe entre unha variable resposta e unha ou máis variables explicativas. Tomando como base o modelo de regresión lineal simple clásico, o obxectivo deste traballo será o de presentar distintas extensións que permitan xeralizar dito modelo e analizar o seu funcionamento. Este traballo de fin de grado terá un forte carácter aplicado, polo tanto, os modelos revisados serán seleccionados e empregados en función da natureza dos datos que se dispoñan.
Recomendacións
Faraway, J.J. (2006). Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models. Chapman and Hall. Sheather, S.J. (2009). A modern approach to regression with R. Springer.
Outras observacións
O tratamento dos datos realizarase empregando o software estadístico de uso libre R (https://www.r-project.org/)

Índice

Resumo

Os modelos de regresión serven para explicar e modelar a relación que existe entre unha variable resposta e unha ou máis variables explicativas. Tomando como base o modelo de regresión lineal simple clásico presentaremos distintas extensións que permitan xeralizar dito modelo. En concreto, expoñeremos dous modelos de regresión sobre unha variable resposta discreta: o modelo de regresión loxística e o modelo de Poisson. É dicir, estes modelos en lugar de presentar unha distribución continua como era a normal para o caso lineal, presentan distribucións discretas como é a de Bernoulli e distribución de Poisson respectivamente. Ademais, cada un destes modelos presenta características e aplicacións particulares que se expoñen ao longo do traballo. O modelo de regresión loxística aplícase cando a nosa variable resposta categórica é dicotómica. Mentres que, o modelo de Poisson é común empregalo para datos de conteo. Ámbolos dous modelos, serán empregados sobre diferentes bases de datos relacionadas co ámbito da saúde e poderemos tratar as mesmas cuestións que para os modelos lineais. Incluso, algunhas destas cuestións se analizarán dun xeito moi similar. Sen embargo, debido a presenza da variable resposta discreta que os caracteriza, acharemos aspectos onde surxirán máis dificultades.

Abstract

Regression models serve to explain and shape the relationship between a response variable and one or more explanatory variables. This essay shows the different extensions that allow us to generalize this model taking the simple classic linear regression model as a reference. Specifically, interpreting two regression models on a discrete response variable: logistic regression model and Poisson's model. Meaning these models instead of presenting a continuous distribution as the normal distribution in the linear case, they present discrete distributions such as the Bernoulli distribution and the Poisson distribution respectively. In addition, each of these models have particular characteristics and applications that will be examined. The logistic regression model

is applied when our categorical response variable is dichotomous. Whereas, Poisson's model is commonly used for count data. Both models will be implemented using different health-related databases and it is possible to address the same questions that for linear models. Even some of those questions can be studied in a very similar way. However, owing to the presence of the discrete response of variable that portrayed them, it is possible to attribute some aspects where more difficulties will emerge.

Introdución

Os modelos de regresión empréganse para explicar a relación que existe entre unha variable resposta e unha ou máis variables explicativas. Tomaremos como punto de partida o modelo de regresión lineal clásico. Este modelo aplícase cando desexamos coñecer unha posible relación lineal entre a resposta e as variables explicativas. O modelo de regresión lineal caracterízase por ter unha variable resposta continua cuxa distribución de probabilidade é normal. Desta forma a media de Y pode ser expresada como función lineal e vén dada pola seguinte expresión:

$$\eta = \beta_0 + \beta_1 X_1 + ... + \beta_q X_q,$$

onde $X_1, ..., X_q$ son q variables predictoras que se corresponden coas variables explicativas presentes no modelo. Recordemos que para que un modelo de regresión lineal sexa correcto debe verificar as hipóteses de linealidade, homocedasticidade, normalidade e independencia.

Pondo atención na estimación dos parámetros, no caso da regresión lineal, recorremos ao método de mínimos cadrados, xa que a idea consiste en escoller aqueles estimadores que dean lugar a erros máis pequenos. Cómpre resaltar que debido a suposición de normalidade, este estimador de mínimos cadrados coincide co estimador de máxima verosimilitude. Ademais, a expresión destos estimadores pódese calcular de forma explícita. Con estas expresións podemos realizar inferencia tanto para o intercepto, como para o valor da pendente, calcular os intervalos de confianza para ambos e realizar contrastes.

Sen embargo, cando a variable resposta non é continua, se non que é discreta, a regresión lineal non é de utilidade para o seu modelaxe. Polo tanto, cómpre recorrer a outros modelos alternativos que permitan explicar a nosa resposta discreta en función dunha serie de variables explicativas, estes modelos serán o modelo de regresión loxística e o modelo de Poisson. A principal diferenza destes modelos con respecto do lineal é a presenza dunha variable resposta discreta. Ademais, esta variable resposta discreta, para o caso do modelo de regresión loxística e o modelo de Poisson, non segue unha distribución normal, se non que segue unha distribución de Bernoulli e de Poisson respectivamente. Estas e máis diferenzas entre a regresión lineal e os modelos alternativos indicados, fan máis complexa a estimación dos parámetros, a inferencia e a selección do modelo para o caso destes últimos.

Nos seguintes capítulos céntraremonos tanto no modelo de regresión loxística como no modelo de Poisson. Estudaremos e analizaremos as súas diferenzas e similitudes co modelo de regresión lineal e aplicaremos os conceptos expostos a datos concretos con obxetivos prácticos.

Nos Capítulos 1 e 3 explicaremos e plantexaremos un modelo de regresión loxística. Primeiro comezaremos con unha descrición teórica do modelo indicado, para posteriormente poder aplicalo a un conxunto de datos concretos empregando o *software* de R (R Core Team, 2021). En particular, o uso deste modelo é habitual en marcos epidemiolóxicos e sanitarios, cando por exemplo se traballa con variables respostas que representan a ausencia ou presenza de certa enfermidade. Unha vez teñamos o axuste para o modelo, analizaremos cómo de correcto é o axuste para os datos e que conclusións podemos obter do mesmo.

Por outra banda, seguindo unha estrutura similar, nos Capítulos 2 e 4, expoñeremos e exemplificaremos un modelo de Poisson. O modelo de Poisson é de gran utilidade para modelar datos de conteo. Máis concretamente, este modelo é aplicable a datos discretos con valores enteiros non negativos que contan algo, como por exemplo a cantidade de veces que ocorre un evento nun período de tempo concreto. De novo, unha vez axustado o modelo cómpre analizar se resulta o máis adecuado para o conxunto de datos en particular.

Capítulo 1

Modelo de Regresión Loxística

Igual que para o modelo lineal contamos cunha variable resposta y e unha serie de variables explicativas $x_1, ..., x_q$, no caso da regresión lineal, máis familiar para nós, a nosa variable resposta era continua. Pola contra neste modelo trataremos un problema de regresión cando a nosa variable resposta é discreta. O modelo de regresión loxística permítenos relacionar unha variable categórica, en particular dicotómica, cunha serie de variables de predicións independentes entre si. É dicir, consideremos unha variable resposta Y que toma valores 0 ou 1 cunha certa probabilidade $(1 - p_i)$ e p_i. O obxectivo deste modelo en concreto é relacionar esta variable resposta descrita con q variables predictoras X_1,...,X_q que se corresponden coas nosas variables explicativas. Neste tipo de regresión plantexar un modelo simplemente lineal sería un erro. Xa que no caso dun modelo simplemente lineal a distribución de probabilidade da nosa variable resposta era normal, de xeito que a media de Y podíase expresar como función lineal e a varianza das observacións era a mesma. Agora, no caso do modelo loxístico a distribución de probabilidade da nosa variable resposta non é normal, xa que é discreta e segue unha distribución de Bernoulli. Ademais dado que a variable resposta é binaria a varianza dependerá da media condicionada, polo que a varianza dependerá das variables explicativas e non terá por que ser constante coma no modelo lineal.

Estas diferenzas son suficientes para xustificar a necesidade de utilizar un modelo distinto do lineal. A regresión loxística é o resultado de combinar un modelo de regresión lineal da forma:

$$\eta = \beta_0 + \beta_1 X_1 + ... + \beta_q X_q, \tag{1.1}$$

cunha función enlace g como:

$$\eta_i = g(p_i).$$

Posto que a relación lineal η_i=p_i non é viable dado que precisamos que $0 \leq p_i \leq 1$ e $\eta_i \in \mathbb{R}$. Debido a esto usaremos una función enlace g como η_i=$g(p_i)$. Precisamos que g sexa monótona e

verifique $0 \leq g^{-1}(\eta) \leq 1$ para calquera valor η. Ainda que podemos pensar en distintas funcións g que satisfagan estas propiedades, a elección máis popular é a función loxística ou función *logit* dada pola seguinte expresión:

$$\eta = g(p) = \ln\left(\frac{p}{1-p}\right),$$

ou equivalentemente

$$p = \frac{\exp^{\eta}}{1+\exp^{\eta}}. \tag{1.2}$$

A combinación do emprego desta función logit xunto cun predictor lineal é o que recibe o nome de Regresión Loxística.

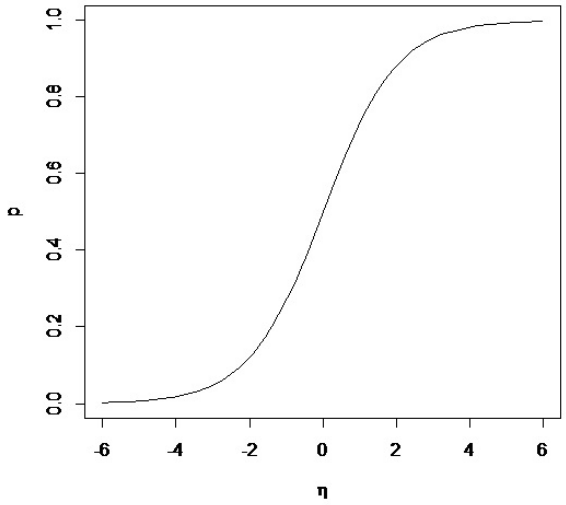


Figura 1.1: Relación entre a probabilidade da resposta, p, e o predictor lineal η.

Na Figura 1.1 podemos apreciar que a curva loxística é case lineal no seu rango medio. Isto quere dicir, que para o modelaxe de respostas con probabilidades próximas a 0,5 o comportamento da regresión loxística e lineal non vai ser moi diferente. Ademais pódese apreciar que nos extremos a curva achégase a cero e a un pero sen chegar nunca a tales límites, o que quere dicir que a regresión loxística non vai predicir algo inevitable ou imposible.

Recapitulando, recordemos que p nas expresións anteriores fai referencia a probabilidade de éxito fronte a $1-p$ que fai referencia a de fracaso. A función loxística consiste en aplicar un logaritmo ao cociente de ambas probabilidades. O cociente desas probabilidades indicadas é o que coñecemos como odds. As odds defínense como a probabilidade de éxito entre a de fracaso

$$\text{odds} = \frac{p}{1-p}$$

ou de xeito equivalente

$$p = \frac{\text{odds}}{1 + \text{odds}}.$$

Estas son unha escala alternativa a probabilidade para representar o azar, pódense ver como unha forma de expresar os pagos das apostas. Así podemos dicir que a probabilidade de que o FC Barcelona gañe o clásico fronte ao Real Madrid é de 2/3 ou o que é o mesmo que a odd vale 2. Falando en términos de apostas diríamos que as apostas están 2 a 1 a favor do FC Barcelona.

Unha importante vantaxe matemática das odds é que poden tomar calquera valor real positivo, mentres que a probabilidade de éxito só podía tomar valores no intervalo [0,1]. Dado que obviamente ao ser un cociente de cantidades positivas as odds van a ser positivas podemos aplicarlle un logaritmo e así transformalas nunha cantidade real calquera.

Empregando o concepto de odds e combinándoo con (1.1), o modelo de regresión loxística é:

$$\eta = \ln \text{odds} = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_q X_q$$

ou

$$\text{odds} = \exp^{\beta_0} \cdot \exp^{\beta_1 X_1} \cdot \exp^{\beta_2 X_2} \cdot ... \cdot \exp^{\beta_q X_q}.$$

De aquí podemos interpretar β_1 como segue: un aumento unitario en X_1 con $X_2, ..., X_q$ fixadas aumenta os $ln(odds)$ de éxito de β_1 ou equivalentemente aumenta as probabilidades de éxito dun factor $\exp^{\beta_1}$. Polo que en certo modo a interpretación dos coeficientes expoñenciais pode ser máis práctica.

1.1. Estimación dos parámetros do modelo

A continuación imos estimar os parámetros do modelo de regresión loxística, para iso usaremos o xa coñecido método de máxima verosimilitude. O estimador de máxima verosimilitude é o valor ou valores dos parámetros que maximiza a función masa de probabilidade ou densidade da mostra nas observacións. Definindo a función de verosimilitude para unha distribución con función masa de probabilidade P_θ que depende dun parámetro θ, como:

$$\alpha(\theta) = P_\theta(X_1 = x_1, ..., X_n = x_n) = \prod_{i=1}^{n} P_\theta(X = x_i),$$

o estimador de máxima verosimilitude é aquel valor $\hat{\theta}$ para o cal temos a seguinte igualdade:

$$\alpha(\hat{\theta}) = \max_{\theta} \alpha(\theta).$$

Como a función logaritmo é monótona crecente e podemos supoñer que $\alpha(\theta)$ é positiva, $\hat{\theta}$ é o estimador de máxima verosimilitude se e só se é un máximo de:

$$l(\theta) = \ln\left(\alpha(\theta)\right). \tag{1.3}$$

A obtención dos candidatos a máximos pódese conseguir derivando (1.3) con respecto de θ.

Céntrandonos agora, no caso que nos ocupa, sabemos que no modelo loxístico a variable resposta segue unha distribución de *Bernoulli* con parámetro p_i, a súa función masa de probabilidade aplicada a unha observación y_i, onde $i \in \{1, ..., n\}$, é :

$$P(Y = y_i; p_i) = p_i^{y_i} \left(1 - p_i\right)^{1-y_i},$$

e a súa función de verosimilitude ten a seguinte expresión:

$$\alpha(\boldsymbol{y}, \boldsymbol{p}) = \prod_{i=1}^{n} p_i^{y_i} (1 - p_i)^{1-y_i} = \prod_{i=1}^{n} \left(\frac{p_i}{1 - p_i}\right)^{y_i} (1 - p_i).$$

Aplicando as funcións expoñencial e logarítmo e empregando as propiedades destes, chegamos a seguinte expresión equivalente:

$$\alpha(\boldsymbol{y}, \boldsymbol{p}) = \left\{\prod_{i=1}^{n} (1 - p_i)\right\} \left\{\prod_{i=1}^{n} \exp\left[\ln\left(\frac{p_i}{1 - p_i}\right)^{y_i}\right]\right\}.$$

Tendo en conta que $\ln\left(\frac{p_i}{1-p_i}\right)^{y_i} = y_i \ln\left(\frac{p_i}{1-p_i}\right) = y_i\eta_i$ e $p_i = \frac{\exp \eta_i}{1+\exp \eta_i}$ e a continuación remplazando $\eta_i = \ln\left(\frac{p_i}{1-p_i}\right)$ e $1 - p_i = \frac{1}{1+\exp \eta_i}$ obtemos a seguinte expresión simplificada:

$$\alpha(\boldsymbol{\eta}) = \left\{\prod_{i=1}^{n} \frac{1}{1 + \exp \eta_i}\right\} \exp\left[\sum_{i=1}^{n} y_i\eta_i\right]. \tag{1.4}$$

Empregando agora o logaritmo para (1.4) e a propiedade do logaritmo dun cociente obténse:

$$l(\boldsymbol{\eta}) = \sum_{i=1}^{n} y_i\eta_i - \ln\left(1 + \exp \eta_i\right). \tag{1.5}$$

Partindo de (1.5) e tendo en conta que $\eta_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_q x_{iq}$ ou o que é o mesmo tomando $\eta_i = \boldsymbol{x_i'}\boldsymbol{\beta}$, sendo $\boldsymbol{x_i'}$ o trasposto do vector de predicciónss $\boldsymbol{x_i}$. Teremos así, a seguinte expresión en función de $\boldsymbol{\beta}$:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} y_i \boldsymbol{x_i'}\boldsymbol{\beta} - \ln(1 + \exp(\boldsymbol{x_i'}\boldsymbol{\beta})). \tag{1.6}$$

Plantexamos a continuación a expresión da derivada da log-verosimilitude. Deste xeito, igualándoas a cero, obteremos o parámetro que será candidato a ser o máximo, e polo tanto o estimador de $\boldsymbol{\beta}$

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} \left[y_i \boldsymbol{x_i'} - \frac{\exp(\boldsymbol{x_i'}\boldsymbol{\beta})}{1 + \exp(\boldsymbol{x_i'}\boldsymbol{\beta})} \boldsymbol{x_i'}\right] = 0. \tag{1.7}$$

En resumo, o método de máxima verosimilitude permítenos obter un estimador, para iso o que se fai é derivar a log-verosimilitude e igualar a cero, chegando a (1.7). Polo que, finalmente para obter o estimador desexado debemos resolver as ecuacións (1.7) e así chegar a unha expresión

para β_j. No caso do modelo lineal, as estimacións dos parámetros obtíñanse a partir do método de mínimos cadrados co cal acadabamos un sistema de n ecuacións e n incógnitas, coñecidas como ecuacións normais de regresión con solucións explícitas. Sen embargo, neste caso a ausencia de solución explícita, fai que teñamos que recurrir a métodos iterativos co fin de achar o estimador buscado. Para resolver numéricamente as ecuacións (1.7) é preciso recorrer a matriz hessiana, dada pola expresión:

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}} = -\sum_{i=1}^{n} \boldsymbol{x_i} \left(\frac{\exp(\boldsymbol{x_i'\beta}) \boldsymbol{x_i'}}{\left(1+\exp(\boldsymbol{x_i'\beta})\right)^2} \right). \tag{1.8}$$

A partir desta expresión (1.8) e tendo en conta que, $\eta_i = \boldsymbol{x_i'\beta}$, $p_i = \frac{\exp \eta_i}{1+\exp \eta_i}$ e $1-p_i = \frac{1}{1+\exp \eta_i}$ podemos reescribir a expresión para a matriz hessiana como segue:

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}} = -\sum_{i=1}^{n} [\boldsymbol{x_i x_i'} p_i (1-p_i)]. \tag{1.9}$$

Dado que precisamos resolver as ecuacións plantexadas pode ser de gran utilidade escribir matricialmente a expresión obtida para a matriz hessiana. Tendo en conta a seguinte notación:

$$X = \begin{pmatrix} 1 \; x_{11} \ldots x_{1q} \\ \vdots \\ 1 \; x_{n1} \ldots x_{nq} \end{pmatrix}, \tag{1.10}$$

onde esta expresión (1.10) non é máis que a matriz que contén os valores das variables explicativas. E denotando por V a matriz diagonal, con valores $p_i(1-p_i)$, temos:

$$V = \begin{pmatrix} p_1(1-p_1) \; 0 \ldots 0 \\ \ddots \\ 0 \ldots 0 \; p_n(1-p_n) \end{pmatrix},$$

finalmente a matriz hessiana pódese escribir:

$$H = \frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}} = -\sum_{i=1}^{n} [\boldsymbol{x_i x_i'} p_i (1-p_i)] = -X'VX. \tag{1.11}$$

No caso que nos ocupa, a partir da expresión para a matriz hessiana (1.11) vemos que é preciso empregar a teoría estándar para resolver un sistema de $n+1$ ecuacións e $n+1$ incógnitas e así obter aproximacións dos erros estándares. Como non se coñece unha solución explícita, temos que recorrer ao uso de métodos iterativos.

1.1.1. Métodos iterativos para o cálculo das estimacións

Para resolver as ecuacións de verosimilitude do modelo loxístico existen diversos métodos iterativos. O primeiro e máis coñecido para nós é o método de Newton-Raphson. Este método

parte dun iterante inicial, $\boldsymbol{\beta^0}$, e obtén sucesivamente o valor do seguinte parámetro mediante a seguinte expresión (Hastie, Tibshirani, e Friedman, 2009):

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} - \left(\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}} \mid \boldsymbol{\beta} = \boldsymbol{\beta^k} \right)^{-1} \frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \mid \boldsymbol{\beta} = \boldsymbol{\beta^k}, \tag{1.12}$$

sendo k o número de iteración. Dada a expresión xeral para o método de Newton-Raphson simplemente queda sustituir en (1.12) para o caso particular que nos ocupa, é dicir, o relativo a regresión loxística. Para iso debemos recordar a expresión matricial para a matriz hessiana descrita en (1.11) e as relacións dadas en (1.1) e en (1.2) así como a expresión da matriz de variables explicativas (1.10). Neste punto podemos expresar as ecuacións de verosimilitude dadas en (1.7) con notación matricial.:

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = X'(\mathbf{y} - \mathbf{p}) = 0, \tag{1.13}$$

onde $\mathbf{y}$ é o vector $\mathbf{y} = (y_1, \ldots, y_n)$, é dicir é a mostra de Y e análogamente $\mathbf{p} = (p_1, \ldots, p_n)$. Tendo en conta estas expresións , volvemos a escritura xeral do método (1.12) e sustituimos polas expresións relativas a matriz hessiana e as ecuacións de verosimilitude ambas en forma matricial, obtendo:

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} + (X' V^k X)^{-1} X'(\mathbf{y} - \boldsymbol{p^k}), \tag{1.14}$$

onde V^k é a matriz diagonal V con valores $p_i(1 - p_i)$ e $\boldsymbol{p^k}$ é o vector $\mathbf{p}$ mudando $\boldsymbol{\beta}$ polo valor de $\boldsymbol{\beta}$ na iteración anterior, é dicir por $\boldsymbol{\beta^k}$.

A partir de (1.14) podemos comezar o proceso iterativo, para o cal, en primeiro lugar debemos partir dun iterante inicial $\boldsymbol{\beta^0}$ e establecer un umbral ϵ . Este valor ϵ permitirános saber se o noso algoritmo converxe ou non. Principalmente o proceso iterativo a seguir consistirá no cálculo de $\mathbf{p}$ usando $\mathbf{p} = \frac{\exp(X\boldsymbol{\beta^k})}{1+\exp(X\boldsymbol{\beta^k})}$ e no cálculo de V cuxa diagonal vén dada por $V_{ii} = p_i(1 - p_i)$. Estos termos calculados iránse introducindo progresivamente na expresión (1.14) para cada iteración. Repetíndose o proceso ata que $\|\boldsymbol{\beta^k} - \boldsymbol{\beta^{k+1}}\| < \epsilon$, cando isto ocorra se é que sucede, usaremos o valor obtido de β^k como estimador do vector de parámetros $\boldsymbol{\beta}$ e rematamos o procedemento. Se isto último, non se da ao cabo dunha cantidade determinada de iteracións, podemos dicir que non hai converxencia.

Sen embargo, a hora de levar a cabo a implementación deste método no *software de R* (R Core Team, 2021) debemos desenvolver manualmente o proceso descrito, o que pode resultar tedioso a hora de escribir o código. Para realizar a regresión loxística, o *software R* (R Core Team, 2021), na función `glm(...,family="binomial")`, non emprega este método por defecto se non que utiliza o método de mínimos cadrados reponderados iterativamente *IRLS* polas súas siglas en inglés (*iteratively reweighted least squares*).

O método IRLS está estreitamente relacionado co método de Newton-Raphson (Agresti, 1990). Para comprender esta relación compre obter a matriz de información de Fisher (Colaboradores

de Wikipedia, 2022), que para o noso caso concreto, esta coincide co oposto da matriz hessiana dada en (1.11):

$$J = X'VX = -\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}}. \tag{1.15}$$

A continuación volvendo a expresión xeral do método de Newton-Raphson (1.12) e neste caso escribindoo en términos da matriz de información de Fisher teremos:

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} + J^{-1}X'(\mathbf{y} - \mathbf{p}). \tag{1.16}$$

Agora multiplicando a ambos lados de (1.16) por $J = X'VX$, chegamos a:

$$\begin{aligned} (X'VX)\boldsymbol{\beta^{k+1}} &= (X'VX)\boldsymbol{\beta^k} + (X'VX)(X'VX)^{-1}X'(\mathbf{y} - \mathbf{p}) \\ X'V\left[X\boldsymbol{\beta^k} + V^{-1}(\mathbf{y} - \mathbf{p})\right] &= X'V\boldsymbol{z^k}, \end{aligned}$$

deste xeito temos que:

$$\boldsymbol{\beta^{k+1}} = (X'VX)^{-1}X'V\boldsymbol{z^k}, \tag{1.17}$$

onde,

$$\boldsymbol{z^k} = X\boldsymbol{\beta^k} + V^{-1}(\mathbf{y} - \mathbf{p}). \tag{1.18}$$

Chegados a isto, xa temos todo o necesario para iniciar o proceso iterativo partindo de (1.17) e avanzando nos valores de V e $\boldsymbol{z^k}$ debido a que en cada iteración o valor de $\mathbf{p}$ varía acorde aos valores de $\boldsymbol{\beta}$ calculados na iteración anterior. Este algoritmo recibe o nome de mínimos cadrados reponderados iterativamente xa que en cada iteración resolvemos un problema de mínimos cadrados ponderados pola matriz V:

$$\arg \min_{\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta}} (\boldsymbol{z} - X\boldsymbol{\beta})'V(\boldsymbol{z} - X\boldsymbol{\beta}). \tag{1.19}$$

1.2. Inferencia sobre os parámetros do modelo

A continuación levaremos a cabo certos procesos de inferencia sobre o noso modelo de regresión loxística, como: realizar un contraste para ver se se pode asumir que os parámetros β_j son iguais a cero e construir intervalos de confianza para ditos parámetros.

1.2.1. Contraste de modelos mediante *deviance*

O obxectivo desta sección será obter o mellor axuste posible para o noso modelo. Con este fin, empregaremos o test de razón de verosimilitudes, este método basease en comparar a verosimilitude baixo unha condición, H_0 coa verosimilitude baixo o modelo con outra condición alternativa H_1. Para o caso que nos ocupa e partindo da función de verosimilitude descrita en (1.4), podemos

escribir esta última en términos dos $\boldsymbol{\beta}$ simplemente tendo en conta que $p_i = \frac{\exp(\boldsymbol{x_i}'\boldsymbol{\beta})}{1+\exp(\boldsymbol{x_i}'\boldsymbol{\beta})}$. Obtendo así, a función de verosimilitude en términos das $\boldsymbol{\beta}$ como:

$$\alpha(\boldsymbol{\beta}|(x_1, y_1) \dots (x_n, y_n)) = \prod_{i=1}^{n} \left[\frac{\exp(\boldsymbol{x_i}'\boldsymbol{\beta})}{1+\exp(\boldsymbol{x_i}'\boldsymbol{\beta})}\right]^{y_i} \left[\frac{1}{1+\exp(\boldsymbol{x_i}'\boldsymbol{\beta})}\right]^{1-y_i}. \tag{1.20}$$

Recorrendo ao método citado, dados dous modelos con distinto número de variables explicativas, un modelo con l variables explicativas e función de verosimilitude α_l, obtida empregando o estimador de $\boldsymbol{\beta}$ na expresión (1.20), e un modelo con s variables explicativas e verosimilitude α_s. Onde o modelo de menor tamaño representa un subconxunto do grande, é dicir, son modelos aniñados ($s > l$). A seguinte relación estadística pódese empregar como un estadístico válido para comparar ambos modelos:

$$-2\log\frac{\alpha_l}{\alpha_s}. \tag{1.21}$$

A distribución deste estadístico pódese aproximar mediante unha ji-cadrado con tantos graos de liberdade como parámetros se perden ao pasar do modelo longo ao máis curto ($s - l$).

Relacionado co estadístico (1.21) aparece o concepto de *deviance*, que mide a desviación do modelo loxístico axustado respecto dun modelo perfecto. Este modelo ideal coñécese como modelo saturado e é un modelo que proporciona predicións da variable resposta que coinciden cos propios valores da mostra. No caso de modelos con variable resposta discreta, como o que nos ocupa, é máis sinxelo obter un modelo saturado xa que a variable resposta ten menos valores posibles.

Dado o modelo axustado e o modelo saturado a *deviance* coincide coa razón de verosimilitudes (1.21) cando se teñen en conta o modelo axustado e o modelo saturado:

$$D_{Modelo} = -2\log\left[\frac{\text{Verosimilitude Modelo}}{\text{Verosimilitude Modelo Saturado}}\right]. \tag{1.22}$$

Empregando por unha parte a expresión para a verosimilitude (1.20) e por outra a definición para a *deviance* do modelo (1.22) e tendo en conta que a verosimilitude do modelo saturado é 1, a *deviance* para o noso modelo loxístico é:

$$D = -2\sum_{i=1}^{n}[y_i\log(\hat{p_i}) + (1-y_i)\log(1-\hat{p_i})], \tag{1.23}$$

onde $\hat{p_i} = \frac{\exp\hat{\eta_i}}{1+\exp\hat{\eta_i}}$ son os valores do modelo axustado, con $\hat{\eta_i} = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_q x_{iq}$.

1.2.2. Intervalos de confianza para os parámetros de regresión

Consideraremos o intervalo de confianza de Wald (Cepeda-Cuervo et al., 2008) para o noso estimador $\hat{\beta}_i$ obtido a partir do método de máxima verosimilitude descrito na Sección 1.1. Neste

punto, acudindo a teoría asintótica sobre os estimadores de máxima verosimilitude ou a un teorema do límite central podemos deducir que a distribución de $\hat{\boldsymbol{\beta}}$ converxe a $N(\boldsymbol{\beta}, (X'VX)^{-1})$. É dicir, $\hat{\boldsymbol{\beta}}$ segue unha distribución normal de media $\boldsymbol{\beta}$ e matriz de varianzas-covarianzas $J^{-1} = (X'VX)^{-1}$. Facendo uso do resultado anterior, o intervalo de confianza para β_i é o seguinte:

$$\hat{\beta}_i \pm z^{\alpha/2} ET(\hat{\beta}_i), \tag{1.24}$$

onde $z^{\alpha/2}$ é o cuantil $1-\alpha/2$ dunha distribución normal estándar e n o tamaño mostral. Con isto, simplemente bastaría obter unha fórmula correcta para o erro típico de $\hat{\beta}_i$. O erro típico áchase partindo de que $\hat{\beta}_i$ segue unha distribución normal cuxa matriz de varianzas-covarianzas coincide precisamente coa matriz de información de Fisher J, definida en (1.15). A continuación, estimando os parámetros $\boldsymbol{\beta}$ e tendo presente tamén que $J = X'VX$, en lugar de V introduciremos $\hat{V}$ sustituíndo en V os parámetros polas súas estimacións. Definindo agora unha matriz $\hat{J} = X'\hat{V}X$ unha forma de obter o erro típico asociado a un β_i concreto consistirá en facer simplemente a raíz cadrada do elemento (i,i) da matriz $\hat{J}^{-1}$. Deste xeito, dado calquera estimador $\hat{\beta}_i$, poderemos obter o seu erro típico e por conseguinte a forma do intervalo de Wald do mesmo.

Teóricamente este intervalo ten, para valores grandes de n, un nivel de confianza aproximado de $100(1-\alpha)\,\%$

1.3. Selección do Modelo

Podemos descubrir que non todas as variables predictoras obtidas son útiles para explicar a resposta polo que tentaremos identificar un subconxunto destas que modele o mellor posible a resposta. Para isto podemos empregar un método de axuste de fácil implementación como o método de eliminación *back-ward*. Para a implementación deste, partimos do modelo completo, con todas as variables predictoras dispoñibles, comparamos secuencialmente este modelo con todos os modelos resultantes de eliminar cada unha destas variables predictoras. Para isto, podemos obter os p-valores correspondentes a realizar o contraste $H_0 : \beta_j = 0$ fronte a $H_1 : \beta_j \neq 0$. Unha estratexia habitual é eliminar aquela variable predictora cuxo p-valor asociado tome o valor máis elevado. Repetimos o proceso ata que non se poidan eliminar máis variables predictoras sen unha pérdida do axuste estadísticamente significativa. A significación de cada variable pódemola ver realizando no *software* de R (R Core Team, 2021) un `summary` do modelo `glm(... , family="binomial")` aplicado anteriormente e ollando o valor obtido do p-valor asociado a cada variable. Con `family="binomial"` especifícase que función de probabilidade utilizamos e `glm` por defecto emprega a función *logit* como función de enlace. A continuación o `summary` emprega a normalidade asintótica introducida na sección (1.2.2) para realizar o contraste mencionado. Deste xeito, se o p-valor obtido no `summary` é moi baixo indicanos que debemos rexeitar a hipótese nula e polo tanto considerar que os nosos coeficientes son distintos de 0. Análogamente tamén

podemos empregar o método *forward*, neste caso en lugar de suprimir o término menos significativo, engadiremos aquela variable predictora, que ao introducila, o modelo resultante sexa máis significativo, é dicir, teña p-valores máis pequenos asociados as súas variables. A pesar de que os algoritmos descritos son de fácil implementación, non son os mellores para identificar o modelo desexado.

Se en lugar de observar a significación de cada coeficiente queremos construír unha medida global para o modelo, podemos empregar un criterio moi popular, o criterio de información de Akaike, AIC. Este criterio para un modelo con verosimilitude l_q e número de parámetros q defínese por:

$$AIC = -2\log(l_q) + 2q. \tag{1.25}$$

Outra opción perfectamente válida para a mesma labor é o emprego do criterio de información bayesiano ou BIC. Este último está estreitamente ligado co criterio anterior e vén dado por:

$$BIC = -2\log(l_q) + q\log(n), \tag{1.26}$$

onde n é o tamaño da mostra. Explorando modelos con distinto número de parámetros (de variables predictoras), finalmente seleccionaremos aquel modelo cun valor máis pequeno tanto de AIC como de BIC. Xa que se o valor do AIC ou BIC é pequeno significa que contamos cun modelo con gran verosimilitude e poucos parámetros. A idea principal é atopar aquel modelo que incorpore variables realmente útiles para así incrementar a verosimilitude.

1.3.1. Detección de datos atípicos

Plantexamonos facer unha diagnose sobre o modelo de regresión loxística para encontrar algún punto inusual. Como no caso dos modelos lineais, os residuos son o máis importante para determinar que tan bos axustes temos para o modelo e onde sería aconsellable unha modificación ou mellora. Podemos calcular os residuos como a diferenza entre os valores observados e axustados. No caso do modelo lineal os residuos presentaban a mesma varianza e a suma residual de cadrados $\sum_{i=1}^{n} \hat{\epsilon_i}^2$, era de gran utilidade para a diagnose do modelo. Pola contra, no modelo loxístico os residuos brutos non teñen a mesma varianza e como cantidade equivalente a suma residual de cadrados pódese empregar a *deviance* residual, $\sum_{i=1}^{n} r_i^2$. Debido a isto, podemos recorrer aos residuos estandarizados, estos son o valor do residuo dividido entre unha estimación da súa desviación estándar. No caso do modelo de regresión loxística os residuos estandarizados son da forma:

$$r_i = \frac{y_i - \hat{p}_i}{\sqrt{\hat{p}_i(1-\hat{p}_i)}}. \tag{1.27}$$

Así, residuos estandarizados demasiado grandes en valor absoluto indican que a observación correspondente pode ser anómala ou atípica. Para o caso dos modelos lineais pódese considerar

como critero para que un dato sexa atípico aquél cun residuo estandarizado maior que 2 ou menor que -2. Ademais, se o modelo axustado é acertado, estos residuos seguirán unha distribución normal estándar. No caso da regresión loxística, a expresión para a *deviance* residual, $\sum_{i=1}^{n} r_i^2$, ten a mesma distribución asintótica que a *deviance*, é dicir, unha chi-cadrado cos mesmos graos de liberdade. A partir dos residuos, tamén podemos elaborar gráficos en R que contrasten os residuos calculados fronte as predicións e interpretalos de forma similar aos modelos lineais.

Pondo especial atención na detección de puntos inusuais, igual que para os modelos lineais, podemos examinar os *leverages* ou apalancamentos. Estos calcúlanse a partir da matriz $H = \hat{V}^{1/2}X(X'\hat{V}X)^{-1}X'\hat{V}^{1/2}$, onde $\hat{V}$ é a matriz obtida a partir de V, sustituíndo os parámetros polos seus estimadores. O valor do *leverage* para a observación i-ésima vén dado polo elemento i-ésimo da diagonal principal da matriz H. Canto máis grande sexa o valor do *leverage* máis pequena será a varianza do residuo. Esta varianza pequena interprétase negativamente, pois as observacións asociadas poderán convertirse en observacións demasiado influíntes no modelo.

1.4. Conclusión

Nun modelo de regresión loxística podemos tratar as mesmas cuestións que para o caso dos modelos lineais. Algunhas destas cuestións resólvense dun xeito moi similar en ambos modelos, mentres que noutros aspectos para a o modelo loxístico surxen máis dificultades. Dado que este último ten unha variable resposta discreta e a interpretación das variables predictoras obtidas pode resultar máis complexa. Deste xeito, a diferenza principal entre ambos modelos é que no caso do modelo de regresión loxística a variable resposta segue unha distribución discreta en concreto unha distribución de Bernoulli en lugar dunha distribución continua como é a normal para o caso lineal. Isto refléxase na saída prevista para a esperanza da resposta, que no caso do modelo loxístico débese atopar no intervalo $[0, 1]$.

Capítulo 2

Modelo de Poisson

O modelo de Poisson é un tipo de modelo de regresión que se utiliza para datos de conteo. Surxe cando a variable resposta é unha cantidade discreta que podemos modelar mediante unha distribución de Poisson e queremos estudar se certas variables explicativas inflúen nela e cómo o fan. Polo tanto, a regresión de Poisson é similar a regresión loxística, que tamén ten unha variable resposta discreta. Sen embargo, no caso da regresión de Poisson a resposta non se limita a uns valores $\{0, 1\}$ como no modelo loxístico.

Recordemos que Y, unha variable aleatoria discreta, segue unha distribución de Poisson con media $\mu > 0$ se a súa función de probabilidade vén dada por:

$$P(Y = y) = \frac{\exp(-\mu)\mu^y}{y!}, \tag{2.1}$$

con $y = 0, 1, 2, ...$ Onde y é o número de veces que ocorre un suceso de interese e μ é un parámetro positivo que coincide coa esperanza. A propiedade principal desta distribución é que a media e a varianza son iguais ao parámetro μ, é dicir, $E(Y) = \mu = Var(Y)$ (Hilbe, 2014). Ademais, a medida que o valor de μ aumenta, a distribución de Poisson vaise volvendo máis simétrica e finalmente aproxímase correctamente a unha distribución normal (Fox, 2008), como podemos ver na Figura 2.1. Dadas estas características a distribución de Poisson é unha boa aproximación: no caso de pequenas probabilidades de éxito e grandes mostras, cando a probabilidade de que ocorra un evento nun intervalo de tempo é proporcional a duración do mesmo e independente de se sucedan ou non outros eventos ou cando o tempo entre sucesos e independente e identicamente distribuido expoñencialmente (Faraway, 2016).

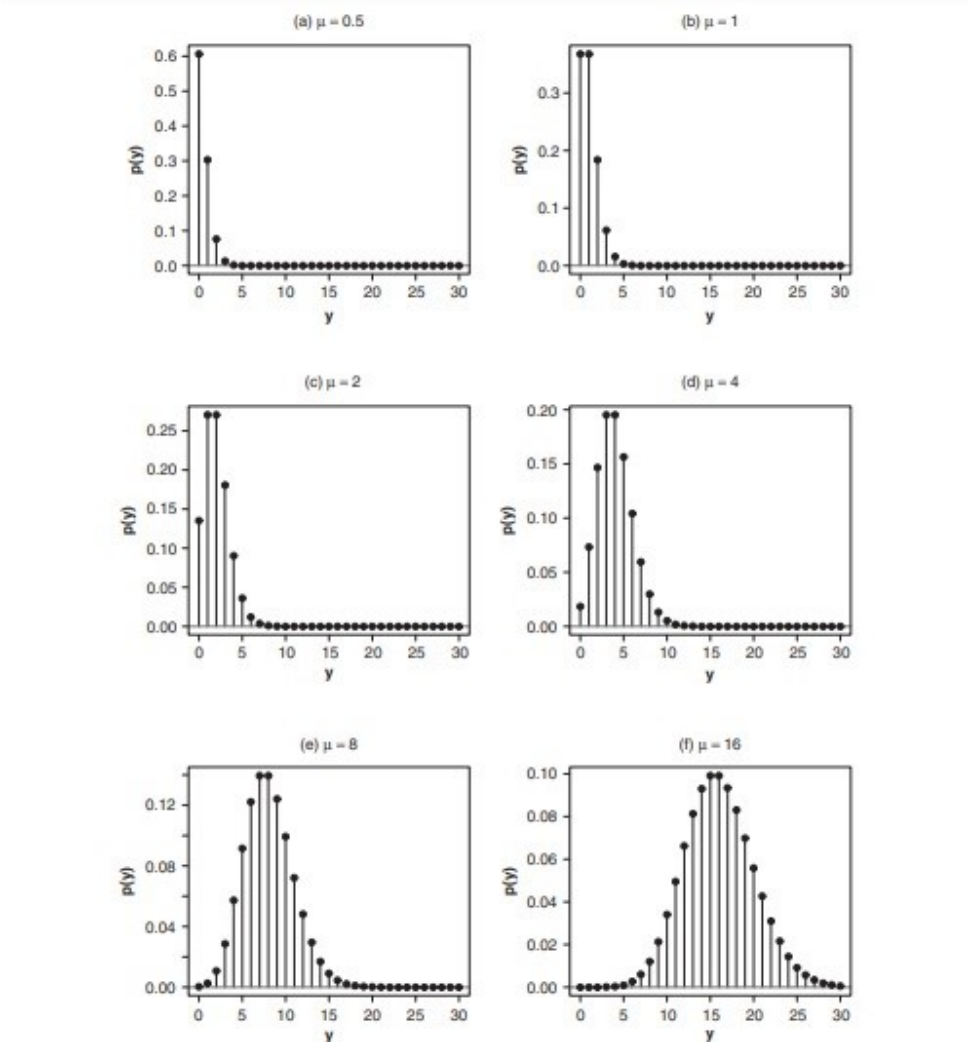


Figura 2.1: Distribución de Poisson para varios valores do parámetro μ

2.1. Plantexando o Modelo de Poisson

Supoñamos que temos un conxunto de respostas y_i, con $i \in \{1, \ldots, n\}$, que queremos modelar en función dun vector de predicións $\boldsymbol{x_i}$ e sabemos ademais que $y_i \sim Pois(\mu_i)$, coma antes queremos expresar μ_i en función dunha combinación lineal de $\boldsymbol{x_i}$, $\eta_i = \boldsymbol{x_i'\beta}$, expresión relativa as predicións. Ademais a función de regresión η_i está no intervalo $(-\infty, +\infty)$ e necesitaremos que $\mu_i \geq 0$. Polo que poderemos usar unha función enlace logarítmica ou *log link* obtendo a seguinte expresión:

$$\log \mu_i = \eta_i = \boldsymbol{x_i'\beta}. \tag{2.2}$$

Deste xeito, a función de regresión do modelo de Poisson exprésase como:

$$\mu_i = \exp\left(\boldsymbol{x_i'\beta}\right). \tag{2.3}$$

A partir desta expresión (2.3) podemos deducir unha interpretación sinxela para os nosos parámetros. Supoñamos que a compoñente explicativa x_j aumenta en n unidades e sabendo que as variables explicativas restantes se manteñen constantes teremos que a media para a variable de Poisson multiplicase pola potencia n-ésima de $\exp \beta_j$.

2.2. Estimación dos parámetros do modelo

Igual que no caso da regresión loxística este modelo tamén conta cun predictor lineal e unha función enlace. Empregaremos de novo o método de máxima verosimilitude explicado no Capítulo 1. No modelo de regresión de Poisson a nosa masa de probabilidade vén dada:

$$\mathbb{P}(Y = y_i; \mu_i) = \frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!}, \tag{2.4}$$

e deste xeito, a función de verosimilitude ten a seguinte expresión:

$$\alpha(\boldsymbol{y}, \boldsymbol{\mu}) = \prod_{i=1}^{n} \frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!}. \tag{2.5}$$

Aplicando a (2.5) a seguinte igualdade, $\mu_i = \exp\left(\boldsymbol{x_i'\beta}\right) = \exp \eta_i$ ou equivalentemente $\log \mu_i = \eta_i$ obtemos:

$$\alpha(\boldsymbol{\eta}) = \prod_{i=1}^{n} \frac{\exp(-\exp(\eta_i))\exp(\eta_i y_i)}{y_i!}, \tag{2.6}$$

empregando a función logarítmo e as súas propiedades para (2.6) chegamos:

$$l(\boldsymbol{\eta}) = \sum_{i=1}^{n} -\exp(\eta_i) + y_i\eta_i - \log(y_i!), \tag{2.7}$$

partindo de (2.7) e simplemente sustituíndo $\eta_i = \boldsymbol{x_i'\beta}$ teremos:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i \boldsymbol{x_i'\beta} - \exp(\boldsymbol{x_i'\beta}) - \log(y_i!)). \tag{2.8}$$

Diferenciando (2.8) respecto de $\boldsymbol{\beta}$, teremos a expresión da derivada da log-verosimilitude. Neste punto, igualando a cero, obteremos finalmente o candidato a máximo , e polo tanto o estimador de $\boldsymbol{\beta}$.

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} y_i \boldsymbol{x_i'} - \exp\left(\boldsymbol{x_i'\beta}\right) \boldsymbol{x_i'} = 0. \tag{2.9}$$

A ecuación (2.9) pódese escribir de forma compacta, tendo en conta (2.3) como:

$$X'\mathbf{y} - X'\boldsymbol{\mu} = 0, \tag{2.10}$$

sendo X a matriz que contén os valores das variables explicativas:

$$X = \begin{pmatrix} 1\ x_{11} \dots x_{1q} \\ \vdots \\ 1\ x_{n1} \dots x_{nq} \end{pmatrix}. \tag{2.11}$$

Finalmente, seguindo o procedemento do método de máxima verosimilitude, para obter o estimador desexado debemos resolver as ecuacións (2.9) e así chegar a unha expresión para β_j. Sen embargo, de xeito análogo que para o modelo loxístico, non existe unha fórmula explícita de $\hat{\boldsymbol{\beta}}$ para a regresión de Poisson, xa que as ecuacións de verosimilitude non son lineais nos parámetros. Polo tanto para o cálculo das estimacións debemos recorrer a procedementos iterativos para atopar unha solución. Igual que no capítulo anterior Newton-Raphson e o IRLS (*Iterative Re-weighted Least Squares*) son exemplos de métodos iterativos que podemos aplicarlle a este modelo para obter as predicións desexadas. Por conseguinte, pode resultar de interese escribir a expresión relativa a matriz hessiana. Esta obtense derivando respecto de $\boldsymbol{\beta}$ a expresión (2.9):

$$H = \frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta'}} = -\sum_{i=1}^{n} \exp(\boldsymbol{x_i'\beta}) \boldsymbol{x_i x_i'}. \tag{2.12}$$

Obtemos así a expresión da matriz hessiana H. Dado isto, e denotando por V a matriz diagonal, con valores $\exp(\boldsymbol{x_i'\beta})$, é dicir:

$$V = \begin{pmatrix} \exp(\boldsymbol{x_1'\beta}) \dots 0 \\ \ddots \\ 0 \dots \exp(\boldsymbol{x_n'\beta}) \end{pmatrix}, \tag{2.13}$$

podemos expresar a matriz hessiana H matricialmente como segue:

$$H = -X'VX. \tag{2.14}$$

2.2.1. Métodos iterativos para o cálculo das estimacións

Para resolver as ecuacións de verosimilitude do modelo de Poisson, igual que para o modelo loxístico, debemos empregar métodos iterativos. De novo, poderemos aplicarlle o método de Newton-Raphson. Este método parte dun iterante inicial $\boldsymbol{\beta^0}$ a partir do cal obteremos sucesivamente o valor do que o segue, mediante esta expresión:

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} - \left(\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}\, \partial \boldsymbol{\beta'}} \mid \boldsymbol{\beta} = \boldsymbol{\beta^k} \right)^{-1} \frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \mid \boldsymbol{\beta} = \boldsymbol{\beta^k}, \tag{2.15}$$

para $k \in \{1, \dots, n\}$. A continuación, simplemente quedaría sustituir na expresión xeral do método de Newton-Raphson (2.15), as expresión relativas a (2.10) e (2.14) para obter:

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} - (H^k)^{-1} \left(X'\boldsymbol{y} - X'\boldsymbol{\mu^k} \right), \tag{2.16}$$

onde H^k e $\boldsymbol{\mu^k}$ son a matriz H dada en (2.14) e o vector $\boldsymbol{\mu}$ trocando $\boldsymbol{\beta}$ polo valor de $\boldsymbol{\beta}$ na iteración previa, é dicir, $\boldsymbol{\beta^k}$.

A partir de (2.16) podemos comezar o proceso iterativo, de xeito totalmente análogo ao capítulo anterior. Temos que partir dun iterante inicial $\boldsymbol{\beta^0}$ e establecer un umbral ϵ. Este ϵ permitirános saber se o algoritmo converxe ou non. O proceso iterativo repetiráse ata que $\|\boldsymbol{\beta^k} - \boldsymbol{\beta^{k+1}}\| < \epsilon$, cando isto ocorra se é que sucede, tomaremos como estimador de $\boldsymbol{\beta}$ o valor obtido $\boldsymbol{\beta^k}$ e rematamos o procedemento. Se isto último, non se da ao cabo dunha cantidade determinada de iteracións, podemos dicir que non hai converxencia.

Sen embargo a función `glm` do *software* de R, igual que no caso anterior, emprega o método IRLS. Partindo da definición da matriz de información de Fisher, tense que neste caso $J = -\frac{\partial^2 l(\beta)}{\partial \beta^2} = -H$ (Colaboradores de Wikipedia, 2022), polo que volvendo a expresión xeral do método de Newton-Raphson (2.15) e escribindoa en términos da matriz de información de Fisher temos:

$$\boldsymbol{\beta^{k+1}} = \boldsymbol{\beta^k} + (J^k)^{-1}\left(X'\boldsymbol{y} - X'\boldsymbol{\mu}\right). \tag{2.17}$$

Agora, se multiplicámos a ambos lados de (2.17) por $J = -H = X'VX$, obtemos:

$$\begin{aligned}(X'VX)\boldsymbol{\beta^{k+1}} &= (X'VX)\boldsymbol{\beta^k} + (X'VX)(X'VX)^{-1}X'(\boldsymbol{y} - \boldsymbol{\mu}) = \\ &= X'V[X\boldsymbol{\beta^k} + V^{-1}(\boldsymbol{y} - \boldsymbol{\mu})] = \\ &= X'V\boldsymbol{z^k}.\end{aligned} \tag{2.18}$$

onde, neste caso $\boldsymbol{z^k} = X\boldsymbol{\beta^k} + V^{-1}(\boldsymbol{y} - \boldsymbol{\mu})$. Chegados a isto, xa temos todo o necesario para comezar o proceso iterativo, partindo da expresión (2.18) e avanzando nos valores de V e $\boldsymbol{z^k}$. Desta forma, en cada iteración, resolveremos un problema de mínimos cadrados ponderados pola matriz V:

$$\arg\min_{\boldsymbol{\beta}}(\boldsymbol{z} - X\boldsymbol{\beta})'V(\boldsymbol{z} - X\boldsymbol{\beta}),$$

o que da nome o algoritmo.

2.3. Inferencia sobre os parámetros do modelo

Nesta sección realizaremos certos procesos de inferencia sobre o modelo de Poisson. En primeiro lugar, pode resultar de utilidade levar a cabo un contraste para ver se os β_j son iguais a cero. Por outra banda tamén podemos construir intervalos de confianza para os nosos parámetros.

2.3.1. Contraste de modelos mediante *deviance*

Análogamente que no Capítulo 1, podemos empregar o test de razón de verosimilitudes. Este consiste en comparar a verosimilitude do modelo baixo unha condición H_0 coa verosimilitude

con outra condición H_1. Neste caso partindo da función de verosimilitude (2.6) e escribindoa en términos das $\boldsymbol{\beta}$, empregando $\eta_i = \boldsymbol{x_i'}\boldsymbol{\beta}$ temos:

$$\alpha(\boldsymbol{\beta}|(x_1, y_1) \dots (x_n, y_n)) = \prod_{i=1}^{n} \frac{\exp(-\exp(\boldsymbol{x_i'}\boldsymbol{\beta}))\exp(\boldsymbol{x_i'}\boldsymbol{\beta} y_i)}{y_i!}. \tag{2.19}$$

De novo como para o modelo loxístico, dados dous modelos con distinto número de variables explicativas, un con l variables explicativas e outro con s, sendo $s > l$, e funcións de verosimilitude α_l e α_s respectivamente. Podemos de novo suxerir o seguinte estadístico para comparar os dous modelos aniñados:

$$-2 \log \frac{\alpha_l}{\alpha_s},$$

o cal se poderá aproximar por unha ji-cadrado con $s - l$ graos de liberdade.

A partir deste estadístico, xorde de novo o concepto de *deviance* definido en (1.22). Empregando a expresión para a verosimilitude (2.19) xunto coa definición (1.22) e tendo en conta que a verosimilitude do modelo saturado é 1, podemos expresar a *deviance* da regresión de Poisson como segue:

$$D = 2 \sum_{i=1}^{n} \left(y_i \log \left(\frac{y_i}{\hat{\mu}_i} \right) - (y_i - \hat{\mu}_i) \right), \tag{2.20}$$

empregando ademais que $\hat{\mu}_i = \exp(\boldsymbol{x_i'}\hat{\boldsymbol{\beta}})$ e onde $\hat{\boldsymbol{\beta}}$ é o estimador de máxima verosimilitude de $\boldsymbol{\beta}$. Un valor elevado deste estadístico pode indicar un axuste pobre para o modelo.

Mediante a diferenza das *deviance* e comparandoas con unha distribución χ^2 con tantos graos de liberdade como a diferenza entre o número de parámetros dos dous modelos, podemos comparar tamén dous modelos aniñados.

Bondade de axuste

Agora podemos testar a bondade de axuste do modelo proposto comparando a *deviance* do modelo fronte unha distribución χ^2 con tantos grados de liberdade como presente o noso modelo (McCullagh e Nelder, 1989). A maiores tamén se pode testar a significación individual dos predictores e construir intervalos de confianza para $\boldsymbol{\beta}$, usando o erro estándar. Unha alternativa a χ^2 é efectuar un contraste. No cal a hipótese nula, H_0, é que o modelo imposto é o correcto, fronte a hipótese alternativa, H_1, que non o sexa. Desta forma, compre calcular un estadístico de contraste a partir da nosa mostra, neste caso o estadístico de Pearson X^2:

$$X^2 = \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}, \tag{2.21}$$

que dado un nivel de significación α, imposto previamente, este estadístico (2.21) permitiranos efectuar o contraste. Dado que, baixo certas condicións de regularidade, a distribución de (2.21)

pódese aproximar por unha ji-cadrado con tantos graos de liberdade como a diferenza entre o número de observacións e a cantidade de parámetros do modelo e tendo isto en conta, poderemos obter un p-valor asociado a este estadístico. Co cal concluiremos o noso contraste, se o p-valor obtido é menor que o criterio de significación α preestablecido, rexeitaremos a hipótese nula; no caso contrario acaptarémola. Cando o número de observacións é o suficientemente grande, a *deviance* e o X^2 son equivalentes.

Unha alternativa, para testar o axuste do modelo é a interpretación do pseudo-R^2 de McFadden. Este vén dado pola seguinte expresión $D^2 = \frac{NullDeviance - ResidualDeviance}{NullDeviance}$, onde *ResidualDeviance* é a diferenza entre a *deviance* do modelo que non depende de ningunha variable menos a do modelo que inclúe as variables explicativas, mentres que a *NullDeviance* é a *deviance* para un modelo que non depende de ningunha variable. Os valores deste coeficiente, D^2, comprendidos entre 0,2 e 0,4 indican según McFadden, un bo axuste para o modelo (Hensher e Stopher, 2021).

2.3.2. Intervalos de confianza para os parámetros de regresión

Os intervalos de confianza para o modelo de Poisson pódense escribir de xeito análogo ao modelo de regresión loxística do Capítulo 1. Polo que consideraremos o intervalo de confianza de Wald (Cepeda-Cuervo et al., 2008) para o noso parámetro $\boldsymbol{\beta_j}$. Ademais , en virtude do teorema do límite central podemos deducir que a distribución de $\hat{\boldsymbol{\beta}}$ converxe a $N(\boldsymbol{\beta}, (X'VX)^{-1})$ (Cameron e Trivedi, 1998), sendo V a matriz definida en (2.13). Con isto, o intervalo de confianza de Wald para $\boldsymbol{\beta_j}$ vén dado como:

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{j}} \pm z^{\alpha/2} ET(\hat{\boldsymbol{\beta}}_{\boldsymbol{j}}), \tag{2.22}$$

onde $z^{\alpha/2}$ é o cuantil $(1-\alpha/2)$ dunha distribución normal estándar. Véxase no Capítulo 1, que simplemente queda obter a fórmula para o erro típico de $\hat{\boldsymbol{\beta}}_{\boldsymbol{j}}$, como indicamos. Denotamos por $\hat{J}$ a matriz $\hat{J} = X'\hat{V}X$, onde $\hat{V}$ é a matriz obtida a partir da expresión de V sustituíndo os parámetros polas súas estimacións. Deste xeito, o erro típico asociado a un $\hat{\boldsymbol{\beta}}_{\boldsymbol{j}}$ concreto vén dado como segue:

$$ET(\hat{\beta}_j) = \left[\hat{J^{-1}_{jj}}\right]^{1/2}. \tag{2.23}$$

Deste modo, para calquera estimador $\hat{\boldsymbol{\beta}}_{\boldsymbol{j}}$ pódese obter o seu erro típico e escribir o intervalo de Wald do parámetro $\boldsymbol{\beta_j}$. Este intervalo ten, para valores grandes de n, un nivel de confianza aproximado de $100(1-\alpha)\,\%$. Sen embargo para modelos con poucas observacións os coeficientes xeralmente non se achegan normalidade asumida.

2.4. Selección do Modelo

En primeiro lugar podemonos plantexar identificar aquelas variables predictoras que non son útiles para modelar a nosa resposta. Para esta labor, podemos acudir aos métodos *back-ward* e *forward* detallados no Capítulo 1, e finalmente obter o modelo cuxas variables predictoras sexan máis significativas.

Co fin de detectar de forma global o conxunto de predicións que mellor modelen a nosa resposta tamén podemos recorrer ao criterio de información de *Akaike*, AIC, ou ao criterio de información bayesiano, BIC, descritos polas expresións (1.25) e (1.26) respectivamente.

Desta forma ao comparar o axuste de modelos, os valores máis baixos tanto de AIC como de BIC indicarán mellores axustes para o noso modelo. Dado que un valor baixo do AIC ou BIC implica maior verosimilitude e menos parámetros.

2.4.1. Detección de datos atípicos

Análogamente ao modelo regresión loxística e ao visto para modelos lineais, se nos plantexamos facer unha diagnose sobre o modelo de Poisson podemos empregar os residuos, que determinarán que tan bos axustes temos para o noso modelo e se sería necesario algunha modificación. No modelo de Poisson, os residuos non teñen a mesma varianza polo que unha forma de medir a diferenza entre os valores observados e axustados neste modelo é a *deviance* residual. Establecemos a *deviance* residual de tal forma que se teña $\sum_{i=1}^{n} r_i^2 = deviance$, onde para este modelo concreto, os residuos da *deviance* veñen dados por:

$$r_i = \text{sign}\left(y_i - \hat{\mu_i}\right)\left[2\left(y_i \log\left(y_i/\hat{\mu_i}\right) - y_i + \hat{\mu_i}\right)\right]^{1/2}, \tag{2.24}$$

onde sign a función é a función que obtén o signo do que tomemos como entrada. Desta forma, se o valor absoluto destes residuos é demasiado alto pode ser debido a que a observación correspondente é atípica. No caso da regresión lineal, podíamos considerar como candidato a dato atípico aquél cun residuo da *deviance* superior a 2 ou inferior a -2.

2.5. Sobredispersión

Como sabemos, unha das características principais da regresión de Poisson é que a media e a varianza coinciden. Sen embargo, cando axustamos un modelo de Poisson a un conxunto de datos determinado, pode darse que estos valores difiran entre si. Isto é o que se coñece como sobredispersión do modelo de Poisson e ten consecuencias cualitativas similares ao incumprimento da hipótese de homocedasticidade no modelo de regresión lineal (Hilbe, 2014).

No paquete `AER` do *software* de R (R Core Team, 2021) hai un test que nos permitirá contrastar a sobredispersión do modelo de Poisson. O comando necesario para este test é `dispersiontest`. Dito test, realiza un contraste onde a hipótese nula H_0 é que o modelo de Poisson é equidisperso, fronte a alternativa H_1 de sobredispersión ou subdispersión, esta última menos frecuente (Hilbe, 2014). Observando o p-valor asociado ao test podemos rexeitar a hipótese nula se este valor é pequeno ou no caso contrario aceptala. No caso en que teñamos sobredispersión para o modelo, pódese empregar como alternativa a regresión de Poisson a regresión binomial negativa (McCullagh e Nelder, 1989).

2.6. Conclusión

O modelo de regresión de Poisson presenta diferenzas significativas respecto do modelo lineal estándar e o modelo de regresión loxística. A diferenza principal é que para o modelo de Poisson a nosa variable resposta segue unha distribución discreta en concreto unha distribución de Poisson en lugar dunha distribución continua como é a distribución normal para o caso lineal. Recordemos que para o modelo loxístico a distribución de probabilidade da nosa variable resposta tamén era discreta pero seguía unha distribución de Bernoulli. Outra diferenza significativa é a varianza do modelo, no caso do modelo lineal esta é constante, mentres que no caso do modelo de regresión loxística e de Poisson non o é, ademais para este último a varianza coincide coa media.

Capítulo 3

Aplicación do Modelo de Regresión Loxística

Chegados a este punto, e recorrendo aos conceptos teóricos explicados no Capítulo 1, aplicaremos o modelo de regresión loxística a un conxunto de datos predeterminado. É dicir, analizaremos un conxunto de datos sobre o cancro de mama en Wisconsin, extraídos do repositorio de aprendizaxe automático UCI que en inglés é coñecido como *UCI Machine Learning Repository* (Dua e Graff, 2017).

3.1. Diagnóstico do cancro de mama

3.1.1. Descrición dos datos

Dispoñemos dun conxunto de datos relativos ao diagnóstico do cancro de mama en Wisconsin. Este conxunto de datos foi doado por Nick Street en 1995 e os seus creadores son: Dr. William H. Wolberg, do departamento de Ciruxía Xeral, W. Nick Street e Olvi L. Mangasarian, do departamento de Ciencias da Computación, todos eles pertencentes a universidade de Wisconsin (Dua e Graff, 2017).

A base de datos recolle un rexistro de 569 casos clínicos particulares, fichados cada un deles polo número de identificación do paciente. O noso obxetivo de traballo será partindo dunha serie de características, determinar se un tumor é maligno (M) ou benigno (B). As características das que dispomos, calculáronse a partir dunha imaxe dixitalizada dunha aspiración con agulla fina (FNA) de cada masa mamaria e describen principalmente particularidades dos núcleos celulares presentes na imaxe.

Centraremonos nas variables indicadas a continuación. Dentro deste conxunto de variables, a variable resposta do noso modelo será *diagnosis*, mentres que as demais serán as variables explicativas que pretenden modelar a resposta:

- Media do radio (*radius_ mean*): media das distancias dende o centro do núcleo celular aos puntos do perímetro do mesmo. Tomará valores como: 18, 20.6, 19.7, 11.4, 20.3, ...
- Media da textura (*texture_ mean*): variable de valores númericos (como 10.4, 17.8, 21.2, 20.4, 14.3, ...) referente a media das desviacións estándar dos valores da escala de grises.
- Media do perímetro (*perimeter_ mean*): media do perímetro dos núcleos celulares, os valores posibles para esta variable poden ser 122.8, 132.9, 130, 77.6, 135.1, ...
- Media da área (*area_ mean*): media da área dos núcleos celulares. Podemos observar os seguintes valores asociados a esta variable: 1001, 1326, 1203, 386, 1297, ...
- Media da uniformidade (*smoothness_ mean*): media da uniformidade, é dicir, media das variacións locais das lonxitudes dos radios, presenta valores como: 0.1184, 0.0847, 0.1096, 0.1425, 0.1003, ...
- Media da compacidade (*compactness_ mean*): media da compacidade que se calcula como $\frac{perimetro^2}{area} - 1$. Esto da lugar aos seguintes posibles valores para dita variable: 0.2776, 0.0786, 0.1599, 0.2839, 0.1328, ...
- Media da concavidade (*concavity_ mean*): media da concavidade, é dicir, severidade das porcións cóncavas do contorno. Contamos cos seguintes valores asociados a variable: 0.3001, 0.0869, 0.1974, 0.2414, 0.198, ...
- Media dos puntos cóncavos (*concave.points_ mean*): media dos puntos cóncavos, é dicir, número de porcións cóncavas do contorno. Pode tomar os seguintes valores: 0.1471, 0.0702, 0.1279, 0.1052, 0.1043, ...
- Media da simetría (*symmetry_ mean*): media das simetrías. Variable númerica con valores como: 0.242, 0.181, 0.207, 0.26, 0.181, ...
- Media da dimensión do fractal (*fractal_ dimension_ mean*): media das dimensións dos fractais. A variable pode tomar os seguintes valores: 0.0787, 0.0567, 0.06, 0.0974, 0.0588, ...

Dado este conxunto de variables que describen características asociadas a cada masa mamaria, o obxetivo é propor o mellor modelo posible para o diagnóstico dun tumor. Polo tanto, podemos plantexar un modelo de regresión loxística que nos permita modelar a nosa resposta dicotómica, de se certo tumor resulta ser maligno ou benigno.

3.1.2. Aplicación do modelo de regresión loxística.

Escritura e selección do modelo.

A regresión loxística permite estimar a probabilidade dunha variable cualitativa binaria en función de certas variables cuantitativas. A saída prevista para a esperanza da nosa resposta atoparase dentro do intervalo $[0, 1]$, mentres que se empregamos un axuste lineal para modelar a nosa resposta é posible que os valores pronosticados para a nosa resposta non se encadren en tal intervalo.

Observando a variable resposta, diagnosis, vemos que do total de diagnósticos a maioría resultarán ser tumores malignos, 357, fronte a 212 que serán benignos. Podemos aplicar aos nosos datos un modelo de regresión loxística, para ver que variables explicativas das descritas inflúen no diagnóstico do cancro. Para iso comezaremos escribindo o modelo formado pola resposta diagnosis e todas as variables explicativas indicadas e faremos un `summary` do mesmo.

```
Call:
glm(formula = diagnosis ~ radius_mean + texture_mean + perimeter_mean +
    area_mean + smoothness_mean + compactness_mean + concavity_mean +
    concave.points_mean + symmetry_mean + fractal_dimension_mean,
    family = binomial, data = datos)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.95590  -0.14839  -0.03943   0.00429   2.91690

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             -7.35952   12.85259  -0.573   0.5669
radius_mean             -2.04930    3.71588  -0.551   0.5813
texture_mean             0.38473    0.06454   5.961  2.5e-09 ***
perimeter_mean          -0.07151    0.50516  -0.142   0.8874
area_mean                0.03980    0.01674   2.377   0.0174 *
smoothness_mean         76.43227   31.95492   2.392   0.0168 *
compactness_mean        -1.46242   20.34249  -0.072   0.9427
concavity_mean           8.46870    8.12003   1.043   0.2970
concave.points_mean     66.82176   28.52910   2.342   0.0192 *
symmetry_mean           16.27824   10.63059   1.531   0.1257
fractal_dimension_mean -68.33703   85.55666  -0.799   0.4244
```

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

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 146.13  on 558  degrees of freedom
AIC: 168.13

Number of Fisher Scoring iterations: 9
```

Comezaremos ollando os p-valores obtidos para cada unha das variables. O contraste que se plantexa ten como hipótese nula, H_0, que o coeficiente asociado a certa variable sexa cero ou equivalentemente que certa variable non explica a nosa resposta, fronte a hipótese alternativa, H_1, que indica o contrario. A interpretación destes p-valores é moi similar a do modelo lineal. É dicir, as variables de interese serán aquelas que teñan uns p-valores máis pequenos que un nivel dado, xa que os p-valores menores indicarán maior significación no modelo da variable asociada. En concreto, no caso que nos ocupa vemos que as únicas variables que teñen p-valores baixos, é dicir, son significativas para un nivel de significación $\alpha = 0,05$, son *texture_mean*, *area_mean*, *smoothness_mean* e *concave.points_mean*. Con isto, descubrimos que non todas as variables preditoras indicadas son de utilidade para explicar a nosa resposta, polo que tentaremos seleccionar un subconxunto de variables que modele o mellor posible a resposta. Para iso, empregaremos o método *backware* descrito na Sección 1.3. A estratexia habitual é eliminar aquela variable predictora cuxo p-valor asociado tome o valor máis elevado. Volvendo a ollar o `summary`, vemos que a variable menos significativa é *compactness_mean* (cun p-valor asociado igual a 0.9427). Se volvemos axustar o modelo de regresión loxística prescindindo desta variable explicativa, obteríamos que *perimeter_mean* é a variable menos significativa, cun p-valor asociado igual a 0.7750. Repetindo iterativamente este proceso ata que os p-valores asociados a todas as variables aleatorias sexan menores que 0.05, vanse eliminando sucesivamente as variables *concavity_mean*, *fractal_dimension_mean* e *symmetry_mean*. Obtendo como resultado, un modelo final no que as únicas variables explicativas serían: *radius_mean*, *texture_mean*, *area_mean*, *smoothness_mean* e *concave.points_mean*. Neste último modelo plantexado, todas as variables predictoras son significativas polo que pomos fin o algoritmo do método *backware*. Sen embargo, podemos observar que non temos significación para o intercepto xa que o p-valor asociado para este é igual a 0.55983.

Como explicamos no Capítulo 1, na Sección 1.3, se en lugar de observar a significación de

cada coeficiente o que queremos é construír unha medida global para a selección do modelo, podemos empregar o criterio de información de Akaike, AIC. Para isto, simplemente debemos comparar os valores de AIC asociados a cada un dos modelos plantexados e quedarnos con aquel que presente un valor de AIC menor, xa que ese será o modelo máis verosímil. O AIC asociado ao primeiro modelo é 168.13, mentres que os asociados aos seguintes son 166.14, 164.22, 163.28, 162.38 e 162.68 respectivamente. Tendo en conta estes valores, o modelo máis plausible acorde a dito criterio é o penúltimo, é dicir, aquel cuxas variables explicativas son: *radius_ mean*, *texture_ mean*, *area_ mean*, *smoothness_ mean*, *concave.points_ mean*, *symmetry_ mean*.

Interpretación dos coeficientes.

Partindo do modelo seleccionado anteriormente, é dicir, aquel cuxas variables explicativas son *radius_ mean*, *texture_ mean*, *area_ mean*, *smoothness_ mean*, *concave.points_ mean* e *symmetry_ mean* podemos por atención nos coeficientes obtidos. A interpretación destes coeficientes difire da do modelo lineal. Como explicamos no Capítulo 1, o modelo de regresión loxística emprega unha función enlace coñecida como función *logit* e dada pola expresión (1.2). En consecuencia, para levar a cabo a interpretación desexada debemos recorrer ao concepto de *odds* definido no Capítulo 1, que para o modelo loxístico é igual a odds $= \exp^{\beta_0} \cdot \exp^{\beta_1 X_1} \cdot \exp^{\beta_2 X_2} \cdot \ldots \cdot \exp^{\beta_q X_q}$. Desta forma, en lugar de interpretar directamente, $\hat{\boldsymbol{\beta}}$, o vector de coeficientes axustados asociados a cada variable, que no noso caso particular ten a seguinte expresión:

$$\hat{\boldsymbol{\beta}} = (-8{,}6108, -2{,}7251, 0{,}3852, 0{,}0430, 58{,}7854, 73{,}7015, 15{,}5621), \tag{3.1}$$

e onde a primeira entrada deste vector correspode ao coeficiente asociado ao intercepto, a segunda ao coeficiente asociado a variable *radius_ mean* e así sucesivamente ata a última entrada correspondente a variable *symmetry_ mean*, pode ser máis útil a interpretación dos valores resultantes de aplicarlle a expoñencial a cada coeficiente. Deste xeito, os coeficientes do modelo de regresión loxística interprétanse como o logarítmo das odds ratio. É dicir, agora β_1 pódese interpretar como segue: un aumento unitario en X_1 con $X_2, \ldots, X_q$ fixados aumenta o logaritmo das *odds* asociado a β_1 ou o que é o mesmo, aumenta as probabilidades de éxito dun factor $\exp \beta_1$. Se nos fixamos no coeficiente axustado asociado a variable *texture_ mean*, 0.3852, este indica que o logaritmo das *odds ratio* de *diagnosis*, aumenta en 0.3852 unidades por cada unidade que aumenta *texture_ mean*. Ou equivalentemente, o aumento dunha unidade da variable *texture_ mean*, permanecendo as demais constantes, implicará que a probabilidade de ter un tumor maligno se multiplique por 1.469941. Por outra banda, o coeficiente relativo a variable *radius_ mean*, -2.72, indícanos que o logaritmo das *odds ratio* de *diagnosis* diminúe en 2.72 unidades por cada unidade que aumenta *radius_ mean*. Ou o que é o mesmo que por cada unidade que aumente *radius_ mean* a probabilidade de ter un tumor maligno multiplícase por 0.065. Deste modo, a interpretación do valor de $\hat{\beta}_j$ asociado a cada variable difire según o signo deste. Se o valor de $\hat{\beta}_j$ é positivo indica

que o aumentar o valor da variable, a probabilidade da nosa resposta, $P(Y = 1)$, increméntase, é dicir, dita variable favorece a aparición do evento en cuestión. Pola contra se o valor é negativo, aumentar o valor da variable implicará que a probabilidade da resposta diminúa, é dicir perxudica a aparición. No caso do noso exemplo aquelas variables que favorecerán a diagnose dun tumor maligno son: *texture_ mean*, *area_ mean*, e sobre todo *smoothness_ mean*, *concave.points_ mean* e *symmetry_ mean*. Por outra banda, a variable *radius_ mean* perxudicará a diagnose dun tumor maligno.

3.1.3. Inferencia sobre o modelo.

Contraste do modelo mediante deviance.

Como explicamos no Capítulo 1, co obxectivo de obter o mellor axuste para o noso modelo podemos recorrer ao test de razón de verosimilitudes. Este método permitiranos comparar dous modelos con distinto número de variables explicativas. Para iso, bastará observar o valor do estadístico que aparece na Ecuación (1.21) cuxa distribución se pode aproximar por unha ji-cadrado.

Nesta sección podemos comparar mediante o test descrito, o modelo inicial, formado por todas as variables explicativas indicadas, co modelo seleccionado, composto polas variables *radius_ mean*, *texture_ mean*, *area_ mean*, *smoothness_ mean*, *concave.points_ mean*, *symmetry_ mean*. Para levar a cabo dita comparación, é necesario obter o valor do estadístico (1.21), na práctica co *software* de R (R Core Team, 2021), basta aproximar o estadístico por unha distribución ji-cadrado cuxos graos de liberdade serán a cantidade de parámetros que se perden o pasar dun modelo a outro, neste caso 4:

```
G2 <- modelo4$deviance - modelo$deviance
1 - pchisq(G2, df = 4),
```

onde o "modelo4" é o modelo coas variables explicativas indicadas no parágrafo anterior, o "modelo" é aquel con todas as variables explicativas descritas inicialmente e G2 é a diferenza entre a *deviance* asociada a cada modelo. O valor do p-valor asociado é 0.688, isto indicanos que non podemos rexeitar a hipótese nula de que os parámetros asociados as variables *compactness_ mean*, *perimeter_ mean*, *concavity_ mean* e *fractal_ dimension_ mean* son nulos. Con isto verificamos que o modelo con menos variables é o máis indicado para o modelaxe da nosa resposta, xa que é un modelo máis sinxelo e que a vez proporciona unha verosimilitude similar.

Tamén podemos calcular este contraste dun xeito máis compacto, empregando a función `anova` da libraría `car` do *software* de R (R Core Team, 2021). Esta función realiza os contrastes

de razón de verosimilitudes sobre os parámetros asociados a cada unha das variables do modelo sen necesidade de especificar os distintos modelos. Os resultados de efectuar dito contraste son os seguintes:

```
Analysis of Deviance Table (Type II tests)

Response: diagnosis
                       LR Chisq Df Pr(>Chisq)
radius_mean               0.306  1    0.57992
texture_mean             49.209  1    2.3e-12 ***
perimeter_mean            0.020  1    0.88749
area_mean                 5.496  1    0.01906 *
smoothness_mean           6.289  1    0.01215 *
compactness_mean          0.005  1    0.94270
concavity_mean            1.103  1    0.29370
concave.points_mean       5.795  1    0.01607 *
symmetry_mean             2.307  1    0.12878
fractal_dimension_mean    0.647  1    0.42136
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

De aquí podemos deducir que as variables cuxos parámetros son significativamente distintos de cero, para un nivel de significación $\alpha = 0{,}05$, son: *texture_mean*, *area_mean*, *smoothness_mean* e *concave.points_mean*. Podemos escribir o modelo resultante e realizar un `summary` do mesmo (R Core Team, 2021).

```
Call:
glm(formula = diagnosis ~ texture_mean + area_mean + smoothness_mean +
    concave.points_mean, family = binomial, data = datos)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.31798  -0.15623  -0.04212   0.01662   2.84201

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -23.677816   3.882774  -6.098 1.07e-09 ***
texture_mean         0.362687   0.060544   5.990 2.09e-09 ***
```

```
area_mean             0.010342   0.002002   5.165 2.40e-07 ***
smoothness_mean      59.471304  25.965153   2.290    0.022 *
concave.points_mean  76.571210  16.427864   4.661 3.15e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 156.44  on 564  degrees of freedom
AIC: 166.44

Number of Fisher Scoring iterations: 8
```

Vemos que todas as variables indicadas para este modelo son significativas para modelar a nosa resposta. Neste caso incluso o valor obtido para o intercepto resulta significativo, o que nos fai pensar que o modelo formado por estas variables modela mellor a nosa resposta que o modelo obtido mediante o método backware ou o modelo obtido tendo en conta o AIC.

Intervalos de confianza para os parámetros do modelo.

Escribiremos o intervalo de confianza de Wald para os nosos parámetros β_j, segundo o indicado na Sección 1.2.2 do Capítulo 1. En primeiro lugar, debemos decidir o nivel de significación para os nosos intervalos, $(1-\alpha)$, esta significación soe expresarse mediante porcentaxes, podemos falar entón de intervalos de confianza ao 90 % ou 95 % segundo $\alpha = 0{,}1$ ou $\alpha = 0{,}05$ respectivamente.

Os intervalos de interese calcúlanse coa función `confint.default` do *software* de R, que por defecto toma un nivel de confianza $1-\alpha = 0{,}95$ (R Core Team, 2021). Podemos calcular estos intervalos para o modelo completo, é dicir o formado polas dez variables explicativas mencionadas. Deste xeito, obtemos os seguintes intervalos de confianza ao 95 %:

```
                          2.5 %       97.5 %
(Intercept)         -32.55012751  17.83109229
radius_mean          -9.33229662   5.23368682
texture_mean          0.25824448   0.51122420
perimeter_mean       -1.06161528   0.91859445
area_mean             0.00698718   0.07260522
smoothness_mean      13.80178675 139.06276076
```

```
compactness_mean         -41.33297960  38.40813510
concavity_mean            -7.44627507  24.38367459
concave.points_mean       10.90575011 122.73776358
symmetry_mean             -4.55732264  37.11380728
fractal_dimension_mean  -236.02499401  99.35094023
```

Se o intervalo inclúe ao 0 significa que para este nivel, pódese asumir que a covariable non ten efecto sobre a resposta. Neste caso particular, os intervalos que inclúen ao cero son os asociados aos coeficientes de *radius_mean*, *perimeter_mean*, *compactness_mean*, *concavity_mean*, *symmetry_mean* e *fractal_dimension_mean*, polo que serían os coeficientes que non aportan nada ao modelo.

3.1.4. Diagnose sobre o modelo.

Igual que ocorría cos modelos lineais, para atopar algún punto inusual debemos calcular os residuos do modelo. Estes residuos permitirannos determinar que tan ben se axustan os datos ao modelo. A continuación, recorreremos ao *software* de R (R Core Team, 2021) para calcular o valor dos residuos e dos residuos estandarizados, explicados na Sección 1.3.1. Para elo, empregaremos as funcións `residuals` e `rstandard` (R Core Team, 2021) respectivamente, e partiremos do modelo seleccionado a partir da función `anova` en 3.1.3, que recordemos, é aquel formado polas variables *texture_mean*, *area_mean*, *smoothness_mean* e *concave.points_mean*. Unha vez, calculados estes residuos podemos considerar como criterio para que certo dato sexa atípico aqueles residuos estandarizados de valor absoluto maior que 2. En ambos casos, estes datos atípicos, supoñen menos dun 2 % do total de residuos. Mostramos a continuación aqueles pacientes cuxos residuos estandarizados son máis elevados, xunto co valor dos mesmos.

```
     298       41      136      172      129       74
7.479692 6.124732 4.144604 3.865401 3.728307 2.789193
```

Pode resultar interesente ver, que probabilidade de diagnose dun tumor maligno, asigna o modelo a cada un destos pacientes. Para isto, empregaremos a función `fitted.values` do *software* de R, esta función extrae a probabilidade asociada ao éxito da resposta, é dicir, para o noso caso particular a probabilidade asociada a diagnose dun tumor maligno.

```
      298
0.01762363
       41
0.02610001
```

```
       136
0.05538029
       172
0.06311982
      129
0.9318822
       74
0.1150489
```

Vemos que ao paciente con número de identificación 298 asóciaselle, unha probabilidade de diagnose dun tumor maligno de menos do 2 %, no caso do número 41 de menos dun 3 %, para o 136 de menos dun 6 %, para o 172 de menos dun 7 %, para o 129 de máis dun 93 % e para o paciente número 74 de menos dun 12 %. Pola contra, se ollamos a base de datos, vemos que todos estos pacientes agás o número 129 foron diagnósticados dun tumor maligno. Por conseguinte, os datos asociadas as variables predictoras destes pacientes deben resultar anómalos, xa que fan que o modelo clasifique aos tumores dos pacientes cunha alta probabilidade na categoría contraria a que realmente pertencen.

Tamén resulta interesante elaborar gráficos en R que contrasten os residuos calculados fronte as predicións dadas polo modelo. Podemos comezar, simplemente debuxando os valores dos residuos para cada paciente empregando a función `plot` do *software* de R e engadir as rectas $y = -2$ e $y = 2$ como podemos observar na Figura 3.1.

Na Figura 3.1, podemos ollar e centrarnos naqueles residuos que presentan un valor elevado, é dicir os representados e numerados na Figura 3.2. Estes valores elevados poden indicar, como dixemos anteriormente, que os datos asociados a certo paciente son atípicos.

Por outra banda, para un modelo de regresión loxística a interpretación dun gráfico de residuos estandarizados fronte aos valores preditos, ao contrario que no caso dos modelos lineais, é complexa dado que contamos cunha variable resposta dicotómica (Agresti, 1990). Coa intención de analizar gráficamente os residuos para un modelo de regresión loxística, podemos recorrer ao paquete `'DHARMa'` (Hartig, 2022). Este emprega un enfoque basado na simulación para crear residuos fácilmente interpretables para modelos lineais xeralizados, de forma que os residuos resultantes están estandarizados a valores entre 0 e 1 e poden interpretarse de forma intuitiva como os residuos da regresión lineal. Observando a Figura 3.3, vemos dous paneis. O panel da esquerda mostra un gráfico de cuantil a cuantil para unha distribución uniforme e superpón probas para problemas de distribución específicos. Unha desas probas é o test de Kolmogorov-Smirnov (KS), para o cal a hipótese nula é que os datos seguen a distribución esperada. O p-valor asociado a

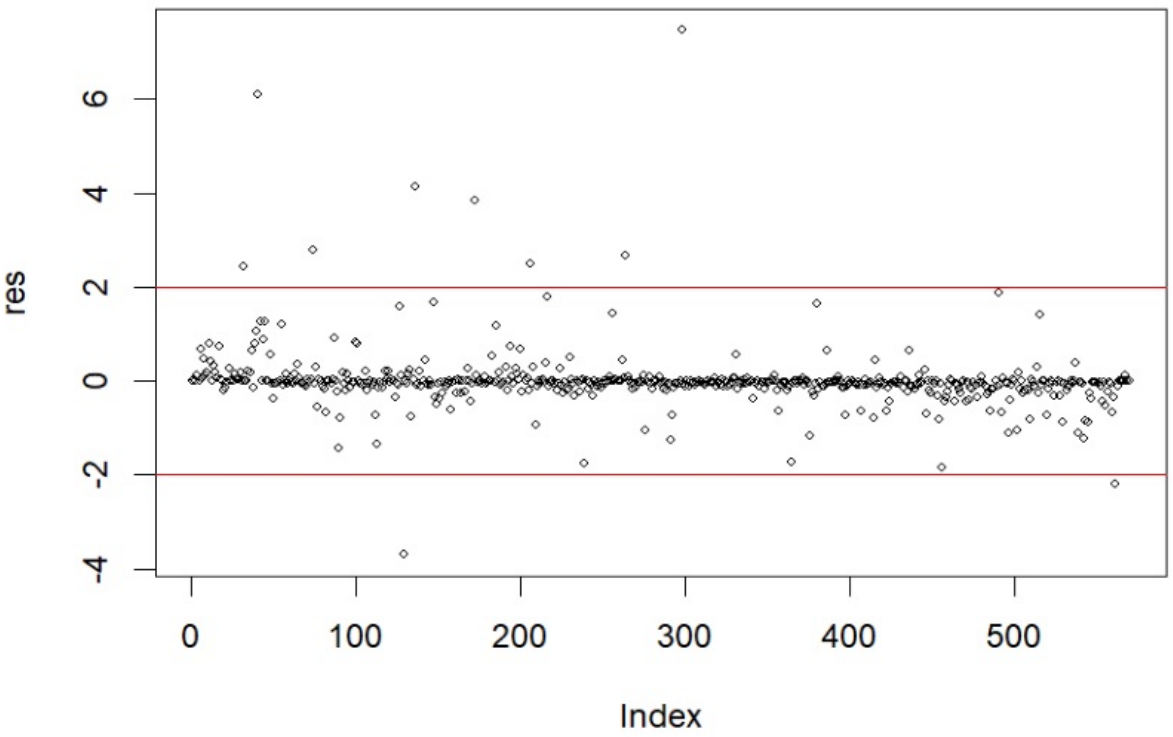


Figura 3.1: Residuos estandarizados do modelo.

dito test é 0.4455, polo que non estamos en condicións de rexeitar a hipótese nula. Por outra banda, o gráfico da dereita mostra os residuos fronte aos valores axustados, no caso de existir valores atípicos estos resaltaríanse en vermello. Ademais, a modo de apoio visual, para detectar desviacións da uniformidade, mostra unhas liñas rectas e discontinuas para os cuantiles por defecto. Estas liñas son as expectativas teóricas de como deberían estar distribuidos uniformemente os residuos para un modelo correcto. Desta forma, o *software* de R, en concreto o paquete `'DHARMa'` (Hartig, 2022), indicará en vermello se a desviación da regresión cuantílica axustada con respecto a expectativa é significativa e mostrarase unha advertencia no gráfico. No noso caso particular vemos que os residuos representados se axustan a distribución esperada, polo tanto o noso modelo semella non presentar ningún problema de sobredispersión ou subdispersión.

Por último, pondo de novo atención na detección de puntos inusuais, como indicamos na Sección 1.3.1 podemos examinar o valor dos *leverages* ou apalancamentos. Para calcular os *leverages* recorremos a función `hatvalues` do *software* estadístico R (R Core Team, 2021). Ademais, ordeando os valores obtidos de maior a menor, temos que os apalancamentos máis elevados son os seguintes:

```
hat.valores<-hatvalues(modelo6)
hat.valores.ma<-sort(hat.valores,decreasing = TRUE)
```

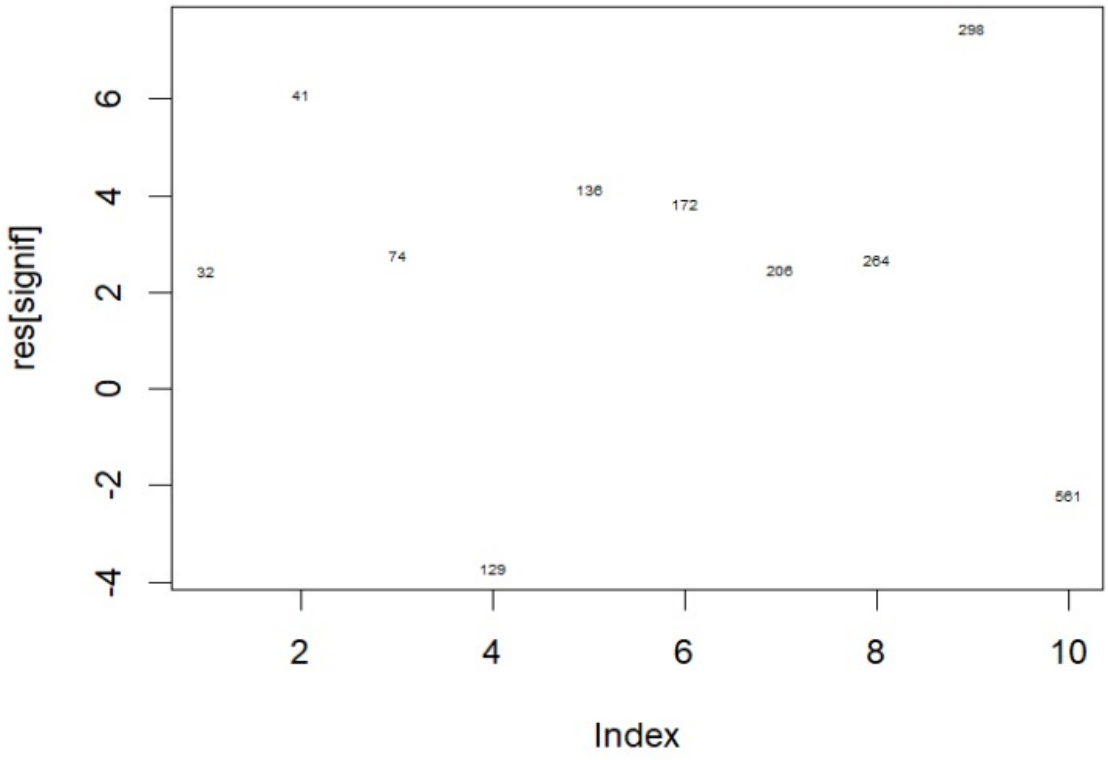


Figura 3.2: Residuos do modelo con valor elevado.

```
head(hat.valores.ma)
       113        492        538        505         77        529
0.18483481 0.10553580 0.10463376 0.09305295 0.06958510 0.06948025
```

Polo tanto no caso que nos ocupa, obtemos en xeral valores pequenos para os *leverages*, o que indica que a varianza do residuo non é pequena. Esto é positivo, dado que unha varianza do residuo pequena pode dar lugar a observacións demasiado influíntes no modelo.

3.1.5. Conclusión.

Partindo do modelo con mellor significación para as variables, é dicir aquel que inclúe como variables explicativas as seguintes: *texture_mean*, *area_mean*, *smoothness_mean* e *concave.points_mean*, será de gran interese comparar as predicións obtidas coas observacións. Desta maneira, veremos que tan bo resulta o noso modelo para predicir a nosa resposta. Para levar a cabo este análise, asumiremos que o modelo clasifica a un paciente cun tumor maligno se $\hat{p}_i \geq 0{,}5$.

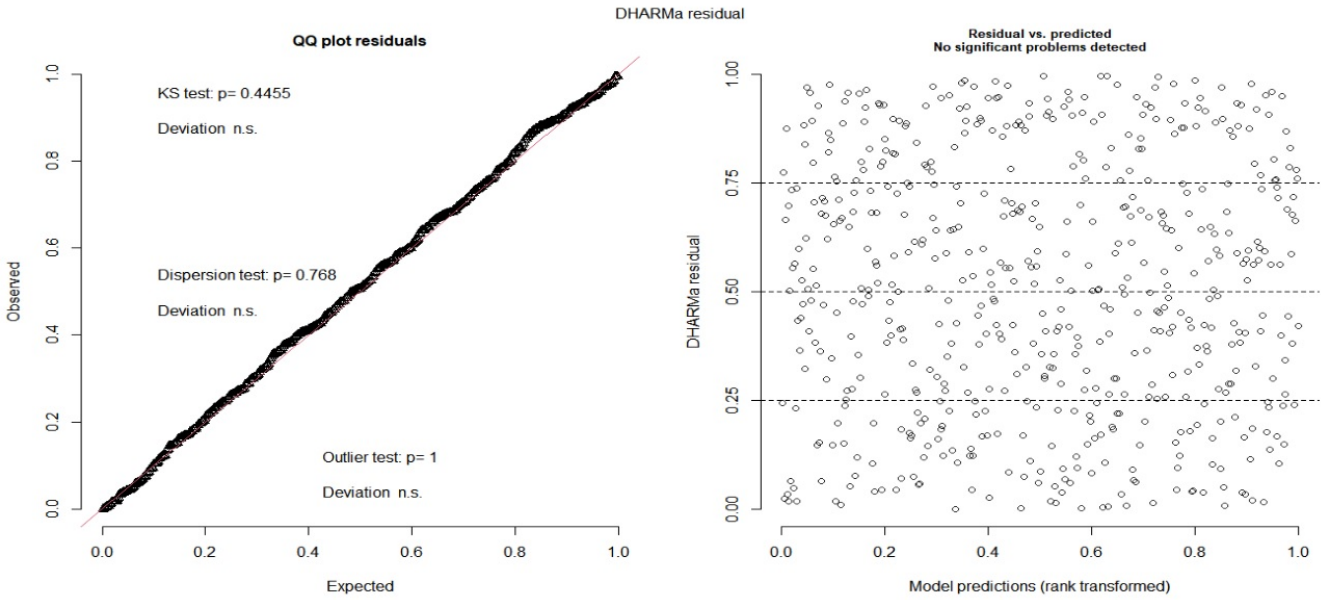


Figura 3.3: Gráficos de residuos empregando o paquete `'DHARMa'`.

Ademais, empregaremos o programa R (R Core Team, 2021), para elaborar unha matriz que reflexe a compación mencionada. Ademais, para maior claridade, tamén podemos representar dita matriz mediante un mapa de calor, como mostra a Figura 3.4 . Deste xeito, os valores en verde, é dicir os da diagonal principal da matriz, correspóndense cos valores estimados correctamente polo modelo. Mentres que a outra diagonal, indicada na Figura 3.4 en vermello, representa os casos onde o modelo se equivoca.

```
                predicciones
observaciones    0   1
            0  343  14
            1   20 192
```

Desta forma o modelo é capaz de clasificar correctamente $\frac{343+192}{343+14+20+192} = 0{,}94024$, aproximadamente un 94 % das observacións ou o que é o mesmo a tasa de erro do modelo é aproximadamente un 6 % (Faraway, 2016). A fracción de pacientes que se prevé que non sexan diagnosticados cun tumor benigno é $\frac{343}{343+14} = 0{,}96078$, o que resulta un valor elevado. Este valor é o que se coñece como especificidad do test (Faraway, 2016). En contraste, a proporción que será diagnosticada dun tumor maligno é o que se coñece como sensibilidade (Faraway, 2016), é será $\frac{192}{20+192} = 0{,}90566$. Polo tanto vemos que é probable que o noso proceso preditivo detecte a presenza dun tumor maligno a partir das variables predictoras das que dispoñemos.

Unha forma alternativa para determinar se o noso modelo obtén as predicións correctas

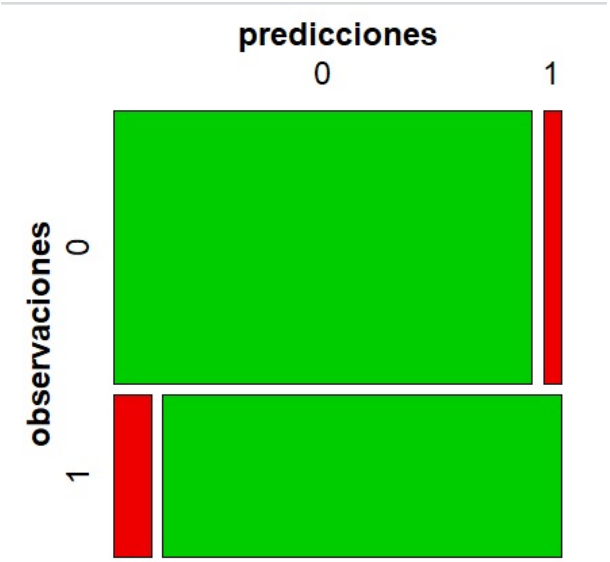


Figura 3.4: Matriz de confusión en forma de mapa de calor.

consiste en elaborar un gráfico que represente a probabilidade dun tumor predita polo modelo, $\hat{p}_i$, fronte aos valores observados. Se o modelo é acertado, este gráfico mostrará valores preto do $(0, 0)$ e do $(1, 1)$. Na Figura 3.5, podemos observar dita representación para o noso modelo en cuestión. Vemos que obtemos unha gran cantidade de valores preto do $(0, 0)$ e do $(1, 1)$, co que podemos concluir que o modelo indicado é correcto para predecir a nosa variable resposta.

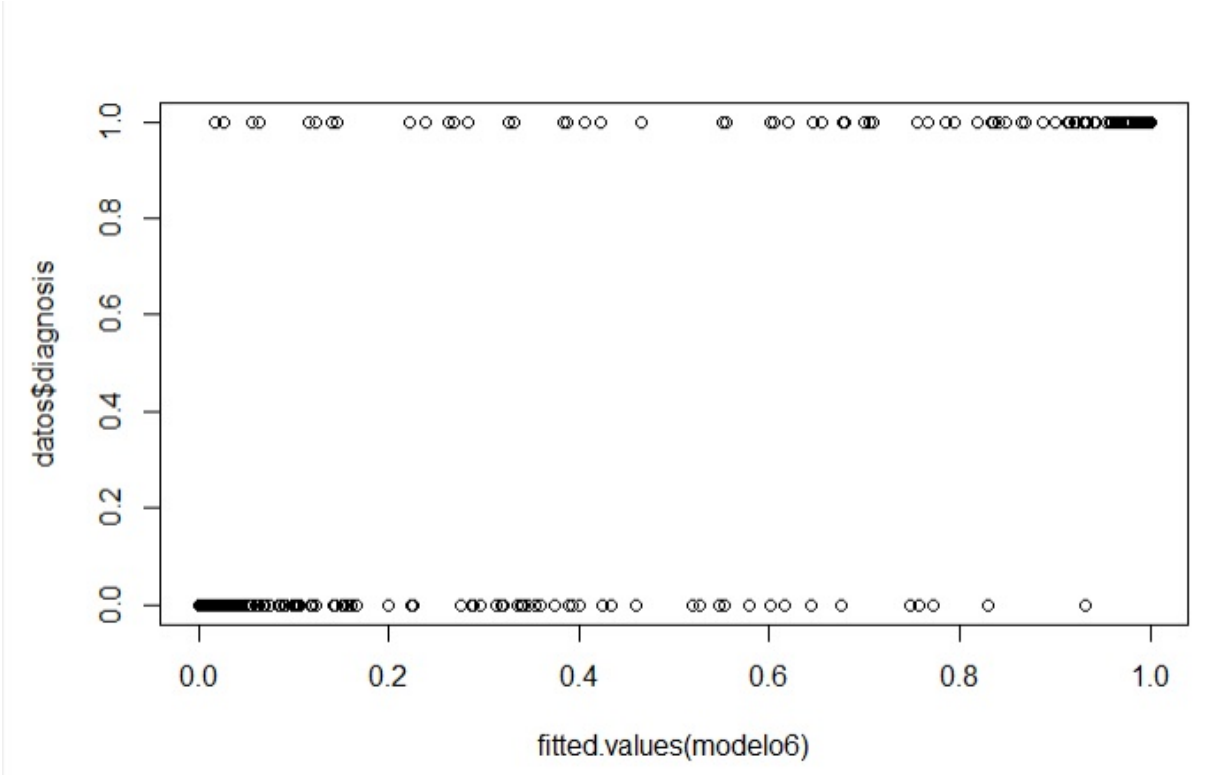


Figura 3.5: Probabilidades preditas fronte a valores observados polo modelo.

Capítulo 4

Aplicación do Modelo de Poisson

Neste capítulo, aplicaremos o modelo de Poisson a un conxunto de datos predeterminado. Para elo, empregaremos as cuestións teóricas explicadas no Capítulo 2, que nos permitirán analizar os nosos datos sobre a demanda de atención médica.

4.1. Demanda de atención médica

4.1.1. Descrición dos datos

Partimos dun conxunto de datos presente no paquete `'AER'` do *software* de R (R Core Team, 2021) e en concreto procedentes da librería `'MASS'`.

A base de datos recolle información transversal procedente da enquisa de saúde australiana dos anos 1977 e 1978. En concreto, o marco de datos contén 5190 observacións sobre 12 variables. Para o noso estudo, tomaremos como variable resposta a variable *visits*, que fai referencia ao número de visitas ao doctor nas últimas dúas semanas. O obxetivo do noso modelo será modelar esta demanda de atención médica en función das seguintes variables explicativas:

- Idade (*age*): idade en anos dividida entre 100.
- Ingresos (*income*): ingresos anuais en decenas de miles de dólares.

Con estas variables, que describen particularidades de cada paciente, o obxetivo é plantexar un modelo para explicar a demanda de atención médica en función das nosas variables explicativas. Dado que a variable resposta é discreta e ademais fai referencia a un número de feitos que ocorreron nun tempo determinado, podemos plantexar un modelo de regresión de Poisson para modelar dita resposta.

4.1.2. Aplicación do modelo de Poisson.

Escritura e selección do modelo.

Como indicamos no Capítulo 2, o modelo de Poisson é un dos modelos que permiten modelar datos de conteo. Para aplicarlle este modelo a nosa resposta, esta debe ser discreta e seguir unha distribución de Poisson. A propiedade principal desta distribución é que a media e a varianza coinciden. Ademais, ao contrario que para a regresión loxística, a resposta para a regresión de Poisson non se limita aos valores $\{0, 1\}$, senón que pode tomar calquer valor enteiro positivo.

Observando a variable dependente, *visits*, vemos que segue aparentemente unha distribución de Poisson como podemos observar na Figura 4.1, cuxa forma se asemella bastante as primeiras gráficas presentes na Figura 2.1. Desta forma, podemos plantexar un modelo de Poisson que

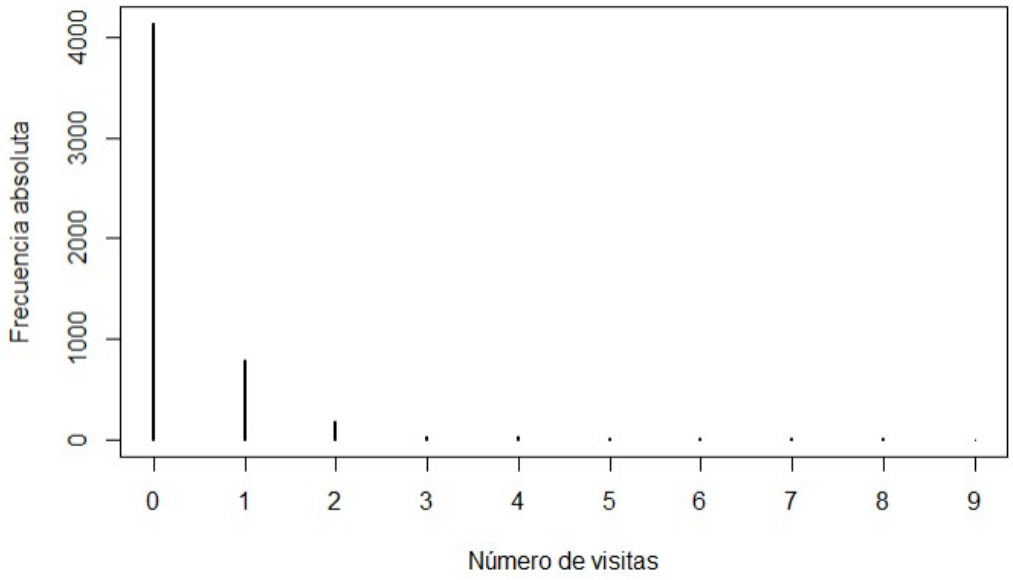


Figura 4.1: Gráfico de barras coa frecuencia absoluta da demanda de atención médica.

pretenda explicar o número de visitas ao consultorio médico en función das variables explicativas indicadas. Polo que escribimos o modelo formado pola resposta *visits* e as variables explicativas *age* e *income* e facemos un `summary` do mesmo.

```
Call:
glm(formula = visits ~ age + income, family = poisson, data = DoctorVisits)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0203  -0.7986  -0.6691  -0.6176   6.3802

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.60661    0.08596 -18.690  < 2e-16 ***
age          1.35292    0.12666  10.681  < 2e-16 ***
income      -0.34165    0.07770  -4.397  1.1e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 5634.8  on 5189  degrees of freedom
Residual deviance: 5449.9  on 5187  degrees of freedom
AIC: 7787.5

Number of Fisher Scoring iterations: 6
```

Comezamos igual que no Capítulo 3, ollando os p-valores obtidos para cada unha das nosas variables. De novo, a interpretación destes valores é bastante semellante a do modelo lineal e a do modelo de regresión loxística. Se este p-valor é baixo indicaranos que a variable asociada é significativa para modelar a resposta. Neste caso particular, vemos que todas as nosas variables resultan ser significativas, para un nivel de significación $\alpha = 0{,}05$. Polo que todas elas son de utilidade para explicar a demanda de atención médica.

Se en lugar de observar a significación de cada coeficiente o que queremos é construír unha medida global para a selección do modelo, podemos empregar o criterio de información de Akaike, AIC. Para isto, simplemente debemos comparar os valores de AIC asociados a cada un dos modelos plantexados e quedarnos con aquel que presente un valor de AIC menor, xa que ese será o modelo máis verosímil. Para isto podemos ter en conta o modelo plantexado anteriormente e composto pola variable resposta visitas e as varibles explicativas idade e ingresos, outro modelo formado só pola variable resposta e como explicativa a variable idade e un último modelo composto pola resposta e a variable explicativa ingresos.

```
Call:
glm(formula = visits ~ age, family = poisson, data = DoctorVisits)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9650  -0.8266  -0.6552  -0.6402   6.5608

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.87944    0.06248  -30.08   <2e-16 ***
age          1.54878    0.12060   12.84   <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: 5634.8  on 5189  degrees of freedom
Residual deviance: 5470.0  on 5188  degrees of freedom
AIC: 7805.6

Number of Fisher Scoring iterations: 6


Call:
glm(formula = visits ~ income, family = poisson, data = DoctorVisits)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9147  -0.8235  -0.7526  -0.6193   6.7789

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.87152    0.04567 -19.082  < 2e-16 ***
income      -0.59991    0.07486  -8.014 1.11e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 5634.8  on 5189  degrees of freedom
Residual deviance: 5566.5  on 5188  degrees of freedom
```

```
AIC: 7902

Number of Fisher Scoring iterations: 6
```

Agora, simplemente ollando no `summary` de cada modelo o valor do AIC obtido, temos que o modelo máis plausible tendo en conta o criterio do AIC é o composto pola resposta e ambas variables explicativas, idade e ingresos. Dado que o AIC asociado a este modelo é 7787,5, mentres que o asociado aos outros modelos indicados é 7805,6 e 7902 respectivamente.

Interpretación dos coeficientes.

Tal e como explicamos na Sección 2.1, o modelo de Poisson emprega unha función enlace logarítmica. Deste xeito poderemos expresar a media da nosa resposta como: $\log(\mu_i) = \eta_i = \boldsymbol{x_i'\beta}$ ou equivalentemente $\mu_i = \exp(\boldsymbol{x_i'\beta})$. Como consecuencia destas expresións descritas, poderemos interpretar os nosos parámetros da forma: se certa variable explicativa x_j aumenta en n unidades e as demais permanecen fixas, a media para a variable de Poisson multiplicase pola potencia n-ésima de $exp(\beta_j)$. Para o caso que nos ocupa, a estimación do vector de coeficientes asociado ao noso modelo vén dado pola seguinte expresión:

$$\hat{\boldsymbol{\beta}} = (-1{,}60, 1{,}35, -0{,}34), \tag{4.1}$$

onde a primeira entrada do vector $\hat{\boldsymbol{\beta}}$, corresponde ao coeficiente asociado ao intercepto e as demais aos coeficientes asociados as respectivas variables explicativas, *age* e *income*. Agora, partindo dos coeficientes obtidos para o noso modelo (4.1) e tendo en conta a función enlace empregada sábese que a interpretación da expoñencial destos coeficientes resulta máis interesante.

$$\mathbf{exp}\left(\hat{\boldsymbol{\beta}}\right) = (0{,}20, 3{,}86, 0{,}71), \tag{4.2}$$

con isto, podemos interpretar que o aumento dunha unidade da variable idade, permanecendo as restantes fixas, implicará que o número medio de visitas ao consultorio se multiplique por 3,86. Por outra banda, no caso da variable *income*, referente ao ingreso familiar anual, o seu aumento diminuirá o número de visitas ao consultorio. Máis concretamente, un aumento unitario da variable ingresos, mantendo as demais variables explicativas constantes, significa que o número de visitas ao consultorio médico se multiplican por 0,71.

En conclusión, teremos que se o valor da expoñencial do coeficiente é maior que 1, dita variable aumentará o valor da media da nosa resposta, como é o caso da variable *age*, mentres que se o valor da expoñencial do coeficiente é menor que 1, como no caso da variable *income*, diminuirá o valor da mesma. No caso de que fose xustamente 1, a variable non terá ningunha influencia para o modelaxe da nosa resposta.

4.1.3. Inferencia sobre o modelo.

Contraste do modelo mediante deviance.

Análogamente ao exposto na Sección 3.1.3, para obter o mellor axuste para o modelo, podemos realizar un test de razón de verosimilitudes. Este consiste en comparar a verosimilitude dun certo parámetro baixo unha condición H_0 coa verosimilitude con outra condición H_1. A hipótese nula, H_0, indica que o parámetro asociado a unha determinada variable é nulo, mentres que a alternativa, H_1, recolle xusto o contrario.

Neste caso particular, dado que todas as variables explicativas resultan significativas, en lugar de comparar modelos aniñados ollando o estadístico, como enunciamos na Sección 2.3, podemos calcular o contraste dun xeito máis compacto e sen especificar os diferentes modelos. Para iso, empregaremos a función `anova` presente na libraría `car` do *software* de R (R Core Team, 2021). Esta función realiza os contrastes de razón de verosimilitudes sobre os parámetros asociados a cada unha das variables do modelo sen necesidade de especificar os distintos modelos.

```
Analysis of Deviance Table (Type II tests)

Response: visits
       LR Chisq Df Pr(>Chisq)
age     116.542  1  < 2.2e-16 ***
income   20.053  1  7.533e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Deste contraste podemos afirmar o que xa intuíamos, que os coeficientes relativos tanto a variable idade como a variable ingresos son significativamente distintos de cero. Polo que todas as variables indicadas resultan significativas para modelar a demanda de atención médica.

Bondade de axuste.

Como explicamos no Capítulo 2, podemos testar a bondade de axuste do modelo realizando un contraste. No cal a hipótese nula, H_0, é que o modelo de regresión de Poisson é correcto, fronte a hipótese alternativa, H_1, que non o sexa. Para levar a cabo tal contraste, compre calcular o estadísitico de Pearson, dado pola expresión (2.21) e que como sabemos a súa distribución pódese aproximar por unha ji-cadrado. Deste xeito, no *software* de R (R Core Team, 2021), poderemos obter o p-valor asociado ao estadístico de Pearson. Este p-valor é igual a cero. Desta forma, para un nivel de significación $\alpha = 0{,}05$, podemos rexeitar a hipótese nula de que o modelo imposto

é o correcto para explicar a nosa resposta. Polo que o modelo de regresión de Poisson non é o adecuado para modelar a demanda de atención médica.

De forma alternativa, podemos realizar un test basado na *deviance* que compare o modelo axustado e o modelo saturado. O estadístico obtido mediante o *software* de R (R Core Team, 2021), para dito test é 0,005453847, polo que para un nivel de significación $\alpha = 0{,}05$, podemos rexeitar a hipótese nula de que o modelo axustado é o correcto. Ademais, en base a *deviance* se recorremos a expresión $D^2 = \frac{NullDeviance - ResidualDeviance}{NullDeviance}$, para o noso caso particular teremos $D^2 = \frac{5634{,}8 - 5449{,}9}{5634{,}8} = 0{,}033$. Os valores de D^2 comprendidos entre 0,2 e 0,4, indican segundo McFadden (Hensher e Stopher, 2021), un excelente axuste para o modelo. Para o noso caso concreto, vemos que o axuste está lonxe de ser o idóneo.

Intervalos de confianza para os parámetros do modelo.

Análogamente ao Capítulo 3, escribimos o intervalo de confianza de Wald para os nosos parámetros β_j. Estes de novo, calcúlanse coa función `confint.default` do *software* de R (R Core Team, 2021), que por defecto toma un nivel de confianza $1 - \alpha = 0{,}95$. Deste xeito, obtemos os seguintes intervalos de confianza ao 95 %:

```
                2.5 %     97.5 %
(Intercept) -1.7750869 -1.4381348
age          1.1046702  1.6011783
income      -0.4939319 -0.1893686
```

Podemos observar que ningún dos intervalos de confianza calculados inclúe o cero. Polo tanto, no noso modelo en particular, ningún dos coeficientes se pode asumir como igual a cero, o que quere dicir que todas as variables explicativas indicadas son de utilidade para modelar a nosa resposta.

4.1.4. Diagnose sobre o modelo.

Análogamente ao modelo de regresión loxística, para levar a cabo unha diagnose do modelo de Poisson, podemos empregar os residuos. O valor destes, determinará que tan bos axustes temos para o noso modelo.

Para o cálculo dos residuos e os residuos estandarizados, igual que no Capítulo 3, recorremos ao *software* de R (R Core Team, 2021), en concreto as funcións `residuals` e `rstandard`. Unha vez calculados estes residuos e de novo considerando como criterio para que certo dato sexa atípico aqueles residuos de valor absoluto maior que 2, vemos que para ámbolos dous, os datos atípicos

supoñen aproximadamente un 5 % do total. Sen embargo, dado que no modelo de Poisson, os residuos non presentan a mesma varianza emprégase a *deviance* residual. Polo que, pode resultar de interese calcular os residuos da *deviance*, dados pola expresión (2.24). Os residuos da *deviance* e residuos da *deviance* estandarizados obtéñense mediante as funcións citadas anteriormente pero especificando en tipo que se trata de residuos da *deviance*. Tomando de novo como atípicos aqueles valores, cun residuo en valor absoluto maior que 2, teremos que aproximadamente o 3 % dos residuos cumplen este criterio.

A continuación, coa intención de analizar gráficamente os residuos calculados, igual que para o modelo de regresión loxística, recorremos ao paquete `'DHARMa'` (Hartig, 2022). Este paquete permitirános interpretar dunha forma máis simple os residuos dun modelo con resposta discreta, como é o caso da regresión de Poisson. Observando a Figura 4.2, análogamente ao Capítulo

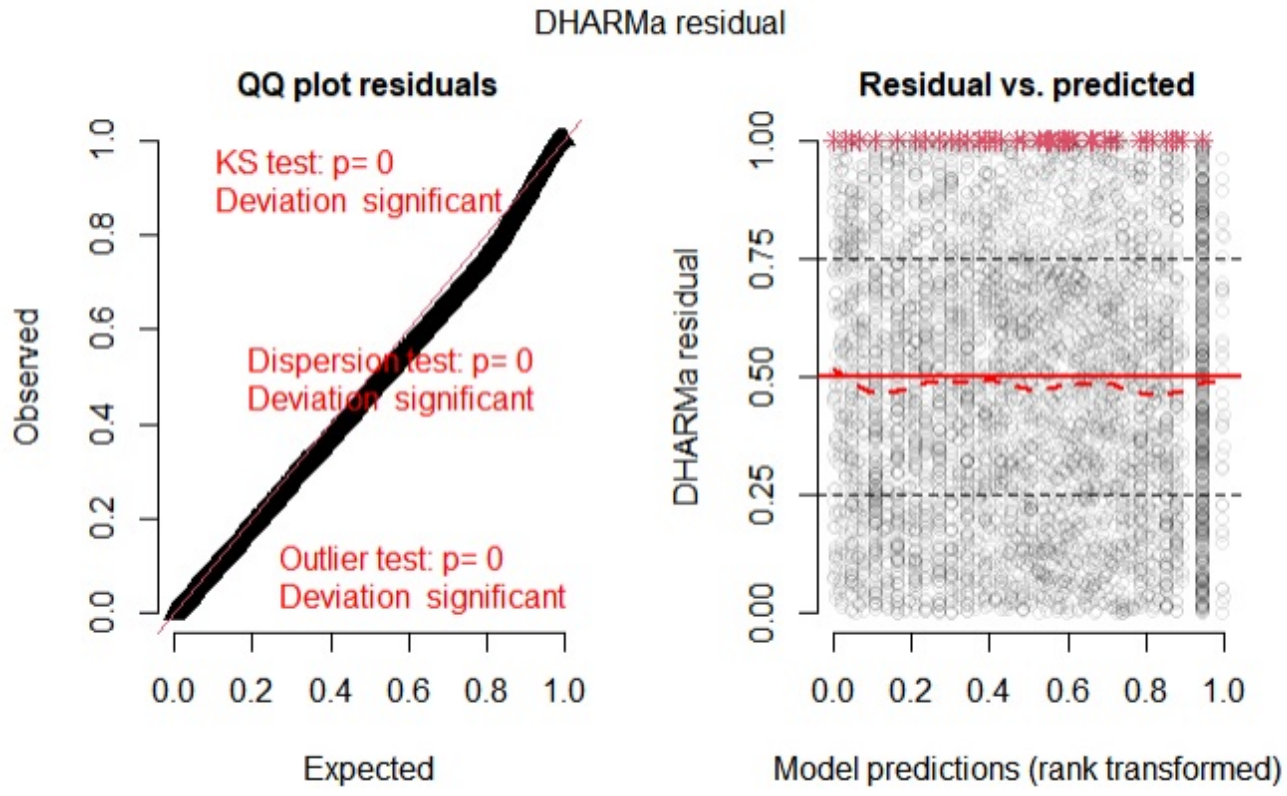


Figura 4.2: Gráficos de residuos empregando o paquete `'DHARMa'`.

3, na gráfica da esquerda vemos un gráfico de cuantil a cuantil xunto con certas probas para problemas de distribución específicos. Neste caso vemos que os residuos calculados non se axustan correctamente a distribución esperada. Dado que segundo o test de Kolmogorov-Smirnov (KS), podemos rexeitar a hipótese nula de que os datos seguen a distribución esperada, xa que o p-

valor asociado a este test é nulo. Por outra banda, ollando o gráfico da dereita, que representa os residuos fronte aos valores axustados, vemos resaltados en vermello e con asteriscos os valores atípicos. Ademais, o *software* de R, en concreto o paquete `'DHARMa'` (Hartig, 2022), resalta en vermello a desviación da regresión cuantílica axustada con respecto a expectativa teórica. Polo que, de novo, poderíamos afirmar que para o noso caso particular, os residuos indicados non se axustan a distribución esperada, é dicir, o noso modelo semella presentar algún problema de sobredispersión ou subdispersión. Na sección seguinte centrarémonos no estudo da dispersión do modelo de Poisson dunha forma máis concreta.

4.1.5. Sobredispersión do modelo.

Recordemos que a sobredispersión do modelo de Poisson ocorre cando a media e a varianza do mesmo non coinciden, é dicir non se cumple a característica principal deste tipo de modelo. Para verificar se esto ocorre ou non para o noso conxunto de datos, empregaremos o paquete `AER` do *software* de R (R Core Team, 2021), en concreto a función `dispersiontest`. Esta función, permítenos realizar un contraste, onde a hipótese nula é que os residuos asociados ao modelo sexan equidispersos, fronte a alternativa de que presente sobredispersión ou subdispersión. Observando a execución deste contraste, vemos que o p-valor asociado ao contraste é moi pequeno, en concreto é $9{,}312e-16$. Polo tanto, para un nivel de significación $\alpha = 0{,}05$, podemos rexeitar a hipótese. Neste caso teremos polo tanto que o noso modelo de Poisson non é equidisperso, senón que é sobredisperso, polo que unha boa alternativa para este conxunto de datos sería unha regresión binomial negativa.

4.1.6. Conclusión

Como sabemos, unha das características principais do modelo de Poisson é a igualdade entre a media e a varianza. Sen embargo, o noso exemplo en concreto presenta sobredispersión o que significa que tal condición non se verifica. Debido a isto, podemos pensar que o modelo de Poisson non será o mellor para explicar a nosa resposta da demanda de atención médica. Unha boa alternativa ao modelo de Poisson e que pode funcionar para os nosos datos sería unha regresión binomial negativa.

A regresión binomial negativa, igual que a regresión de Poisson, é un tipo de modelo de regresión útil para modelar unha resposta que representa un conteo discreto. Se a varianza é aproximadamente igual a media, teremos que en xeral un modelo de regresión de Poisson se axustará ben aos datos. Sen embargo, se a varianza é significativamente maior ca media, entón o modelo de regresión binomial negativa resultará unha boa alternativa. Desta forma, a distribución binomial negativa ten dous parámetros, μ, que é o valor medio ou esperado da distribución e k

que é o parámetro de sobredispersión. Cando $k = 0$, a distribución binomial negativa é a mesma que a distribución de Poisson. Poderemos atopar máis información sobre este modelo en (Hilbe, 2014).

Por último, na práctica, para implementar no *software* de R un modelo de regresión binomial negativa, recorremos ao paquete `'MASS'`, en concreto a función `glm.nb`, para máis detalles teóricos e sobre a súa implementación acudir a (Venables e Ripley, 2002) e (Cameron e Trivedi, 1998).

```
Call:
glm.nb(formula = visits ~ age + income, data = DoctorVisits,
    init.theta = 0.4239459284, link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8230  -0.6932  -0.5995  -0.5617   3.6708

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.62405    0.10929 -14.860  < 2e-16 ***
age          1.37043    0.16723   8.195  2.5e-16 ***
income      -0.32376    0.09872  -3.280  0.00104 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Negative Binomial(0.4239) family taken to be 1)

    Null deviance: 3099.2  on 5189  degrees of freedom
Residual deviance: 2992.9  on 5187  degrees of freedom
AIC: 7076.4

Number of Fisher Scoring iterations: 1


              Theta:  0.4239
          Std. Err.:  0.0309

 2 x log-likelihood:  -7068.3970
```

Finalmente, vemos que neste modelo, todas as variables resultan significativas para un nivel

de significación $\alpha = 0{,}05$. Ademais, se o comparamos co modelo de Poisson indicado ao comezo do Capítulo 4, vemos que a estimación dos coeficientes é moi similar. Así mesmo, o modelo de regresión binomial negativo, ten un valor de AIC menor, 7076,4, fronte a 7787,5 que tiña o modelo de regresión de Poisson. Polo que dito modelo de regresión binomial negativa pode ser máis adecuado para modelar a demanda de atención médica.

Bibliografía

Agresti, A. G. (1990). *Categorical data analysis.* New York: John Wiley and Sons.

Cameron, A. C., e Trivedi, P. K. (1998). *Regression analysis of count data.* Cambridge: Cambridge University Press.

Cepeda-Cuervo, E., Aguilar, W., Cervantes, V., Corrales, M., Díaz, I., e Rodríguez, D. (2008). Intervalos de confianza e intervalos de credibilidad para una proporción. *Revista Colombiana de Estadística*, *31*(2), 211-228.

Colaboradores de Wikipedia. (2022). *Información de Fisher — Wikipedia, la enciclopedia libre.* `https://en.wikipedia.org/w/index.php?title=Fisher_information&oldid=1079985740`. ([En línea; consultado el 2 de mayo de 2022])

Dua, D., e Graff, C. (2017). *UCI Machine Learning Repository.* `http://archive.ics.uci.edu/ml`. ([En línea; consultado el 12 de mayo de 2022])

Faraway, J. J. (2016). *Extending the linear model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models.* Boca Raton: CRC Press.

Fox, J. (2008). *Applied regression analysis and generalized linear models* (2nd ed.). Los Angeles: Sage.

Hartig, F. (2022). *DHARMa: Residual Diagnostics for Hierarchical (Multi-Level / Mixed) Regression Models* [Manual]. Obtido de:`https://CRAN.R-project.org/package=DHARMa` (R package version 0.4.5)

Hastie, T. J., Tibshirani, R. J., e Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction* (2nd ed ed.). New York: Springer.

Hensher, D., e Stopher, P. (2021). *Behavioural travel modelling.* Taylor & Francis. Obtido de:`https://books.google.com.do/books?id=Z_UlEAAAQBAJ`

Hilbe, J. M. (2014). *Modeling count data.* New York: Cambridge University Press.

McCullagh, P., e Nelder, J. A. (1989). *Generalized linear models* (2nd ed.). London: Chapman and Hall.

R Core Team. (2021). *R: A Language and Environment for Statistical Computing* [Manual]. Vienna, Austria. Obtido de:`https://www.R-project.org/`

Rivera, J. I. y. G. (n.d.). *Chapter 8 regresión de poisson | modelos lineales generaliza-*

dos con r. Obtido de:https://bookdown.org/jaimeisaacp/bookglm/regresi%C3%B3n-de-poisson.html

Venables, W., e Ripley, B. (2002). *Modern applied statistics with S* (4th ed.). Berlin: Springer.

Anexos

O análise de datos desenrolado ao longo dos Capítulos 3 e 4, realizouse mediante o *software* estatístico de R. Para maior precisión, adxunto o código empregado para replicar todos os resultados expostos.

```
#Lectura dos datos
library(readxl)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

datos <- read.csv("data.csv")
attach(datos)

#Recodificación da variable diagnosis
levels(datos$diagnosis) <- c("0", "1")
levels(datos$diagnosis)

## [1] "0" "1"

datos$diagnosis<- factor(datos$diagnosis)
table(datos$diagnosis)

##
##   0   1
## 357 212

#Escribimos o modelo

modelo<-glm(diagnosis~radius_mean+texture_mean+perimeter_mean+area_mean+
smoothness_mean+compactness_mean+concavity_mean+concave.points_mean+
```

```
symmetry_mean+fractal_dimension_mean,data=datos,family=binomial)

summary(modelo)

##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + perimeter_mean +
##     area_mean + smoothness_mean + compactness_mean + concavity_mean +
##     concave.points_mean + symmetry_mean + fractal_dimension_mean,
##     family = binomial, data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.95590  -0.14839  -0.03943   0.00429   2.91690
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -7.35952   12.85259  -0.573   0.5669
## radius_mean              -2.04930    3.71588  -0.551   0.5813
## texture_mean              0.38473    0.06454   5.961  2.5e-09 ***
## perimeter_mean           -0.07151    0.50516  -0.142   0.8874
## area_mean                 0.03980    0.01674   2.377   0.0174 *
## smoothness_mean          76.43227   31.95492   2.392   0.0168 *
## compactness_mean         -1.46242   20.34249  -0.072   0.9427
## concavity_mean            8.46870    8.12003   1.043   0.2970
## concave.points_mean      66.82176   28.52910   2.342   0.0192 *
## symmetry_mean            16.27824   10.63059   1.531   0.1257
## fractal_dimension_mean -68.33703   85.55666  -0.799   0.4244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 146.13  on 558  degrees of freedom
## AIC: 168.13
##
## Number of Fisher Scoring iterations: 9
```

```r
#Selección do modelo (método backware)

modelo1<-glm(diagnosis~radius_mean+texture_mean+perimeter_mean+
area_mean+smoothness_mean+concavity_mean+
concave.points_mean+symmetry_mean+fractal_dimension_mean,
data=datos,family=binomial)
summary(modelo1)

##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + perimeter_mean +
##     area_mean + smoothness_mean + concavity_mean + concave.points_mean +
##     symmetry_mean + fractal_dimension_mean, family = binomial,
##     data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.97463  -0.14756  -0.03947   0.00428   2.92261
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -7.04893   12.08068  -0.583   0.5596
## radius_mean              -1.89732    3.05194  -0.622   0.5342
## texture_mean              0.38463    0.06455   5.959 2.54e-09 ***
## perimeter_mean           -0.09813    0.34325  -0.286   0.7750
## area_mean                 0.03999    0.01649   2.424   0.0153 *
## smoothness_mean          76.08009   31.50588   2.415   0.0157 *
## concavity_mean            8.48311    8.10558   1.047   0.2953
## concave.points_mean      66.77285   28.49046   2.344   0.0191 *
## symmetry_mean            16.16553   10.51539   1.537   0.1242
## fractal_dimension_mean -71.99700   68.75680  -1.047   0.2950
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 146.14  on 559  degrees of freedom
```

```
## AIC: 166.14
##
## Number of Fisher Scoring iterations: 9

modelo2<-glm(diagnosis~radius_mean+texture_mean+area_mean+
smoothness_mean+concavity_mean+concave.points_mean+symmetry_mean+
fractal_dimension_mean,data=datos,family=binomial)
summary(modelo2)

##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + area_mean +
##     smoothness_mean + concavity_mean + concave.points_mean +
##     symmetry_mean + fractal_dimension_mean, family = binomial,
##     data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.96847  -0.15195  -0.04024   0.00409   2.93549
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -5.27847   10.31074  -0.512  0.60869
## radius_mean              -2.68473    1.32326  -2.029  0.04247 *
## texture_mean              0.38262    0.06413   5.966 2.42e-09 ***
## area_mean                 0.04157    0.01554   2.675  0.00747 **
## smoothness_mean          78.22119   30.57445   2.558  0.01052 *
## concavity_mean            8.25689    8.04476   1.026  0.30472
## concave.points_mean      64.07659   26.75842   2.395  0.01664 *
## symmetry_mean            16.02120   10.47671   1.529  0.12621
## fractal_dimension_mean -82.21451   58.85970  -1.397  0.16248
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 146.22  on 560  degrees of freedom
```

```
## AIC: 164.22
##
## Number of Fisher Scoring iterations: 9

modelo3<-glm(diagnosis ~ radius_mean+texture_mean+area_mean+
smoothness_mean+concave.points_mean+
symmetry_mean+fractal_dimension_mean,
data=datos,family=binomial)
summary(modelo3)

##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + area_mean +
##     smoothness_mean + concave.points_mean + symmetry_mean + fractal_dimension_mean,
##     family = binomial, data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.00824  -0.15647  -0.04323   0.00326   2.83457
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -3.14334    9.86273  -0.319  0.74995
## radius_mean              -3.06595    1.24925  -2.454  0.01412 *
## texture_mean              0.38122    0.06374   5.981 2.22e-09 ***
## area_mean                 0.04562    0.01485   3.072  0.00212 **
## smoothness_mean          60.39599   23.93789   2.523  0.01163 *
## concave.points_mean      83.46572   19.16188   4.356 1.33e-05 ***
## symmetry_mean            17.04684   10.23216   1.666  0.09571 .
## fractal_dimension_mean -49.23528   47.54527  -1.036  0.30041
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 147.28  on 561  degrees of freedom
## AIC: 163.28
```

```
##
## Number of Fisher Scoring iterations: 9


modelo4<-glm(diagnosis~radius_mean+texture_mean+area_mean+
smoothness_mean+concave.points_mean+symmetry_mean,
data=datos,family=binomial)
summary(modelo4)


##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + area_mean +
##     smoothness_mean + concave.points_mean + symmetry_mean, family = binomial,
##     data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.94562  -0.15248  -0.04346   0.00366   2.89274
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -8.61085    8.33550  -1.033  0.30159
## radius_mean          -2.72515    1.17554  -2.318  0.02044 *
## texture_mean          0.38522    0.06430   5.991 2.09e-09 ***
## area_mean             0.04308    0.01428   3.017  0.00255 **
## smoothness_mean      58.37855   23.49622   2.485  0.01297 *
## concave.points_mean 73.70154   16.21489   4.545 5.49e-06 ***
## symmetry_mean        15.56212   10.25705   1.517  0.12921
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 148.39  on 562  degrees of freedom
## AIC: 162.39
##
## Number of Fisher Scoring iterations: 9
```

```r
modelo5<-glm(diagnosis~radius_mean+texture_mean+area_mean+
smoothness_mean+concave.points_mean,data=datos,family=binomial)
summary(modelo5)
```

```
##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + area_mean +
##     smoothness_mean + concave.points_mean, family = binomial,
##     data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.94769  -0.16164  -0.04965   0.00378   2.77371
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -4.64306    7.96274  -0.583  0.55983
## radius_mean         -2.88770    1.17018  -2.468  0.01360 *
## texture_mean         0.37317    0.06258   5.963 2.47e-09 ***
## area_mean            0.04400    0.01422   3.094  0.00197 **
## smoothness_mean     63.18020   23.87993   2.646  0.00815 **
## concave.points_mean 81.11419   15.91206   5.098 3.44e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 150.68  on 563  degrees of freedom
## AIC: 162.68
##
## Number of Fisher Scoring iterations: 9
```

```r
#Medida global de significación:Selección do modelo, criterio de AIC.
drop1(modelo,test="Chi")
```

```
## Single term deletions
```

```
##
## Model:
## diagnosis ~ radius_mean + texture_mean + perimeter_mean + area_mean +
##     smoothness_mean + compactness_mean + concavity_mean + concave.points_mean +
##     symmetry_mean + fractal_dimension_mean
##                         Df Deviance    AIC    LRT Pr(>Chi)
## <none>                       146.13 168.13
## radius_mean              1   146.44 166.44  0.306  0.57992
## texture_mean             1   195.34 215.34 49.209  2.3e-12 ***
## perimeter_mean           1   146.15 166.15  0.020  0.88749
## area_mean                1   151.63 171.63  5.496  0.01906 *
## smoothness_mean          1   152.42 172.42  6.289  0.01215 *
## compactness_mean         1   146.14 166.14  0.005  0.94270
## concavity_mean           1   147.23 167.23  1.103  0.29370
## concave.points_mean      1   151.93 171.93  5.795  0.01607 *
## symmetry_mean            1   148.44 168.44  2.307  0.12878
## fractal_dimension_mean   1   146.78 166.78  0.647  0.42136
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

drop1(modelo1,test="Chi")

## Single term deletions
##
## Model:
## diagnosis ~ radius_mean + texture_mean + perimeter_mean + area_mean +
##     smoothness_mean + concavity_mean + concave.points_mean +
##     symmetry_mean + fractal_dimension_mean
##                         Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                       146.14 166.14
## radius_mean              1   146.52 164.52  0.386   0.53417
## texture_mean             1   195.48 213.48 49.341 2.151e-12 ***
## perimeter_mean           1   146.22 164.22  0.082   0.77483
## area_mean                1   152.00 170.00  5.864   0.01546 *
## smoothness_mean          1   152.45 170.45  6.316   0.01197 *
## concavity_mean           1   147.25 165.25  1.111   0.29191
## concave.points_mean      1   151.93 169.93  5.798   0.01604 *
## symmetry_mean            1   148.46 166.46  2.328   0.12711
```

```
## fractal_dimension_mean  1   147.27 165.27  1.138   0.28613
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(modelo2,test="Chi")
```

```
## Single term deletions
##
## Model:
## diagnosis ~ radius_mean + texture_mean + area_mean + smoothness_mean +
##     concavity_mean + concave.points_mean + symmetry_mean + fractal_dimension_mean
##                         Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                       146.22 164.22
## radius_mean              1   150.19 166.19  3.971  0.046293 *
## texture_mean             1   195.93 211.93 49.713 1.779e-12 ***
## area_mean                1   153.35 169.35  7.131  0.007576 **
## smoothness_mean          1   153.39 169.39  7.169  0.007419 **
## concavity_mean           1   147.28 163.28  1.062  0.302661
## concave.points_mean      1   152.19 168.19  5.971  0.014546 *
## symmetry_mean            1   148.52 164.52  2.302  0.129190
## fractal_dimension_mean   1   148.30 164.30  2.082  0.149027
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(modelo3,test="Chi")
```

```
## Single term deletions
##
## Model:
## diagnosis ~ radius_mean + texture_mean + area_mean + smoothness_mean +
##     concave.points_mean + symmetry_mean + fractal_dimension_mean
##                         Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                       147.28 163.28
## radius_mean              1   153.04 167.04  5.765  0.016347 *
## texture_mean             1   197.40 211.40 50.115 1.450e-12 ***
## area_mean                1   156.64 170.64  9.359  0.002219 **
## smoothness_mean          1   153.61 167.61  6.331  0.011862 *
```

```
## concave.points_mean     1   170.51 184.51 23.233 1.435e-06 ***
## symmetry_mean           1   150.03 164.03  2.750  0.097259 .
## fractal_dimension_mean  1   148.38 162.38  1.105  0.293075
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(modelo4,test="Chi")
```

```
## Single term deletions
##
## Model:
## diagnosis ~ radius_mean + texture_mean + area_mean + smoothness_mean +
##     concave.points_mean + symmetry_mean
##                     Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                   148.38 162.38
## radius_mean          1   153.44 165.44  5.055  0.024558 *
## texture_mean         1   198.83 210.83 50.444 1.226e-12 ***
## area_mean            1   157.38 169.38  8.997  0.002705 **
## smoothness_mean      1   154.44 166.44  6.050  0.013904 *
## concave.points_mean  1   172.31 184.31 23.927 1.001e-06 ***
## symmetry_mean        1   150.68 162.68  2.290  0.130174
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(modelo5,test="Chi")
```

```
## Single term deletions
##
## Model:
## diagnosis ~ radius_mean + texture_mean + area_mean + smoothness_mean +
##     concave.points_mean
##                     Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                   150.68 162.68
## radius_mean          1   156.44 166.44  5.764  0.016355 *
## texture_mean         1   199.83 209.83 49.156 2.365e-12 ***
## area_mean            1   160.32 170.32  9.645  0.001899 **
## smoothness_mean      1   157.39 167.39  6.712  0.009577 **
```

```
## concave.points_mean  1   180.85 190.85 30.174 3.949e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#Interpretación dos coeficientes
options(width = 60)
(beta1<-coef(modelo4))

##         (Intercept)         radius_mean        texture_mean
##         -8.61084826         -2.72515193           0.38522221
##           area_mean     smoothness_mean concave.points_mean
##          0.04308238         58.37854510          73.70154094
##       symmetry_mean
##         15.56211857

exp(beta1)

##         (Intercept)         radius_mean        texture_mean
##        1.821194e-04        6.553625e-02        1.469941e+00
##           area_mean     smoothness_mean concave.points_mean
##        1.044024e+00        2.256732e+25        1.018996e+32
##       symmetry_mean
##        5.735116e+06

#INFERENCIA SOBRE O MODELO
#Contraste do modelo  mediante deviance
#Test de razón de verosimilitudes
modelo<-glm(diagnosis~radius_mean+texture_mean+perimeter_mean
+area_mean+smoothness_mean+compactness_mean+concavity_mean
+concave.points_mean+symmetry_mean+fractal_dimension_mean,
data=datos,family=binomial)
modelo4<-glm(diagnosis~radius_mean+texture_mean+area_mean
+smoothness_mean+concave.points_mean+symmetry_mean,
data=datos,family=binomial)
G2 <- modelo4$deviance - modelo$deviance
G2

## [1] 2.254874
```

```r
1 - pchisq(G2, df = 4)
```

```
## [1] 0.6889965
```

```r
#Contraste empregando función anova
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
Anova(modelo)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: diagnosis
##                         LR Chisq Df Pr(>Chisq)
## radius_mean                0.306  1    0.57992
## texture_mean              49.209  1    2.3e-12 ***
## perimeter_mean             0.020  1    0.88749
## area_mean                  5.496  1    0.01906 *
## smoothness_mean            6.289  1    0.01215 *
## compactness_mean           0.005  1    0.94270
## concavity_mean             1.103  1    0.29370
## concave.points_mean        5.795  1    0.01607 *
## symmetry_mean              2.307  1    0.12878
## fractal_dimension_mean     0.647  1    0.42136
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
modelo6<-glm(diagnosis~texture_mean+area_mean+smoothness_mean
+concave.points_mean,data=datos,family = binomial)
summary(modelo6)
```

```
##
## Call:
## glm(formula = diagnosis ~ texture_mean + area_mean + smoothness_mean +
##     concave.points_mean, family = binomial, data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.31798  -0.15623  -0.04212   0.01662   2.84201
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -23.677816   3.882774  -6.098 1.07e-09
## texture_mean          0.362687   0.060544   5.990 2.09e-09
## area_mean             0.010342   0.002002   5.165 2.40e-07
## smoothness_mean      59.471304  25.965153   2.290    0.022
## concave.points_mean  76.571210  16.427864   4.661 3.15e-06
##
## (Intercept)         ***
## texture_mean        ***
## area_mean           ***
## smoothness_mean     *
## concave.points_mean ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 751.44  on 568  degrees of freedom
## Residual deviance: 156.44  on 564  degrees of freedom
## AIC: 166.44
##
## Number of Fisher Scoring iterations: 8

#Intervalos de confianza para os parámetros do modelo
confint.default(modelo)

##                                 2.5 %       97.5 %
```

```
## (Intercept)             -32.55012751  17.83109229
## radius_mean              -9.33229662   5.23368682
## texture_mean              0.25824448   0.51122420
## perimeter_mean           -1.06161528   0.91859445
## area_mean                 0.00698718   0.07260522
## smoothness_mean          13.80178675 139.06276076
## compactness_mean        -41.33297960  38.40813510
## concavity_mean           -7.44627507  24.38367459
## concave.points_mean      10.90575011 122.73776358
## symmetry_mean            -4.55732264  37.11380728
## fractal_dimension_mean -236.02499401  99.35094023

#Diagnose sobre o modelo
#Cálculo de residuos de Pearson
res <- residuals(modelo6, type = "pearson")
head(res) #residuos de Pearson dos 6 primeiros pacientes

##           1           2           3           4           5
## 0.012625601 0.031857290 0.001676578 0.120231091 0.011750045
##           6
## 0.689101952

res.sig <- abs(res) > 2 #Residuos de Pearson significativos
table(res.sig)

## res.sig
## FALSE  TRUE
##   559    10

res.orde <- sort(abs(res[res.sig]), decreasing = TRUE) #mostramos os máis altos
head(res.orde)

##      298       41      136      172      129       74
## 7.466056 6.108532 4.130008 3.852646 3.698711 2.773438

#Probabilidades de diagnose de tumor maligno asociadas
fitted.values(modelo6)[298]
```

```
##         298
## 0.01762363

fitted.values(modelo6)[41]

##          41
## 0.02610001

fitted.values(modelo6)[136]

##         136
## 0.05538029

fitted.values(modelo6)[172]

##         172
## 0.06311982

fitted.values(modelo6)[129]

##       129
## 0.9318822

fitted.values(modelo6)[74]

##         74
## 0.1150489

#Cálculo de residuos estandarizados
res.std <- rstandard(modelo6, type = "pearson")
res.std.sig <- abs(res.std) > 2 #Residuos estandarizados significativos
table(res.std.sig)

## res.std.sig
## FALSE  TRUE
##   559    10
```

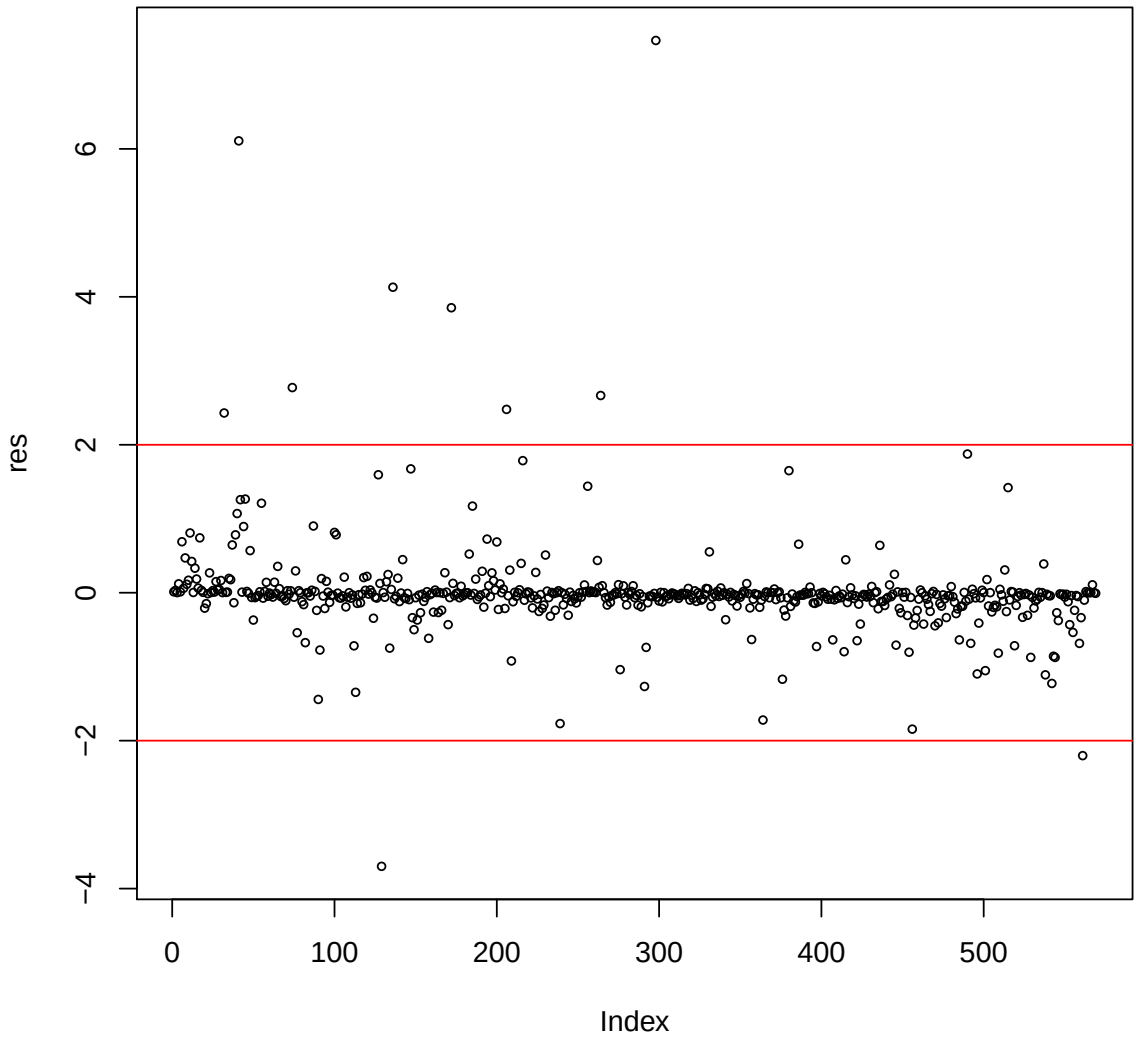
```r
head(res.std[res.std.sig])
```

```
##        32        41        74       129       136       172
##  2.448381  6.124732  2.789193 -3.728307  4.144604  3.865401
```

```r
res.ordeest <- sort(abs(res.std[res.std.sig]), decreasing = TRUE)
# mostramos só os máis altos
head(res.ordeest)
```

```
##      298       41      136      172      129       74
## 7.479692 6.124732 4.144604 3.865401 3.728307 2.789193
```

```r
#Gráficos para os residuos do modelo
plot(res, cex = 0.6)
abline(h = c(-2, 2), col = "red")
```




```
#Residuos con valor absoluto elevado
signif <- which(abs(res) > 2)
plot(res[signif], type = "n")
text(1:length(signif), res[signif], label = signif, cex = 0.4)

#Gráficos de residuos empregando DHARMa
#install.packages("DHARMa",repos = "http://cran.us.r-project.org")
#library(DHARMa)
#install.packages("glmmTMB",repos ="http://cran.us.r-project.org")
#library(glmmTMB)
```

```r
#resDhar <- simulateResiduals(modelo6)
#plot(resDhar)

#Observamos valores dos leverages
hat.valores <- hatvalues(modelo6)
head(hat.valores)
```

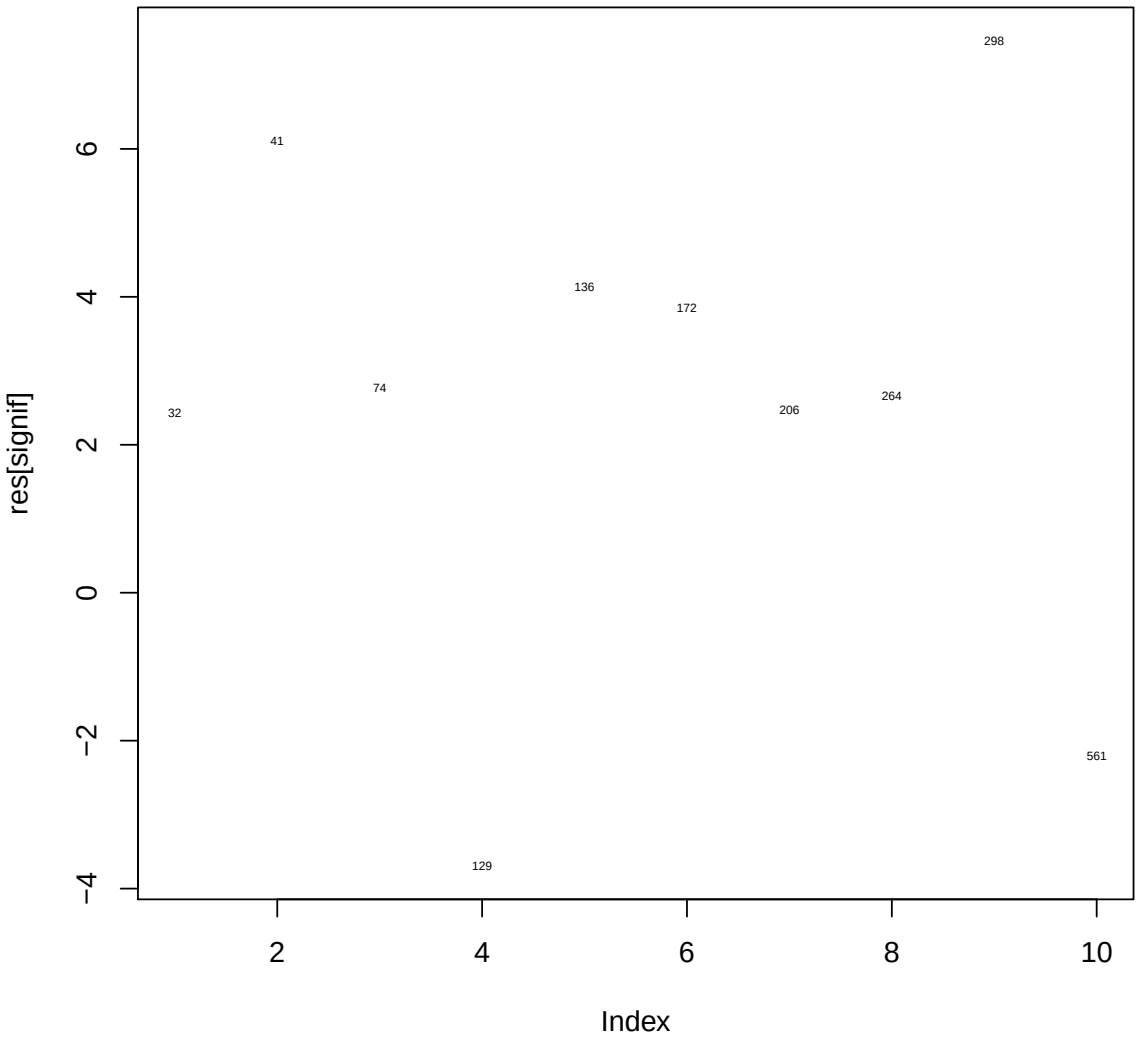
```
##            1            2            3            4
## 2.745494e-04 1.365718e-03 6.419484e-06 1.171571e-02
##            5            6
## 2.180533e-04 6.057120e-02
```

```r
#Conclusión
predicciones <- ifelse(test = modelo6$fitted.values > 0.5, yes = 1, no = 0)
matriz_confusion <- table(modelo6$model$diagnosis, predicciones,
                          dnn = c("observaciones", "predicciones"))
matriz_confusion
```

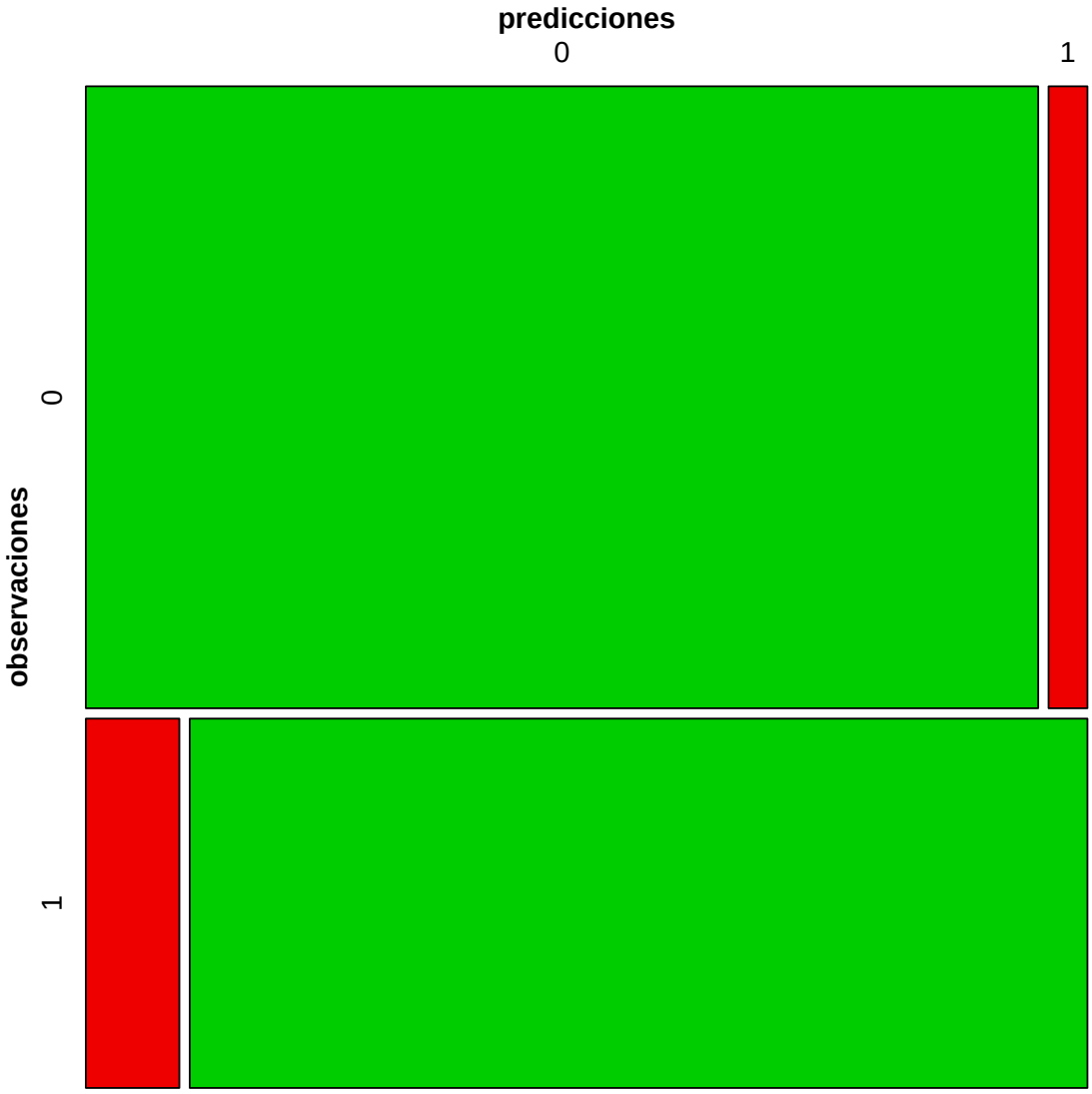
```
##              predicciones
## observaciones   0   1
##             0 343  14
##             1  20 192
```

```r
library(vcd)
```

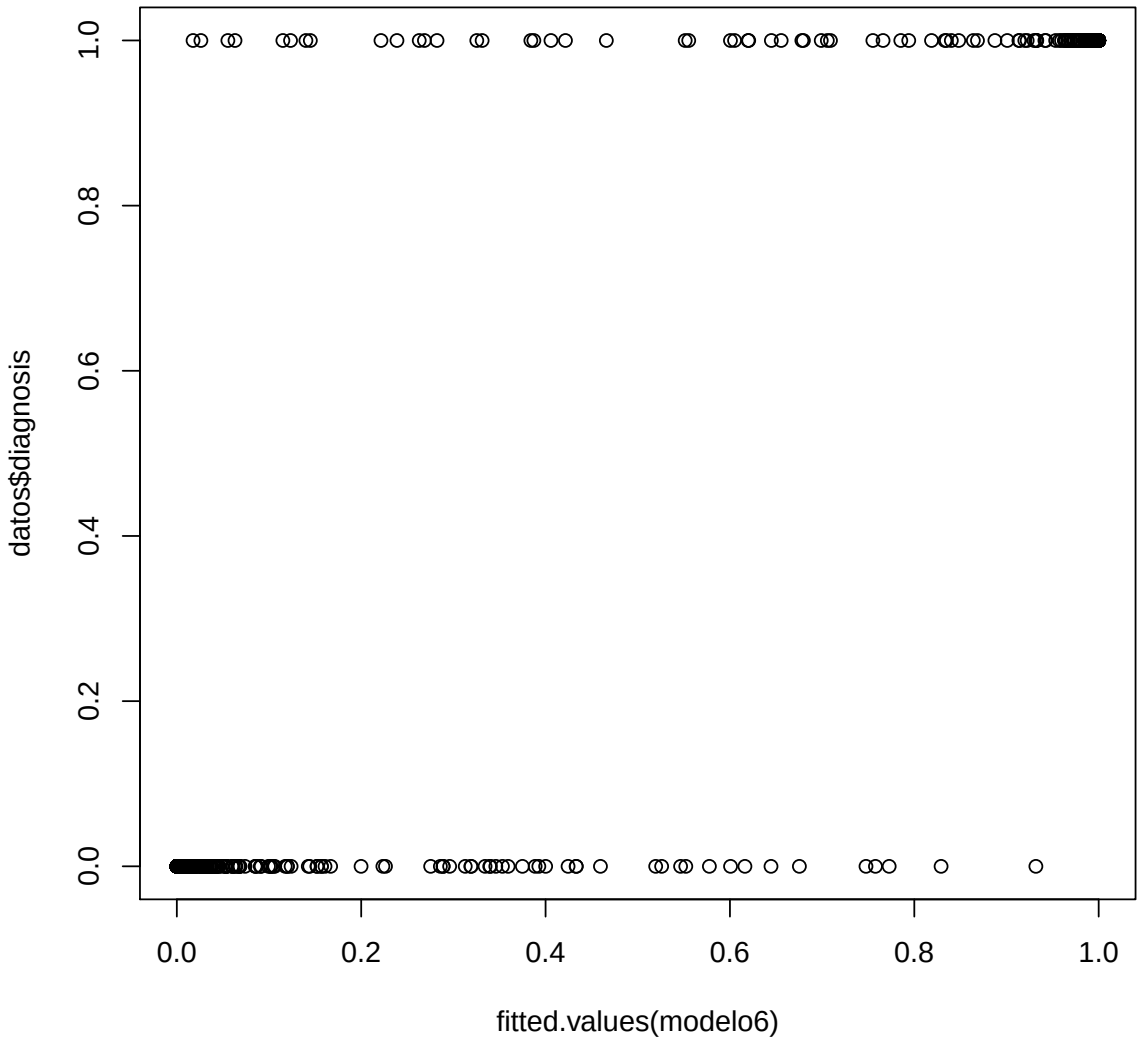
```
## Loading required package: grid
```




```
mosaic(matriz_confusion, shade = T, colorize = T,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)))
```




```
#CODIFICACIÓN 0 1 DA VARIABLE RESPOSTA
datos$diagnosis <- as.character(datos$diagnosis)
datos$diagnosis <- as.numeric(datos$diagnosis)
###
plot(fitted.values(modelo6),datos$diagnosis)
```



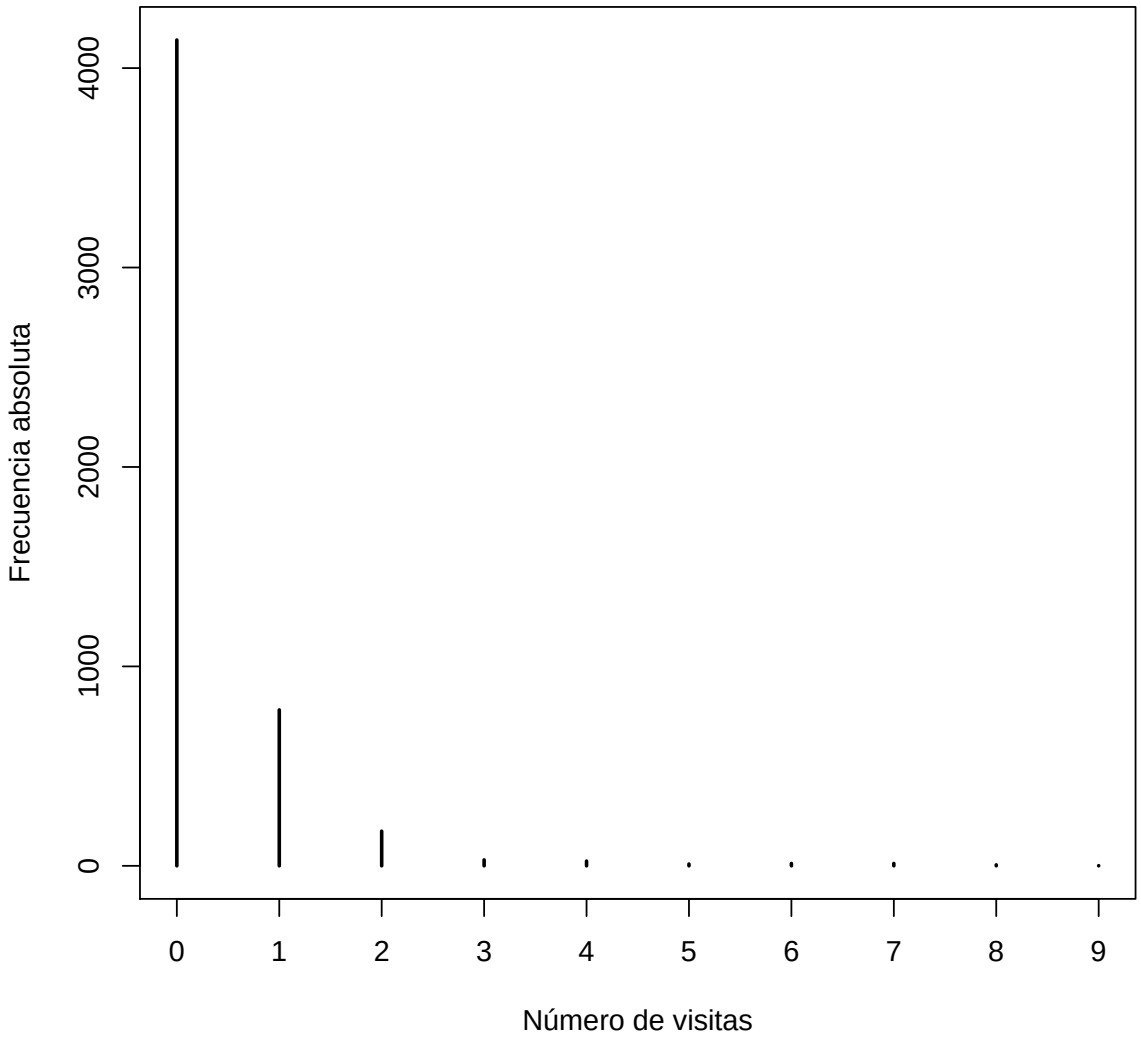

```
#Aplicación do modelo de regresión de Poisson
#install.packages("AER", repos = "https://cran.r-project.org/package=AER")
library("MASS")

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
data("DoctorVisits", package = "AER")
library("MASS")

data("DoctorVisits")
attach(DoctorVisits)
#Escritura e selección do modelo
plot(table(DoctorVisits$visits),
  xlab = "Número de visitas",
  ylab = "Frecuencia absoluta")
```



```r
modelo<-glm(visits~age+income,data = DoctorVisits, family = poisson)
summary(modelo)
```

```
##
## Call:
## glm(formula = visits ~ age + income, family = poisson, data = DoctorVisits)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0203  -0.7986  -0.6691  -0.6176   6.3802
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.60661    0.08596 -18.690  < 2e-16 ***
## age          1.35292    0.12666  10.681  < 2e-16 ***
## income      -0.34165    0.07770  -4.397  1.1e-05 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5634.8  on 5189  degrees of freedom
## Residual deviance: 5449.9  on 5187  degrees of freedom
## AIC: 7787.5
##
## Number of Fisher Scoring iterations: 6
```

```r
modelo_2<-glm(visits~age,data = DoctorVisits, family = poisson)
summary(modelo_2)
```

```
##
## Call:
## glm(formula = visits ~ age, family = poisson, data = DoctorVisits)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -0.9650  -0.8266  -0.6552  -0.6402   6.5608
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.87944    0.06248  -30.08   <2e-16 ***
## age          1.54878    0.12060   12.84   <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: 5634.8  on 5189  degrees of freedom
## Residual deviance: 5470.0  on 5188  degrees of freedom
## AIC: 7805.6
##
## Number of Fisher Scoring iterations: 6

modelo_3<-glm(visits~income,data = DoctorVisits, family = poisson)
summary(modelo_3)

##
## Call:
## glm(formula = visits ~ income, family = poisson, data = DoctorVisits)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9147  -0.8235  -0.7526  -0.6193   6.7789
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.87152    0.04567 -19.082  < 2e-16 ***
## income      -0.59991    0.07486  -8.014 1.11e-15 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
```

```
##
##     Null deviance: 5634.8  on 5189  degrees of freedom
## Residual deviance: 5566.5  on 5188  degrees of freedom
## AIC: 7902
##
## Number of Fisher Scoring iterations: 6

#Interpretación dos coeficientes
beta<-coef(modelo)
beta

## (Intercept)         age      income
##  -1.6066109   1.3529242  -0.3416503

exp(beta)

## (Intercept)         age      income
##   0.2005662   3.8687221   0.7105967

#INFERENCIA SOBRE O MODELO
#Contraste do modelo mediante deviance
library(car)
Anova(modelo)

## Analysis of Deviance Table (Type II tests)
##
## Response: visits
##        LR Chisq Df Pr(>Chisq)
## age     116.542  1  < 2.2e-16 ***
## income   20.053  1  7.533e-06 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#Bondade de axuste
(residuos.pearson <- sum(residuals(modelo, type = "pearson")^2))
```

```
## [1] 10523.84
```

```
1 - pchisq(residuos.pearson, modelo$df.residual)
```

```
## [1] 0
```

```
pchisq(modelo$deviance,df=modelo$df.residual,lower.tail = FALSE)
```

```
## [1] 0.005453847
```

```
#Intervalos de confianza para os parámetros do modelo
confint.default(modelo)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.7750869 -1.4381348
## age          1.1046702  1.6011783
## income      -0.4939319 -0.1893686
```

```
#Diagnose sobre o modelo
#Cálculo residuos
res <- residuals(modelo, type = "pearson")
head(res)
```

```
##        1        2        3        4        5        6
## 1.693427 1.648905 1.853233 1.518175 1.648905 1.604865
```

```
res.sig <- abs(res) > 2
table(res.sig)
```

```
## res.sig
## FALSE  TRUE
##  4920   270
```

```
res.orde <- sort(abs(res[res.sig]), decreasing = TRUE) # mostramos os máis altos
head(res.orde)
```

```
##       48      334      630      355      663      115
## 15.90619 14.72410 14.64670 14.32798 13.13571 12.94933

#Cálculo de residuos estandarizados
res.std <- rstandard(modelo, type = "pearson")
res.std.sig <- abs(res.std) > 2
table(res.std.sig)

## res.std.sig
## FALSE  TRUE
##  4920   270

res.ordeest <- sort(abs(res.std[res.std.sig]), decreasing = TRUE)
#mostramos os máis altos
head(res.ordeest)

##       48      334      630      355      663      115
## 15.91152 14.72677 14.66276 14.33012 13.13888 12.95648

#Cálculo dos residuos da deviance
res.d <- residuals(modelo, type = "deviance")
res.d.sig <- abs(res.d) > 2
table(res.d.sig)

## res.d.sig
## FALSE  TRUE
##  5010   180

res.dev.std <- rstandard(modelo, type = "deviance")
table(abs(res.dev.std) > 2)

##
## FALSE  TRUE
##  5010   180

#Gráficos para os residuos empregando DHARMa
#install.packages("DHARMa")
```

```r
#library(DHARMa)
#install.packages("glmmTMB")
#library(glmmTMB)
#resDhar <- simulateResiduals(modelo)
#plot(resDhar)

#Sobredispersión modelo
library(AER)

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: survival

dispersiontest(modelo)

##
##  Overdispersion test
##
## data:  modelo
## z = 7.9502, p-value = 9.312e-16
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##   2.027443

#Modelo de regresión binomial negativa
modelo_bn<-glm.nb(visits~age+income,data = DoctorVisits)
summary(modelo_bn)

##
```

```
## Call:
## glm.nb(formula = visits ~ age + income, data = DoctorVisits,
##     init.theta = 0.4239459284, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8230  -0.6932  -0.5995  -0.5617   3.6708
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.62405    0.10929 -14.860  < 2e-16 ***
## age          1.37043    0.16723   8.195  2.5e-16 ***
## income      -0.32376    0.09872  -3.280  0.00104 **
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4239) family taken to be 1)
##
##     Null deviance: 3099.2  on 5189  degrees of freedom
## Residual deviance: 2992.9  on 5187  degrees of freedom
## AIC: 7076.4
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4239
##           Std. Err.:  0.0309
##
##  2 x log-likelihood:  -7068.3970
```