Práctica 2

***Alumno**: Francisco Javier Piqueras Martínez*
***Asignatura**: Modelado Estadístico de Datos*
***Fecha de entrega**: 7 de enero de 2020*

Índice

1. Descripción del documento

Este documento consiste en la realización de la Práctica 2 de la asignatura de MED (Modelado Estadístico de Datos).

2. Ejercicios

2.1. Ejercicio 1

1. **Enunciado**: *Se ha realizado un estudio para ver si la utilización del biomarcador "acid phosphatase" en sangre (x100) (exp) influye a la hora de detectar la presencia de nódulos infectados (rta=1: sí, rta=0: no). Para ello se han tomado 20 individuos con nódulos infectados y 33 sin nódulos infectados. Los datos experimentales se dan en Le (2006) y se han reproducido en la tabla 1. Además, dichos datos se pueden encontrar en el fichero le.txt alojado en el curso virtual. Se pide rellenar la tabla 2 y calcular a partir de dicha tabla el auc. Adicionalmente, utilizando R, dibujar la curva roc.*

   Resolución:

Para la resolución de este ejercicio, vamos a basarnos en la siguiente tabla:

	X=1	X=0	
Y=1	a	b	r_1
Y=0	c	d	r_2
	s_1	s_0	n

Completaremos esta tabla para cada valor de 'c'.

Por ejemplo, para c=75.5, la tabla quedaría así:

	X=1	X=0	
Y=1	**10**	10	20
Y=0	7	**26**	33
	17	36	53

En la diagonal principal, obtendremos los valores correspondientes a "Enfermo" y "Sano" respectivamente.

Además, podemos calcular ffp(c) siguiendo la fórmula:

$$ffp(c) = 1 - \frac{d}{r_0} = 1 - \frac{d}{33}$$

Y también podemos calcular ffp(c) siguiendo la fórmula:

$$fvp(c) = se = \frac{a}{r_1} = \frac{a}{20}$$

A partir de estos dos valores ffp, fvp, también podemos calcular las columnas "Sano" y "Enfermo" siguiendo la ecuación que hemos hecho hasta ahora:

$$Sano = d = r_0(1 - ffp(c)) = 33(1 - ffp(c)) = 33 - 33ffp(c)$$
$$Enfermo = a = 20fvp(c)$$

Por lo tanto, para completar la tabla, aplicaremos estas fórmulas en función de la columna que necesitemos calcular sin necesidad de que tengamos que completar la tabla inicial.

Finalmente, para calcular el $\widehat{auc(c)}$:

$$\widehat{auc(c)} = (ffp_i - ffp_{i-1})\frac{se_i + se_{i-1}}{2} = (ffp_i - ffp_{i-1})\frac{fvp_i + fvp_{i-1}}{2}$$

A la hora se completar la tabla, se va a mostrar un ejemplo de cálculo para cada columna, suponiendo que el resto se han calculado siguiendo la misma fórmula:

c	Sano	Enf	ffp(c)	fvp(c)	$\widehat{auc(c)}$
39.0	**0**	20	1	1	0.061
43.0	**2**	20	0.939	1	0.030
46.5	**3**	20	0.909	1	0.030
47.5	**4**	20	0.879	1	0.059
48.5	**6**	19	0.818	0.950	0.056
49.5	**8**	18	0.758	0.900	0.136
50.5	**13**	18	0.606	0.900	**0.000**
51.5	**13**	17	0.606	0.850	**0.052**
53.5	**15**	17	0.545	0.850	**0.051**
55.5	**17**	17	0.485	0.850	**0.025**
57.5	18	**16**	0.455	0.800	**0.025**
60.5	19	**16**	0.424	0.800	**0.048**
62.5	21	**16**	0.364	0.800	**0.025**
64.0	22	**16**	0.333	0.800	**0.024**
65.5	23	**16**	0.303	0.800	**0.024**
66.5	24	**16**	0.273	0.800	**0.000**
68.5	24	**13**	0.273	0.650	0.000
70.5	24	**11**	0.273	0.550	0.017
71.5	25	**11**	0.242	0.550	0.000
73.5	25	**10**	0.242	0.500	0.015
75.5	26	**10**	0.212	0.500	0.014
77.0	27	9	**0.182**	0.450	0.013

79.5	28	8	**0.152**	0.400	0.000
81.5	28	7	**0.152**	0.350	0.000
82.5	28	5	**0.152**	0.250	0.008
83.5	29	5	**0.121**	0.250	0.000
86.5	29	4	**0.121**	0.200	0.000
92.0	29	3	**0.121**	0.150	0.005
96.5	30	3	**0.091**	0.150	0.005
98.5	31	3	**0.061**	0.150	0.000
100.5	31	2	**0.061**	0.100	0.003
114.0	32	2	0.030	**0.100**	0.000
131.0	32	1	0.030	**0.050**	0.000
161.5	32	0	0.030	**0.000**	0.000
188.0	32	0	0.000	**0.000**	0.000

Ejemplo de cálculo “Sano” para c=47.5:

$$Sano = d = 33 - 33ffp(47.5) = 33 - 33 * 0.879 = 3.993 \approx 4$$

Ejemplo de cálculo “Enf” para c=62.5:

$$Enf = a = 20fvp(62.5) = 20 * 0.800 = 16$$

Ejemplo de cálculo “ffp” para c=79.5:

$$ffp(79.5) = 1 - \frac{d}{33} = 1 - \frac{28}{33} = 0.152$$

Ejemplo de cálculo “fvp” para c=114.0:

$$fvp(114) = \frac{a}{20} = \frac{2}{20} = 0.100$$

Ejemplo de cálculo “$\widehat{auc(c)}$” para c=51.5:

$$\widehat{auc(51.5)} = \big(ffp(51.5) - ffp(53.5)\big)\frac{fvp(51.5) + fvp(53.5)}{2} = 0.061\frac{1.7}{2} = 0.052$$

Para calcular el auc, solo tenemos que hacer el sumatorio de $\widehat{auc(c)}$ para todo c:

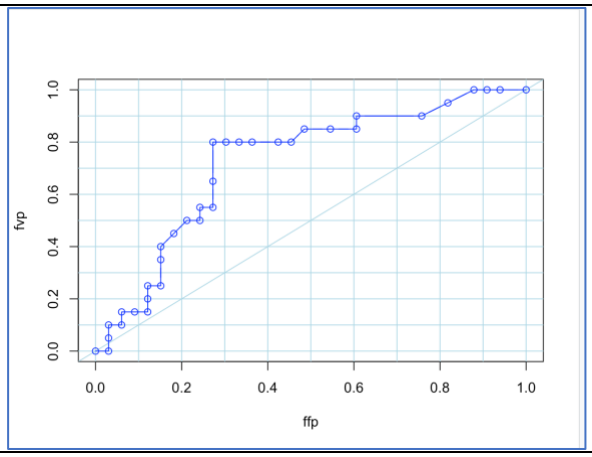
$$AUC = \sum \widehat{auc(c)} = 0.726$$

Finalmente, se ha hecho uso de R para dibujar la curva ROC:

```
rm(list=ls())
# We read the data
datos=read.table("./UNED/MASTER-INGENIERIA-CIENCIA-DATOS/MED/tp2/data/le.txt",
header=T)
attach(datos)
# We load the library pROC
library(pROC)

# We create the ROC object
roc_obj <- roc(datos$rta, datos$exp)
# We print the area under curve
auc(roc_obj)
# and we plot the roc curve
roc_df <- data.frame(
  fvp=rev(roc_obj$sensitivities),
  ffp=rev(1 - roc_obj$specificities))
plot(0:10/10, 0:10/10, type='n', xlab="ffp", ylab="fvp")
abline(h=0:10/10, col="lightblue")
abline(v=0:10/10, col="lightblue")
abline(coef = c(0,1), col="lightblue")
with(roc_df, {
  lines(ffp, fvp, type='l', lwd=1, col="blue")
  lines(ffp, fvp, type='b', lwd=1, col="blue")
})
```

Que imprime lo siguiente:

```
> rm(list=ls())
> # We read the data
>       datos=read.table("./UNED/MASTER-INGENIERIA-CIENCIA-DATOS/MED/tp2/data/le.txt",
header=T)
> attach(datos)
The following objects are masked from datos (pos = 5):

    exp, rta

> # We load the library pROC
> library(pROC)
>
> # We create the ROC object
> roc_obj <- roc(datos$rta, datos$exp)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> # We print the area under curve
> auc(roc_obj)
Area under the curve: 0.725
> # and we plot the roc curve
> roc_df <- data.frame(
+   fvp=rev(roc_obj$sensitivities),
+   ffp=rev(1 - roc_obj$specificities))
> plot(0:10/10, 0:10/10, type='n', xlab="ffp", ylab="fvp")
> abline(h=0:10/10, col="lightblue")
> abline(v=0:10/10, col="lightblue")
> abline(coef = c(0,1), col="lightblue")
> with(roc_df, {
+   lines(ffp, fvp, type='l', lwd=1, col="blue")
+   lines(ffp, fvp, type='b', lwd=1, col="blue")
+ })
>
```



2.2. Ejercicio 2

Enunciado: *Se pide analizar los datos de la tabla 1 con regresión logística evaluando su comportamiento a través de la curva roc. ¿Cuál es la relación con el ejercicio anterior?*

Resolución:

Para analizar los datos de la tabla 1 con regresión logística, se va a hacer uso de R.

Este código contiene las explicaciones comentadas de lo que se ha ido realizando.

En primer lugar, se leen los datos de la ruta donde está ubicado el archivo de datos y se cargan las librerías que se van a necesitar para la realización del ejercicio.

```
rm(list=ls())
# We load the 'InformationValue' library
library(InformationValue)
# We load the 'pROC' library
library(pROC)
# First of all, we read the file that contains the data
train_data=read.table('./UNED/MASTER-INGENIERIA-CIENCIA-
DATOS/MED/tp2/data/le.txt',header=T)
# and we print it to take a look at the data
head(train_data)
table(train_data)
```

Se entrena un modelo de regresión logística con una variable explicativa(exp) y una variable respuesta(rta).

```
# We train our logistic regression model, binomial as a family indicates R to run a
logistic regression
model.lr=glm(train_data$rta ~ train_data$exp, data = train_data, family = "binomial")
# and we print the coefficients of our model
coef(model.lr)
#(Intercept) train_data$exp
#-1.92703240     0.02040076
```

Se va a tratar de realizar una predicción con los mismos datos con los que se ha entrenado el modelo para ver las probabilidades que el modelo nos devuelve para cada variable explicativa.

```
# Now, let's predict our model.
model.prob=predict(model.lr, train_data, type="response")
# and we take a look at them
model.prob
```

A continuación, se va a tomar como decisión de corte o "threshold" el porcentaje del 50% que decidirá si se predice un individuo como enfermo o no.

```
# Now, let's transform all probabilities > 0.5 into 1, and all probabilities <0.5 into 0
model.pred=rep(0, 53)
model.pred[model.prob > 0.5] = 1
```

Como se puede apreciar en la matriz de confusión, a pesar de lo acertada que sea la predicción, para la variable respuesta 1(el individuo está enfermo) se están prediciendo 17 individuos como sanos cuando realmente no lo son y solamente 3 como enfermo. Es decir, la diagonal principal no contiene valores muy elevados comparados con la diagonal opuesta, por lo que los valores ffp y fvp no van a ser nada buenos.

```
# and we take a look at the confussion matrix
table(model.pred, rta)
#             rta
# model.pred  0  1
#          0  29 17
#          1  4  3
# In the main diagonal represent the correct predictions.
# The off-diagonal represents the incorrect predictions.
```

	X=1	X=0	
Y=1	**3**	17	20
Y=0	4	**29**	33
	7	**46**	53

$$es = \frac{d}{r_0} = 0.879$$

$$se = \frac{a}{r_1} = 0.150$$

Es necesario encontrar el "threshold" óptimo.

A continuación, se ve cual es el error de clasificación errónea: 36.5.

```
# Let's compute the fraction of correct classifications
mean(model.pred == rta)
# 0.6037736
# And the missclassification error
misClassError(train_data, model.prob, threshold = 50)
# 36.5
```

Para realizar una mejora en la predicción de enfermedad, se va a ver cual es el punto óptimo de corte para decidir si un individuo está enfermo o no dadas las probabilidades:

```
# Now, let's compute what would be the optimal cut-off to reduce the missclassiffication
error
model.optCutOff=optimalCutoff(train_data, model.prob)
model.optCutOff
# 0.3585171
```

Y se realiza una nueva predicción con este nuevo "threshold".

```
# So let's see the results with this new threshold
model.pred2=rep(0, 53)
model.pred2[model.prob > model.optCutOff] = 1
```

Como se puede observar, a pesar de que el error de clasificación no haya sido reducido en gran valor, hemos mejorado bastante:

```
# and we take a look at the confussion matrix
table(model.pred2, rta)
#             rta
# model.pred  0  1
#          0  23 4
#          1  10  16
# The main diagonal represents the correct predictions.
# The off-diagonal represents the incorrect predictions.
```

	X=1	X=0	
Y=1	**16**	4	20
Y=0	10	**23**	33
	26	**27**	53

$$es = \frac{d}{r_0} = 0.697$$

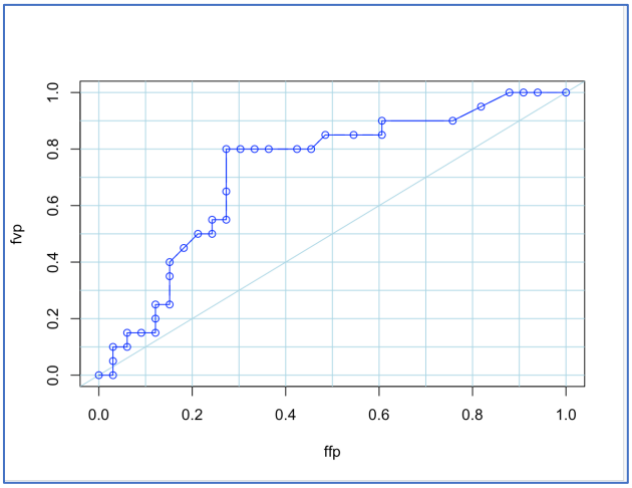
$$se = \frac{a}{r_1} = 0.800$$

```
# Let's compute the fraction of correct classifications
mean(model.pred2 == rta)
# 0.7358491
# And the missclassification error
misClassError(train_data, model.prob, threshold = model.optCutOff)
# 33.5
```

A continuación, se va a calcular cual es el AUC y se va a imprimir la curva ROC basado en las probabilidades en lugar de en los valores que puede tomar x tal y como hemos hecho en el ejercicio anterior.

```
Roc_obj=roc(train_data$rta, model.prob)
auc(roc_obj)
# Area under the curve: 0.725
```

```
Roc_df <- data.frame(
  fvp=rev(roc_obj$sensitivities),
  ffp=rev(1 – roc_obj$specificities))
plot(0:10/10, 0:10/10, type='n', xlab="ffp", ylab="fvp")
abline(h=0:10/10, col="lightblue")
abline(v=0:10/10, col="lightblue")
abline(coef = c(0,1), col="lightblue")
with(roc_df, {
  lines(ffp, fvp, type='l', lwd=1, col="blue")
  lines(ffp, fvp, type='b', lwd=1, col="blue")
})
```



Como se puede observar, el resultado es exactamente el mismo, la única diferencia es que esta vez, para dibujar la curva roc nos hemos basado en las probabilidades en lugar de en los valores de 'x', no obstante, el valor de la curva va a depender de los predictores y la variable respuesta.

2.3. Ejercicio 3

Enunciado: *Se pide calcular el or y su IC para los datos del fichero d_d_3.txt, que se encuentran resumidos en la tabla 3. Adicionalmente, utilizando R, analícese dichos datos con regresión logística ¿Cuál es la relación con el cálculo del or y de su IC? Compruébese que:*

$$\exp(\widehat{\beta_0}) = lrn\frac{prev.muestral}{1 - prev.muestral}$$

Resolución:

	X=1	X=0	
Y=1	**21**	16	37
Y=0	8	**31**	39
	29	47	76

$$OR \equiv Odds\ ratio$$

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

Utilizando el teorema de Bayes, es fácil ver que:

$$\widehat{OR} = \frac{\frac{\widehat{\pi_{11}}}{1-\widehat{\pi_{11}}}}{\frac{\widehat{\pi_{10}}}{1-\widehat{\pi_{10}}}} \rightarrow \widehat{or} = \frac{\frac{\frac{a}{s_1}}{\frac{c}{s_1}}}{\frac{\frac{b}{s_0}}{\frac{d}{s_0}}} = \frac{ad}{bc} \rightarrow \widehat{or} = \frac{21*31}{16*8} = 5.086$$

Al ser $\widehat{or} \neq 0$, se descarta la hipótesis nula h_0 y se puede afirmar que la variable explicativa influye en la variable respuesta.

Para el cálculo del intervalo de confianza:

$$IC(1-\alpha)\%(or) = \left(\exp\left(Ln\left(\frac{ad}{bc}\right) \pm z_{1-\frac{\alpha}{2}}\sqrt{\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}}\right)\right)$$
$$= \left(\exp\left(Ln(5.086) \pm 1.96\sqrt{\frac{1}{21}+\frac{1}{16}+\frac{1}{8}+\frac{1}{31}}\right)\right)$$
$$= (\exp(1.6265 \pm 1.96 * 0.5171)) = \exp(2.64), \exp(0.613)$$
$$= 14.0132, 1.84596$$

A continuación, utilizando R, tras hacer un modelo de regresión logística, se obtienen los siguientes coeficientes:

```
rm(list=ls())
# We load the 'InformationValue' library
library(InformationValue)
# We load the 'pROC' library
library(pROC)
# First of all, we read the file that contains the data
train_data=read.table('./UNED/MASTER-INGENIERIA-CIENCIA-
DATOS/MED/tp2/data/d_d_3.txt',header=T)
# and we print it to take a look at the data
head(train_data)
table(train_data)
# We train our logistic regression model, binomial as a family indicates R to run a
logistic regression
model.lr=glm(train_data$rta ~ train_data$exp, data = train_data, family = "binomial")
# and we print the summary of our model
summary(model.lr)
```

Que proporciona el siguiente resultado:

```
Call:
glm(formula = train_data$rta ~ train_data$exp, family = "binomial",
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6049  -0.9123  -0.9123   0.8035   1.4680

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.6614     0.3078  -2.149  0.03167 *
train_data$exp   1.6265     0.5171   3.145  0.00166 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 105.306  on 75  degrees of freedom
Residual deviance:  94.446  on 74  degrees of freedom
AIC: 98.446

Number of Fisher Scoring iterations: 4
```

$$\beta_0 = -0.6614$$
$$\beta_1 = 1.6265$$

y el z-value es:

$$z = 3.145,$$

que coincide con:

$$z = \frac{\beta_1}{SE(\beta_1)} = \frac{1.6265}{0.5171} = 3.145$$

Como se puede observar, el valor absoluto de z no se aproxima a 0. Por lo tanto, se rechaza la hipótesis nula h_0 y se llega a la conclusión de que la variable explicativa influye en la variable respuesta.

En cuanto a la relación de éste con el $\widehat{or}$, y su intervalo de confianza, es que gracias a ambos se puede deducir que la variable explicativa influye en la variable respuesta tanto tras realizar un análisis de regresión logística como obteniendo el valor $\widehat{or}$ y su intervalo de confianza, en el que vemos que la variable explicativa es una variable de riesgo sobre la variable respuesta. Además, el cálculo del $\widehat{or}$, también puede hacerse de la siguiente manera:

$$\widehat{or} = e^{\beta_1} = 5.086$$
$$IC(1-\alpha)\%(or) = (\beta_1 \pm 1.96 * SE(\beta_1)) = 14.0132, 1.84596$$

Para finalizar, compruébese que:

$$\exp(\widehat{\beta_0}) = lrn\frac{prev.muestral}{1-prev.muestral} = \frac{1-se}{es}\frac{prev.muestral}{1-prev.muestral} = \frac{b/r_1}{c/r}\frac{r_1/n}{1-r_1/n} = 0.5440 * 0.9487 = 0.5161 = e$$

2.4. Ejercicio 4

Enunciado: *Se supone que los datos experimentales están dados por la tabla 4. En esta situación, se pide demostrar que la función de verosimilitud en el modelo de regresión logística con una única variable explicativa dicotómica es:*

$$l(\beta_0,\beta_1) = (\frac{1}{1+e^{-(\beta_0+\beta_1)}})^a(\frac{1}{1+e^{-\beta_0}})^b(1-\frac{1}{1+e^{-(\beta_0+\beta_1)}})^c(1-\frac{1}{1+e^{-\beta_0}})^d$$

Resolución:

Partimos de la siguiente fórmula:

$$l(\beta_0,\beta_1) = \prod_{i:y=1} p(x_i) \prod_{i':y'=0} \left(1-p(x'_{i'})\right)$$

Y de la siguiente tabla (puesto que estamos en un modelo con una única variable explicativa dicotómica):

	X=1	X=0	
Y=1	a	b	r_1
Y=0	c	d	r_2
	s_1	s_0	n

Sustituyendo la fórmula inicial, se obtienen cuatro partes. Dos correspondientes al primer operador (en los que Y=1) y dos correspondientes al segundo operador (en los que Y=0).

Por un lado, tenemos $p(x_1)$, *a* veces para Y=1, $p(x_0)$, b veces para Y=1, $p(x'_1)$, c veces para Y=0 y $p(x'_1)$, d veces para Y=0:

$$l(\beta_0,\beta_1) = (p(x_1))^a(p(x_0))^b(1-p(x'_1))^c(1-p(x'_0))^d$$

Tomando por separado cada una de las cuatro partes para su simplificación:

$$\left(p(x_1)\right)^a = \left(\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)^a = \left(\frac{e^{\beta_0+\beta_1}}{1+e^{\beta_0+\beta_1}}\right)^a = \left(\frac{\frac{e^{\beta_0+\beta_1}}{e^{\beta_0+\beta_1}}}{\frac{1+e^{\beta_0+\beta_1}}{e^{\beta_0+\beta_1}}}\right)^a$$
$$= \left(\frac{1}{1+e^{-(\beta_0+\beta_1)}}\right)^a$$

$$\left(p(x_0)\right)^b = \left(\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)^b = \left(\frac{e^{\beta_0}}{1+e^{\beta_0}}\right)^b = \left(\frac{\frac{e^{\beta_0}}{e^{\beta_0}}}{\frac{1+e^{\beta_0}}{e^{\beta_0}}}\right)^b = \left(\frac{1}{1+e^{-\beta_0}}\right)^b$$

$$\left(p(x_1)\right)^c = \left(1-\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)^c = \left(1-\frac{e^{\beta_0+\beta_1}}{1+e^{\beta_0+\beta_1}}\right)^c = \left(1-\frac{\frac{e^{\beta_0+\beta_1}}{e^{\beta_0+\beta_1}}}{\frac{1+e^{\beta_0+\beta_1}}{e^{\beta_0+\beta_1}}}\right)^c$$
$$= \left(1-\frac{1}{1+e^{-(\beta_0+\beta_1)}}\right)^c$$

$$\left(p(x_0)\right)^d = \left(1-\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)^d = \left(1-\frac{e^{\beta_0}}{1+e^{\beta_0}}\right)^d = \left(1-\frac{\frac{e^{\beta_0}}{e^{\beta_0}}}{\frac{1+e^{\beta_0}}{e^{\beta_0}}}\right)^d$$
$$= \left(1-\frac{1}{1+e^{-\beta_0}}\right)^d$$

Juntándolo todo, quedaría demostrado que:

$$l(\beta_0,\beta_1) = (\frac{1}{1+e^{-(\beta_0+\beta_1)}})^a(\frac{1}{1+e^{-\beta_0}})^b(1-\frac{1}{1+e^{-(\beta_0+\beta_1)}})^c(1-\frac{1}{1+e^{-\beta_0}})^d$$

2.5. Ejercicio 5

Enunciado: *Se pide demostrar que el modelo de regresión logística es lineal en el logit.*

Resolución:

Véase la siguiente demostración:

$$\begin{aligned}\log\left(\frac{p(x)}{1-p(x)}\right) &= \log\big(p(x)\big) - \log\big(1-p(x)\big)\\ &= \log\left(\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right) - \log\left(1-\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)\\ &= \log\big(e^{\beta_0+\beta_1 X}\big) - \log\big(1+e^{\beta_0+\beta_1 X}\big) - \log\left(\frac{1+e^{\beta_0+\beta_1 X}-e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\right)\\ &= \beta_0+\beta_1 X - \log\big(1+e^{\beta_0+\beta_1 X}\big) + \log\big(1+e^{\beta_0+\beta_1 X}\big) = \beta_0+\beta_1 X\end{aligned}$$

2.6. Ejercicio 6

Enunciado: *Se ha realizado un estudio para ver si el hecho de fumar (fumar=1: Sí, fumar=0: No), tomar café (cafe=1: Sí, cafe=0: No) o tomar un fármaco (trat=1: Sí, trat=0: No) influye en la presencia de una enfermedad (enf=1: Sí, enf=0: No). Han participado en el estudio 90 individuos. Los datos experimentales están en el fichero d ddd.txt alojado en el curso virtual.*

Se pide:

- *En un análisis de regresión logística bivariante, variable a variable explicativa, ¿qué ocurre?*
- *En un análisis de regresión logística multivariante con todas las variables, ¿qué ocurre?*
- *Aplicar al análisis de regresión logística multivariante un proceso de selección automático, ¿qué ocurre?*
- *¿Hay "confusión"? ¿Hay "interacción"?*
- *¿Cuál es la curva roc del modelo final? ¿Cuál es su auc? ¿Cuál es el punto de corte óptimo para utilizar el modelo?*
- *¿Se puede aplicar el análisis discriminante a estos datos? ¿Qué técnica es mejor en este caso?*

Resolución:

En un análisis de regresión logística bivariante, variable a variable explicativa, ¿qué ocurre?

```
rm(list=ls())

# First of_all, we read the file that contains the data
train_data=read.table('./UNED/MASTER-INGENIERIA-CIENCIA-
DATOS/MED/tp2/data/d_ddd.txt',header=T)
```

```
# We train our logistic regression model, binomial as a family indicates R to run a logistic regression
# and we train the model with different predictors one by one. In this case, we are going to train the model
# three times: enf<-fumar, enf<-cafe and enf<-trat
model.lr.smoke=glm(train_data$enf ~ train_data$fumar, data = train_data, family = "binomial")
model.lr.coffee=glm(train_data$enf ~ train_data$cafe, data = train_data, family = "binomial")
model.lr.treatment=glm(train_data$enf ~ train_data$trat, data = train_data, family = "binomial")
# and we print the summary of our models
summary(model.lr.smoke)
summary(model.lr.coffee)
summary(model.lr.treatment)
```

Esto produce el siguiente resultado:

```
> summary(model.lr.smoke)

Call:
glm(formula = train_data$enf ~ train_data$fumar, family = "binomial",
    data = train_data)



Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9728  -0.6335  -0.6335   0.5553   1.8465

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       -1.5041     0.3496  -4.302 1.69e-05 ***
train_data$fumar   3.2958     0.5963   5.527 3.25e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.653  on 89  degrees of freedom
Residual deviance:  80.863  on 88  degrees of freedom
AIC: 84.863

Number of Fisher Scoring iterations: 4

> summary(model.lr.coffee)

Call:
glm(formula = train_data$enf ~ train_data$cafe, family = "binomial",
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.4660  -0.7852  -0.7852   0.9140   1.6290

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.0186     0.3236  -3.148 0.001645 **
train_data$cafe   1.6753     0.4617   3.629 0.000285 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.65  on 89  degrees of freedom
Residual deviance: 109.34  on 88  degrees of freedom
AIC: 113.34

Number of Fisher Scoring iterations: 4

> summary(model.lr.treatment)
```

```
Call:
glm(formula = train_data$enf ~ train_data$trat, family = "binomial",
    data = train_data)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.259  -1.259  -0.840   1.098   1.558

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)      0.1892     0.2760   0.686   0.4929
train_data$trat  -1.0494     0.4533  -2.315   0.0206 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.65  on 89  degrees of freedom
Residual deviance: 118.03  on 88  degrees of freedom
AIC: 122.03

Number of Fisher Scoring iterations: 4
```

Para la variable explicativa `fumar`, se determina que ésta influye en la variable respuesta ya que tiene un valor absoluto de z elevado, distante de cero.

Para la variable explicativa `cafe`, se determina que ésta también influye en la variable respuesta ya que tiene un valor absoluto de z elevado, distante de cero, aunque en menor medida que la variable fumar.

Finalmente, para la variable explicativa `trat`, se determina que ésta también influye, aunque en menor medida que las anteriores. También tiene un valor absoluto de z elevado y distante de cero.

Por lo tanto, se concluye que en un análisis bivariante, todas ellas influyen de forma independiente en la variable respuesta, siendo `fumar` la que más influye y `trat` la que menos.

En un análisis de regresión logística multivariante con todas las variables, ¿qué ocurre?

```
# Now, let's train the model with multiple predictors (enf<-fumar,cafe,trat)
model.lr.multiple=glm(train_data$enf    ~    train_data$fumar    +    train_data$cafe    +
train_data$trat, data  = train_data, family = "binomial")
# and we print the summary of the model
summary(model.lr.multiple)
```

Que produce este resultado:

```
> summary(model.lr.multiple)

Call:
glm(formula = train_data$enf ~ train_data$fumar + train_data$cafe +
    train_data$trat, family = "binomial", data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9838  -0.6443  -0.6140   0.5487   1.9106
```

```
Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.38998    0.52906  -2.627  0.00861 **
train_data$fumar  3.28380    0.74257   4.422 9.77e-06 ***
train_data$cafe  -0.07661    0.73584  -0.104  0.91708
train_data$trat  -0.18290    0.62241  -0.294  0.76887
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.653  on 89  degrees of freedom
Residual deviance:  80.777  on 86  degrees of freedom
AIC: 88.777

Number of Fisher Scoring iterations: 4
```

Aquí, como era de esperar, las cosas han cambiado. Anteriormente todas las variables explicativas eran significativas para la variable respuesta. Ahora, no lo son, o no en tal medida. Predomina la influencia de la variable explicativa `fumar`, que tiene un valor z de 4.422. Mientras que las otras dos tienen un valor z de -0.104 y -0.294.

Aplicar al análisis de regresión logística multivariante un proceso de selección automático, ¿qué ocurre?

```
# The next step is to apply to the multivariate logisitic regression model, un automatic
selection process
model.lr.multiple.step = step(model.lr.multiple)
```

Que produce el siguiente resultado:

```
Start:  AIC=88.78
train_data$enf ~ train_data$fumar + train_data$cafe + train_data$trat

                  Df Deviance     AIC
- train_data$cafe   1   80.788  86.788
- train_data$trat   1   80.863  86.863
<none>                  80.777  88.777
- train_data$fumar  1  108.332 114.332

Step:  AIC=86.79
train_data$enf ~ train_data$fumar + train_data$trat

                  Df Deviance     AIC
- train_data$trat   1   80.863  84.863
<none>                  80.788  86.788
- train_data$fumar  1  118.034 122.034

Step:  AIC=84.86
train_data$enf ~ train_data$fumar

                  Df Deviance     AIC
<none>                  80.863  84.863
- train_data$fumar  1  123.653 125.653
```

Para el proceso de selección automático `step()`, lo que se ve es que se aplica un proceso de selección de variables para el modelo. En este caso se analiza un modelo con las tres variables, cuyo AIC es 88.78. Seguidamente, se descarta la variable menos significativa y se vuelve a entrenar el modelo con las variables `fumar` y `trat`, cuyo AIC es 86.79. Finalmente, solo con la variable `fumar`. El AIC de este último es 84.86.

El que menor AIC tiene es el modelo que solamente considera como variable significativa `fumar`, mientras que el que mayor AIC tiene es el que considera las tres variables.

¿Hay "confusión"? ¿Hay "interacción"?

En primer lugar, se estudia la interacción, para ello, se va a crear una nueva variable producto de `fumar*cafe` y `fumar*trat`. Seguidamente, se va a entrenar dos nuevos modelos: uno con las variables aleatorias `fumar`, `cafe` y `fumar*cafe` y otro con las variables aleatorias `fumar`, `trat` y `fumar*trat`. Para ver si existe interacción entre ellas, se deberá ver si el coeficiente del término producto de ambas es distinto de cero. Una vez hecho esto, se estudiará la confusión.

Para estudiar la interacción con el café, se plantea la siguiente hipótesis:

$$h_o: \beta_3 = 0$$
$$h_1: \beta_3 \neq 0$$

Para ello, se plantea el análisis de un nuevo modelo de regresión logística con las variables aleatorias `fumar`, `cafe` y `fumar*cafe`.

```
# We train the model with fumar*cafe looking for some interaction between variables
model.lr.smokeIcoffee=glm(train_data$enf ~ train_data$fumar + train_data$cafe +
train_data$fumar*train_data$cafe, data = train_data, family = "binomial")
summary(model.lr.smokeIcoffee)
```

Que proporciona la siguiente salida:

```
Call:
glm(formula = train_data$enf ~ train_data$fumar + train_data$cafe +
    train_data$fumar * train_data$cafe, family = "binomial",
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9905  -0.6416  -0.6039   0.5448   1.8930

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                      -1.4759     0.3919  -3.766 0.000166 ***
train_data$fumar                  3.0853     1.1634   2.652 0.008003 **
train_data$cafe                  -0.1335     0.8681  -0.154 0.877749
train_data$fumar:train_data$cafe  0.3567     1.4979   0.238 0.811785
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.653  on 89  degrees of freedom
Residual deviance:  80.807  on 86  degrees of freedom
AIC: 88.807

Number of Fisher Scoring iterations: 4
```

Por lo tanto:

$$\widehat{or}_0 = e^{3.0853} = 21.874$$
$$\widehat{or}_1 = e^{3.0853-0.1333} = 19.1442$$

Como se observa, no hay grandes diferencias significativas entre ellas. También se puede observar que el coeficiente β_3 es muy cercano a 0.

Para ver si este es significativo:

$$|z| = \left|\frac{\beta_3}{\widehat{SE}_{\beta_3}}\right| = 0.7164 < 1.96$$

Aceptamos la hipótesis h_0 y concluimos que no hay interacción.

Para ver si existe confusión, veamos el cambio exponencial de los coeficientes de `fumar`. Tenemos en primer lugar el coeficiente 3.2958 (para el modelo que solo contiene la variable `fumar`) y 3.0853 (para el modelo que estudia la interacción). Al no haber una diferencia mayor a un 10%, se concluye que tampoco existe interacción.

Para estudiar la interacción con el tratamiento, se plantea la siguiente hipótesis de nuevo:

$$h_o: \beta_3 = 0$$
$$h_1: \beta_3 \neq 0$$

Para ello, se plantea el análisis de un nuevo modelo de regresión logística con las variables aleatorias `fumar`, `trat` y `fumar*trat`.

```
# We train the model with fumar*treatment looking for some interaction between variables
model.lr.smokeItreatment=glm(train_data$enf ~ train_data$fumar + train_data$trat +
train_data$fumar*train_data$trat, data = train_data, family = "binomial")
summary(model.lr.smokeItreatment)
```

Que proporciona la siguiente salida:

```
Call:
glm(formula = train_data$enf ~ train_data$fumar + train_data$trat +
    train_data$fumar * train_data$trat, family = "binomial",
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5816  -0.7876  -0.4084   0.2697   2.2475

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                      -2.4423     0.7372  -3.313 0.000923 ***
train_data$fumar                  5.7382     1.2572   4.564 5.01e-06 ***
train_data$trat                   1.4307     0.8449   1.693 0.090398 .
train_data$fumar:train_data$trat  -5.0143     1.5278  -3.282 0.001031 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 123.653  on 89  degrees of freedom
Residual deviance:  66.922  on 86  degrees of freedom
AIC: 74.922

Number of Fisher Scoring iterations: 6
```

Por lo tanto:

$$\widehat{or}_0 = e^{5.7382} = 310.505$$
$$\widehat{or}_1 = e^{5.7382+1.4307} = 1298.4156$$

Como se observa, hay grandes diferencias significativas entre ellas. También se puede observar que el coeficiente β_3 es bastante elevado.

Para ver si este es significativo:

$$|z| = \left|\frac{\beta_3}{\widehat{SE}_{\beta_3}}\right| = 3.282 > 1.96$$

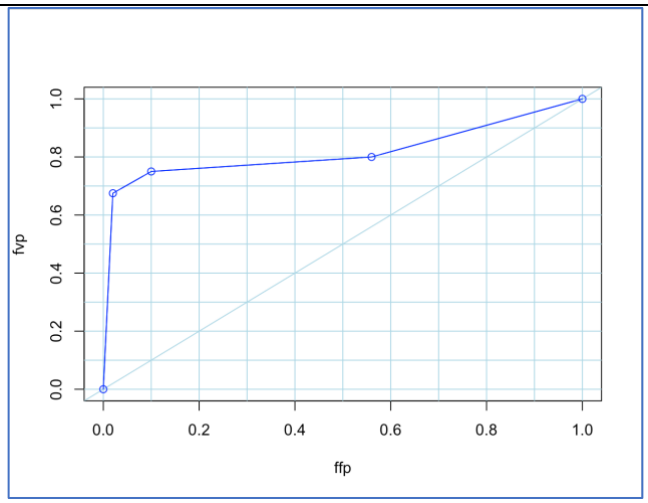
Rechazamos la hipótesis h_0 y concluimos que hay interacción.

Al haber interacción, no tiene sentido estudiar la confusión.

¿Cuál es la curva roc del modelo final? ¿Cuál es su auc? ¿Cuál es el punto de corte óptimo para utilizar el modelo?

El modelo final es el que considera las variables aleatorias fumar y tratamiento.

```
# Final model roc and AUC
model.lr.multiple.smokeTreatment=glm(train_data$enf      ~      train_data$fumar      +
train_data$trat, data  = train_data, family = "binomial")
summary(model.lr.multiple.smokeTreatment)
# Now, let's predict our model.
model.lr.multiple.smokeTreatmentProb=predict(model.lr.multiple.smokeTreatment,
train_data, type="response")
# and we take a look at them
model.lr.multiple.smokeTreatmentProb
# plot roc
roc_obj=roc(train_data$enf, model.lr.multiple.smokeTreatmentProb)
auc(roc_obj)
# Area under the curve: 0.8162

roc_df <- data.frame(
  fvp=rev(roc_obj$sensitivities),
  ffp=rev(1 - roc_obj$specificities))
plot(0:10/10, 0:10/10, type='n', xlab="ffp", ylab="fvp")
abline(h=0:10/10, col="lightblue")
abline(v=0:10/10, col="lightblue")
abline(coef = c(0,1), col="lightblue")
with(roc_df, {
  lines(ffp, fvp, type='l', lwd=1, col="blue")
  lines(ffp, fvp, type='b', lwd=1, col="blue")
})
```



El AUC es 0.8162.

Hay muchas formas de calcular el punto de corte óptimo, de hecho, dependerá de cual es nuestro interés conociendo el modelo. Sin embargo, el que menos error de clasificación tiene calculándolo con R es el siguiente:

```
# Now, let's compute what would be the optimal cut-off to reduce the missclassiffication
error
model.optCutOff=optimalCutoff(train_data, model.lr.multiple.smokeTreatmentProb)
model.optCutOff
```

Que produce la siguiente salida:

```
> model.optCutOff
[1] 0.2013392
```

¿Se puede aplicar el análisis discriminante a estos datos? ¿Qué técnica es mejor en este caso?

El objetivo del análisis discriminante es encontrar la combinación lineal de las variables independientes que mejor permite diferenciar (discriminar) a los grupos.

Una vez encontrada esa combinación (la función discriminante) podrá ser utilizada para clasificar nuevos casos. Se trata de una técnica de análisis multivariante que es capaz de aprovechar las relaciones existentes entre una gran cantidad de variables independientes para maximizar la capacidad de discriminación.

Una de las condiciones para aplicarlo es que las variables aleatorias explicativas deben ser continuas. En este caso, son todas dicotómicas, por lo que éste no sería aplicable.