Mayo 2018

Cristina Gil Martínez

MÉTODOS DE CLASIFICACIÓN: REGRESIÓN LOGÍSTICA

Apuntes personales sobre regresión logística simple y múltiple y ejemplo en R

CONTENIDO

INTRODUCCIÓN

Los métodos de clasificación permiten predecir **variables cualitativas** o categóricas. Tres de los clasificadores más usados son:

- Regresión logística
- Análisis discriminante lineal y cuadrático
- K-vecinos más cercanos *(K-nearest neighbours)*

El método de regresión logística es el recomendado cuando se trabaja con una variable cualitativa con dos niveles, tanto con uno (regresión logística simple) como con múltiples predictores (regresión logística múltiple).

Al igual que en el caso de regresión, en los problemas de clasificación contamos con un set de observaciones de entrenamiento $(x_1, y_1), ..., (x_n, y_n)$ que usamos para generar el clasificador. El objetivo es que nuestro modelo funcione bien no sólo con las observaciones de entrenamiento, sino con nuevas observaciones.

¿Por qué no regresión lineal?

Para una variable respuesta binaria (dos niveles) podríamos crear dos variables *dummy* (0/1) y predecir la variable codificada como 1 si $\hat{Y} > 0{,}5$ (o usando un límite mayor o menor dependiendo del interés). En este caso no importaría que nivel fuera codificado como 0 o 1, el modelo de regresión lineal daría el mismo resultado. Si la variable cualitativa contara con más de dos niveles, el orden en que se establecieran las variables *dummy* (no habiendo un criterio que indique en que forma se tienen que ordenar) influiría en el modelo resultante, por lo que este enfoque no se considera apto para estas situaciones. Por otro lado, algunos de los valores estimados mediante una recta de mínimos cuadrados pueden ser < 0 o > 1, lo que entra en conflicto con el hecho de que toda probabilidad está comprendida entre [0, 1].

REGRESIÓN LOGÍSTICA SIMPLE

Dada una variable respuesta categórica con dos niveles, la regresión logística modela la **probabilidad** de que *Y* pertenezca a una categoría o nivel particular, dados los valores de **un único predictor** *X*. La clasificación depende del límite o *threshold* que se establezca.

$$\Pr(Y = k \mid X = x)$$

En regresión logística utilizamos la **función logística**:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

que siempre producirá una curva en forma de S, comprendiéndose los valores de *Y* entre [0, 1]. La ecuación anterior puede reestructurarse como

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

donde *p(X) / [1 – p(X)]* corresponde a los ***odds***, pudiendo tomar cualquier valor entre 0 (muy baja probabilidad de éxito) y ∞ (muy alta probabilidad de éxito). Este ratio, pues, indica cuanto más probable es el éxito que el fracaso.

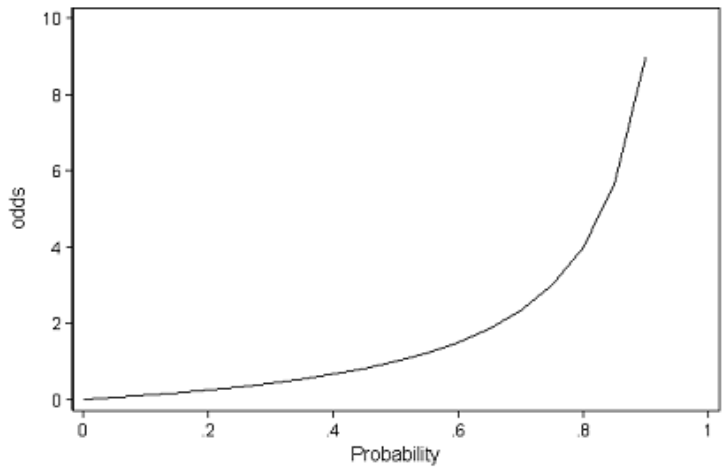
Introduciendo el logaritmo en ambos lados de la ecuación, obtenemos una función lineal

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X$$

La parte izquierda de la ecuación es lo que se conoce como logaritmo de *odds* (**log-*odds***) o ***logit***.

La transformación de probabilidad a *odds* es monotónica, lo que significa que los *odds* aumentan conforme aumenta la probabilidad, y viceversa:

p	odds
0.0010	0.0010010
0.0100	0.0101010
0.1500	0.1764706
0.2000	0.2500000
0.2500	0.3333333
0.3000	0.4285714
0.3500	0.5384616
0.4000	0.6666667
0.4500	0.8181818
0.5000	1.0000000
0.5500	1.2222220
0.6000	1.5000000
0.6500	1.8571430
0.7000	2.3333330
0.7500	3.0000000
0.8000	4.0000000
0.8500	5.6666670
0.9000	9.0000000
0.9990	999.0000000
0.9999	9999.0000000



Todas estas transformaciones se implementan para evitar la restricción del rango de probabilidad [0, 1] en la variable respuesta, ya que transformación logística (logaritmo de *odds*) permite mapear desde menos infinito hasta más infinito.

Interpretación de los coeficientes de regresión

Mientras que en regresión lineal β_1 se corresponde con el cambio promedio en *Y* asociado a un incremento de una unidad en *X*, en regresión logística **β_1** es el valor que indica cuanto cambia el logaritmo de *odds* cuando *X* se incrementa en una unidad, o equivalentemente, multiplica los *odds* por $e^{\beta 1}$. La cantidad con la que *p*(*X*) cambia debido a un cambio en *X* dependerá del valor actual de *X*, pero independientemente de ello, si β_1 es positivo, entonces aumentar *X* provocará un aumento de *p*(*X*). La intersección **β_0** corresponde con el resultado predicho para el nivel de referencia.

Estimación de los coeficientes de regresión

Los coeficientes β_0 y β_1 de la ecuación logística son desconocidos, y han de estimarse a partir de los datos de entrenamiento. Mientras que en regresión lineal los coeficientes del modelo se estiman por mínimos cuadrados, en regresión logística se utiliza el método de **máxima verosimilitud** (*máximum likelihood*): se buscan coeficientes tales que la probabilidad prevista $\hat{p}(x_i)$ de éxito se aproxime lo máximo posible a las observaciones reales.

Los coeficientes estimados por el modelo para las variables se corresponden al valor del logaritmo de *odds*, o lo que es lo mismo, multiplica los *odds* por $e^{\beta 1}$.

Podemos medir la precisión de los coeficientes estimados a partir de sus **errores estándar**. Además, en este modelo se emplea el **estadístico *Z*** para obtener el nivel de significancia del predictor (*p-value)*, a diferencia del estadístico *t* en regresión lineal, aunque juegan el mismo papel. Por ejemplo, el estadístico *z* asociado a β_1 sería igual a

$$\hat{\beta}_1 / SE(\hat{\beta}_1)$$

Un valor alto (absoluto) de Z indica la evidencia en contra de la **hipótesis nula**

$$H_0 : \beta_1 = 0$$

la cual implica que la probabilidad de éxito no depende de la variable independiente *X*:

$$p(X) = \frac{e^{\beta_0}}{1+e^{\beta_0}}$$

Si el *p-value* es menor que el nivel de significancia establecido, podemos deducir que hay una relación entre el predictor *X* y la probabilidad de éxito.

La ordenada en el origen β_0 estimada en el modelo no suele ser de interés.

REGRESIÓN LOGÍSTICA MÚLTIPLE

La regresión logística múltiple es una extensión del modelo de regresión logística simple en el que se predice una respuesta binaria en función de **múltiples predictores**, que pueden ser tanto continuos como categóricos. La ecuación con la que podemos obtener las predicciones en este caso es

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

donde $X = (X_1, ..., X_p)$ son los *p* predictores.

De nuevo usamos el método de máxima verosimilitud para estimar los coeficientes $\beta_0, \beta_1, ..., \beta_p$. Cada coeficiente se interpreta manteniendo fijos al resto.

Al igual que en el caso de la regresión lineal, los resultados obtenidos usando solo un predictor pueden diferir respecto a aquellos obtenidos usando múltiples predictores, especialmente cuando existe correlación entre ellos. Este fenómeno se conoce como confusión (***confounding***).

CONDICIONES DEL MODELO LOGÍSTICO

La regresión logística no requiere de ciertas condiciones como linealidad, normalidad y homocedasticidad de los residuos que sí lo son para la regresión lineal. Las principales condiciones que este modelo requiere son:

- **Respuesta binaria**: La variable dependiente ha de ser binaria.
- **Independencia**: las observaciones han de ser independientes.

- **Multicolinealidad**: se requiere de muy poca a ninguna multicolinealidad entre los predictores (para regresión logística múltiple).
- **Linealidad** entre la variable independiente y el logaritmo natural de *odds*.
- **Tamaño muestral**: como regla general, se requiere un mínimo de 10 casos con el resultado menos frecuente para cada variable independiente del modelo.

EJEMPLO EN R

En este ejemplo trabajaremos con el set de datos *Weekly*, que forma parte del paquete *ISLR*. Este set de datos contiene información sobre el rendimiento porcentual semanal del índice bursátil S&P 500 entre los años 1990 y 2010. Trataremos de ajustar un modelo logístico simple para predecir el rendimiento (positivo o negativo), usando *Lag2* (porcentaje de retorno en las dos semanas previas) como el único predictor.

1. ANÁLISIS EXPLORATORIO DE LOS DATOS

- **head()** -> Muestra las primeras observaciones de un vector, matriz, tabla, data frame o función
- **glimpse()** -> Muestra de manera compacta la estructura interna de un objeto (muy similar a str())
- **summary()** -> Resumen de los datos
- **pairs()** -> Matriz con gráficos de dispersión de las variables cuantitativas
- **cor()** -> Calcula los coeficientes de correlación entre variables (solo acepta vectores numéricos)
- **plot()**

Como primer paso generaremos un resumen numérico/gráfico para detectar posibles relaciones entre variables y obtener información útil:

```
library(ISLR)
library(tidyverse)

head(Weekly)
```

```
##   Year   Lag1   Lag2   Lag3   Lag4   Lag5    Volume  Today Direction
## 1 1990  0.816  1.572 -3.936 -0.229 -3.484 0.1549760 -0.270      Down
## 2 1990 -0.270  0.816  1.572 -3.936 -0.229 0.1485740 -2.576      Down
## 3 1990 -2.576 -0.270  0.816  1.572 -3.936 0.1598375  3.514        Up
## 4 1990  3.514 -2.576 -0.270  0.816  1.572 0.1616300  0.712        Up
## 5 1990  0.712  3.514 -2.576 -0.270  0.816 0.1537280  1.178        Up
## 6 1990  1.178  0.712  3.514 -2.576 -0.270 0.1544440 -1.372      Down
```

```
glimpse(Weekly)
```

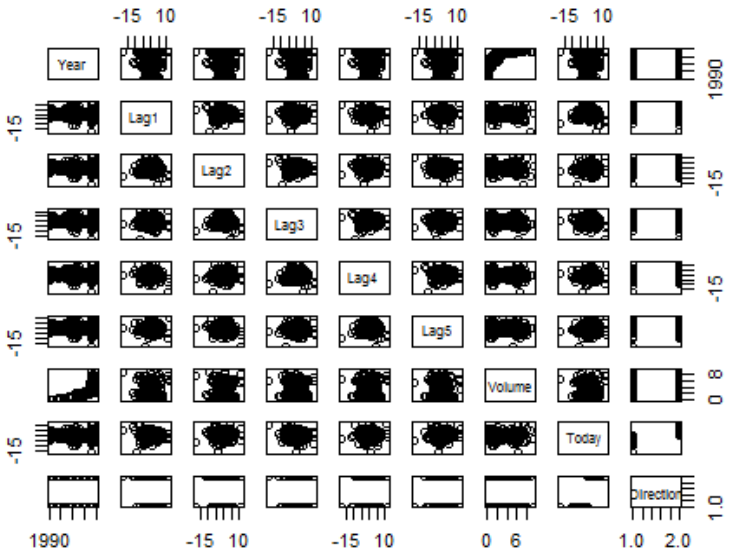
```
## Observations: 1,089
## Variables: 9
## $ Year      <dbl> 1990, 1990, 1990, 1990, 1990, 1990, 1990, 1990, 1990, 199...
## $ Lag1      <dbl> 0.816, -0.270, -2.576, 3.514, 0.712, 1.178, -1.372, 0.807...
## $ Lag2      <dbl> 1.572, 0.816, -0.270, -2.576, 3.514, 0.712, 1.178, -1.372...
## $ Lag3      <dbl> -3.936, 1.572, 0.816, -0.270, -2.576, 3.514, 0.712, 1.178...
## $ Lag4      <dbl> -0.229, -3.936, 1.572, 0.816, -0.270, -2.576, 3.514, 0.71...
## $ Lag5      <dbl> -3.484, -0.229, -3.936, 1.572, 0.816, -0.270, -2.576, 3.5...
## $ Volume    <dbl> 0.1549760, 0.1485740, 0.1598375, 0.1616300, 0.1537280, 0....
## $ Today     <dbl> -0.270, -2.576, 3.514, 0.712, 1.178, -1.372, 0.807, 0.041...
## $ Direction <fct> Down, Down, Up, Up, Up, Down, Up, Up, Up, Down, Down, Up,...
```

Contamos con un set de datos con 9 variables (8 numéricas y 1 categórica que será nuestra variable respuesta: *Direction*) y 1089 observaciones.

```
summary(Weekly)
```

```
##       Year           Lag1               Lag2               Lag3          
##  Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.1950  
##  1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.1580  
##  Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.2410  
##  Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.1472  
##  3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.4090  
##  Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.0260  
##       Lag4               Lag5              Volume            Today         
##  Min.   :-18.1950   Min.   :-18.1950   Min.   :0.08747   Min.   :-18.1950  
##  1st Qu.: -1.1580   1st Qu.: -1.1660   1st Qu.:0.33202   1st Qu.: -1.1540  
##  Median :  0.2380   Median :  0.2340   Median :1.00268   Median :  0.2410  
##  Mean   :  0.1458   Mean   :  0.1399   Mean   :1.57462   Mean   :  0.1499  
##  3rd Qu.:  1.4090   3rd Qu.:  1.4050   3rd Qu.:2.05373   3rd Qu.:  1.4050  
##  Max.   : 12.0260   Max.   : 12.0260   Max.   :9.32821   Max.   : 12.0260  
##  Direction
##  Down:484
##  Up  :605
##
```

```
pairs(Weekly)
```

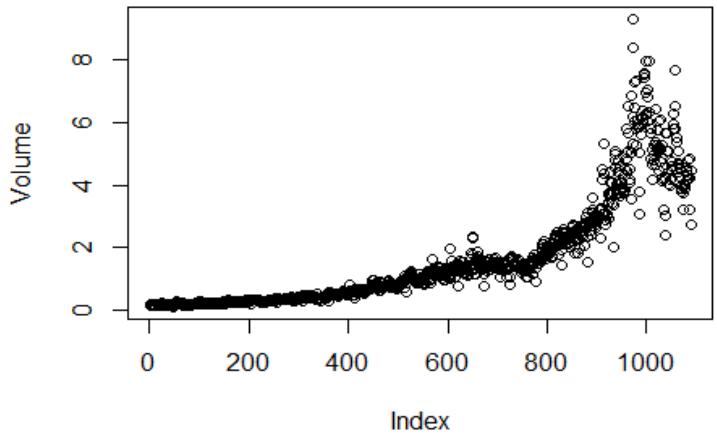


```
# Correlación entre parejas de predictores
cor(Weekly[, -9])
```

```
##                Year         Lag1        Lag2        Lag3         Lag4
## Year    1.00000000 -0.032289274 -0.03339001 -0.03000649 -0.031127923
## Lag1   -0.03228927  1.000000000 -0.07485305  0.05863568 -0.071273876
## Lag2   -0.03339001 -0.074853051  1.00000000 -0.07572091  0.058381535
## Lag3   -0.03000649  0.058635682 -0.07572091  1.00000000 -0.075395865
## Lag4   -0.03112792 -0.071273876  0.05838153 -0.07539587  1.000000000
## Lag5   -0.03051910 -0.008183096 -0.07249948  0.06065717 -0.075675027
## Volume  0.84194162 -0.064951313 -0.08551314 -0.06928771 -0.061074617
## Today  -0.03245989 -0.075031842  0.05916672 -0.07124364 -0.007825873
##                Lag5      Volume        Today
## Year   -0.030519101  0.84194162 -0.032459894
## Lag1   -0.008183096 -0.06495131 -0.075031842
## Lag2   -0.072499482 -0.08551314  0.059166717
## Lag3    0.060657175 -0.06928771 -0.071243639
## Lag4   -0.075675027 -0.06107462 -0.007825873
## Lag5    1.000000000 -0.05851741  0.011012698
## Volume -0.058517414  1.00000000 -0.033077783
## Today   0.011012698 -0.03307778  1.000000000
```

Las correlaciones entre las variables *Lag* (valores de mercado en semanas anteriores) y el valor del día actual (*Today*) son próximas a 0. La única correlación destacada se da entre el año (*Year*) y el volumen (*Volume*). De hecho, si representamos la variable *Volume* podemos ver como ésta aumenta con el tiempo:

```
attach(Weekly)
plot(Volume)
```



2. CÁLCULO DEL MODELO LOGÍSTICO

- **glm()** -> Función para ajustar modelos lineales generalizados
- **summary(**modelo**)** -> Resumen del ajuste del modelo
- **contrasts(**variable**)** -> Variables dummy que R ha generado para cada nivel de una variable cualitativa
- **confint()** -> Cálculo de intervalos de confianza para uno o más parámetros de un modelo ajustado

Modelo logístico MÚLTIPLE

Para ajustar este modelo hacemos uso de la función *glm()* para modelos lineales generalizados, una clase de modelos en los que se incluye el modelo logístico. Para ello especificamos el argumento *family = binomial*.

```
# Modelo con todos los predictores, excluyendo "Today"
modelo.log.m <- glm(Direction ~ . -Today, data = Weekly, family = binomial)
summary(modelo.log.m)
```

```
##
## Call:
## glm(formula = Direction ~ . - Today, family = binomial, data = Weekly)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7071  -1.2578   0.9941   1.0873   1.4665
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 17.225822  37.890522   0.455   0.6494
## Year        -0.008500   0.018991  -0.448   0.6545
## Lag1        -0.040688   0.026447  -1.538   0.1239
## Lag2         0.059449   0.026970   2.204   0.0275 *
## Lag3        -0.015478   0.026703  -0.580   0.5622
## Lag4        -0.027316   0.026485  -1.031   0.3024
## Lag5        -0.014022   0.026409  -0.531   0.5955
## Volume       0.003256   0.068836   0.047   0.9623
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1496.2  on 1088  degrees of freedom
## Residual deviance: 1486.2  on 1081  degrees of freedom
## AIC: 1502.2
##
## Number of Fisher Scoring iterations: 4
```

```
contrasts(Direction)
```

```
##      Up
## Down  0
## Up    1
```

Los valores "Null deviance" y "Residual deviance" hacen referencia a los residuos del modelo nulo sin predictores y al modelo completo, respectivamente. Los grados de libertad se corresponden con nº obs – nº predictores.

Además del valor de las estimaciones de los coeficientes parciales de correlación del modelo, conviene obtener sus correspondientes intervalos de confianza:

```
confint(object = modelo.log.m, level = 0.95)
```

```
##                     2.5 %      97.5 %
## (Intercept) -56.985558236 91.66680901
## Year         -0.045809580  0.02869546
## Lag1         -0.092972584  0.01093101
## Lag2          0.007001418  0.11291264
## Lag3         -0.068140141  0.03671410
## Lag4         -0.079519582  0.02453326
## Lag5         -0.066090145  0.03762099
## Volume       -0.131576309  0.13884038
```

Según el modelo resultante, el valor del mercado en las dos semanas previas (*Lag2*) es el único predictor estadísticamente significativo (β_1 = 0,0594, *p-value* = 0,027). Su valor positivo indica que si el valor en el mercado fue positivo hace dos semanas, es más probable que lo sea en el día actual. Más concretamente, por cada unidad que se incrementa la variable *Lag2*, se espera que el **logaritmo de *odds*** de la variable *Direction* se incremente en promedio 0,0594 unidades.

En este caso, aplicando la inversa del logaritmo natural para *Lag 2* obtenemos los *odds*:

$$e^{0,0594} = 1,06$$

Es decir, por cada unidad que se incrementa la variable *Lag2*, los ***odds*** de que *Direction* sea "Up" se incrementan en promedio 1,06 unidades. No hay que confundir esto último con que la probabilidad aumente en un 1,06%.

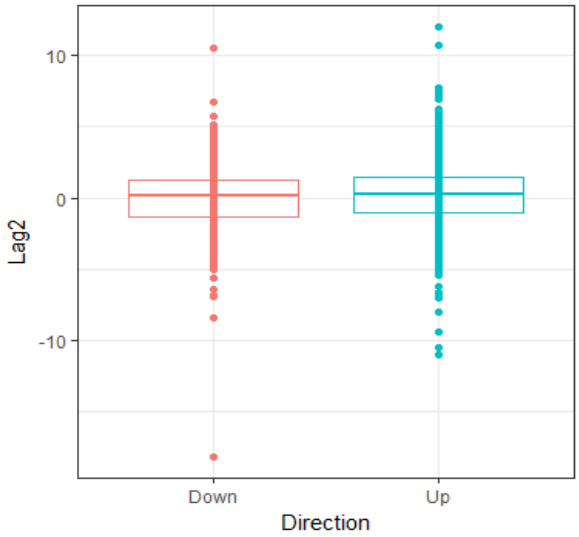
Esto corresponde a una probabilidad de que el mercado tenga un valor positivo en el día de hoy de

$$p = \frac{e^{0,0594}}{1 + e^{0,0594}} = \mathbf{0,51}$$

Modelo logístico SIMPLE

Procederemos a ajustar un modelo logístico simple utilizando solamente el predictor que resultó significativo en el modelo múltiple, *Lag2*.

```
ggplot(data = Weekly, mapping = aes(x = Direction, y = Lag2)) +
geom_boxplot(aes(color = Direction)) +
geom_point(aes(color = Direction)) +
theme_bw() +
theme(legend.position = "null")
```



Esta vez vamos a dividir nuestro set de datos en observaciones de entrenamiento (para ajustar el modelo) y observaciones de test (para evaluar el mismo):

```
# Training: observaciones desde 1990 hasta 2008
datos.entrenamiento <-  (Year < 2009)
# Test: observaciones de 2009 y 2010
datos.test <-  Weekly[!datos.entrenamiento, ]

nrow(datos.entrenamiento) + nrow(datos.test)

## integer(0)

# Ajuste del modelo logístico simple
modelo.log.s <- glm(Direction ~ Lag2, data = Weekly, family = binomial,
                    subset = datos.entrenamiento)


summary(modelo.log.s)
## Call:
## glm(formula = Direction ~ Lag2, family = binomial, data = Weekly,
##     subset = datos.entrenamiento)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.536  -1.264   1.021   1.091   1.368
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.20326    0.06428   3.162  0.00157 **
## Lag2         0.05810    0.02870   2.024  0.04298 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1354.7  on 984  degrees of freedom
## Residual deviance: 1350.5  on 983  degrees of freedom
## AIC: 1354.5
```

```
##
## Number of Fisher Scoring iterations: 4
```

Conclusiones del modelo:

El coeficiente estimado para *Lag2* (β_1 = 0,058) en el modelo simple apenas difiere del estimado en el modelo múltiple. En este caso, el valor esperado del *logaritmo de odds* es

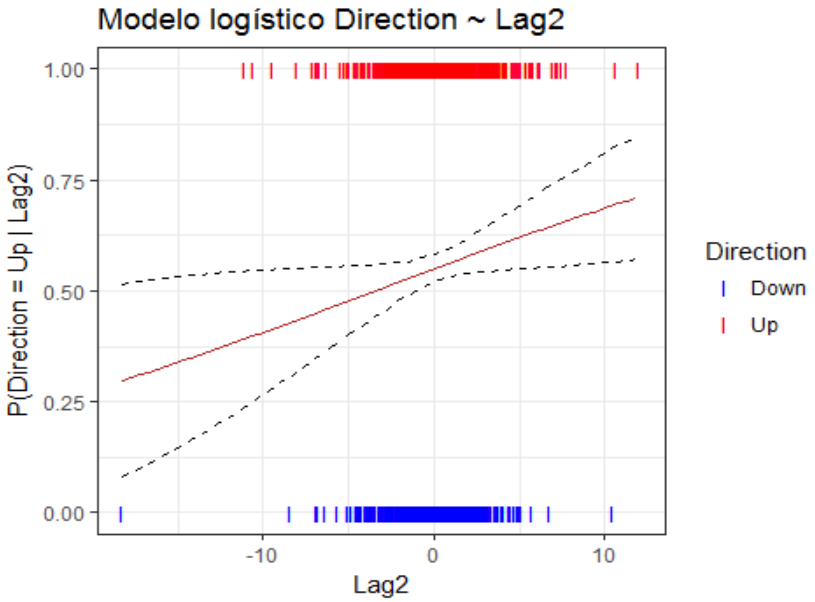
$$e^{0,0581} = 1,05$$

3. REPRESENTACIÓN GRÁFICA DEL MODELO

Dado que las unidades con las que el modelo logístico devuelve las predicciones se corresponden con el logaritmo de *odds*, es necesario convertirlas en probabilidad. Con la función *predict( )* y especificando el argumento *type = "response"* podemos obtener directamente las probabilidades en lugar del logaritmo de *odds*. Con *ggplot2* podemos representar el gráfico de la siguiente manera:

```
# Vector con nuevos valores interpolados en el rango del predictor Lag2
nuevos_puntos <- seq(from = min(Weekly$Lag2), to = max(Weekly$Lag2), by = 0.5)

# Predicción de los nuevos puntos según el modelo. La función predict() calcula la
 probabilidad de que la variable respuesta pertenezca al nivel de referencia (en e
ste caso "Up")
predicciones <- predict(modelo.log.s, newdata = data.frame(Lag2 = nuevos_puntos),
                        se.fit = TRUE, type = "response")

names(predicciones)

## [1] "fit"            "se.fit"         "residual.scale"

# Límites del intervalo de confianza (95%) de las predicciones
CI_inferior <- predicciones$fit - 1.96 * predicciones$se.fit
CI_superior <- predicciones$fit + 1.96 * predicciones$se.fit

# Matriz de datos con los nuevos puntos y sus predicciones
datos_curva <- data.frame(Lag2 = nuevos_puntos, probabilidad = predicciones$fit,
                          CI.inferior = CI_inferior, CI.superior = CI_superior)

head(datos_curva,4)

##      Lag2 probabilidad CI.inferior CI.superior
## 1 -18.195    0.2986393  0.08140726   0.5158713
## 2 -17.695    0.3047588  0.09092464   0.5185929
## 3 -17.195    0.3109481  0.10069231   0.5212038
## 4 -16.695    0.3172057  0.11070742   0.5237040



# Codificación 0,1 de la variable respuesta Direction
Weekly$Direction <- ifelse(Weekly$Direction == "Down", yes = 0, no = 1)
```

```r
ggplot(Weekly, aes(x = Lag2, y = Direction)) +
geom_point(aes(color = as.factor(Direction)), shape = "I", size = 3) +
geom_line(data = datos_curva, aes(y = probabilidad), color = "firebrick") +
geom_line(data = datos_curva, aes(y = CI.superior), linetype = "dashed") +
geom_line(data = datos_curva, aes(y = CI.inferior), linetype = "dashed") +
labs(title = "Modelo logístico Direction ~ Lag2",
     y = "P(Direction = Up | Lag2)",
     x = "Lag2") +
scale_color_manual(labels = c("Down", "Up"), values = c("blue", "red")) +
guides(color=guide_legend("Direction")) +
theme(plot.title = element_text(hjust = 0.5)) +
theme_bw()
```



4. EVALUACIÓN DEL MODELO

- **anova()** -> Análisis de la varianza. Comparar modelos (anidados)
- **predict()**-> Predicciones a partir de los resultados de diversas funciones de ajuste de modelos
- **table()**-> Tabla de contingencia de los conteos en cada combinación de niveles de factores
- **mosaic()**-> Diagrama de mosaico para visualizar la asociación entre dos variables categóricas

Para evaluar si el modelo logístico es válido, se analiza tanto el modelo en su conjunto como los predictores que lo forman. El modelo se considerará útil si es capaz de mejorar la predicción de las observaciones respecto al modelo nulo sin predictores. Para ello se analiza la significancia de la diferencia ("Deviance") de residuos entre ambos modelos ("Null deviance" y "Residual deviance"), con un estadístico

que sigue la distribución **chi-cuadrado** con grados de libertad correspondientes a la diferencia de los grados de libertad de ambos modelos.

```
anova(modelo.log.s, test ='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Direction
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                  984     1354.7
## Lag2  1   4.1666       983     1350.5  0.04123 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

En este caso, el modelo con el predictor *Lag2* sí es significativo respecto al modelo nulo.

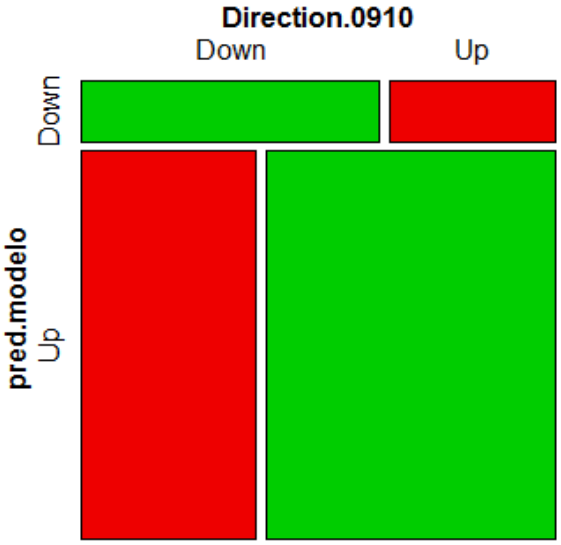
Es importante también analizar el porcentaje de predicciones correctas además de los falsos positivos y falsos negativos que hace nuestro modelo para evaluar su potencial. Para este ejemplo utilizaremos un *threshold* de 0,5. Si la probabilidad predicha de que el valor del mercado sea positivo es mayor de 0.5, la observación se asignará al nivel 1 ("Up"), y si es menor se asignará al nivel 0 ("Down"). Además de evaluar el test-error global, es conveniente identificar como se reparte este error entre falsos positivos y falsos negativos, ya que puede ocurrir que un modelo sea mucho mejor prediciendo en una dirección que en otra. Esto se ve directamente influenciado por límite de clasificación o *threshold* establecido.

```
# Cálculo de la probabilidad predicha por el modelo con los datos de test
prob.modelo <- predict(modelo.log.s, newdata = datos.test, type = "response")

# Vector de elementos "Down"
pred.modelo <- rep("Down", length(prob.modelo))
# Sustitución de "Down" por "Up" si la p > 0.5
pred.modelo[prob.modelo > 0.5] <- "Up"

Direction.0910 = Direction[!datos.entrenamiento]

# Matriz de confusión
matriz.confusion <- table(pred.modelo, Direction.0910)
matriz.confusion
```

```
##            Direction.0910
## pred.modelo Down Up
```

```
##         Down     9  5
##         Up      34 56
```

```
library(vcd)
mosaic(matriz.confusion, shade = T, colorize = T,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)))
```



El porcentaje de predicciones correctas sobre los datos de test es

```
mean(pred.modelo == Direction.0910)
```

```
## [1] 0.625
```

$$\frac{9+56}{104} = \mathbf{62,5}\%$$

Por tanto, el *test error rate* es

100 – 62,5 = **37,5%**

Además, para aquellas semanas con un valor de mercado al alza, el modelo clasifica correctamente el

$$\frac{56}{56+5} = \mathbf{91,8}\% \text{ de las observaciones}$$

mientras que para las semanas con un valor de mercado a la baja, el modelo acierta en un

$$\frac{9}{9+34} = \mathbf{20,93}\% \text{ de las veces}$$

MODELO FINAL:

$$Logit(Direction) = 0{,}20326 + 0{,}05810 * Lag2$$

$$p(Direction) = \frac{e^{2{,}20326 + 0{,}05810 * \mathrm{Lag2}}}{1 + e^{2{,}20326 + 0{,}05810 * \mathrm{Lag2}}}$$

COMPARACIÓN ENTRE REGRESIÓN LOGÍSTICA, LDA, QDA Y KNN

La **regresión logística** y el **LDA** son métodos muy próximos: ambos casos son funciones lineales de *x*, por lo que ambos producen límites de decisión lineales, aunque LDA se aplica para casos en los que la variable respuesta cuenta con predictores con más de dos clases. La única diferencia es que en regresión logística, β_0 y β_1 se estiman mediante el método de máxima verosimilitud, mientras que en el caso del LDA los estimadores se corresponden a la media y varianza de una distribución normal. Así, la regresión logística puede dar mejores resultados si la condición de normalidad no se cumple.

LDA es un método mucho menos flexible que **QDA** y sufre de menos varianza. Ello puede suponer una mejora en la predicción, pero hay un inconveniente: si la asunción del LDA de que todas las clases comparten la misma matriz de covarianza no es correcta en realidad, el LDA puede sufrir un alto *bias* o sesgo. Visto de otra manera, LDA suele ser mejor que QDA si contamos con relativamente pocas observaciones de entrenamiento y reducir la varianza es importante. Por el contrario, se recomienda QDA si el set de observaciones de entrenamiento es muy grande y la varianza del clasificador no supone un problema, o si el supuesto de una matriz de covarianza común entre las clases claramente no se cumple.

Si el verdadero límite de Bayes es lineal, LDA será una aproximación más precisa que QDA. Si por el contrario no es lineal, QDA será una mejor opción.

Por otro lado, el método **KNN** adquiere un enfoque distinto a la hora de clasificar: identifica las K observaciones más cercanas a *x* para clasificar dicha observación. La observación es clasificada en la clase mayoritaria a la que pertenecen las K observaciones vecinas más cercanas. Además, se trata de un método no paramétrico, ya que no asume ninguna forma sobre el límite de decisión. Por lo tanto, cuando los límites de decisión son altamente no lineales, KNN puede superar el QDA si el número de observaciones no es limitado. Una desventaja del KNN cuando *p* aumenta es que hay muy pocas observaciones "cerca" de cualquier observación de test, por lo que es importante que si aumentan el número de predictores, lo haga también el número de observaciones (este fenómeno se conoce como ***curse of dimensionality***).

Paramétrico	Límite de decisión **LINEAL**	**Regresión Logística**	K = 2, no se requiere normalidad
		LDA	K > 2, distribución normal y varianza común entre clases
	Límite de decisión **NO LINEAL**	**QDA**	K > 2, normalidad, ≠ matriz de covarianza entre clases, más flexible que LDA
NO paramétrico		**KNN**	Límite de decisión altamente no lineal, más flexible que QDA

*Los métodos lineales pueden flexibilizarse añadiendo transformaciones e interacciones de los predictores.

BIBLIOGRAFÍA

OpenIntro Statistics: Third Edition, David M Diez, Christopher D Barr, Mine Çetinkaya-Rundel

An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics)