Regresión logística simple y múltiple

Joaquín Amat Rodrigo *j.amatrodrigo@gmail.com*

Agosto, 2016

Índice

Regresión logística simple

Idea intuitiva

La Regresión Logística Simple, desarrollada por David Cox en 1958, es un método de regresión que estima la probabilidad de una variable cualitativa en función de una variable cuantitativa. Permite estudiar en qué medida variaciones de una variable continua independiente influyen en una variable cualitativa dependiente.

Una de las principales aplicaciones de la regresión logística es el análisis discriminante, en el que una observación se clasifica un grupo u otro dependiendo del valor que tome la variable empleada como predictor. Por ejemplo, clasificar a un individuo desconocido como hombre o mujer en función del tamaño de la mandíbula. Si bien es posible emplear regresión logística para variables cualitativas con más de dos niveles, no es recomendable, en su lugar es preferible emplear *Lineal Discriminant Analysis (LDA), Quadratic Discriminat Analysis (QDA)* o *K-NearestNeighboirs (K-NN).*

La existencia de una relación significativa entre una variable cualitativa con dos niveles y una variable continua se puede estudiar mediante otros test estadísticos tales como *t-test* o *ANOVA* (un ANOVA de dos grupos es equivalente al t-test). Sin embargo, la regresión logística permite calcular además la probabilidad de que la variable dependiente pertenezca a cada una de las dos categorías en función del valor que adquiera la variable independiente.

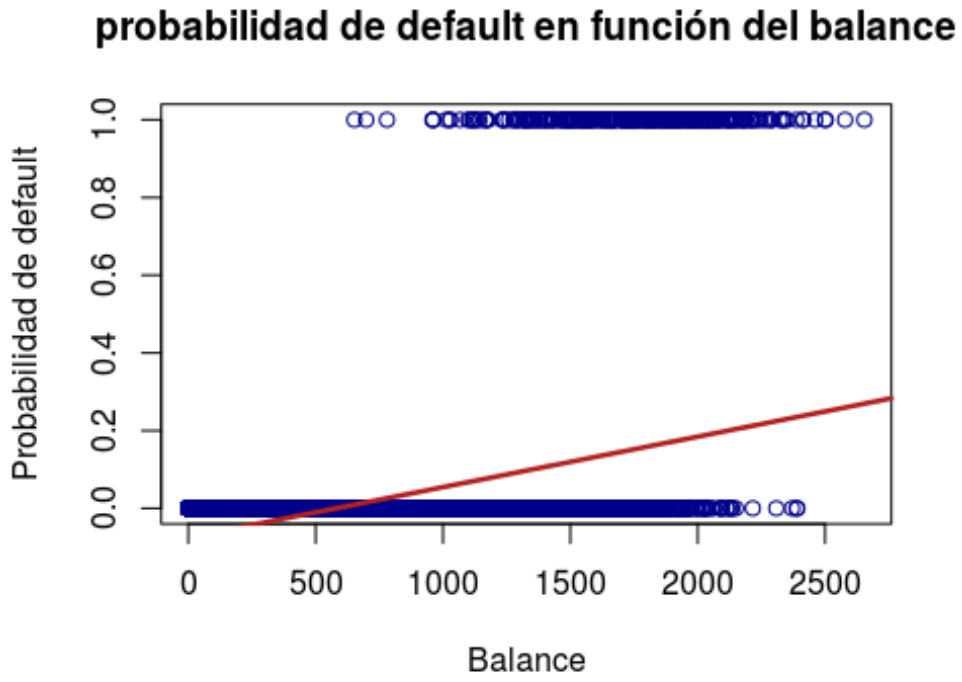
Supóngase que se quiere estudiar la relación entre los niveles de colesterol y los ataques de corazón. Para ello se mide el colesterol de un grupo de personas y durante los siguientes 20 años se monitoriza que individuos han sufrido un ataque. Un *t-test* entre los niveles de colesterol de las personas que han sufrido ataque *vs* las que no lo han sufrido permitiría contrastar la hipótesis de que el colesterol y los ataques al corazón están asociados. Si además se desea conocer la probabilidad de que una persona con un determinado nivel de colesterol sufra un infarto en los próximos 20 años, o poder conocer cuánto tiene que reducir el colesterol un paciente para no superar un 50% de probabilidad de padecer un infarto en los próximos 20 años, se tiene que recurrir a la regresión logística.

A modo orientativo: Si se considera que variaciones en la variable cualitativa influyen en la variable continua, es decir, que la variable cualitativa es la independiente y la variable continua la dependiente, se recomienda utilizar t-test o sus alternativas no paramétricas. Cuando se considera que variaciones en la variable continua causan variaciones en la probabilidad de la variable cualitativa, la variable cualitativa es la dependiente y la continua la independiente, se emplea regresión logística.

¿Por qué regresión logística y no lineal para variables cualitativas?

Si una variable cualitativa con dos niveles se codifica como *1* y *0*, es posible ajustar un modelo de regresión lineal. Al hacerlo, se estaría obteniendo la probabilidad de que la variable dependiente *Y* pertenezca al nivel de referencia empleado por el modelo dado un determinado valor del predictor *X*. El principal problema de esta aproximación es que al tratarse de una recta, para valores extremos del predictor, se obtienen valores de *Y* menores que 0 o mayores que 1, lo que entra en contradicción con el hecho de que las probabilidades siempre están dentro del rango [0,1].

```
require(ISLR)

levels(Default$default) <- c("0", "1")
Default$default <- as.character(Default$default)
Default$default <- as.numeric(Default$default)
modelo_lineal <- lm(default ~ balance, data = Default)
plot(x = Default$balance, y = Default$default, col = "darkblue", main =
"probabilidad de default en función del balance", xlab = "Balance", ylab =
"Probabilidad de default")
abline(modelo_lineal, lwd = 2.5, col = "firebrick")
```



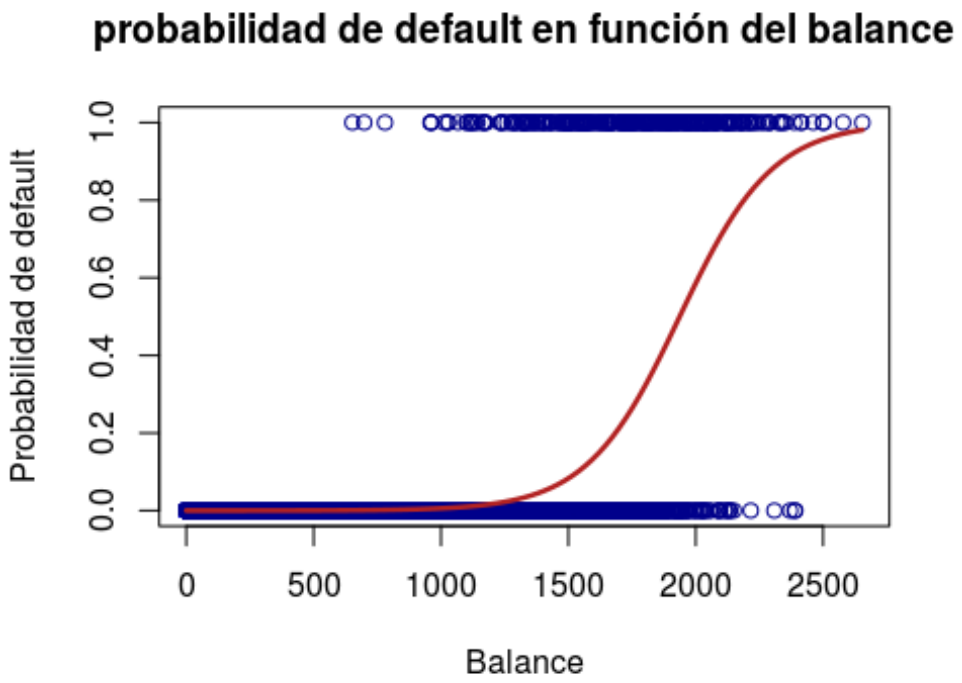
En el caso de que existan múltiples observaciones de la variable dependiente cualitativa para cada valor de la variable independiente continua, se puede calcular la proporción de eventos verdaderos en cada valor y realizar una regresión lineal entre la nueva variable continua (proporción) y el predictor. Sin embargo, esta aproximación tiene la limitación de no tener en cuenta el número de observaciones en cada caso (no se le debe dar el mismo peso a una proporción de 0.5 que proceda de *2/4* que a una de *8/16*).

Para evitar estos problemas, la regresión logística modela la probabilidad de Y usando una función cuyo resultado está siempre comprendido entre *0* y *1* para todos los posibles valores del predictor X. Existen multitud de funciones que cumplen esta descripción, la utilizada en este caso es la función logística:

$$Pr(Y = k|X = x) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

$Pr(Y = k|X = x)$ puede interpretarse como: la probabilidad de que la variable cualitativa Y adquiera el valor k (el nivel de referencia, comúnmente codificado como 1), dado que el predictor X tiene el valor x.

```
modelo_logistico <- glm(default ~ balance, data = Default, family = "binomial")
plot(x = Default$balance, y = Default$default, col = "darkblue", main =
"probabilidad de default en función del balance", xlab = "Balance", ylab =
"Probabilidad de default")
curve(predict(modelo_logistico, data.frame(balance = x), type = "response"), add =
TRUE, col = "firebrick", lwd = 2.5)
```



Para obtener una función lineal a partir de esta ecuación se recurre a logaritmos, generando lo que se conoce como *LOG of ODDs*.

$$log(\frac{p(Y = k|X = x)}{1 - p(Y = k|X = x)}) = \beta_0 + \beta_1 X$$

Concepto de ODDS o razón de probabilidad, ratio de ODDS y logaritmo de ODDS

En el caso de la regresión lineal, se modela el valor de la variable dependiente *Y* en función del valor de la variable independiente *X*. En la regresión logística se modela la probabilidad de que la variable respuesta *Y* pertenezca al nivel elegido como referencia en función del valor que adquieran los predictores, mediante el uso de *LOG of ODDs*.

Supóngase que la probabilidad de un evento sea verdadero es de *0.8*, por lo que la probabilidad de evento falso es de *1 - 0.8 = 0.2*. Los *odds* o *razón de probabilidad* de verdadero se definen como el ratio entre la probabilidad de evento verdadero y la probabilidad de evento falso $\frac{p}{q}$. En este caso los *odds* de verdadero son *0.8 / 0.2 = 4*, lo que equivale a decir que se esperan 4 eventos verdaderos por cada evento falso.

La trasformación de probabilidades a *odds* es monotónica, si la probabilidad aumenta también lo hacen los *odds* y viceversa. El rango de valores que pueden tomar los *odds* es de [0,∞]. Dado que el valor de una probabilidad está acotado entre [0,1] se recurre a una trasformación *logit* (existen otras) que consiste en el logaritmo natural (ln) de los *odds*. Esto permite convertir el rango de probabilidad previamente limitado a [0,1] a [−∞,+∞].

p	odds	logodds
0.001	0.001001	-6.906755
0.01	0.010101	-4.59512
0.2	0.25	-1.386294
0.3	0.4285714	-0.8472978
0.4	0.6666667	-0.4054651
0.5	1	0
0.6	1.5	0.4054651
0.7	2.333333	0.8472978
0.8	4	1.386294
0.9	9	2.197225
0.999	999	6.906755
0.9999	9999	9.21024

Los *odds* y el *logaritmo de odds* cumplen que:

- Si p(verdadero) = p(falso), entonces odds(verdadero) = 1
- Si p(verdadero) < p(falso), entonces odds(verdadero) < 1
- Si p(verdadero) > p(falso), entonces odds(verdadero) > 1
- A diferencia de la probabilidad que no puede exceder el 1, los *odds* no tienen límite superior.
- Si odds(verdadero) = 1, entonces logit(p) = 0
- Si odds(verdadero) < 1, entonces logit(p) < 0
- Si odds(verdadero) > 1, entonces logit(p) > 0
- La transformación *logit* no existe para p = 0

Ajuste del modelo

Una vez obtenida la relación lineal entre el logaritmo de los *odds* y la variable predictora X se tienen que estimar los parámetros β_0 y β_1. Para ello se recurre al *maximum likelihood methos* (en regresión lineal se emplea mínimos cuadrados). El método de *maximum likelihood (ML)* es un proceso computacional/matemático que busca el valor de los parámetros β_0 y β_1 para los cuales se maximiza la probabilidad de obtener las observaciones.

El método de *maximum likelihood*, traducido como máxima verosimilitud, está ampliamente extendido en la estadística aunque su interpretación no siempre es trivial. En el enlace http://www.seh-lelha.org/maxverosim.htm se puede obtener una descripción muy buena sobre este concepto. Y una descripción más detallada en http://statgen.iop.kcl.ac.uk/bgim/mle/sslike_1.html.

Evaluación del modelo

Existen diferentes técnicas estadísticas para calcular la significancia de un modelo logístico en su conjunto (*p-value* del modelo). Todos ellos consideran que el modelo es útil si es capaz de mostrar una mejora respecto a lo que se conoce como modelo nulo, el modelo sin predictores, solo con β_0. Dos de los más empleados son:

- *Wald chi-square*: está muy expandido pero pierde precisión con tamaños muestrales pequeños.

- *Lokelihood ratio*: usa la diferencia entre la probabilidad de obtener los valores observados con el modelo logístico creado y las probabilidades de hacerlo con un modelo sin relación entre las variables. Para ello, calcula la significancia de la diferencia de residuos entre el modelo con predictores y el modelo nulo (modelo sin predictores). El estadístico tiene una distribución *chi-cuadrado* con grados de libertad equivalentes a la diferencia de grados de libertad de los dos modelos comparados. Si se compara respecto al modelo nulo, los grados de libertad equivalen al número de predictores del modelo generado. *En el libro Handbook for biological statistics se recomienda usar este*.

Para determinar la significancia individual de cada uno de los predictores introducidos en un modelo de regresión logística se emplea el estadístico *Z* y el test *Wald chi-test*. En R, este es el método utilizado para calcular los *p-values* que se muestran al hacer `summary()` del modelo.

Interpretación del modelo

A diferencia de la regresión lineal en la que β_1 se corresponde con el cambio promedio en la variable dependiente *Y* debido al incremento en una unidad del predictor *X*, en regresión logística, β_1 indica el cambio en el logaritmo de *odds* debido al incremento de una unidad de *X*, o lo que es lo mismo, multiplica los *odds* por e^{β_1}. Dado que la relación entre *p(Y)* y *X* no es lineal, β_1 no se corresponde con el cambio en la probabilidad de *Y* asociada con el incremento de una unidad de *X*. Cuánto se incremente la probabilidad de *Y* por unidad de *X* depende del valor de *X*, es decir, de la posición en la curva logística en la que se encuentre.

Condiciones

- Independencia: las observaciones tienen que ser independientes unas de otras.
- Relación lineal entre el logaritmo natural de *odds* y la variable continua: patrones en forma de U son una clara violación de esta condición.
- La regresión logística no precisa de una distribución normal de la variable continua independiente.
- Número de observaciones: no existe una norma establecida al respecto, pero se recomienda entre 50 a 100 observaciones.

Predicciones

Una vez estimados los coeficientes del modelo logístico, es posible conocer la probabilidad de que la variable dependiente pertenezca al nivel de referencia dado un determinado valor del predictor. Para ello se emplea la ecuación del modelo:

$$\hat{p}\,(Y = 1|X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}}$$

Comparación de clasificación predicha y observaciones

Una de las principales aplicaciones de un modelo de regresión logística es predecir la clasificación de la variable cualitativa en función de valor que tome el predictor. Para conseguir esta clasificación es necesario establecer un *threshold* de probabilidad a partir de la cual se considera que la variable pertenece a uno de los niveles. Una forma de evaluar la capacidad del modelo para acertar en las clasificaciones es empleando *matrices de confusión*, en las que se recoge el número de verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos. Como se describe en el capítulo *Validación de modelos de regresión*, emplear las predicciones de las observaciones con las que se ha creado el modelo (*trainign error*) no es lo más adecuado, pero aun así ayuda a evaluar la capacidad del modelo.

Ejemplo

Un estudio quiere establecer un modelo que permita calcular la probabilidad de obtener una matrícula de honor al final del bachillerato en función de la nota que se ha obtenido en matemáticas. La variable matrícula está codificada como 0 si no se tiene matrícula y 1 si se tiene.

```
matricula <- as.factor(c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0,
    0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0,
    1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1,
    0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 1, 1,
    1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1))
matematicas <- c(41, 53, 54, 47, 57, 51, 42, 45, 54, 52, 51, 51, 71, 57, 50,
    43, 51, 60, 62, 57, 35, 75, 45, 57, 45, 46, 66, 57, 49, 49, 57, 64, 63,
    57, 50, 58, 75, 68, 44, 40, 41, 62, 57, 43, 48, 63, 39, 70, 63, 59, 61,
    38, 61, 49, 73, 44, 42, 39, 55, 52, 45, 61, 39, 41, 50, 40, 60, 47, 59,
    49, 46, 58, 71, 58, 46, 43, 54, 56, 46, 54, 57, 54, 71, 48, 40, 64, 51,
    39, 40, 61, 66, 49, 65, 52, 46, 61, 72, 71, 40, 69, 64, 56, 49, 54, 53,
    66, 67, 40, 46, 69, 40, 41, 57, 58, 57, 37, 55, 62, 64, 40, 50, 46, 53,
    52, 45, 56, 45, 54, 56, 41, 54, 72, 56, 47, 49, 60, 54, 55, 33, 49, 43,
    50, 52, 48, 58, 43, 41, 43, 46, 44, 43, 61, 40, 49, 56, 61, 50, 51, 42,
    67, 53, 50, 51, 72, 48, 40, 53, 39, 63, 51, 45, 39, 42, 62, 44, 65, 63,
    54, 45, 60, 49, 48, 57, 55, 66, 64, 55, 42, 56, 53, 41, 42, 53, 42, 60,
    52, 38, 57, 58, 65)
```

```r
datos <- data.frame(matricula, matematicas)
head(datos, 4)
```

```
##   matricula matematicas
## 1         0          41
## 2         0          53
## 3         0          54
## 4         0          47
```

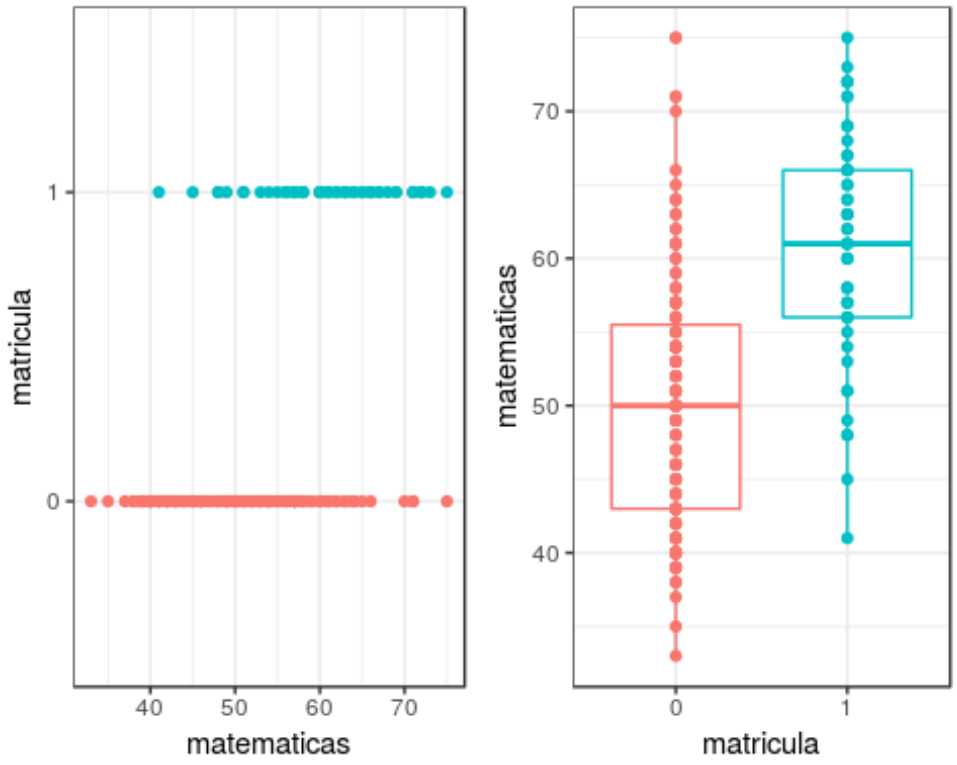
1.Representación de las observaciones

Representar las observaciones es útil para intuir si la variable independiente escogida está relacionada con la variable respuesta y por lo tanto puede ser un buen predictor.

```r
require(ggplot2)
require(gridExtra)
table(datos$matricula)
```

```
##
##   0   1
## 151  49
```

```r
p1 <- ggplot(data = datos, aes(x = matematicas, y = matricula)) +
  geom_point(aes(color = matricula)) +
  theme_bw() + theme(legend.position = "null")

p2 <- ggplot(data = datos, aes(x = matricula, y = matematicas)) +
  geom_boxplot(aes(color = matricula)) +
  geom_point(aes(color = matricula)) +
  theme_bw() +
  theme(legend.position = "null")

grid.arrange(p1, p2, nrow = 1)
```



Parece existir una diferencia entre la nota de las personas con matrícula y sin matrícula.

2.Generar el modelo de regresión logística

```
modelo <- glm(matricula ~ matematicas, data = datos, family = "binomial")
summary(modelo)
```

```
##
## Call:
## glm(formula = matricula ~ matematicas, family = "binomial", data = datos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0332  -0.6785  -0.3506  -0.1565   2.6143
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.79394    1.48174  -6.610 3.85e-11 ***
## matematicas  0.15634    0.02561   6.105 1.03e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 222.71  on 199  degrees of freedom
## Residual deviance: 167.07  on 198  degrees of freedom
## AIC: 171.07
##
## Number of Fisher Scoring iterations: 5
```

El coeficiente estimado para la intersección es el valor esperado del *logaritmo de odds* de que un estudiante obtenga matrícula teniendo un 0 en en matemáticas. Como es de esperar, los *odds* son muy bajos $e^{-9.793942} = 5.579e^{-5}$, lo que se corresponde con una probabilidad de obtener matrícula de $p = \frac{e^{-9.793942}}{1+e^{-9.793942}} = 5.579e^{-5}$.

Acorde al modelo, el logaritmo de los *odds* de que un estudiante tenga matrícula está positivamente relacionado con la puntuación obtenida en matemáticas (coeficiente de regresión = 0.1563404). Esto significa que, por cada unidad que se incrementa la variable matemáticas, se espera que el logaritmo de *odds* de la variable matrícula se incremente en promedio 0.1563404 unidades. Aplicando la inversa del logaritmo natural se obtiene que ($e^{0.1563404} = 1.169$) por cada unidad que se incrementa la variable matemáticas los *odds* de obtener matrícula se incremente en promedio 1.169 unidades. No hay que confundir esto último con que la probabilidad de matrícula se incremente un 1.169 %.

A diferencia de la regresión lineal en la que β_1 se corresponde con el cambio promedio en la variable dependiente *Y* debido al incremento en una unidad del predictor *X*, en regresión logística, β_1 indica el cambio en el logaritmo de *odds* debido al incremento de una unidad de *X*, o lo que es lo mismo, multiplica los *odds* por e^{β_1}. Dado que la relación entre *p(Y)* y *X* no es lineal, β_1 no se corresponde con el cambio en la probabilidad de *Y* asociado con el incremento de una unidad de *X*. Cuánto se incremente la probabilidad de *Y* por unidad de *X* depende del valor de *X*, es decir, de la posición en la curva logística en la que se encuentre.

Además del valor de las estimaciones de los coeficientes parciales de correlación del modelo, es conveniente calcular sus correspondientes intervalos de confianza. En el caso de regresión logística, estos intervalos suelen calcularse empleando el método de *profile likelihood* (en R es el método por defecto si se tiene instalado el paquete MASS). Para una descripción más detallada ver: http://www.math.umt.edu/patterson/ProfileLikelihoodCI.pdf

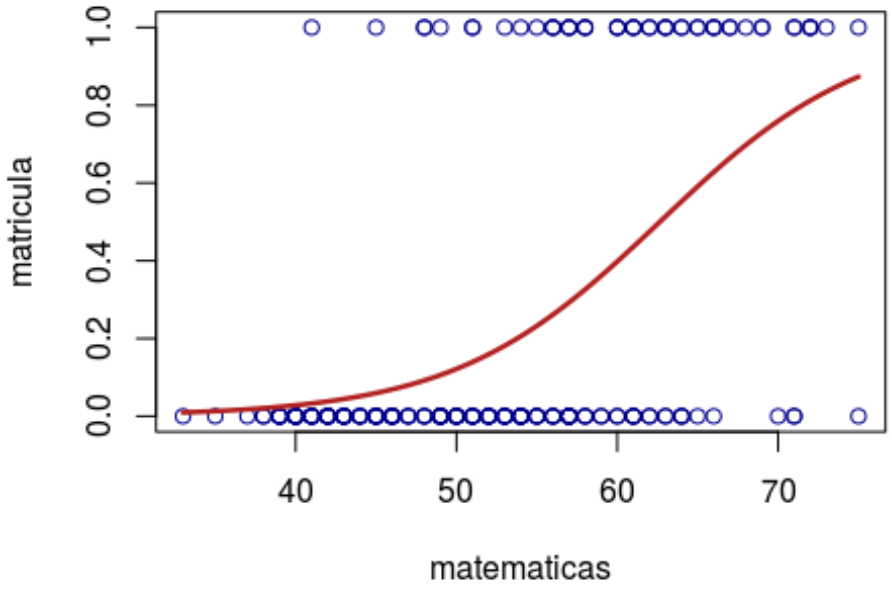
```
confint(object = modelo, level = 0.95)
```

```
## Waiting for profiling to be done...
```

```
##                    2.5 %     97.5 %
## (Intercept) -12.9375208 -7.0938806
## matematicas   0.1093783  0.2103937
```

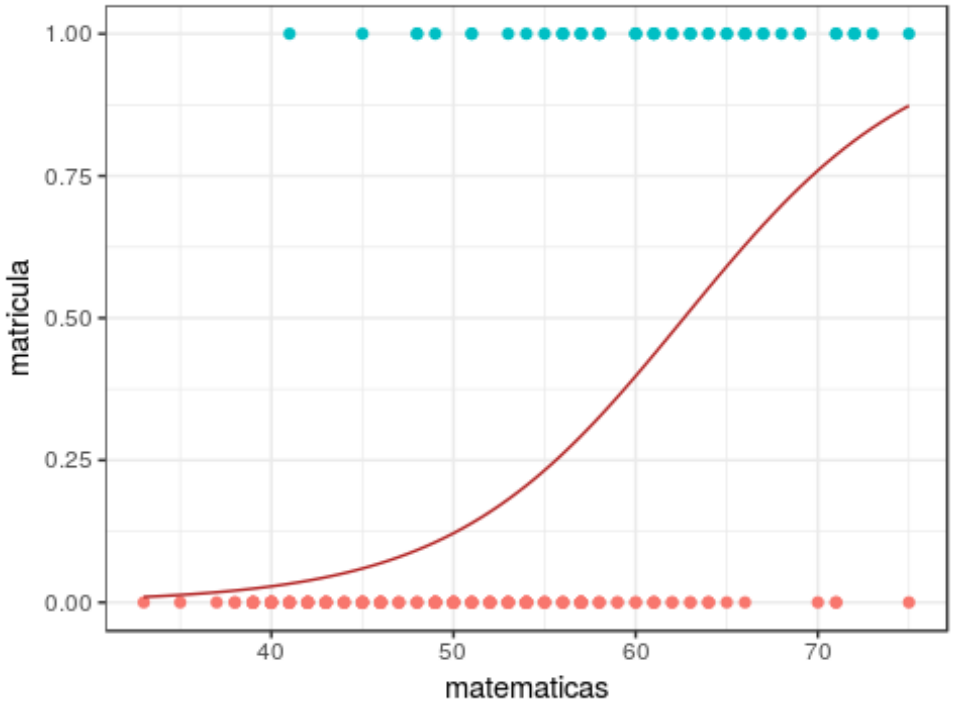
3.Gráfico del modelo

```
# Mediante base graphics
datos$matricula <- as.character(datos$matricula)
datos$matricula <- as.numeric(datos$matricula)
plot(matricula ~ matematicas, datos, col = "darkblue", main = "Modelo regresión
logística matrícula ~ nota matemáticas")
curve(predict(modelo, data.frame(matematicas = x), type = "response"), col =
"firebrick", lwd = 2.5, add = TRUE)
```




```
# Mediante base ggplot2
datos$matricula <- as.character(datos$matricula)
datos$matricula <- as.numeric(datos$matricula)

# Se crea un vector con nuevos valores interpolados en el rango de observaciones
nuevos_puntos <- seq(from = min(datos$matematicas), to = max(datos$matematicas),
                     by = 0.5)
# predicciones de los nuevos puntos según el modelo. type = 'response'
# devuelve las predicciones en forma de probabilidad en lugar de en log_ODDs
predicciones <- predict(object = modelo, newdata = data.frame(matematicas =
nuevos_puntos), type = "response")
# Se crea un data frame con los nuevos puntos y sus predicciones para graficar la
curva
datos_curva <- data.frame(matematicas = nuevos_puntos, matricula = predicciones)
```

```r
ggplot(datos, aes(x = matematicas, y = matricula)) +
  geom_point(aes(color = as.factor(matricula))) +
  geom_line(data = datos_curva, aes(y = matricula), color = "firebrick") +
  theme_bw() +
  labs(title = "Modelo regresión logística matrícula ~ nota matemáticas") +
  theme(legend.position = "null")
```



4.Evaluación del modelo

A la hora de evaluar la validez y calidad de un modelo de regresión logística, se analiza tanto el modelo en su conjunto como los predictores que lo forman.

Se considera que el modelo es útil si es capaz de mostrar una mejora explicando las observaciones respecto al modelo nulo (sin predictores). El test *Likelihood ratio* calcula la significancia de la diferencia de residuos entre el modelo de interés y el modelo nulo. El

estadístico sigue una distribución *chi-cuadrado* con grados de libertad equivalentes a la diferencia de grados de libertad de los dos modelos.

```
# Diferencia de residuos En R, un objeto glm almacena la 'deviance' del
# modelo, así como la 'deviance' del modelo nulo.  Diferencia de residuos
dif_residuos <- modelo$null.deviance - modelo$deviance
# Grados libertad
df <- modelo$df.null - modelo$df.residual
# p-value
p_value <- pchisq(q = dif_residuos, df = df, lower.tail = FALSE)
paste("Diferencia de residuos:", round(dif_residuos, 4))
```

```
## [1] "Diferencia de residuos: 55.6368"
```

```
paste("Grados de libertad:", df)
```

```
## [1] "Grados de libertad: 1"
```

```
paste("p-value:", p_value)
```

```
## [1] "p-value: 8.71759108087093e-14"
```

```
# El mismo cálculo se puede obtener directamente con: anova(modelo, test =
# 'Chisq')
```

En este caso, el modelo obtenido sí es significativo.

Para determinar si los predictores introducidos en un modelo de regresión logística contribuyen de forma significativa se emplea el estadístico *Z* y el test *Wald chi-test*. Este es el método utilizado para calcular los *p-values* que se muestran al hacer *summary()* del modelo. El predictor matemáticas sí contribuye de forma significativa (*p-value* = 1.03e-09).

A diferencia de los modelos de regresión lineal, en los modelos logísticos no existe un equivalente a R^2 que determine exactamente la varianza explicada por el modelo. Se han desarrollado diferentes métodos conocidos como $pseudoR^2$ que intentan aproximarse al concepto de R^2 pero que, aunque su rango oscila entre 0 y 1, no se pueden considerar equivalentes.

- McFadden's: $R^2_{McF} = 1 - \frac{ln\hat{L}(modelo)}{ln\hat{L}(modelo\ nulo)}$, siendo $\hat{L}$ el valor de likelihood de cada modelo. La idea de esta formula es que $ln(\hat{L})$ tiene un significado análogo a la suma de cuadrados de la regresión lineal. De ahí que se le denomine $pseudoR^2$.
- Otra opción bastante extendida es el test de *Hosmer-Lemeshow*. Este test examina mediante un *Pearson chi-square test* si las proporciones de eventos observados son similares a las probabilidades predichas por el modelo, haciendo subgrupos.

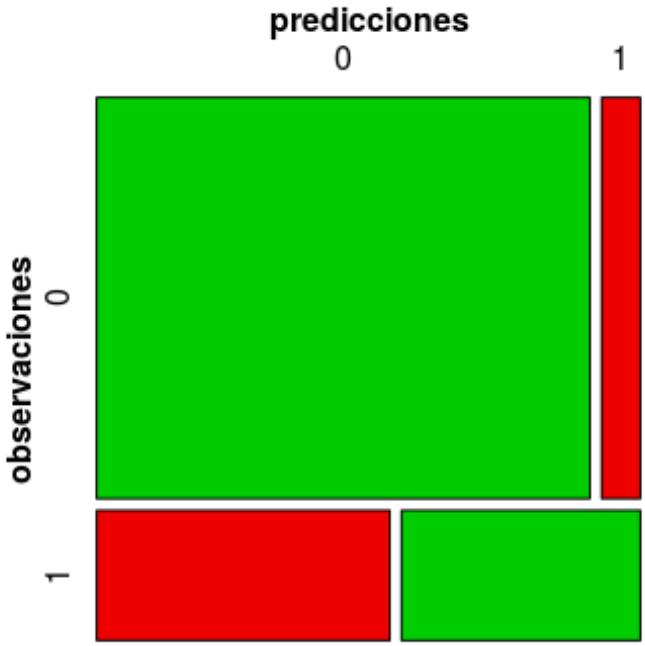
5.Comparación de clasificación predicha y observaciones

Para este estudio se va a emplear un *threshold* de 0.5. Si la probabilidad de que la variable adquiera el valor 1 (matricula) es superior a 0.5 se asigna a este nivel, si es menor se asigna al 0 (no matrícula).

```
library(vcd)
predicciones <- ifelse(test = modelo$fitted.values > 0.5, yes = 1, no = 0)
matriz_confusion <- table(modelo$model$matricula, predicciones, dnn =
("observaciones", "predicciones"))
matriz_confusion
```

```
##              predicciones
## observaciones   0   1
##             0 140  11
##             1  27  22
```

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



El modelo es capaz de clasificar correctamente $\frac{140+22}{140+22+27+11} = 0.81\%$ de las observaciones cuando se emplea el *trainig data set*.

6.Conclusión

El modelo logístico creado para predecir la probabilidad de que un alumno obtenga matrícula de honor a partir de la nota de matemáticas es en conjunto significativo acorde al *Likelihood ratio* (*p-value* = 8.717591e-14). El *p-value* del predictor *matematicas* es significativo (*p-value* = 1.03e-09).

$$logit(matricula) = -9.793942 + 0.1563404 * nota\ matematicas$$

$$P(matricula) = \frac{e^{-9.793942+0.1563404*nota\ matematicas}}{1 + e^{-9.793942+0.1563404*nota\ matematicas}}$$

Regresión logística múltiple

Idea intuitiva

La regresión logística múltiple es una extensión de la regresión logística simple. Se basa en los mismos principios que la regresión logística simple (explicados anteriormente) pero ampliando el número de predictores. Los predictores pueden ser tanto continuos como categóricos.

$$\ln(\frac{p}{1-p}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i$$

$$logit(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i$$

El valor de la probabilidad de *Y* se puede obtener con la inversa del logaritmo natural:

$$p(Y) = \frac{e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i}}{1 + e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i}}$$

A la hora de evaluar la validez y calidad de un modelo de regresión logística múltiple se analiza tanto el modelo en su conjunto como los predictores que lo forman. Se considera que el modelo es útil si es capaz de mostrar una mejora respecto al modelo nulo, el modelo sin predictores. Existen 3 test estadísticos que cuantifican esta mejora mediante la comparación de los residuos: *likelihood ratio*, *score* y *Wald test*. No hay garantías de que los 3 lleguen a la misma conclusión, cuando esto ocurre parece ser recomendable basarse en el *likelihood ratio*. http://www.ats.ucla.edu/stat/mult_pkg/faq/general/nested_tests.htm .

Ejemplo1

Un estudio considera que existe relación entre el hecho de que un estudiante asista a clases de repaso de lectura (sí = 1, no = 0), la nota que obtiene en un examen de lectura estándar (realizado antes de iniciar las clases de repaso) y el sexo (hombre = 1, mujer = 0). Se quiere generar un modelo en el que a partir de las variables puntuación del examen y género, prediga la probabilidad de que el estudiante tenga que asistir a clases de repaso

1.Análisis de las observaciones

Las tablas de frecuencia así como representaciones gráficas de las observaciones son útiles para intuir si las variables independientes escogidas están relacionadas con la variable respuesta y por lo tanto ser un buen predictor.

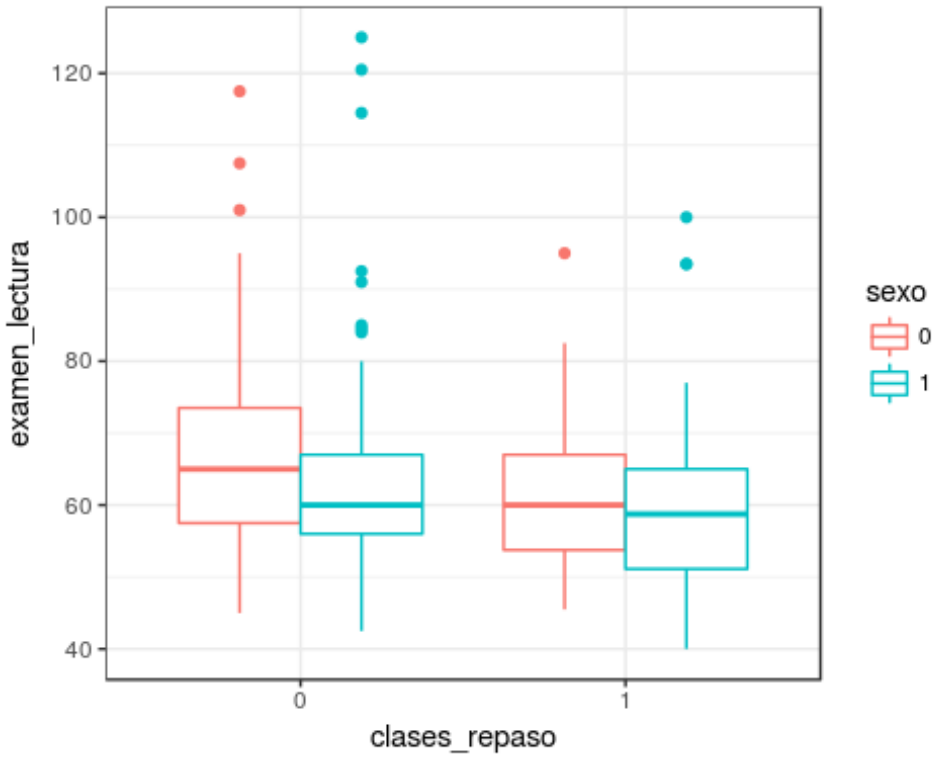
```
require(ggplot2)
tabla <- table(datos$clases_repaso, datos$sexo, dnn = c("cases de repaso", "sexo"))
addmargins(tabla)
```

```
##                sexo
## cases de repaso   0   1 Sum
##              0    73  57 130
##              1    23  36  59
##              Sum  96  93 189
```

```
tabla_frecuencias <- prop.table(tabla) * 100
addmargins(tabla_frecuencias)
```

```
##                sexo
## cases de repaso         0         1       Sum
##              0    38.62434  30.15873  68.78307
##              1    12.16931  19.04762  31.21693
##              Sum  50.79365  49.20635 100.00000
```

```
ggplot(data = datos, mapping = aes(x = clases_repaso, y = examen_lectura, colour =
sexo)) +
    geom_boxplot() + theme_bw()
```



El número de estudiantes en la muestra es semejante para ambos sexos (96, 93). Parece ser mayor el porcentaje de hombres que necesitan clases de repaso (19.04762, 12.16931). El promedio de las notas de lectura de los estudiantes que requieren de clases particulares es menor que el de los que no requieren clases. En vista de estos datos, tiene sentido considerar el sexo y la nota como posibles predictores.

2.Generar el modelo de regresión logística

```
modelo <- glm(clases_repaso ~ examen_lectura + sexo, data = datos, family =
"binomial")
summary(modelo)
```

```
##
## Call:
## glm(formula = clases_repaso ~ examen_lectura + sexo, family = "binomial",
##     data = datos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2079  -0.8954  -0.7243   1.2592   2.0412
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.53616    0.81088   0.661   0.5085
## examen_lectura -0.02617    0.01223  -2.139   0.0324 *
## sexo1           0.64749    0.32484   1.993   0.0462 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 234.67  on 188  degrees of freedom
## Residual deviance: 224.64  on 186  degrees of freedom
## AIC: 230.64
##
## Number of Fisher Scoring iterations: 4
```

Acorde al modelo, el *logaritmo de odds* de que un estudiante necesite clases de repaso esta negativamente relacionado con la puntuación obtenida en el examen de lectura (coeficiente parcial = -0.02617), siento significativa esta relación (*p-value* = 0.0324). También existe una relación significativa positiva entre el *logaritmo de odds* de necesitar clases de repaso y el género del estudiante (*p-value* = 0.0462), siendo, para un mismo resultado en el examen de lectura, mayor si el estudiante en hombre. En concreto los *odds* de que un hombre requiera clases de repaso son $e^{0.64749} = 1.910739$ mayores que los de las mujeres. (Esto se puede ver gráficamente representando el modelo para hombres y mujeres).

Además del valor estimado de los coeficientes parciales de correlación calculados por el modelo, es conveniente generar sus correspondientes intervalos de confianza. En el caso de regresión logística, estos intervalos suelen calcularse basados en *profile likelihood* (en R es el método por defecto si se tiene instalado el paquete MASS).

```
confint(modelo)
# En caso de querer los intervalos basados en el error estandar
# confint.default(modelo)
```

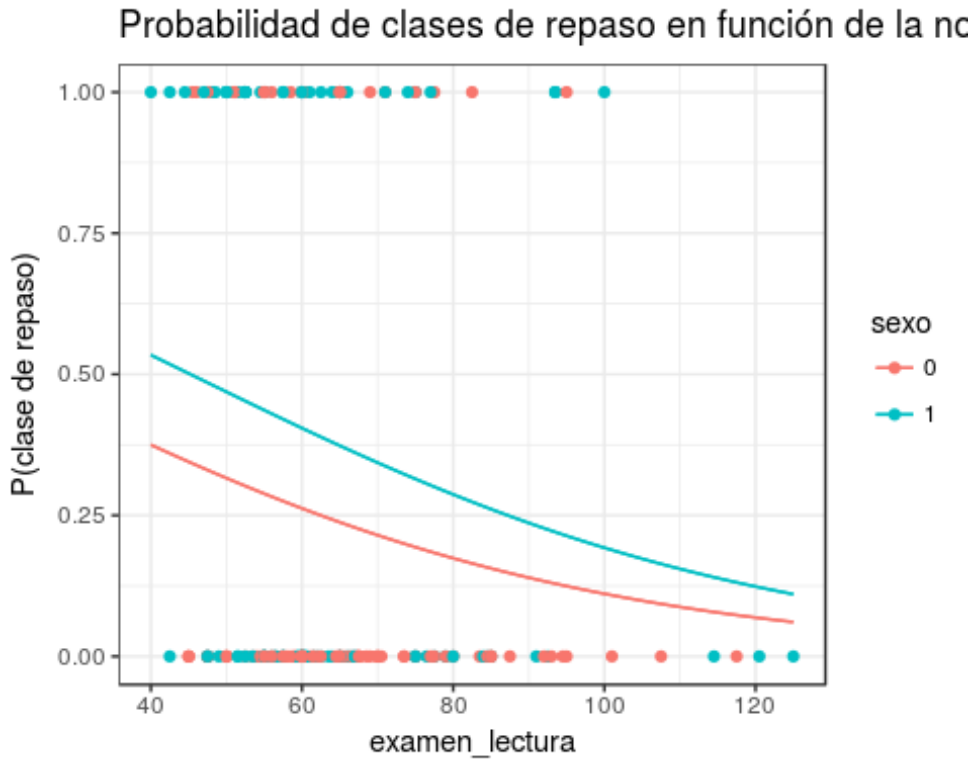
```
##                       2.5 %       97.5 %
## (Intercept)    -0.99992945  2.196547544
## examen_lectura -0.05180290 -0.003536647
## sexo1           0.01565893  1.293121628
```

3.Representación gráfica del modelo

Al tratarse de un modelo con 2 predictores, no se puede obtener una representación en 2D en la que se incluyan ambos predictores a la vez. Sí es posible representar la curva del modelo logístico cuando se mantiene constante uno de los dos predictores. Por ejemplo, al representar las predicciones del modelo diferenciando entre hombres y mujeres (fijando el valor del predictor sexo) se aprecia que la curva de los hombres (sexo=1) siempre está por encima. Esto se debe a que, como indica el coeficiente parcial de correlación del predictor sexo, para una misma nota en el examen de lectura el *logaritmo de ODDs* de necesitar clases de repaso es 0.64749 veces mayor en hombres.

```
require(ggplot2)
# Para graficar los valores en ggplot junto con la curva, la variable
# respuesta tiene que ser numérica en lugar de factor.
datos$clases_repaso <- as.numeric(as.character(datos$clases_repaso))

# Se crea un data frame que contenga la probabilidad de que se necesiten
# clases de repaso dada una determinada nota en el examen de lectura y
# siendo hombre (sex=1).

# vector con nuevos valores interpolados en el rango de observaciones
nuevos_valores_examen <- seq(from = min(datos$examen_lectura), to =
max(datos$examen_lectura), by = 0.5)
sexo <- as.factor(rep(x = 1, length(nuevos_valores_examen)))


# predicciones de los nuevos puntos según el modelo. type = 'response'
# devuelve las predicciones en forma de probabilidad en lugar de en log_ODDs
predicciones <- predict(object = modelo, newdata = data.frame(examen_lectura =
nuevos_valores_examen, sexo = sexo), type = "response")


# Se crea un data frame con los nuevos puntos y sus predicciones para
# graficar la curva
datos_curva_hombre <- data.frame(examen_lectura = nuevos_valores_examen, sexo =
sexo, clases_repaso = predicciones)
```

```
# Mismo proceso para mujeres (sexo = 0)
nuevos_valores_examen <- seq(from = min(datos$examen_lectura), to =
max(datos$examen_lectura), by = 0.5)
sexo <- as.factor(rep(x = 0, length(nuevos_valores_examen)))
predicciones <- predict(object = modelo, newdata = data.frame(examen_lectura =
nuevos_valores_examen, sexo = sexo), type = "response")
datos_curva_mujer <- data.frame(examen_lectura = nuevos_valores_examen, sexo =
sexo, clases_repaso = predicciones)

# Se unifican los dos dataframe
datos_curva <- rbind(datos_curva_hombre, datos_curva_mujer)

ggplot(data = datos, aes(x = examen_lectura, y = as.numeric(clases_repaso),
    color = sexo)) +
  geom_point() +
  geom_line(data = datos_curva, aes(y = clases_repaso)) +
  geom_line(data = datos_curva, aes(y = clases_repaso)) +
  theme_bw() +
 labs(title = "Probabilidad de clases de repaso en función de la nota en lectura y
sexo", y = "P(clase de repaso)")
```




```
# Otra opción para graficarlo es: qplot(x = modelo$data$examen_lectura, y =
# modelo$fitted.values, geom = c('point','line'), colour = modelo$data$sexo,
# ylim = c(0,1))
```

4.Evaluación del modelo

Likelihood ratio:

```
# Diferencia de residuos
dif_residuos <- modelo$null.deviance - modelo$deviance
# Grados libertad
df <- modelo$df.null - modelo$df.residual
# p-value
p_value <- pchisq(q = dif_residuos, df = df, lower.tail = FALSE)

paste("Diferencia de residuos:", round(dif_residuos, 4))
```

```
## [1] "Diferencia de residuos: 10.0334"
```

```
paste("Grados de libertad:", df)
```

```
## [1] "Grados de libertad: 2"
```

```
paste("p-value:", round(p_value, 4))
```

```
## [1] "p-value: 0.0066"
```

El modelo en conjunto sí es significativo y acorde a los *p-values* mostrados en el *summary(modelo)* también es significativa la contribución al modelo de ambos predictores.

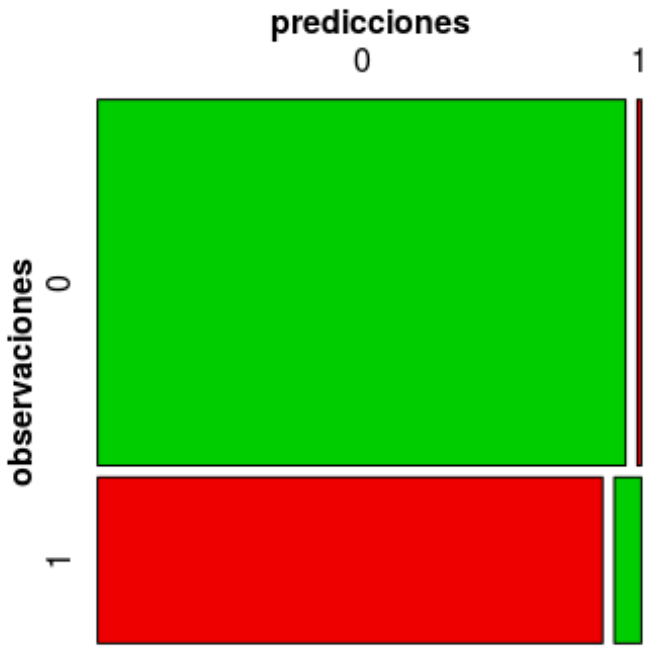
5.Comparación las predicciones con las observaciones

Para este estudio se va a emplear un *threshold* de 0.5. Si la probabilidad predicha de asistir a clases de repaso es superior a 0.5 se asigna a al nivel 1 (sí asiste), si es menor se asigna al nivel 0 (no clases de repaso).

```
predicciones <- ifelse(test = modelo$fitted.values > 0.5, yes = 1, no = 0)
matriz_confusion <- table(modelo$model$clases_repaso, predicciones, dnn =
c("observaciones", "predicciones"))
matriz_confusion
```

```
##              predicciones
## observaciones   0   1
##             0 129   1
##             1  56   3
```

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



El modelo es capaz de clasificar correctamente $\frac{129+3}{129+3+56+1} = 0.698\%$ de las observaciones cuando se emplea el *trainig data set*. Si se analiza en detalle cómo se distribuye el error, se aprecia que el modelo solo ha sido capaz de identificar correctamente a 3 de los 59 alumnos que realmente asisten a clases de repaso. El porcentaje de falsos negativos es muy alto. Seleccionar otro *threshold* puede mejorar la exactitud del modelo.

```
library(vcd)
predicciones <- ifelse(test = modelo$fitted.values > 0.45, yes = 1, no = 0)
matriz_confusion <- table(modelo$model$clases_repaso, predicciones, dnn =
c("observaciones", "predicciones"))
matriz_confusion
```

```
##              predicciones
## observaciones   0   1
##             0 122   8
##             1  45  14
```

6.Conclusión

El modelo logístico creado para predecir la probabilidad de que un alumno tenga que asistir a clases de repaso a partir de la nota obtenida en un examen de lectura y el sexo del alumno es en conjunto significativo acorde al *Likelihood ratio* (*p-value* = 0.0066). El *p-value* de ambos predictores es significativo (examen_lectura = 0.0324, sexo1 = 0.0462). El ratio de error obtenido empleando las observaciones con las que se ha entrenado el modelo muestra un porcentaje de falsos negativos muy alto.

$$logit(clases\ de\ repaso) = 0.53616 - 0.02617 examen\ lectura + 0.64749 sexo$$

$$P(clases\ de\ repaso) = \frac{e^{0.53616 - 0.02617 examen\ lectura + 0.64749 sexo}}{1 + e^{0.53616 - 0.02617 examen\ lectura + 0.64749 sexo}}$$

Ejemplo2

Se dispone de un registro que contiene cientos de emails con información de cada uno de ellos. El objetivo de estudio es intentar crear un modelo que permita filtrar que emails son "spam" y cuáles no, en función de determinadas características. Ejemplo extraído del libro OpenIntro Statistics.

```
library(openintro)
data(email)
str(email)
```

```
## 'data.frame':    3921 obs. of  21 variables:
##  $ spam        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ to_multiple : num  0 0 0 0 0 0 1 1 0 0 ...
##  $ from        : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ cc          : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ sent_email  : num  0 0 0 0 0 0 1 1 0 0 ...
##  $ time        : POSIXct, format: "2012-01-01 07:16:41" "2012-01-01 08:03:59"
...
##  $ image       : num  0 0 0 0 0 0 0 1 0 0 ...
##  $ attach      : num  0 0 0 0 0 0 0 1 0 0 ...
##  $ dollar      : num  0 0 4 0 0 0 0 0 0 0 ...
##  $ winner      : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ inherit     : num  0 0 1 0 0 0 0 0 0 0 ...
##  $ viagra      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ password    : num  0 0 0 0 2 2 0 0 0 0 ...
##  $ num_char    : num  11.37 10.5 7.77 13.26 1.23 ...
##  $ line_breaks : int  202 202 192 255 29 25 193 237 69 68 ...
##  $ format      : num  1 1 1 1 0 0 1 1 0 1 ...
##  $ re_subj     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ exclaim_subj: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ urgent_subj : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ exclaim_mess: num  0 1 6 48 1 1 1 18 1 0 ...
##  $ number      : Factor w/ 3 levels "none","small",..: 3 2 2 2 1 1 3 2 2 2 ...
```

En este caso se van a emplear únicamente como posibles predictores variables categóricas. Esto se debe a que los *outliers* complican bastante la creación de estos modelos y en el data set que se emplea como ejemplo las variables cuantitativas son muy asimétricas. En particular, las variables que se van a estudiar como posibles predictores son:

- spam: si el email es spam (1) si no lo es (0)
- to_multiple: si hay más de una persona en la lista de distribución.
- format: si está en formato HTLM
- cc: si hay otras direcciones en copia.
- attach: si hay archivos adjuntos
- dollar: si el email contiene la palabra dollar o el símbolo $.
- inherit: si contiene la palabra inherit
- winner: si el email contiene la palabra winner.
- password: si el email contiene la palabra password.
- re_subj: si la palabra "Re:" está escrita en el asunto del email.
- exclaim_subj: si se incluye algún signo de exclamación en el email.

En primer lugar se genera el modelo completo introduciendo todas las variables como predictores:

```
modelo_completo <- glm(spam ~ to_multiple + format + cc + attach + dollar +
    winner + inherit + password + re_subj + exclaim_subj, data = email, family =
binomial())
summary(modelo_completo)
```

```
##
## Call:
## glm(formula = spam ~ to_multiple + format + cc + attach + dollar +
##     winner + inherit + password + re_subj + exclaim_subj, family = binomial(),
##     data = email)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6348  -0.4325  -0.2566  -0.0945   3.8846
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.79976    0.08935  -8.950  < 2e-16 ***
## to_multiple  -2.84097    0.31158  -9.118  < 2e-16 ***
## format       -1.52284    0.12270 -12.411  < 2e-16 ***
## cc            0.03134    0.01895   1.654 0.098058 .
## attach        0.20351    0.05851   3.478 0.000505 ***
## dollar       -0.07304    0.02306  -3.168 0.001535 **
## winneryes     1.83103    0.33641   5.443 5.24e-08 ***
## inherit       0.32999    0.15223   2.168 0.030184 *
## password     -0.75953    0.29597  -2.566 0.010280 *
## re_subj      -3.11857    0.36522  -8.539  < 2e-16 ***
## exclaim_subj  0.24399    0.22502   1.084 0.278221
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2437.2  on 3920  degrees of freedom
## Residual deviance: 1936.2  on 3910  degrees of freedom
## AIC: 1958.2
##
## Number of Fisher Scoring iterations: 7
```

Se mejora el modelo mediante el proceso basado en *p-values*. El resultado es el siguiente modelo:

```
modelo_final <- glm(spam ~ to_multiple + format + attach + dollar + winner +
    inherit + password + re_subj, data = email, family = binomial())
summary(modelo_final)
```

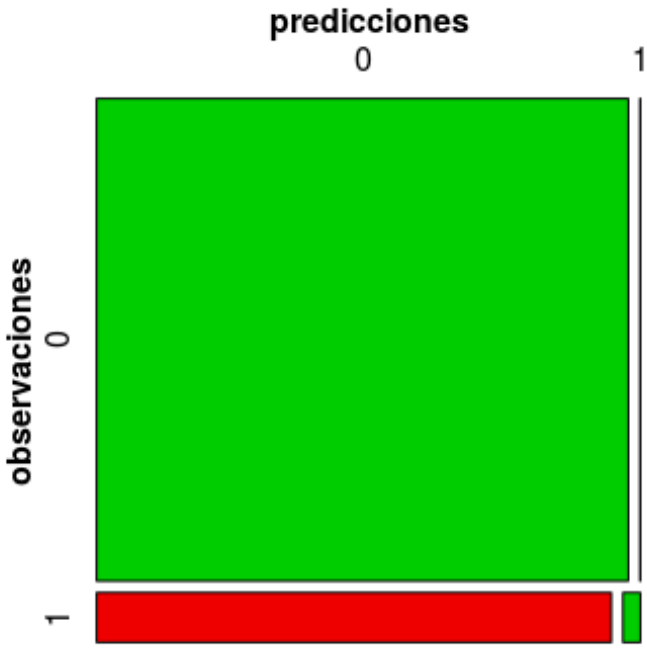
```
##
## Call:
## glm(formula = spam ~ to_multiple + format + attach + dollar +
##     winner + inherit + password + re_subj, family = binomial(),
##     data = email)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6591  -0.4373  -0.2544  -0.0944   3.8707
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.78138    0.08860  -8.820  < 2e-16 ***
## to_multiple -2.77682    0.30752  -9.030  < 2e-16 ***
## format      -1.51770    0.12226 -12.414  < 2e-16 ***
## attach       0.20419    0.05789   3.527  0.00042 ***
## dollar      -0.06970    0.02239  -3.113  0.00185 **
## winneryes    1.86675    0.33652   5.547  2.9e-08 ***
## inherit      0.33614    0.15073   2.230  0.02575 *
## password    -0.76035    0.29680  -2.562  0.01041 *
## re_subj     -3.11329    0.36519  -8.525  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2437.2  on 3920  degrees of freedom
## Residual deviance: 1939.6  on 3912  degrees of freedom
## AIC: 1957.6
##
## Number of Fisher Scoring iterations: 7
```

Con este modelo podemos saber la probabilidad, dadas unas determinadas características, de que el email sea *espam* (valor 1 de la variable). Para evaluar el modelo, se puede comparar el valor real (si realmente es spam) con el predicho por el modelo.

```
library(vcd)
predicciones <- ifelse(test = modelo_final$fitted.values > 0.5, yes = 1, no = 0)
matriz_confusion <- table(modelo_final$model$spam, predicciones, dnn =
c("observaciones", "predicciones"))
matriz_confusion
```

```
##              predicciones
## observaciones    0    1
##             0 3551    3
##             1  355   12
```

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



El modelo generado no es capaz de diferenciar bien entre *spam* y no *spam*, solo 12 de los 367 correos *spam* se han identificado correctamente. Si fuese capaz, la probabilidad calculada por el modelo para aquellos emails del *dataset* que sí son spam deberían estar por encima del 0.5.

Bibliografía

Introduction to Statistical Learning

OpenIntro Statistics

An introduction to Logistic Regresion Analysis and Reporting. Chao-Ying Joanne Peng

http://www.ats.ucla.edu/stat/r/dae/logit.htm

http://ww2.coastal.edu/kingw/statistics/R-tutorials/index.html