Modelos estadísticos en R

Entornos de Computación Estadística

Máster en Estadística Aplicada

María Dolores Martínez Miranda
mmiranda@ugr.es

Departamento de Estadística e I.O.
Universidad de Granada

Índice general

Capítulo 1

Modelos Estadísticos en R

1.1. Modelos estadísticos y modelos lineales

All models are wrong, but some are useful. [George E.P. Box]

En Estadística se formulan modelos estadísticos con la finalidad de describir (y/o predecir) el comportamiento de un cierto proceso. Se trata de modelos con componentes estocásticas que representan la incertidumbre, debida entre otras cosas a no disponer de la suficiente información sobre las variables que influyen en el fenómeno en estudio. La inferencia estadística proporciona herramientas para ajustar y evaluar la validez de los modelos estadísticos a partir de los datos observados.

En este tema nos centramos en la definición y tratamiento en R de modelos estadísticos donde una variable, denominada variable de respuesta, se pretender describir o explicar en términos de un conjunto de variables explicativas (o predictoras).[1] Dos ejemplos de este tipo, fundamentales en Estadística, son los modelos de regresión y el análisis de la varianza (ANOVA). Ambos casos particulares de los denominados modelos lineales cuya formulación teórica se muestra a continuación:

Dadas n observaciones independientes de una variable aleatoria Y, $\{Y_1, \ldots, Y_n\}$, se dice que

[1] Al final se incluyen algunas referencias adecuadas para extender estos apuntes. Por ejemplo Faraway (2004, 2006) proporciona más detalles de la metodología incluyendo aplicaciones con datos en R.

siguen un modelo lineal si

$$Y_i = x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{im}\beta_m + \epsilon_i, \quad i = 1, \ldots, n \tag{1.1}$$

donde $\beta_1, \ldots, \beta_m$ son parámetros (poblacionales) desconocidos, x_{ij} son valores conocidos, cada uno de los cuales representa situaciones experimentales distintas, y ϵ_i son errores aleatorios. En forma matricial el modelo anterior se escribe como

$$\underbrace{\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}}_{\mathbf{Y}=\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\epsilon}}.$$

La matriz $\mathbf{X}$ se denomina matriz del modelo y su rango constituye el rango del modelo lineal.

El modelo anterior se denomina modelo lineal de Gauss-Markov cuando se verifican las condiciones de Gauss-Markov:

$$\mathbb{E}[\boldsymbol{\epsilon}] = \mathbf{0} \quad \mathbb{V}(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I}_n$$

siendo $\mathbf{0}$ un vector de ceros y $\mathbf{I}_n$ la matriz identidad de dimensión n.

El modelo (1.1) es general y admite como casos particulares algunos modelos básicos de la Estadística como son:

- Modelo de regresión lineal simple:

  $$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \quad (i = 1, \ldots, n)$$

  En este caso la matriz del modelo (también llamada matriz de regresión) tiene dimensión $n \times 2$ y se escribe como:

  $$\mathbf{X} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix}$$

  Los parámetros del modelo son el término constante u ordenada en el origen β_0 y la pendiente β_1.

- Modelo de regresión lineal múltiple:

$$Y_i = \beta_0 + x_{i1}\beta_1 + \cdots + x_{ik}\beta_k + \epsilon_i$$

En este caso la matriz de regresión tiene dimensión $n \times (k+1)$ siendo k el número de variables independientes y se escribe como

$$\mathbf{X} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,k} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,k} \end{pmatrix}$$

y los parámetros β_j $(j = 1, \ldots, k)$ se denominan coeficientes de regresión o efectos de las variables explicativas.

- Regresión polinomial simple:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_p x_i^p + \epsilon_i$$

con matriz de regresión

$$\mathbf{X} = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^p \\ 1 & x_2 & x_2^2 & \cdots & x_2^p \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^p \end{pmatrix}$$

- Modelo de análisis de la varianza (ANOVA) de una vía. El objetivo es estudiar el efecto de un supuesto factor de variación sobre una variable de respuesta[2]. Si el factor tiene k niveles el modelo se escribe como:

$$y_{ij} = \mu_i + \epsilon_{ij} = \mu + \alpha_i + \epsilon_{ij} \quad (i = 1, \ldots, k; j = 1, \ldots, n_i)$$

donde μ_i es la media de la variable de respuesta para el grupo i-ésimo, que se descompone como un factor común a todos los grupos, μ, más un factor específico de grupo, α_i. Aquí n_i representa el número de observaciones tomadas de dicho grupo, con lo que el

[2]El nombre abreviado ANOVA viene del inglés *Analysis of Variance* y se utiliza porque la idea es descomponer la variabilidad total de la variable de respuesta en una parte debida al factor de clasificación y otra de error.

total de observaciones es $n = \sum_{i=1}^{k} n_i$. El vector $\mathbf{Y}$, la matriz $\mathbf{X}$ (denominada en este contexto matriz de diseño) y el vector de parámetros en este caso tienen la forma:

$$\mathbf{Y} = \begin{pmatrix} Y_{1,1} \\ \vdots \\ Y_{1,n_1} \\ \hline Y_{2,1} \\ \vdots \\ Y_{2,n_2} \\ \hline \vdots \\ \hline Y_{k,1} \\ \vdots \\ Y_{k,n_k} \end{pmatrix} \qquad \mathbf{X} = \left(\begin{array}{cccc} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \\ \hline 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & 0 \\ \hline \vdots & \vdots & \vdots & \vdots \\ \hline 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{array}\right) \qquad \boldsymbol{\beta} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_k \end{pmatrix}$$

El problema del análisis de la varianza de una vía se puede reducir a un contraste de igualdad de medias del tipo:

$$\begin{aligned} H_0 : &\quad \mu_1 = \mu_2 = \cdots = \mu_k \\ H_1 : &\quad \mu_i \neq \mu_l \quad \text{para algún } i \neq l \end{aligned}$$

donde la hipótesis nula es equivalente a $\alpha_1 = \cdots = \alpha_k = 0$.

El análisis de la varianza de una vía se puede generalizar a más vías considerando más factores de clasificación que a su vez pueden interaccionar entre sí. Este tipo de modelos forman parte de lo que se denominan *diseños experimentales* o factoriales.

1.2. Definición de un modelo estadístico en R

Desde el punto de vista del lenguaje, en tratamiento en R de este tipo de modelos es muy similar.

Para definir un modelo estadístico en R se suelen emplear fórmulas[3] del tipo:

```
respuesta ~ modelo
```

[3]También para generar algunos gráficos como por ejemplo los diagramas de cajas múltiples con `boxplot`.

donde `modelo` especifica la expresión que describe la `respuesta`. Algunos ejemplos pueden ser:

`y ~ x` Modelo de regresión lineal simple, $Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$.
Otra fórmula equivalente sería `y ~ 1 + x`.

`y ~ x - 1` Regresión lineal simple pasando por el origen de coordenadas ($\beta_0 = 0$).
Otra fórmula equivalente sería `y ~ 0 + x`.

`y ~ x + I(x ^ 2)` Regresión polinomial de grado 2, $Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon$.
Una versión usando polinomios ortogonales es `y ~ poly(x,2)`.

`y ~ A` Análisis de la varianza de una vía, $Y_{ij} = \mu_i + \epsilon_{ij}$. Aquí `A` define los k grupos (por ejemplo un factor con k niveles).

`y ~ A * B` Diseño experimental con dos factores de clasificación, `A` y `B`.
Otra fórmula equivalente sería `y ~ A + B + A:B`.

`y ~ (A + B + C)^ 2` Diseño experimental con tres factores de clasificación, `A`, `B` y `C`, pero solo se consideran interacciones de orden 2.
Otra fórmula equivalente sería `y ~ A*B*C - A:B:C`.

`y ~ A * x` Modelos de regresión lineal simple separados para cada nivel de `A`.
Otra fórmula equivalente sería `y ~ A/x`.

Las fórmulas anteriores se pueden escribir de forma general como:

```
respuesta ~ op_1 term_1 op_2 term_2 op_3 term_3 ···
```

donde

- `response` es un vector (o una matriz) con las observaciones de la(s) variable(s) de respuesta;
- `op_1`, `op_2`, ..., son operadores de fórmula (con un significado especial en este contexto, ver Tabla 1.1);
- `term_1`, `term_2`, ..., son alguno de: vector, matriz, factor, o una expresión en términos de factores, vectores o matrices conectados por operadores.

Operador	Descripción
`+ , -`	incluye, excluye efectos principales
`1`	término constante (por defecto se incluye siempre)
`* , :`	efectos principales más interacciones
	`a*b = a+b+a:b`
`/ , \%in\%`	efectos anidados
	`a/b = a + b \%in\% a = a + a:b`
`^n`	interacciones hasta nivel n
	`(a+b)^2 = a+b+a:b`
`I()`	función identidad
	`y ~ I(x^2)` en lugar de `y ~ x^2 = y ~ x` (la interacción `x:x=x`)
`poly()`	polinomios ortogonales

Tabla 1.1: Operadores para escribir fórmulas en R.

1.3. Ajuste de un modelo lineal

El vector de parámetros $\boldsymbol{\beta}$ del modelo lineal se estima por mínimos cuadrados o máxima verosimilitud. En el caso de errores ϵ con distribución Normal ambos métodos son equivalentes, siendo el estimador mínimo-cuadrático y máximo-verosímil de $\boldsymbol{\beta}$ el definido por la expresión:

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

Al proceso de estimación de los parámetros no referimos comúnmente como "ajuste del modelo". En R la función básica para esto es la función `lm` cuya sintaxis puede ser simplemente:

```
lm(formula, data)
```

donde `data` es un objeto de tipo data frame que incluye las variables usadas en la fórmula[4].

La función `lm` devuelve un objeto de tipo lista de la clase `lm`, con al menos las siguientes componentes:

[4]Es posible omitir dicho data frame en la evaluación de la función pero en dicho caso es necesario que las variables usadas en la fórmula estén en el espacio de trabajo (o en general en la lista de búsqueda de R).

- `coefficients`: vector de coeficientes del modelo ajustado, $\widehat{\boldsymbol{\beta}}$.
- `fitted.values`: vector con los valores ajustados, $\widehat{\mathbf{Y}} = \mathbf{X}\widehat{\boldsymbol{\beta}}$.
- `residuals`: vector con los residuos del ajuste $\mathbf{e} = \mathbf{Y} - \widehat{\mathbf{Y}}$.
- `rank`: rango del modelo (rango de $\mathbf{X}$).
- `df.residual`: los grados de libertad de los residuos.

Es posible además especificar entre otros los siguientes argumentos opcionales: `subset`, para especificar un subconjunto de los datos para el ajuste; `weights`, para ajustar el modelo usando un criterio de mínimos cuadrados ponderados; y `offset`, que permite especificar una componente del modelo conocida a priori, en cuyo caso se restará a la respuesta.

1.4. Funciones genéricas para extraer información adicional del ajuste

Un objeto de tipo `lm` contiene diversa información del ajuste que puede mostrarse, representarse gráficamente, así como extenderse a través de varias funciones genéricas como las que se resumen en la Tabla 1.2. El uso y la utilidad de estas funciones lo ilustramos a continuación (y en la sesión de prácticas) a través de ejemplos.

Función	Descripción
`fitted`	valores ajustados
`coef`	coeficientes estimados (y errores estándar)
`confint`	intervalos de confianza para los coeficientes
`residuals`	residuos
`summary`	resumen detallado del modelo estimado
`predict`	calcula predicciones para nuevos datos
`anova`	tablas ANOVA (y comparación de modelos)
`vcov`	matriz de covarianzas de los parámetros estimados
`plot`	gráficos de diagnóstico
`termplot`	gráfico de efectos parciales
`step`	selección de modelos organizados jerárquicamente

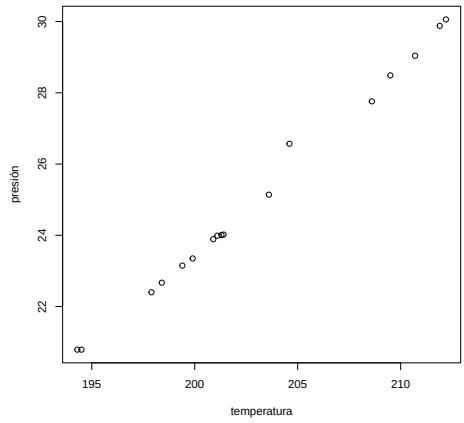
Tabla 1.2: Funciones genéricas para extraer información de un ajuste con `lm`.

1.5. Ejemplo 1: Regresión lineal simple

Entre 1840 y 1850 el físico escocés James D. Forbes desarrolló una serie de experimentos con el fin de estimar la altitud sobre el nivel del mar a partir de medidas sobre el punto de ebullición del agua. Él sabía que la altitud podía determinarse a partir de la presión atmosférica medida a través de un barómetro, con presiones menores correspondientes a elevadas altitudes. En aquella época los barómetros eran instrumentos muy frágiles y Forbes pensó que sería posible reemplazar las medidas de la presión con medidas de la temperatura de ebullición del agua. Forbes recogió datos en 17 lugares de los Alpes y los montes de Escocia. En cada lugar midió con un barómetro la presión en pulgadas de mercurio (`pres`) y con un termómetro la temperatura de ebullición del agua en grados Farenheit (`bp`).

Los datos recogidos por Forbes están disponibles en el data frame `forbes` del paquete `MASS`. Como primer paso construimos un diagrama de dispersión de los datos para explorar la relación que existe entre las variables:

```
> library(MASS)
> plot(forbes$bp,forbes$pres, xlab = 'temperatura', ylab = 'presión')
```



Del gráfico vemos que parece existir una fuerte relación lineal. Nuestro objetivo es ajustar un modelo lineal que permita describir la presión en función de la temperatura, esto es,

$$\text{presión} = \beta_0 + \beta_1 \times \text{temperatura} + \epsilon$$

donde ϵ representa los errores del modelo verificando las condiciones del modelo lineal Normal de Gauss-Markov.

Usamos la función `lm` para ajustar el modelo que escribimos como `pres~bp`:

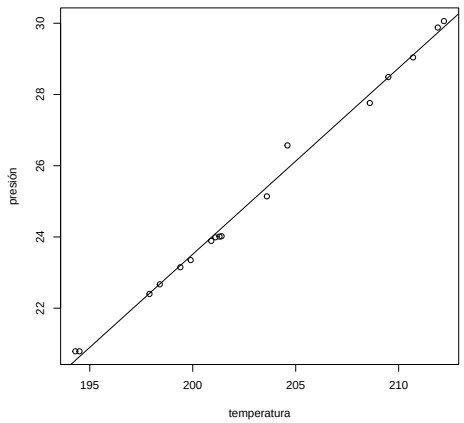
```
> lm(pres~bp,data=forbes)



Call:
lm(formula = pres ~ bp, data = forbes)

Coefficients:
(Intercept)           bp
   -81.0637       0.5229
```

La función muestra los coeficientes estimados $\widehat{\beta}_0 = -81.0637$ y $\widehat{\beta}_1 = 0.5229$. Para ver todos los valores que devuelve dicha función asignamos su valor a un objeto y visualizamos su contenido:

```
> fit<-lm(pres~bp,data=forbes)
> typeof(fit); class(fit)

[1] "list"
[1] "lm"

> names(fit)

 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.values"
 [6] "assign"        "qr"            "df.residual"   "xlevels"       "call"
[11] "terms"         "model"
```

El ajuste podemos representarlo sobre el diagrama de dispersión anterior usando la función `abline`:

```
> plot(forbes$bp,forbes$pres, xlab = 'temperatura', ylab = 'presión')
> abline(fit)
```



Para evaluar la bondad del ajuste usamos la función `summary`:

```
> summary(fit)


Call:
```

```
lm(formula = pres ~ bp, data = forbes)

Residuals:
     Min       1Q   Median       3Q      Max
-0.25717 -0.11246 -0.05102  0.14283  0.64994

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -81.06373    2.05182  -39.51   <2e-16 ***
bp            0.52289    0.01011   51.74   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2328 on 15 degrees of freedom
Multiple R-squared:  0.9944,Adjusted R-squared:  0.9941
F-statistic:  2677 on 1 and 15 DF,  p-value: < 2.2e-16
```

Entre otros resultados localizamos el valor del coeficiente de determinación $R^2 = 0.9944$, y el error estándar residual $\widehat{\sigma}_R = 0.2328$. El primero nos indica que el modelo ajusta bastante bien a los datos.

También podemos ver los resultados de los contrastes de significación de la pendiente y de la ordenada en el origen. Estos contrastes se realizan bajo el supuesto de que se cumplen las condiciones de Gauss-Markov además de que los errores del modelo siguen una distribución Normal[5].

Para la pendiente se ha formulado el problema $H_0 : \beta_1 = 0$ vs $H_1 : \beta_1 \neq 0$. Los resultados muestran el cálculo del contraste como sigue:

- $\widehat{\beta}_1 = 0.5229$ y su error estándar $s.e.(\widehat{\beta}_1) = 0.0101$
- Estadístico de contraste: $t = \frac{\widehat{\beta}_1 - 0}{s.e.(\widehat{\beta}_1)} = 51.7408$
- P-valor≈ 0, lo que indica que se debe rechazar H_0, y por tanto la temperatura contribuye de manera significativa a explicar la presión.

[5]Esto habrá que comprobarlo pero lo dejamos para la sesión de prácticas.

Para la ordenada en el origen se ha formulado el problema $H_0 : \beta_0 = 0$ vs $H_1 : \beta_0 \neq 0$, y los resultados mostrados son:

- $\widehat{\beta}_0 = -81.0637$ y su error estándar $s.e.(\widehat{\beta}_0) = 2.0518$
- Estadístico de contraste: $t = \frac{\widehat{\beta}_0 - 0}{s.e.(\widehat{\beta}_0)} - 39.5082$
- P-valor≈ 0, lo que indica que se debe rechazar H_0, y por tanto incluir un término constante en la ecuación del modelo parece adecuado.

Entre los resultados también se muestra la solución al contraste de regresión basado en la descomposición de la variabilidad. El valor del estadístico de contraste es $F = 2677.1$. Se trata de un valor muy elevado que lleva a rechazar la hipótesis nula (observa que el p-valor correspondiente es aproximadamente 0) que en este caso coincide con la del contraste de significación de la pendiente que hemos descrito antes.
Los cálculos intermedios del contraste de regresión se pueden recoger en la denominada tabla ANOVA para la regresión, que podemos obtener con la función `anova`:

```
> anova(fit)

Analysis of Variance Table

Response: pres
          Df  Sum Sq Mean Sq F value    Pr(>F)
bp         1 145.125 145.125  2677.1 < 2.2e-16 ***
Residuals 15   0.813   0.054
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

En la tabla ANOVA que nos da R se muestran las dos de las fuentes de variación del problema, VNE=0.813, y VE=145.125, con sus respectivos grados de libertad (15 y 1). Con ellas se calcula el valor del estadístico de contraste es $F = 2677.1$.

Usando de nuevo la hipótesis de normalidad de los errores se calculan intervalos de confianza para la pendiente, β_1, y la ordenada en el origen, β_0, usando la función `confint`:

```
> confint(fit)

                 2.5 %      97.5 %
(Intercept) -85.437080 -76.6903740
bp            0.501352   0.5444328
```

El resultado muestra los límites inferiores y superiores de cada intervalo. Para la pendiente se obtiene $(0.5014; 0.5444)$, y para la ordenada $(-85.4371; -76.6904)$. Por defecto nos muestra intervalos con nivel de confianza $1 - \alpha = 0.95$, pero se puede cambiar si se desea usando el argumento `level`.

A partir del modelo ajustado podemos plantearnos también hacer predicciones que era el objetivo del estudio de Forbes. A continuación usamos la función `predict` para calcular un intervalo de confianza para la presión que se esperaría (por término medio) para una localización en la que la temperatura de ebullición del agua es de 200:

```
> predict(fit,newdata=data.frame(bp=200),interval='confidence',se.fit=TRUE)

$fit
       fit      lwr      upr
1 23.51475 23.37862 23.65089

$se.fit
[1] 0.06386993

$df
[1] 15

$residual.scale
[1] 0.2328294
```

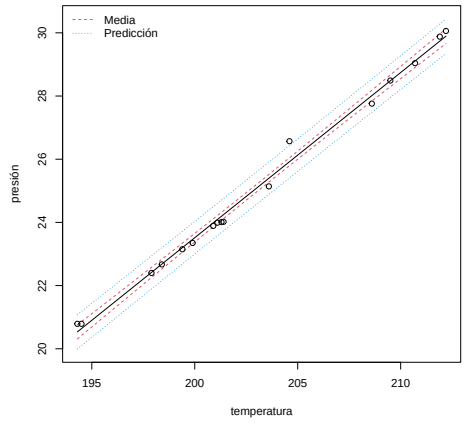
La presión esperada es de 23.5147 con un error estándar de 0.0639, lo que conduce al intervalo $(23.3786; 23.6509)$ con una confianza del 95 %. Observa que el intervalo es bastante estrecho.

Si en lugar de un intervalo para la presión media esperada buscamos un intervalo de predicción entonces el resultado sería:

```
> predict(fit,newdata=data.frame(bp=200),interval='prediction')

       fit      lwr      upr
1 23.51475 23.00016 24.02935
```

Podemos finalmente representar las bandas de confianza para la media de la presión a partir del modelo y las bandas de predicción (al 95 %). Para ello usamos de nuevo la función `predict` pero hacemos variar el punto donde se realiza la predicción dentro del rango de valores observados de la temperatura. El código que permite hacer esto y representar las bandas obtenidas se muestra a continuación junto con el resultado.

```
> x0<-data.frame(bp=seq(min(forbes$bp),max(forbes$bp),length.out=20))
> pred.m<-predict(fit,newdata=x0,interval='confidence',se.fit=T)
> pred.p<-predict(fit,newdata=x0,interval='prediction',se.fit=T)
> matplot(x0$bp,cbind(pred.m$fit,pred.p$fit[,-1]),lty=c(1,2,2,3,3),
+   col=c(1,2,2,4,4),type='l',xlab='temperatura',ylab='presión',main='')
> legend('topleft',c('Media','Predicción'), lty=c(2,3),col=c(2,4),bt='n')
> points(forbes$bp,forbes$pres)
```

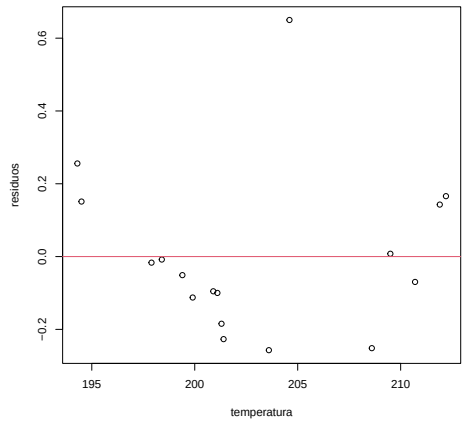


El estudio anterior se basa en el supuesto de que se verifican las hipótesis de Gauss-Markov, además de la normalidad de los errores del modelo. La verificación de estos supuesto constituye

lo que se denomina el análisis de los residuos o diagnósticos del modelo, y se realiza sobre los residuos `fit$residuals` ($e_i = y_i - \widehat{y}_i$) que constituyen estimaciones de los errores del modelo ϵ. La función `plot` también nos permite mostrar algunos gráficos de diagnóstico (`plot(fit)`). Aunque el tema de los diagnósticos del modelo lo describiremos con más detalle en el ejemplo 3, para el caso de la regresión lineal múltiple, a continuación incluimos algunos resultados básicos que nos llevarán a una variación del modelo inicial.

Con el siguiente código creamos un gráfico de los residuos frente a los valores de la temperatura:

```
> plot(forbes$bp,fit$residuals, xlab = 'temperatura', ylab = 'residuos')
> abline(h=0,col=2)
```



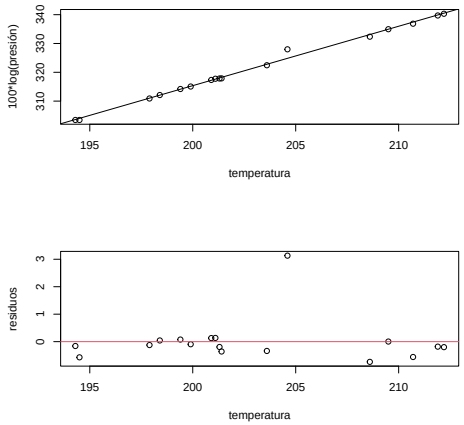
El gráfico indica claramente que algo falla en el modelo ya que en otro caso deberíamos observar una nube de puntos aleatoria. En concreto se puede observar que los puntos describen una forma curvilínea. Esto es indicativo de que la relación entre las variables no es lineal (observa que esto no era evidente del gráfico inicial de los datos). Forbes era consciente de esta no linealidad y propuso transformar la variable de respuesta tomando logaritmos. Esto como veremos a continuación va a permitir resolver el problema en gran medida.

En lugar del modelo que consideramos antes formulamos ahora un modelo para explicar el logaritmo de la presión a partir de la temperatura de ebullición del agua. Consideramos el logaritmo natural multiplicado por 100 para evitar trabajar con números pequeños:

$$100 \times \log(\text{presión}) = \beta_0 + \beta_1 \times \text{temperatura} + \epsilon$$

Repetimos los pasos anteriores para estimar el modelo y representar el nuevo ajuste:

```
> ly<-100*log(forbes$pres)
> x<-forbes$bp
> fit2<-lm(ly~x)
> par(mfrow=c(2,1))
> plot(x,ly, xlab = 'temperatura', ylab = '100*log(presión)')
> abline(fit2)
> plot(x,fit2$residuals, xlab = 'temperatura', ylab = 'residuos')
> abline(h=0,col=2)
```



El gráfico del ajuste muestra un resultado similar al que teníamos con el modelo inicial, sin embargo el gráfico de los residuos es bastante diferente. En concreto se puede deducir que cuando la variable de respuesta es el logaritmo de la presión, en lugar de la presión, a partir de la temperatura la relación sí parece ser de tipo lineal. El gráfico de los residuos no muestra ninguna forma curvilínea en este caso. Lo único a resaltar es que parece haber un dato anómalo, se trata de la observación número 12 que tiene un residuo de $e_{12} = 3.13$ (mucho mayor que el resto). En este ejemplo vamos a ignorar esta observación y obtener el resumen del nuevo ajuste:

```
> summary(fit2)


Call:
lm(formula = ly ~ x)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.73622 -0.33863 -0.15865  0.04322  3.13139

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -97.08662    7.69377  -12.62 2.17e-09 ***
x             2.06224    0.03789   54.42  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.873 on 15 degrees of freedom
Multiple R-squared:  0.995,Adjusted R-squared:  0.9946
F-statistic:  2962 on 1 and 15 DF,  p-value: < 2.2e-16
```

Se muestra la estimación de los coeficientes de la nueva recta de regresión y sus errores estándar. La interpretación de dichos coeficientes en términos de las variables originales es distinta de la que se hacía en el modelo inicial. En concreto la pendiente en un modelo donde se toma el logaritmo de la respuesta, $\log(y) = \beta_0 + \beta_1 x + \epsilon$, la pendiente, β_1, indica el *incremento porcentual* (entre 0 y 1) *en la variable de respuesta cuando la variable independiente aumenta en una unidad*. Para los datos que estamos considerando $\widehat{\beta}_1 = 2.0622$, lo que podemos interpretar como: *por cada grado que aumente la temperatura de ebullición del agua, la presión aumenta en media en aproximadamente un* 2.0622 *%*. De forma similar la varianza residual tiene ahora una interpretación interesante. En el caso del modelo sin transformar, $y = \beta_0 + \beta_1 x + \epsilon$, la varianza residual mide el error de la estimación promedio de la respuesta y. En el modelo transformado, $\log(y) = \beta_0 + \beta_1 x + \epsilon$, la varianza residual cuantifica el error relativo o porcentual (entre 0 y 1) que se obtendría para la respuesta original y. En este caso, observamos de los resultados anteriores que el error estándar (raíz cuadrada de la varianza residual) es $\widehat{\sigma}_R = 0.873$, con lo cual las estimaciones de la presión desde el nuevo modelo tienen un error promedio del $0.873\,\%$.

1.6. Ejemplo 2: Análisis de la varianza de una vía

Los siguientes datos corresponden a 24 tiempos de coagulación sanguínea en ratas[6]. Los 24 animales se asignaron aleatoriamente a 4 dietas diferentes y las muestras se tomaron al azar.

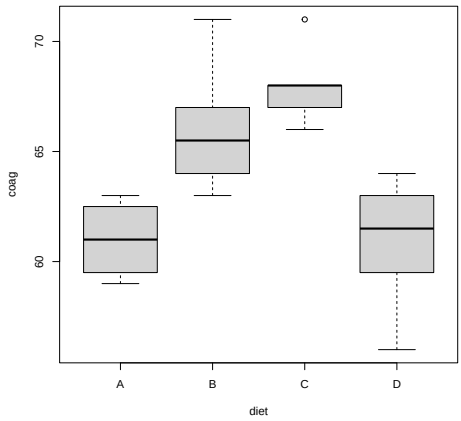
A	B	C	D
60	65	71	62
59	66	66	63
63	67	68	60
62	63	68	61
	64	67	64
	71	68	63
			56
			59

El objetivo es comprobar si existen diferencias significativa entre los tiempos de coagulación para las 4 dietas. Para ello formulamos un modelo ANOVA con el tiempo de coagulación como variable de respuesta y la dieta como factor de clasificación.

Los datos anteriores están disponibles en un data frame (`coagulation`) dentro del paquete *faraway*. Comenzamos cargando los datos y construyendo un diagrama de cajas múltiple que nos permita apreciar visualmente las posibles diferencias entre las dietas:

```
> library(faraway)
> data(coagulation)
> boxplot(coag~diet,data=coagulation)
```

[6] Box, G.P, Hunter, W.G. y Hunter, J.S. (1978) *Statistics for Experimenters*. Wiley.



Este gráfico parece mostrar diferencias entre los grupos. Por otro lado de este gráfico también podemos tener una primera impresión sobre la validez de las hipótesis del modelo, en concreto verificar que no se observa nada de los siguiente:

- Datos anómalos (*outliers*).
- Asimetría (que sería incompatible con el supuesto de normalidad).
- Varianzas desiguales (lo que correspondería a cajas de dimensiones notablemente diferentes).

En este caso no parece haber problemas en relación a ninguno de las tres aspectos anteriores, teniendo en cuenta el reducido número de observaciones en algunos grupos[7].

A continuación procedemos al ajuste del modelo. Para ello podemos hacerlo de nuevo usando la función `lm`:

```
> lm(coag~diet,data=coagulation)


Call:
lm(formula = coag ~ diet, data = coagulation)
```

[7]Observa que en este caso las diferencias en variabilidad pueden explicarse por los reducidos tamaños muestrales junto con que hay valores repetidos.

```
Coefficients:
(Intercept)        dietB        dietC        dietD
  6.100e+01    5.000e+00    7.000e+00    2.719e-15
```

Observamos que nos devuelve los efectos del modelo con término constante, esto es, $\mu = 61$, $\alpha_B = 5$, $\alpha_C = 7$, $\alpha_D = 2.7194799 \times 10^{-15}$ [8]. Otra posibilidad sería estimar directamente los parámetros $\mu_i = \mu + \alpha_i$ para lo cual escribiríamos:

```
> lm(coag~diet-1,data=coagulation)


Call:
lm(formula = coag ~ diet - 1, data = coagulation)

Coefficients:
dietA  dietB  dietC  dietD
   61     66     68     61
```

Una vez ajustado el modelo (con cualquiera de las dos opciones anteriores) resolvemos el problema de contraste:

$$H_0: \quad \mu_A = \mu_B = \mu_C = \mu_D$$
$$H_1: \quad \mu_i \neq \mu_l \quad \text{para algún } i \neq l$$

para lo que utilizamos la función `anova` evaluada en el objeto resultante del ajuste:

```
> fit<-lm(coag~diet,data=coagulation)
> anova(fit)

Analysis of Variance Table

Response: coag
```

[8] Con esta parametrización del modelo se impone que $\sum n_i \alpha_i = 0$ por lo que solo es necesario estimar tres de los α's.

```
          Df Sum Sq Mean Sq F value    Pr(>F)
diet       3    228    76.0  13.571 4.658e-05 ***
Residuals 20    112     5.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

El p-valor 4.658471×10^{-5} nos indica que podemos rechazar claramente la hipótesis nula lo que supondría que la dieta tiene un efecto significativo en el tiempo de coagulación.

Otra forma de ajustar el modelo ANOVA y resolver el problema es usando la función `aov` en lugar de `lm`. Su uso y resultado en este caso sería:

```
> fit2<-aov(coag~diet,data=coagulation)
> summary(fit2)

            Df Sum Sq Mean Sq F value   Pr(>F)
diet         3    228    76.0   13.57 4.66e-05 ***
Residuals   20    112     5.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cuando encontramos diferencias significativas entre los factores es interesante realizar un análisis posterior que permite descubrir la raíz de estas diferencias. Una primera idea sería hacer una comparación de los grupos dos a dos usando la función `pairwise.t.test`:

```
> pairwise.t.test(coagulation$coag,coagulation$diet)


Pairwise comparisons using t tests with pooled SD

data:  coagulation$coag and coagulation$diet

  A       B       C
B 0.01141 -       -
```

```
C 0.00090 0.31755 -
D 1.00000 0.00345 0.00014

P value adjustment method: holm
```

Un estudio más adecuado sería a través de las comparaciones múltiples de Tukey que podemos obtener usando la función `TukeyHSD`:

```
> TukeyHSD(aov(coag~diet, coagulation))

  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = coag ~ diet, data = coagulation)

$diet
    diff         lwr       upr     p adj
B-A    5   0.7245544  9.275446 0.0183283
C-A    7   2.7245544 11.275446 0.0009577
D-A    0  -4.0560438  4.056044 1.0000000
C-B    2  -1.8240748  5.824075 0.4766005
D-B   -5  -8.5770944 -1.422906 0.0044114
D-C   -7 -10.5770944 -3.422906 0.0001268
```

Como resultado nos muestra intervalos de confianza para las misma diferencias $\alpha_i - \alpha_j$, junto con una corrección del p-valor adecuada para el problema de comparaciones múltiples formulado.

Para finalizar tenemos que de nuevo hacer un comentario importante y es que los análisis anteriores se han hecho bajo el supuesto de que se cumplen las hipótesis del modelo. Evaluar si se verifican dichas hipótesis supondría un análisis de residuos similar al caso de los modelos de regresión. Este aspecto no lo recogemos aquí, si bien puede consultarse por ejemplo en el libro de Faraway (2004) cuya referencias se puede ver al final del pdf.

En el caso en que no se pueda asumir la hipótesis de igualdad de varianzas[9] se puede utilizar la versión que implementa la función `oneway.test`:

```
> oneway.test(coag~diet,data=coagulation)


One-way analysis of means (not assuming equal variances)

data:  coag and diet
F = 16.728, num df = 3.0000, denom df = 9.9533, p-value = 0.0003249
```

También existe una versión no paramétrica para el caso en que no se pueda asumir la hipótesis de normalidad:

```
> kruskal.test(coag~diet,data=coagulation)


Kruskal-Wallis rank sum test

data:  coag by diet
Kruskal-Wallis chi-squared = 17.015, df = 3, p-value = 0.0007016
```

1.7. Ejemplo 3: Regresión lineal múltiple

El contexto del análisis es un estudio de mercado donde se obtuvieron datos de 100 empresas, clientes de un gran distribuidor industrial (denominado HATCO). Los datos se recogen en el fichero `hatco2.csv`, disponible en PRADO, que contiene para cada empresa información acerca de las siguientes variables:

[9] Que se puede comprobar por ejemplo con un contraste usando la función `bartlett.test`.

Variable	Descripción
empresa	Identificador de la empresa
tamano	Tamaño de la empresa
adquisic	Estructura de adquisición
tindustr	Tipo de industria
tsitcomp	Tipo de situación de compra
velocidad	Velocidad de entrega
precio	Nivel de precios
flexprec	Flexibilidad de precios
imgfabri	Imagen del fabricante
servconj	Servicio conjunto
imgfvent	Imagen de fuerza de ventas
calidadp	Calidad de producto
fidelidad	Porcentaje de compra a HATCO
nfidelidad	Nivel de compra a HATCO
nsatisfac	Nivel de satisfacción

Comenzamos cargando los datos en R y almacenándolos en un data frame con nombre `hatco`. Para ello usamos la función `read.csv` con el argumento `as.is` de modo que las variables de tipo factor se identifiquen como tal.

```
> hatco<-read.csv('hatco2.csv',header=TRUE,as.is=NA)
```

De las 16 variables que componen el data frame `hatco`, `velocidad`, `precio`, `flexprec`, `imgfabri`, `servconj`, `imgfvent` y `calidadp`, constituyen percepciones del cliente (la empresa) acerca de la distribuidora y sus productos en relación a distintos aspectos. Estas percepciones han sido evaluadas en una escala métrica entre 0 (pobre) y 10 (excelente). La variable `fidelidad` se mide como el porcentaje que se compra al distribuidor del total del producto de la empresa.

Nuestro objetivo con los datos anteriores es predecir la fidelidad al distribuidor por parte de los clientes, tomando con base las percepciones que estos tienen del mismo y de sus productos, así como identificar los factores que llevan al aumento de la utilización del producto. Para ello

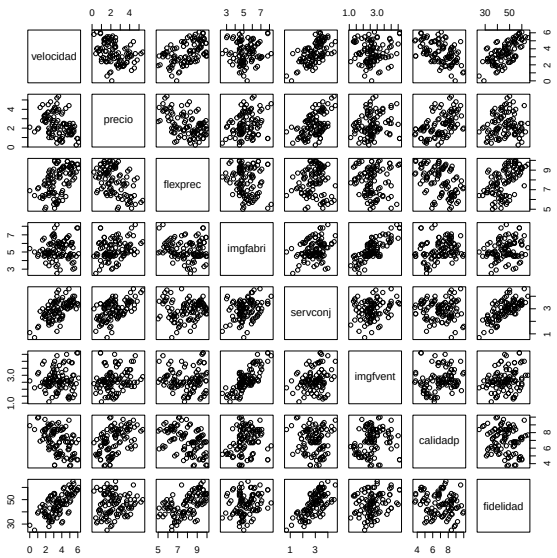
se propone un modelo de regresión lineal múltiple:

$$\begin{aligned} Y_i &= \beta_0 + x_{i1}\beta_1 + \cdots + x_{ik}\beta_k + \epsilon_i \quad (i = 1, \ldots, n) \\ &\quad \text{asumiendo que } \epsilon_i \rightsquigarrow N(0, \sigma^2) \text{ (incorreladas)} \end{aligned}$$

donde la variable de respuesta Y es la fidelidad por parte del cliente (`fidelidad`), y como variables explicativas se consideran las $k = 7$ percepciones medidas: `velocidad`, `precio`, `flexprec`, `imgfabri`, `servconj`, `imgfvent` y `calidadp`.

Antes de ajustar el modelo anterior representamos gráficamente los datos para observar la posible relación entre la respuesta y las variables explicativas.

```
> plot(hatco[,c(6:13)])
```



El gráfico obtenido proporciona diagramas de dispersión de las variables consideradas dos a dos. De este gráfico, y siempre de una manera aproximada, es posible:

- Advertir si existe relación lineal entre la respuesta y cada una de las variables explicativas.
- Descubrir si hay variables explicativas que sean aproximadamente colineales al resto.

Ajustamos ahora el modelo de regresión lineal múltiple a las $n = 100$ observaciones que tenemos de las variables, usando la función `lm`, y almacenando el resultado en un objeto con nombre `mod1`:

```
> mod1<-lm(fidelidad~velocidad+precio+flexprec+imgfabri+servconj+
+            imgfvent+calidadp,hatco)
> mod1


Call:
lm(formula = fidelidad ~ velocidad + precio + flexprec + imgfabri +
    servconj + imgfvent + calidadp, data = hatco)

Coefficients:
(Intercept)    velocidad       precio     flexprec     imgfabri     servconj
  -10.16148     -0.04352     -0.67891      3.36197     -0.04101      8.34537
   imgfvent     calidadp
    1.29147      0.56295
```

El resultado nos muestra los coeficientes estimados, $\widehat{\beta}_0, \ldots, \widehat{\beta}_7$, de donde los valores ajustados (estimados) de la fidelidad para cada cliente se obtienen a partir de la siguiente expresión lineal:

$$\widehat{Y}_i = -10.161 - 0.044x_{i1} - 0.679x_{i2} + 3.362x_{i3} - 0.041x_{i4} + 8.345x_{i5} + 1.291x_{i6} + 0.563x_{i7}$$

Los coeficientes estimados $\widehat{\beta}_j$ representan la magnitud del efecto que cada percepción del cliente ejerce en su fidelidad a la distribuidora. Por ejemplo, considerando la flexibilidad de precios (x_3) podemos decir que por cada unidad que aumenta la percepción que el cliente tiene de dicha flexibilidad, su fidelidad se incrementa en 3.362 unidades, supuesto que el resto de percepciones permanece constante.

Inferencia sobre el modelo

Estudiamos ahora en qué medida las 7 percepciones de forma conjunta consiguen describir la fidelidad de los clientes y hacemos una valoración global del ajuste. Para ello calculamos el contraste de regresión y los coeficientes R^2 (coeficiente de determinación) y $\bar{R}^2$ (versión corregida para la regresión múltiple). Todo esto lo podemos obtener con la función `summary`.

```
> summary(mod1)


Call:
lm(formula = fidelidad ~ velocidad + precio + flexprec + imgfabri +
    servconj + imgfvent + calidadp, data = hatco)

Residuals:
     Min       1Q   Median       3Q      Max
-12.9759  -1.9491   0.5896   2.8144   6.7565

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -10.16148    4.97696  -2.042   0.0440 *
velocidad    -0.04352    2.01273  -0.022   0.9828
precio       -0.67891    2.09025  -0.325   0.7461
flexprec      3.36197    0.41125   8.175 1.56e-12 ***
imgfabri     -0.04101    0.66683  -0.061   0.9511
servconj      8.34537    3.91830   2.130   0.0359 *
imgfvent      1.29147    0.94720   1.363   0.1761
calidadp      0.56295    0.35544   1.584   0.1167
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.424 on 92 degrees of freedom
Multiple R-squared:  0.775,Adjusted R-squared:  0.7578
F-statistic: 45.26 on 7 and 92 DF,  p-value: < 2.2e-16
```

El contraste de regresión se muestra al final de la salida anterior. Este contraste formula la hipótesis nula $H_0 : \beta_1 = \beta_2 = \cdots = \beta_7 = 0$, lo que implicaría que las variables explicativas de forma conjunta no tienen ningún efecto en la variable de respuesta. Esto parece poco probable atendiendo a la naturaleza de las variables, y se confirma con el estadístico de contraste obtenido $F = 45.26$ y el p-valor asociado de aproximadamente 0. Por tanto podemos concluir

que el modelo de regresión ajustado es útil y tiene sentido, ya que las variables explicativas (percepciones de los clientes en este caso) consideradas de modo conjunto permiten explicar la variable de respuesta (fidelidad).

En relación a la bondad del ajuste observamos los valores `Multiple R-squared` y `Adjusted R-squared`. El primero es el coeficiente de determinación, R^2, que en este caso vale 0.775). Lo podemos interpretar diciendo que aproximadamente el 77.5 % de la variabilidad total de la fidelidad de los clientes queda explicada por las 7 percepciones a través del modelo lineal ajustado. El segundo de los valores es el coeficiente de determinación corregido que resulta 0.7578. Este valor da una medida adecuada[10] de la bondad del ajuste, corregido por el número de variables explicativas.

La tabla ANOVA nos muestra la descomposición de la variabilidad utilizada para el contraste de regresión descrito antes y el coeficiente R^2. La obtenemos usando la función `anova`:

```
> anova(mod1)

Analysis of Variance Table

Response: fidelidad
           Df Sum Sq Mean Sq  F value    Pr(>F)
velocidad   1 3659.8  3659.8 187.0071 < 2.2e-16 ***
precio      1  927.9   927.9  47.4128 6.932e-10 ***
flexprec    1 1346.3  1346.3  68.7912 8.779e-13 ***
imgfabri    1  100.4   100.4   5.1311   0.02585 *
servconj    1   71.5    71.5   3.6517   0.05913 .
imgfvent    1   44.9    44.9   2.2963   0.13311
calidadp    1   49.1    49.1   2.5084   0.11667
Residuals  92 1800.5    19.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

[10] El R^2 tiende a sobrevalorar el ajuste tomando valores mayores a medida que incluímos variables explicativas adicionales, aunque no tengan ninguna relación con la respuesta.

La columna `Sum Sq` nos da al final la variabilidad no explicada (`Residuals`) y en las filas anteriores la variabilidad explicada de forma secuencial para cada variable explicativa (sumándolas tendríamos la variabilidad explicada por el modelo ajustado).

Estudiamos ahora la influencia individual que cada una de las 7 percepciones consideradas tiene en la fidelidad. Para ello formulamos contrastes de significación individuales para cada percepción x_j ($j = 1, \ldots, 7$), de tipo $H_0 : \beta_j = 0$ frente a $H_1 : \beta_j \neq 0$. Se trata por tanto de 7 problemas de contraste de hipótesis donde en cada caso rechazar la hipótesis nula supondrá concluir que la percepción correspondiente tiene influencia significativa en la fidelidad, considerada en el contexto de la regresión lineal múltiple (donde están presentes las 7 percepciones). Los resultados de cada contraste se mostraron anteriormente como resultado aplicar la función `summary` (observa los contrastes en la tabla de los coeficientes que entre otra información te ofrecía dicha función antes).

Por ejemplo para la primera variable explicativa, x_1, correspondiente a la percepción que el cliente tiene de la velocidad de entrega, el estadístico de contraste[11] sería: $t_1 = \widehat{\beta}_1 / s.e.(\widehat{\beta}_1) = -0.0435/2.0127 = -0.0216$. Observando el p-valor asociado, 0.983, se concluye que no hay evidencia suficiente de los datos para rechazar H_0 al 5 % de significación, y por tanto la percepción que el cliente tiene de la velocidad de la entrega no parece influir significativamente en su fidelidad a la distribuidora[12]

Además de los efectos de cada percepción β_j ($j = 1, \ldots, 7$), el modelo que hemos ajustado incluye un término constante, el cual se interpreta en general como el valor medio de la respuesta cuando todas las variables explicativas toman el valor 0. En el ajuste realizado la estimación de este término es $\widehat{\beta}_0 = -10.1615$ con un error estándar de 4.977. Podemos estudiar su significación formulando el problema de contraste $H_0 : \beta_0 = 0$ frente a $H_1 : \beta_0 \neq 0$. Los resultados están de nuevo en la tabla de coeficientes que proporcionaba la función `summary` (fila correspondiente a `Intercept`).

[11] Bajo H_0 el estadístico de contraste sigue una t de Student con $n - k - 1$ grados de libertad (en este caso serían 92 grados de libertad).

[12] Esta conclusión sin embargo debe entenderse en el contexto del modelo ajustado, donde están presentes las 7 percepciones, y no nos lleva a concluir que podemos prescindir de x_1 en el modelo, sino que es posible que el modelo se pueda simplificar ya que alguna(s) variable(s) puede(n) ser redundante(s).

Diagnósticos del modelo

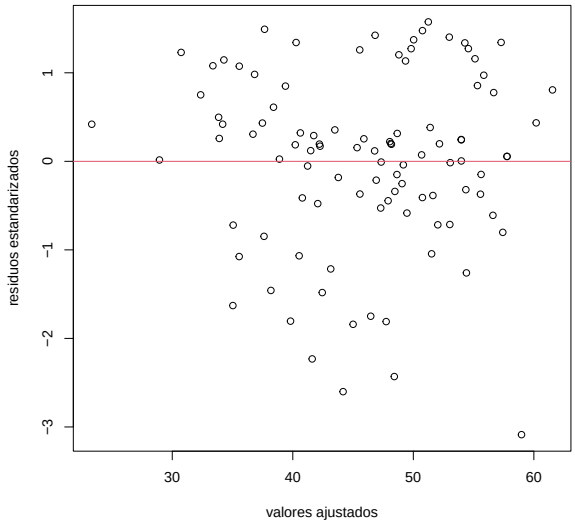
A continuación verificamos las hipótesis del modelo de regresión que hemos considerado inicialmente.

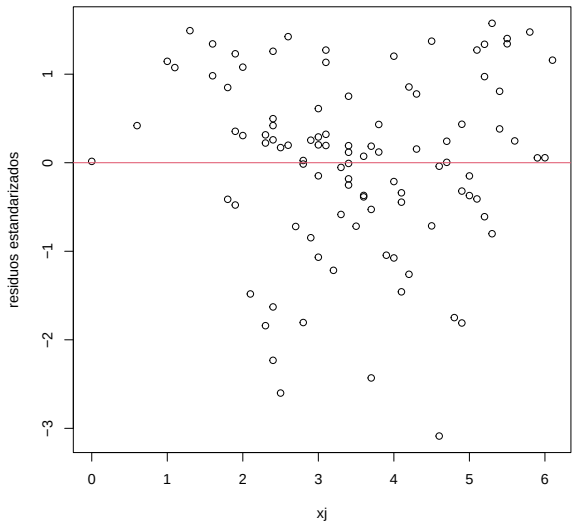
Los diagnósticos del modelo incluyen en primer lugar un análisis de los residuos para comprobar que se verifican las hipótesis del modelo de regresión lineal múltiple: linealidad de la relación y normalidad, homocedasticidad e incorrelación de los errores del modelo. A través de este análisis también comprobamos si existen observaciones anómalas e influyentes[13]. Finalmente se hará un estudio de la posible multicolinealidad entre las variables explicativas del modelo.

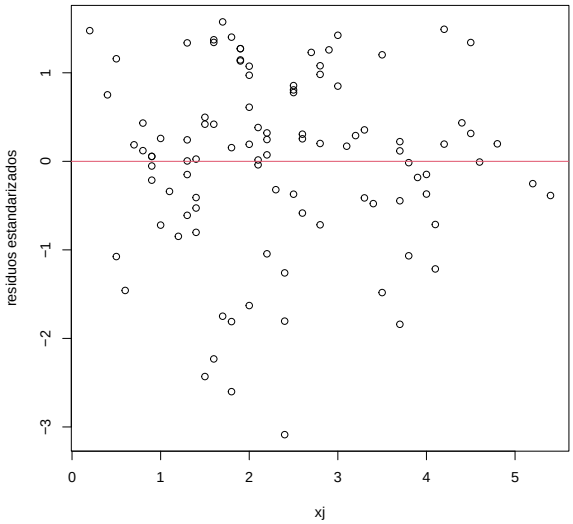
La hipótesis de homocedasticidad implica que los errores del modelo tienen varianza constante. Dado que los errores del modelo no se observan, estudiamos dicha hipótesis sobre los residuos $e_i = Y_i - \widehat{Y}_i$, o su versión estandarizada, r_i (media 0 y varianza 1), o estudentizada (media 0, varianza 1 y distribución t de Student), $\widehat{t}_i$. En este caso vamos a calcular los residuos estandarizados que se pueden calcular con la función `rstandard`, y valoraremos la hipótesis de homocedasticidad representando gráficos de estos residuos frente a valores ajustados primero, y luego frente a cada una de las percepciones. Patrones no aleatorios en estos gráficos nos alertarían de desviaciones de la hipótesis de homocedasticidad.

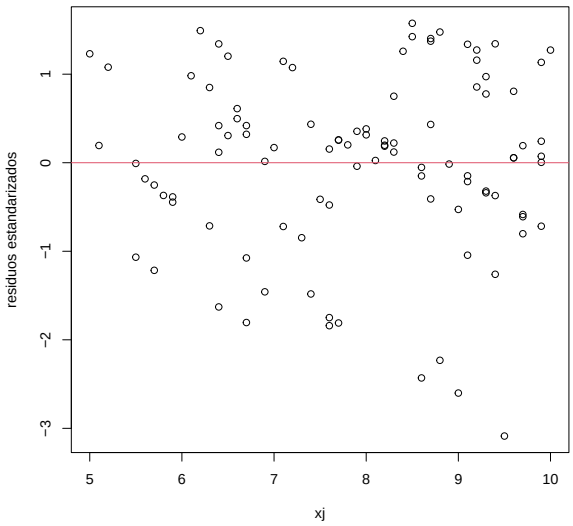
```
> ei.std<-rstandard(mod1)
> plot(mod1$fitted.values,ei.std,main="",xlab="valores ajustados", ylab="residuos estan
> abline(h=0,col=2)
> for (j in 6:12) {
+   plot(hatco[,j],ei.std,main="",xlab="xj",
+        ylab="residuos estandarizados")
+   abline(h=0,col=2)
+ }
```

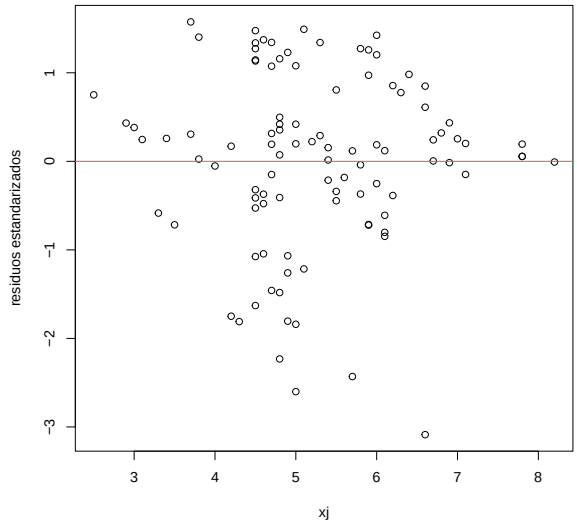
[13] Algunos gráficos para estos diagnósticos se pueden obtener escribiendo `plot(mod1)`. Puedes echar un vistazo a los mismos no obstante en esta práctica construiremos nuestros propios gráficos siguiendo las indicaciones proporcionadas.

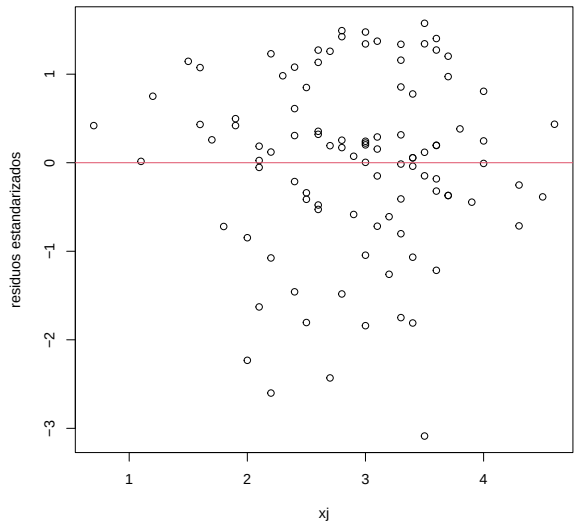

residuos estandarizados
valores ajustados
1
0
−1
−2
−3
30
40
50
60

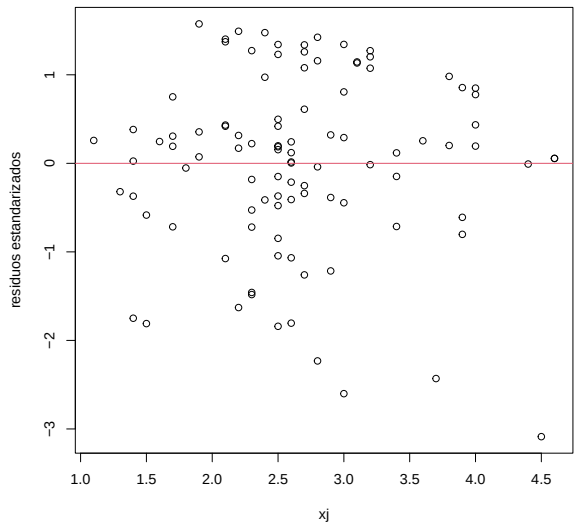

residuos estandarizados
xj
1
0
−1
−2
−3
0
1
2
3
4
5
6

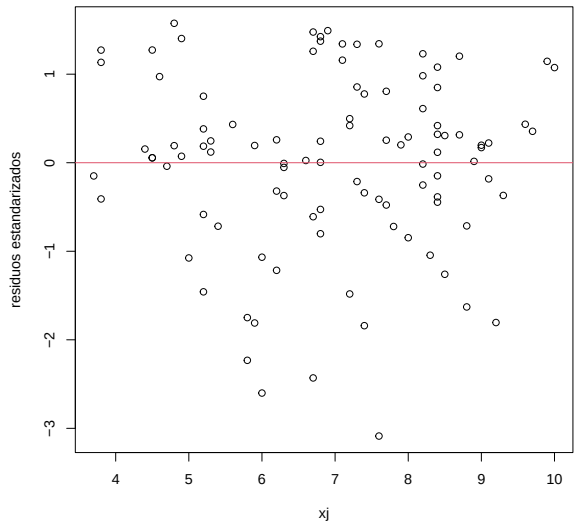

residuos estandarizados
xj
1
0
−1
−2
−3
0
1
2
3
4
5


residuos estandarizados
xj
1
0
−1
−2
−3
5
6
7
8
9
10


residuos estandarizados
xj
1
0
−1
−2
−3
3
4
5
6
7
8

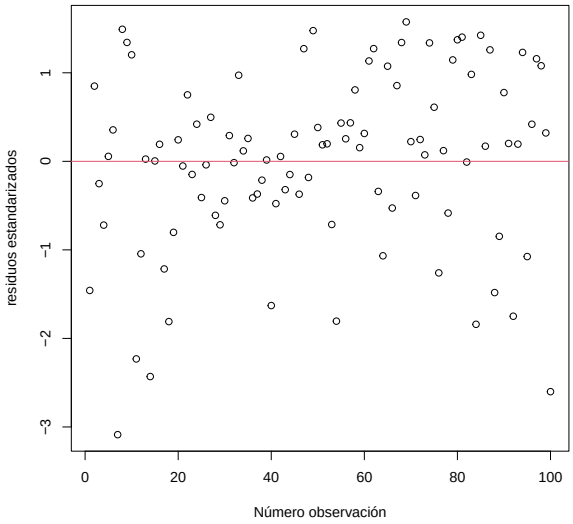

residuos estandarizados
xj
1
0
−1
−2
−3
1
2
3
4





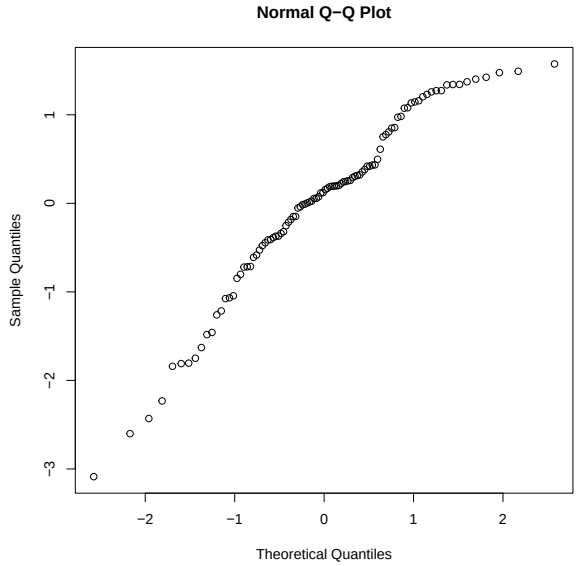
Los errores del modelo de regresión lineal múltiple que hemos ajustado se asumen incorrelados. Para verificar esta hipótesis podemos construir un gráfico de residuos (e.g. los estandarizados) frente al número de orden de cada observación (variable `empresa` en el fichero). Un patrón no aleatorio en este gráfico nos alertaría de posibles desviaciones de esta hipótesis. La impresión visual del gráfico la podemos confirmar con el test de Durbin-Watson que se puede obtener usando la función `dwtest` del paquete *lmtest*.

```
> plot(hatco$empresa,ei.std,main="",xlab="Número observación",
+      ylab="residuos estandarizados")
> abline(h=0,col=2)
> library(lmtest)

Warning: package 'lmtest' was built under R version 4.3.3

Loading required package: zoo

Warning: package 'zoo' was built under R version 4.3.3


Attaching package: 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric

> dwtest(mod1)


Durbin-Watson test

data:  mod1
DW = 1.8924, p-value = 0.3179
alternative hypothesis: true autocorrelation is greater than 0
```



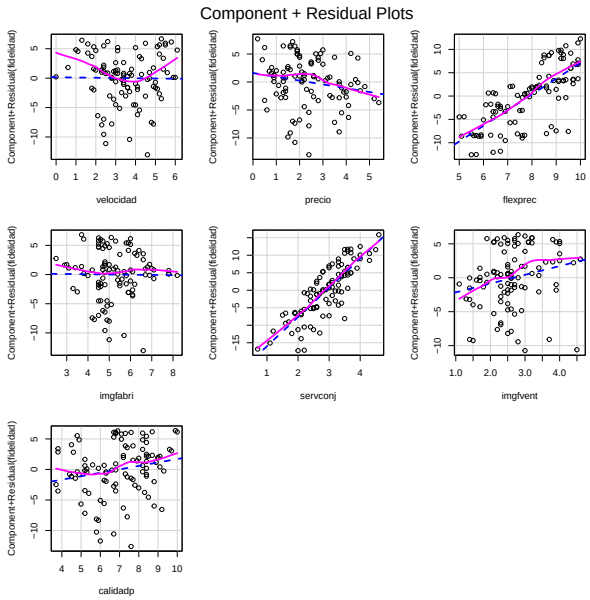
Los errores de modelo se asumen normales y dicha hipótesis es esencial para desarrollar la inferencia del modelo que hemos descrito antes. Verificar esta hipótesis es por tanto crucial. Dado que el tamaño de muestra es relativamente grande ($n = 100$) un contraste de normalidad recomendable puede ser el test de Kolmogorov-Smirnov (función `ks.test`). El resultado del test lo podemos completar representando un gráfico probabilístico normal (función `qqnorm`).

```
> ks.test(ei.std,pnorm)


Asymptotic one-sample Kolmogorov-Smirnov test

data:  ei.std
D = 0.098903, p-value = 0.282
alternative hypothesis: two-sided

> qqnorm(ei.std)
```



La posible falta de linealidad en la relación entre la respuesta y las variables explicativas se ha podido investigar en un primer momento observando los diagramas de dispersión entre la respuesta y cada una de las variables explicativas. Allí se observaban pautas de relación aproximadamente lineales salvo en algunos casos donde no parecía existir mucha relación. No obstante los gráficos más adecuados para detectar posible no linealidad son los gráficos de componente más residuo que podemos obtener usando la función `crPlots` del paquete *car*.

```
> library(car)

Warning: package 'car' was built under R version 4.3.3
Loading required package: carData
Warning: package 'carData' was built under R version 4.3.3

Attaching package: 'car'
The following objects are masked from 'package:faraway':
   logit, vif

> crPlots(mod1)
```



La falta de linealidad se identifica en estos gráficos buscando patrones no lineales. Para ello se muestra en azul la línea que representaría el ajuste lineal a la nube de puntos y en rosa una curva de suavizado de los puntos (representando de forma más flexible la nube de puntos). En general en estos gráficos se puede ver que las nubes de puntos son aproximadamente lineales (linea azul muy próxima a la curva rosa), salvo quizá para la primera percepción x_1 donde por otra parte no parece existir mucha relación con la variable dependiente. Por tanto podemos concluir que asumir que la relación entre la fidelidad de los clientes y sus percepciones individuales es lineal parece una hipótesis adecuada para estos datos.

Terminamos el análisis de residuos identificando posibles datos anómalos e influyentes. Datos anómalos son aquellos que tienen un residuo asociado cuya magnitud es excesivamente grande. Para detectar estos puntos consideramos residuos estandarizados o estudentizados y localizamos valores que estén fuera del rango (-2,2), siendo muy extremos aquellos que están fuera de (-3,3).

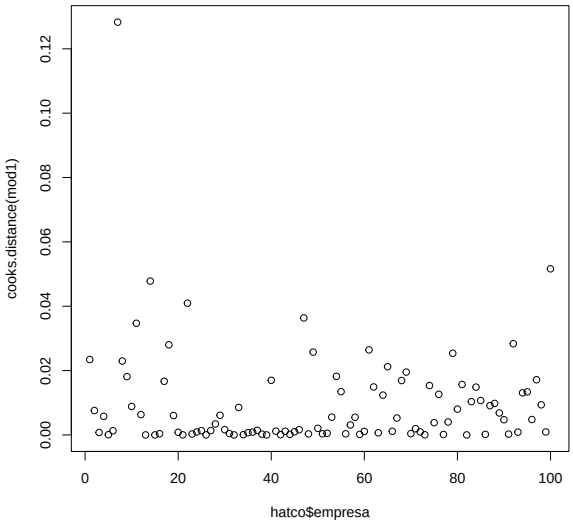
```
> which(abs(ei.std)>2.5)

  7 100
  7 100
```

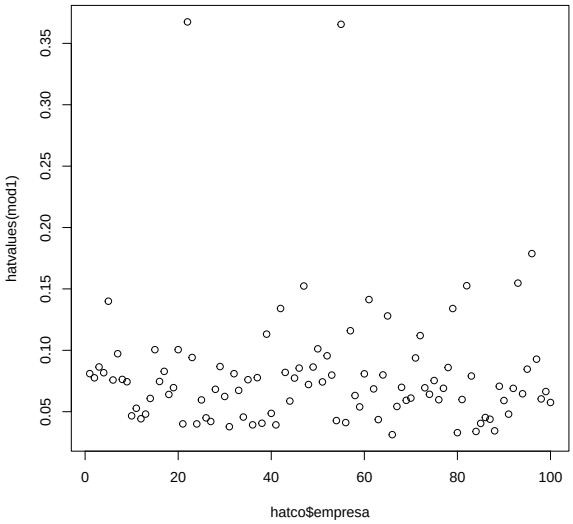
Datos influyentes son aquellos que tienen un impacto desproporcionado sobre los resultados de

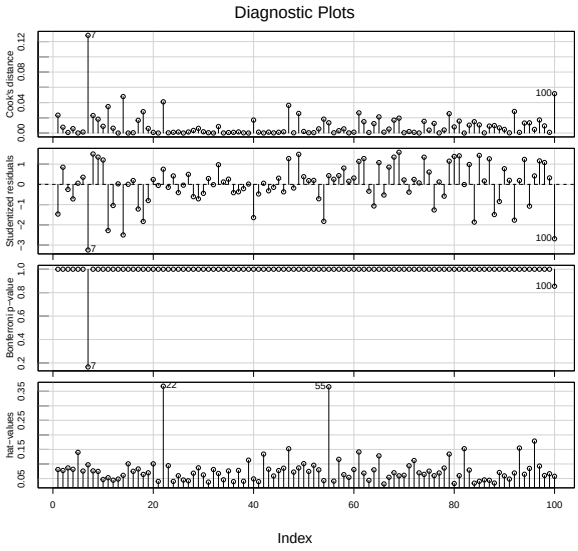
la regresión. Si además se trata de datos aislados estos datos deberían tratarse. Para medir la influencia de las observaciones utilizamos la distancia de Cook D_i (función `cooks.distance`) y para medir el aislamiento (*leverage*) utilizamos los valores en la diagonal, de la matriz hat $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ (función `hatvalues`). Medidas de influencia notablemente superiores a las del resto de observaciones corresponderían a datos influyentes (igual para el aislamiento).

El paquete *car* dispone de la función `influenceIndexPlot` que permite reproducir gráficos para hacer de una manera más sencilla esta tarea.

```
> plot(hatco$empresa,cooks.distance(mod1))
> plot(hatco$empresa,hatvalues(mod1))
> influenceIndexPlot(mod1)
```







Los datos anómalos y muy influyentes deben ser tratados. Cuando se tiene acceso a la fuente de los datos es necesario comprobar si corresponden a errores de medida o procesado de los datos, o bien si indican alguna situación más complicada en relación al diseño del estudio. En este caso no tenemos acceso a la fuente ni más información que la que se ha ofrecido de modo que procederemos a eliminar dichos datos[14] Si has hecho la tarea anterior correctamente habrás podido identificar 2 observaciones anómalas y/o muy influyentes que son las empresas

[14]Lo ideal sería hacer una eliminación gradual, uno a uno, empezando con el dato más problemático y observando en cada paso los cambios que se producen en el modelo.

7 y 100.

```
> hatco<-hatco[-c(7,100),]
> mod2<-lm(fidelidad~velocidad+precio+flexprec+imgfabri+servconj+
+            imgfvent+calidadp,hatco)
```

Una vez completado el análisis de residuos procedemos a hacer un estudio de la multicolinealidad. Los contrastes de significación individual de cada percepción (x_j) nos advierten de que hay variables redundantes en el modelo. Es importante que confirmemos que esto supone un problema serio de multicolinealidad. Para descartar la existencia de multicolinealidad seguimos los siguientes pasos:

- Calculamos la matriz de correlaciones $\mathbf{R}$ entre las percepciones consideradas dos a dos. Esto se puede hacer escribiendo `R<-cor(hatco[,6:12])`. Tenemos que descartar que existan correlaciones muy elevadas.
- Calculamos el índice de condicionamiento de la matriz $\mathbf{R}$ y comprobamos que está por debajo de 30. Para obtener dicho índice puedes escribir:

  ```
  ai<-eigen(R)$values    # autovalores de R
  sqrt(max(ai)/min(ai))  # el índice IC
  ```

- Para cada una de las 7 percepciones calculamos el factor de inflado de varianza VIF. Todos deben estar por debajo de 10, y preferentemente por debajo de 5. Calcula dichos valores usando la función `vif` del paquete `car`. Para ello escribe `vif(mod2)`.

```
> R<-cor(hatco[,6:12])
> ai<-eigen(R)$values    # autovalores de R
> sqrt(max(ai)/min(ai))  # el índice IC

[1] 16.67168

> vif(mod2)

velocidad    precio  flexprec  imgfabri  servconj  imgfvent  calidadp
35.695288 31.696869  1.646636  2.884906 43.396533  2.700327  1.614434
```

Simplificación del modelo: selección de variables

El modelo que hemos ajustado anteriormente incorpora 7 variables explicativas. Ahora buscamos posibles simplificaciones del modelo considerando solo las variables que realmente suponen una contribución significativa a la hora de describir la variable de respuesta. Esto lo podemos hacer utilizando métodos automáticos de selección de variables como los algoritmos paso a paso (*stepwise*).

A continuación vamos a aplicar un algoritmo de este tipo basado en el criterio de Akaike (AIC) para la selección de modelos. La función `step` implementa el algoritmo en el que paso a paso se permite que las variables entren y salgan atendiendo a los valores del AIC. El algoritmo termina ofreciendo el mejor modelo que será aquel que tenga el menor valor del AIC.

Con el último ajuste realizado (objeto `mod2`) escribimos:

```
> step(mod2)

Start:  AIC=281.44
fidelidad ~ velocidad + precio + flexprec + imgfabri + servconj +
    imgfvent + calidadp

            Df Sum of Sq    RSS    AIC
- velocidad  1      1.38 1472.2 279.54
- precio     1      4.63 1475.5 279.75
- imgfabri   1      6.88 1477.7 279.90
<none>                  1470.8 281.44
- calidadp   1     47.79 1518.6 282.58
- servconj   1    103.69 1574.5 286.12
- imgfvent   1    104.01 1574.8 286.14
- flexprec   1   1477.38 2948.2 347.59

Step:  AIC=279.54
fidelidad ~ precio + flexprec + imgfabri + servconj + imgfvent +
    calidadp

           Df Sum of Sq    RSS    AIC
```

```
- imgfabri  1      6.11 1478.3 277.94
- precio    1     12.43 1484.6 278.36
<none>                  1472.2 279.54
- calidadp  1     47.69 1519.9 280.66
- imgfvent  1    102.87 1575.1 284.16
- flexprec  1   1478.35 2950.6 345.67
- servconj  1   1805.21 3277.4 355.96

Step:  AIC=277.94
fidelidad ~ precio + flexprec + servconj + imgfvent + calidadp

           Df Sum of Sq    RSS    AIC
- precio    1     13.05 1491.4 276.80
<none>                  1478.3 277.94
- calidadp  1     46.45 1524.8 278.97
- imgfvent  1    165.64 1644.0 286.35
- flexprec  1   1500.56 2978.9 344.60
- servconj  1   1807.08 3285.4 354.20

Step:  AIC=276.8
fidelidad ~ flexprec + servconj + imgfvent + calidadp

           Df Sum of Sq    RSS    AIC
<none>                  1491.4 276.80
- calidadp  1     33.89 1525.3 277.01
- imgfvent  1    169.54 1660.9 285.35
- flexprec  1   2126.24 3617.6 361.64
- servconj  1   2893.83 4385.2 380.50

Call:
lm(formula = fidelidad ~ flexprec + servconj + imgfvent + calidadp,
    data = hatco)
```

```
Coefficients:
(Intercept)     flexprec     servconj     imgfvent     calidadp
   -13.3173       3.7927       7.5142       1.8377       0.4223
```

Observa que el algoritmo por defecto comienza con el modelo completo (7 variables explicativas). Nos muestra el AIC correspondiente, así como los valores que tomaría eliminando una de las variables (observa el signo – delante del nombre de la variable explicativa). Si hay alguna reducción, se pasa a la segunda iteración eliminado la variable que produce la mayor reducción del AIC. El algoritmo continua en la medida en que se observen reducciones del AIC, terminado cuando no se produzcan. En este caso el algoritmo termina seleccionando 4 variables explicativas.

El algoritmo que hemos aplicado antes es del tipo denominado *backward*, esto es, comienza con el modelo completo y paso a paso va eliminado variables. Esto corresponde al valor por defecto del argumento `direction` de la función `stepwise`. A continuación utilizamos un procedimiento combinado *backward-forward* usando el argumento `direction='both'`.

```
> step(mod2,direction='both')

Start:  AIC=281.44
fidelidad ~ velocidad + precio + flexprec + imgfabri + servconj +
    imgfvent + calidadp

            Df Sum of Sq    RSS    AIC
- velocidad  1      1.38 1472.2 279.54
- precio     1      4.63 1475.5 279.75
- imgfabri   1      6.88 1477.7 279.90
<none>                  1470.8 281.44
- calidadp   1     47.79 1518.6 282.58
- servconj   1    103.69 1574.5 286.12
- imgfvent   1    104.01 1574.8 286.14
- flexprec   1   1477.38 2948.2 347.59

Step:  AIC=279.54
```

```
fidelidad ~ precio + flexprec + imgfabri + servconj + imgfvent +
    calidadp

            Df Sum of Sq    RSS    AIC
- imgfabri   1      6.11 1478.3 277.94
- precio     1     12.43 1484.6 278.36
<none>                   1472.2 279.54
- calidadp   1     47.69 1519.9 280.66
+ velocidad  1      1.38 1470.8 281.44
- imgfvent   1    102.87 1575.1 284.16
- flexprec   1   1478.35 2950.6 345.67
- servconj   1   1805.21 3277.4 355.96

Step:  AIC=277.94
fidelidad ~ precio + flexprec + servconj + imgfvent + calidadp

            Df Sum of Sq    RSS    AIC
- precio     1     13.05 1491.4 276.80
<none>                   1478.3 277.94
- calidadp   1     46.45 1524.8 278.97
+ imgfabri   1      6.11 1472.2 279.54
+ velocidad  1      0.61 1477.7 279.90
- imgfvent   1    165.64 1644.0 286.35
- flexprec   1   1500.56 2978.9 344.60
- servconj   1   1807.08 3285.4 354.20

Step:  AIC=276.8
fidelidad ~ flexprec + servconj + imgfvent + calidadp

            Df Sum of Sq    RSS    AIC
<none>                   1491.4 276.80
- calidadp   1     33.89 1525.3 277.01
+ precio     1     13.05 1478.3 277.94
```

```
+ velocidad  1     10.39 1481.0 278.12
+ imgfabri   1      6.73 1484.6 278.36
- imgfvent   1    169.54 1660.9 285.35
- flexprec   1   2126.24 3617.6 361.64
- servconj   1   2893.83 4385.2 380.50

Call:
lm(formula = fidelidad ~ flexprec + servconj + imgfvent + calidadp,
    data = hatco)

Coefficients:
(Intercept)     flexprec     servconj     imgfvent     calidadp
   -13.3173       3.7927       7.5142       1.8377       0.4223
```

Por otro lado también es posible utilizar la función `update` y realizar una selección manual paso a paso, incluyendo o descartando variables. Esta opción es muy interesante sobre todo si se detectan relaciones no lineales en algunas variables explicativas, incluyendo por ejemplo términos cuadráticos, o interacciones. El libro de Crawley (2015) que mencionamos a continuación proporciona ejemplos donde se hace uso de estas opciones.

Inclusión de una variable cualitativa en el modelo

Además de las variables que hemos utilizado en la regresión, había un factor con dos niveles que identifica el tipo de cliente de la muestra. Los clientes son empresas y se ha registrado si corresponden a empresas pequeñas o grandes. Es razonable pensar que el nivel de fidelidad a la distribuidora pueda ser diferente dependiendo de si se trata de una empresa pequeña o grande. Por tanto vamos a incluir esta nueva variable en la regresión y comprobaremos el efecto que tiene en los niveles de fidelidad:

```
mod3<-lm(fidelidad~flexprec+servconj+
           imgfvent+calidadp+tamano,hatco)
summary(mod3)

##
```

```
## Call:
## lm(formula = fidelidad ~ flexprec + servconj + imgfvent + calidadp +
##     tamano, data = hatco)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.334  -2.653   0.084   2.134   8.187
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.8114     4.0896  -3.133 0.002323 **
## flexprec      4.4137     0.3668  12.032  < 2e-16 ***
## servconj      7.9067     0.5459  14.484  < 2e-16 ***
## imgfvent      2.1354     0.5457   3.913 0.000174 ***
## calidadp     -0.2626     0.3472  -0.756 0.451429
## tamanopeque  -4.3023     1.3192  -3.261 0.001556 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.812 on 92 degrees of freedom
## Multiple R-squared:  0.8292,Adjusted R-squared:  0.8199
## F-statistic: 89.33 on 5 and 92 DF,  p-value: < 2.2e-16
```

Cuando la función `lm` se encuentra con un factor de dos niveles, como es el caso de `tamano`, crea una nueva variable basada en el segundo nivel del factor. En este caso, podemos ver que el segundo nivel corresponde a empresas pequeñas:

```
levels(hatco$tamano)

## [1] "grande" "peque"
```

Con lo que en el resumen del modelo podemos leer `tamanopeque`. Internamente se está trabajando con una variable binaria que toma el valor 1 cuando la empresa es pequeña y 0 en otro caso. Esto implica que el coeficiente de `tamanopeque` tiene una interpretación diferente

al de las otras variables. Por ejemplo, con esta codificación, un valor negativo del coeficiente asociado nos indica que el nivel de fidelidad es menor (alrededor de un 4.302 %) en las empresas pequeñas que en las grandes. Por otro lado, tenemos que la ecuación del modelo ajustado para las empresas pequeñas vendría dada por:

$$\begin{aligned}\widehat{fidelidad} &= -12.811 + 4.414 * flexprec + 7.907 * servconj + 2.135 * imgfvent \\ &\quad -0.263 * calidadp - 4.302 \\ &= -17.113 + 4.414 * flexprec + 7.907 * servconj + 2.135 * imgfvent \\ &\quad -0.263 * calidadp\end{aligned}$$

comparada con la de las empresas grandes que sería:

$$\begin{aligned}\widehat{fidelidad} &= -12.811 + 4.414 * flexprec + 7.907 * servconj + 2.135 * imgfvent \\ &\quad -0.263 * calidadp\end{aligned}$$

Una última observación es que la variable `calidadp` parece haber perdido significación. Sería deseable aplicar de nuevo el método de selección stepwise anterior para comprobar si la inclusión del nuevo factor nos permite eliminar dicha variable del modelo.

Otros ejemplos y herramientas adicionales

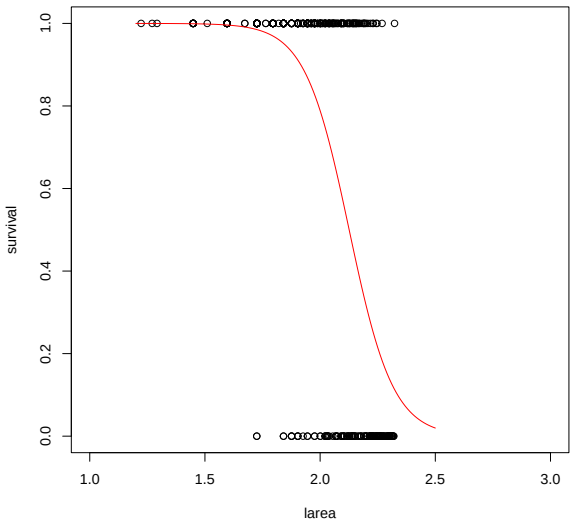
El análisis realizado en este ejemplo sigue los pasos habituales en un análisis de regresión lineal múltiple. Sin embargo hay herramientas adicionales que se pueden utilizar por ejemplo en caso en que existan interacciones complejas entre las variables explicativas o relaciones no lineales. Algunos ejemplos de datos describiendo el uso de herramientas apropiadas en estos casos se puede ver en el capítulo 10 de Crawley (2015) (ver referencia al final del pdf).

1.8. Ejemplo 4: Regresión logística

En los ejemplos de regresión simple y múltiple anteriores se supone que la variable de respuesta es continua, y la inferencia se desarrolla bajo la hipótesis de normalidad de los errores. En el análisis de datos reales es habitual encontrar situaciones donde la respuesta es de tipo binario.

En estas situaciones consideramos una versión más general del modelo lineal que describíamos antes, los denominados modelos lineales generalizados (GLM[15]). La referencia básica de este tipo de modelos es McCullagh y Nelder (1989). Otra referencia útil con un enfoque práctico en R puede ser el libro de Crawley (2015), además del libro Faraway (2006). Ver referencias completas al final del pdf.

A continuación trabajamos con un ejemplo de este tipo. Se trata de datos de 435 pacientes con quemaduras severas para los que se proporciona la variable binaria `survival` asociada a la recuperación (1) o muerte (0). La variable `larea` corresponde al logaritmo del área quemada. El objetivo es describir la supervivencia de los pacientes (variable de respuesta) a partir la variable `larea`.



La gráfica anterior muestra una curva en rojo que podría describir bien la supervivencia según el logaritmo del área afectada (realmente se trata de un ajuste que obtendremos más abajo). Se trata de una función monótona decreciente de 1 a 0, que coincide con la intuición de que cuanto mayor sea el área afectada, menos probabilidad habrá de que el paciente sobreviva. Existe una familia paramétrica (dos parámetros) con dicha forma y que viene dada por la siguiente expresión:

$$p(x) = \frac{1}{1 + \exp(-\alpha - \beta x)}$$

[15]Siglas del inglés *Generalized linear models*

Se trata de la función sigmoide:

$$\begin{aligned} p(x = -\alpha/\beta) &= 0.5 \\ 1 - p(x) &= \frac{\exp(-\alpha - \beta x)}{1 + \exp(-\alpha - \beta x)} \\ \frac{p(x)}{1 - p(x)} &= \exp(\alpha + \beta x) \\ \log\left(\frac{p(x)}{1 - p(x)}\right) &= \alpha + \beta x. \end{aligned}$$

Comparando con el modelo de regresión lineal que consideramos antes, $Y = \alpha + \beta x + \epsilon$, lo que equivale bajo las condiciones habituales sobre el error a, $E[Y|X = x] = \alpha + \beta x$. Dado que ahora la esperanza de la respuesta binaria es una probabilidad, denotando por $\pi(x) = E[Y|X = x]$, el modelo anterior se escribe como:

$$\text{logit}(\pi(x)) = \log \frac{\pi(x)}{1 - \pi(x)} = \alpha + \beta x$$

La función *logit* sirve de enlace (*link*) entre la esperanza condicional y la función lineal del modelo, $\alpha + \beta x$. El cociente $\pi/(1 - \pi)$ corresponde a la probabilidad de "éxito" (*odds*) y tenemos que:

$$o = \frac{\pi}{1 - \pi} \leftrightarrow \pi = \frac{o}{1 - o}$$

En R disponemos de la función `glm` para ajustar este tipo de modelos. Su uso es similar al de la función `lm` que hemos descrito antes. Para el obtener el ajuste que mostrábamos en la figura anterior escribiríamos:

```
bdat <- read.table("burns.dat",header=TRUE)
bfit <- glm(survival~larea, family=binomial(link=logit),data=bdat)
plot(bdat, xlim=c(1,3))
x <- 120:250/100
y <- bfit$coeff[[1]] + bfit$coeff[[2]]*x
y <- exp(y)/(1+exp(y))
lines(x,y,col="red")
```

El siguiente ejemplo es un poco más complejo, involucra variables explicativas que son de tipo factor, y además utilizamos un procedimiento de selección de variables usando la función `step`. Los datos corresponden a 37 pacientes de la enfermedad "injerto-contra-receptor" (GVHD),

que algunos pacientes desarrollan después de recibir un trasplante de médula como tratamiento para la leucemia. Las variables en el conjunto de datos son:

- **recage** edad del receptor en años
- **recsex** sexo del receptor (0=masculino, 1=femenino)
- **donage** edad del donante en años
- **donmfp** código para tipo de donante (0=masculino, 1=femenino, 2=embarazada)
- **type** tipo de leucemia (codificada como 1,2,3)
- **indx** un cociente de dos medidas clínicas
- **gvhd** variable de respuesta (0 = receptor no desarrolla gvhd, 1 = receptor desarrolla gvhd)

A continuación cargamos los datos del fichero `gvhd.dat` y describimos el ajuste del modelo logístico:

```
> gvdat <- read.table("gvhd.dat",header=TRUE)
> gvdat<-within(gvdat,
+               {ftype<-factor(type)
+               fdonmfp<-factor(donmfp)
+               frecsex<-factor(recsex)
+               lindx<-log(indx)})
> glm2 <- glm(gvhd~recage+frecsex+donage+fdonmfp+ftype+lindx,
+             family=binomial(link=logit),data=gvdat)
> summary(glm2)


Call:
glm(formula = gvhd ~ recage + frecsex + donage + fdonmfp + ftype +
    lindx, family = binomial(link = logit), data = gvdat)

Coefficients:
```

```
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -7.98191    4.43781  -1.799   0.0721 .
recage       0.04587    0.09505   0.483   0.6294
frecsex1    -1.39391    1.32308  -1.054   0.2921
donage       0.17627    0.12256   1.438   0.1504
fdonmfp1     1.99698    2.26161   0.883   0.3772
fdonmfp2     1.53591    1.35150   1.136   0.2558
ftype2      -0.68835    1.49453  -0.461   0.6451
ftype3       2.37233    1.64296   1.444   0.1488
lindx        2.14589    0.99726   2.152   0.0314 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 51.049  on 36  degrees of freedom
Residual deviance: 23.504  on 28  degrees of freedom
AIC: 41.504

Number of Fisher Scoring iterations: 6
```

La desvianza (*deviance*) es la medida de la bondad del ajuste en modelos lineales generalizados. Sería equivalente a la suma de cuadrados residual de un modelo lineal, y valores más altos indican peor ajuste. La desvianza base (*null deviance*) sería la del modelo con tan solo la constante y la desvianza residual (*residual deviance*) la del modelo ajustado con las variables explicativas indicadas.

La razón de ventajas (*odds ratios*) permite cuantificar el efecto de las variables explicativas en la respuesta. Considerando el modelo anterior tendríamos:

```
> exp(coef(glm2))

 (Intercept)       recage     frecsex1       donage     fdonmfp1     fdonmfp2
3.415848e-04 1.046941e+00 2.481032e-01 1.192757e+00 7.366801e+00 4.645555e+00
```

```
      ftype2       ftype3        lindx
5.024056e-01 1.072235e+01 8.549654e+00
```

La posible simplificación del modelo la podemos hacer de nuevo usando la función `step` (o hacer actualizaciones de forma manual usando la función `update`):

```
> glm3 <- step(glm2,~recage+frecsex+donage+fdonmfp+ftype+lindx,
+                  dir="both")

Start:  AIC=41.5
gvhd ~ recage + frecsex + donage + fdonmfp + ftype + lindx

          Df Deviance    AIC
- fdonmfp  2   25.340 39.340
- recage   1   23.743 39.743
- frecsex  1   24.691 40.691
<none>         23.504 41.504
- ftype    2   27.710 41.710
- donage   1   25.927 41.927
- lindx    1   30.745 46.745

Step:  AIC=39.34
gvhd ~ recage + frecsex + donage + ftype + lindx

          Df Deviance    AIC
- recage   1   25.868 37.868
- ftype    2   28.480 38.480
<none>         25.340 39.340
- donage   1   27.929 39.929
- frecsex  1   28.540 40.540
+ fdonmfp  2   23.504 41.504
- lindx    1   33.816 45.816

Step:  AIC=37.87
```

```
gvhd ~ frecsex + donage + ftype + lindx

          Df Deviance    AIC
- ftype    2   29.157 37.157
<none>         25.868 37.868
- frecsex  1   28.812 38.812
+ recage   1   25.340 39.340
+ fdonmfp  2   23.743 39.743
- donage   1   31.358 41.358
- lindx    1   35.708 45.708

Step:  AIC=37.16
gvhd ~ frecsex + donage + lindx

          Df Deviance    AIC
- frecsex  1   31.068 37.068
<none>         29.157 37.157
+ ftype    2   25.868 37.868
+ recage   1   28.480 38.480
+ fdonmfp  2   28.047 40.047
- donage   1   35.174 41.174
- lindx    1   45.007 51.007

Step:  AIC=37.07
gvhd ~ donage + lindx

          Df Deviance    AIC
<none>         31.068 37.068
+ frecsex  1   29.157 37.157
+ recage   1   30.668 38.668
+ fdonmfp  2   28.750 38.750
+ ftype    2   28.812 38.812
- donage   1   37.740 41.740
```

```
- lindx    1   45.476 49.476

> summary(glm3)


Call:
glm(formula = gvhd ~ donage + lindx, family = binomial(link = logit),
    data = gvdat)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.45399    2.08147  -2.620  0.00879 **
donage       0.14594    0.06465   2.257  0.02399 *
lindx        2.17773    0.78986   2.757  0.00583 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 51.049  on 36  degrees of freedom
Residual deviance: 31.068  on 34  degrees of freedom
AIC: 37.068

Number of Fisher Scoring iterations: 5
```

Con la función `plot` se pueden crear gráficos para la diagnosis del modelo, así como con las funciones del paquete *car* que usábamos en el ejemplo 3. La interpretación difiere un poco en este caso y se puede consultar en las referencias proporcionadas al final del documento.

1.9. Ejemplo 5: Regresión de Poisson

La denominada regresión de Poisson permite describir variables de respuesta correspondientes a recuentos. Asumir que este tipo de datos proceden de una distribución de Poisson se basa

por lo general en consideraciones teóricas, o en análisis exploratorio de los datos. Formalmente se asume que las observaciones de la variable de respuesta, $Y_i \sim \text{Poisson}(\cdot)$, con $E[Y_i] = V[Y_i] = \mu_i$, y se considera el siguiente modelo lineal:

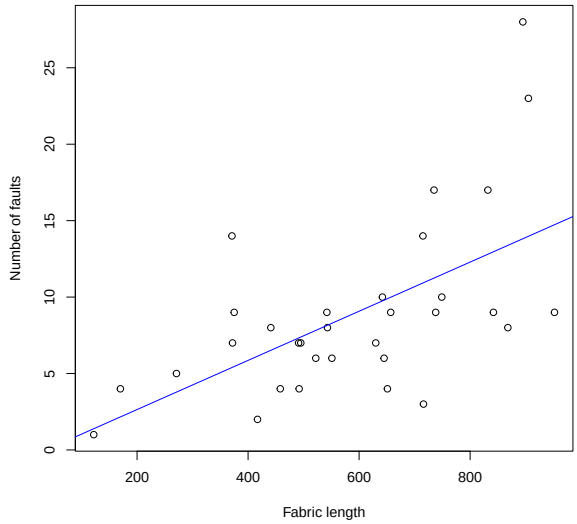
$$E[Y_i|x_i] = \mu_i = \alpha + \beta x_i$$

Este tipo de modelo se puede ver también como un modelo lineal generalizado y se puede ajustar utilizando la función `glm`. De nuevo una referencia útil con un enfoque práctico en R es Crawley (2015) y Faraway (2006). Aquí describimos un ejemplo con un conjunto de datos consistente en el número de defectos (`faults`) encontrados en muestras de telas de distinta longitud (`len`), que se producen en una determinada fábrica. Los datos se recogen en fichero `fabric.dat` y se muestran a continuación:

len	faults	len	faults	len	faults	len	faults
551	6	543	8	491	7	738	9
651	4	842	9	372	7	371	14
832	17	905	23	645	6	735	17
375	9	542	9	441	8	749	10
715	14	522	6	895	28	495	7
868	8	122	1	458	4	716	3
271	5	657	9	642	10	952	9
630	7	170	4	492	4	417	2

Veamos primero el resultado que obtendríamos si ignoramos que nuestros datos no son de tipo continuo y ajustamos un modelo de regresión estándar como en el ejemplo 1:

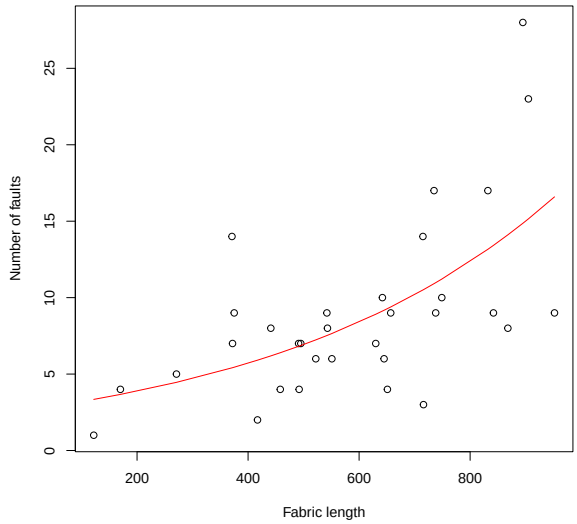
```
> fabric<-read.table("fabric.dat",header=TRUE)
> plot(fabric,xlab="Fabric length", ylab="Number of faults")
> mf<-lm(faults~len,data=fabric)
> abline(mf, col="blue")
```



Ahora ajustamos un modelo de regresión de Poisson usando la función `glm` como sigue:

```
> fabfit<-glm(faults ~ len, family = poisson(link = log),data=fabric)
> summary(fabfit)


Call:
glm(formula = faults ~ len, family = poisson(link = log), data = fabric)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) 0.9717506  0.2124693   4.574 4.79e-06 ***
len         0.0019297  0.0003063   6.300 2.97e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 103.714  on 31  degrees of freedom
Residual deviance:  61.758  on 30  degrees of freedom
AIC: 189.06
```

```
Number of Fisher Scoring iterations: 4

> # Representamos el ajuste con los datos
> plot(fabric,xlab="Fabric length", ylab="Number of faults")
> x<-sort(fabric$len)
> py<-predict.glm(fabfit,data.frame(len=x),type="response")
> lines(x,py,col="red")
```



El modelo que hemos ajustado es un GLM con el logaritmo como función de enlace (*link*), esto nos asegura que todos los valores ajustados son positivos. Además el modelo asume que los errores tienen distribución de Poisson lo que parece adecuado con datos de tipo entero, donde se puede asumir que la varianza sea igual a la media. Otra posibilidad, en caso de sobre-dispersión sería considerar errores de tipo *quasipoisson*. En Crawley (2015, capítulo 13) se describe un ejemplo sencillo de este tipo, además de herramientas para la evaluación del ajuste y el diagnóstico del modelo.

1.10. Otros modelos en R

En R es posible ajustar modelos de regresión más flexibles que los que hemos descrito antes. Algunas de las funciones y paquetes disponibles se citan a continuación:

- Ajuste de modelos paramétricos no lineales: Función `nls`.
- Modelos con efectos aleatorios: Paquete `nlme`.
- Métodos de regularización (regresión ridge, Lasso): Paquete `glmnet`.
- Modelos no paramétricos y suavizado: Función `loess`, paquetes `KernSmooth`, `sm` y `np`, entre otros.
- Modelos aditivos generalizados (GAM): Función `gam` del paquete `mgcv`.
- Métodos de Machine Learning (Random Forest, Boosting, redes neuronales etc.): Paquetes `rpart`, `randomForest`, `xgboost`, etc.

Se recomienda también inspeccionar la herramienta *CRAN task views* para una descripción más completa y actualizada de estos y otros paquetes de R, así como referencias relacionadas con modelos de regresión y Machine Learning. En particular `https://cran.r-project.org/web/views/Econometrics.html` y `https://cran.r-project.org/web/views/MachineLearning.html`.

1.11. Referencias y enlaces

1. Crawley, M.J. (2015). Statistics: An Introduction Using R. Kindle Edition.

   Un pdf gratuito del libro completo se puede descargar en `https://minerva.it.manchester.ac.uk/~saralees/statbook4.pdf`. Los datos usados en los ejemplos, así como el código en R, se pueden descargar en `http://www.bio.ic.ac.uk/research/crawley/statistics/`.

2. Drapper, N.R. y Smith, H. (1981). Applied Regression Analysis. John Wiley & Sons, New York.

3. Faraway, J.J. (2004). *Linear Models with R*. Chapman & Hall/CRC Texts in Statistical Science.

   Un pdf gratuito del libro completo se puede descargar en `https://www.utstat.toronto.edu/~brunner/books/LinearModelsWithR.pdf`.

4. Faraway, J.J. (2006). Extending the linear model with R. Chapman& Hall.

   Materiales adicionales del autor y el paquete *faraway* se pueden descargar en `https://julianfaraway.github.io/faraway/ELM/`. Una versión abreviada está disponible en pdf `https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf`. Un pdf gratuito del libro completo, datos y otros materiales relacionados se pueden descargar en `https://github.com/robjhyndman/ETC3580/tree/master`.

5. McCullagh, P. y Nelder, J.A. (1983, 1989). Generalized Linear Models, Chapman&Hall, London.

   Un pdf gratuito del libro completo se puede descargar en `https://www.utstat.toronto.edu/brunner/oldclass/2201s11/readings/glmbook.pdf`