Actividad 7: Regresión Logísitica

Daniela Jiménez Téllez

2024-11-05

Problema

Trabaja con el set de datos Weekly, que forma parte de la librería ISLR. Este set de datos contiene información sobre el rendimiento porcentual semanal del índice bursátil S&P 500 entre los años 1990 y 2010. Se busca predecir el tendimiento (positivo o negativo) dependiendo del comportamiento previo de diversas variables de la bolsa bursátil S&P 500.

Encuentra un modelo logístico para encontrar el mejor conjunto de predictores que auxilien a clasificar la dirección de cada observación.

Se cuenta con un set de datos con 9 variables (8 numéricas y 1 categórica que será nuestra variable respuesta: Direction). Las variables Lag son los valores de mercado en semanas anteriores y el valor del día actual (Today). La variable volumen (Volume) se refiere al volumen de acciones.

Instrucciones

1. El análisis de datos. Estadísticas descriptivas y coeficiente de correlación entre las variables.

```
data(Weekly)
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)
```

```
## Rows: 1,089
## Columns: 9
## $ Year      <dbl> 1990, 1990, 1990, 1990, 1990, 1990, 1990, 1990, 1990, 1990, …
## $ Lag1      <dbl> 0.816, -0.270, -2.576, 3.514, 0.712, 1.178, -1.372, 0.807, 0…
## $ Lag2      <dbl> 1.572, 0.816, -0.270, -2.576, 3.514, 0.712, 1.178, -1.372, 0…
## $ 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, 0.712, …
## $ Lag5      <dbl> -3.484, -0.229, -3.936, 1.572, 0.816, -0.270, -2.576, 3.514,…
## $ Volume    <dbl> 0.1549760, 0.1485740, 0.1598375, 0.1616300, 0.1537280, 0.154…
## $ Today     <dbl> -0.270, -2.576, 3.514, 0.712, 1.178, -1.372, 0.807, 0.041, 1…
## $ Direction <fct> Down, Down, Up, Up, Up, Down, Up, Up, Up, Down, Down, Up, Up…
```

```
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
##
##
##
##
```

2. Formula un modelo logístico con todas las variables menos la variable “Today”. Calcula los intervalos de confianza para las β_i. Detecta variables que influyen y no influyen en el modelo. Interpreta el efecto de la variables en los odds (momios).

```
# Modelo logístico

modelo.log.m <- glm(Direction ~ . -Today, data = Weekly, family = binomial)

# Resumen

summary(modelo.log.m)
```

```
##
## Call:
## glm(formula = Direction ~ . - Today, family = binomial, data = Weekly)
##
## 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
```

```
# Intervalos de confianza

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

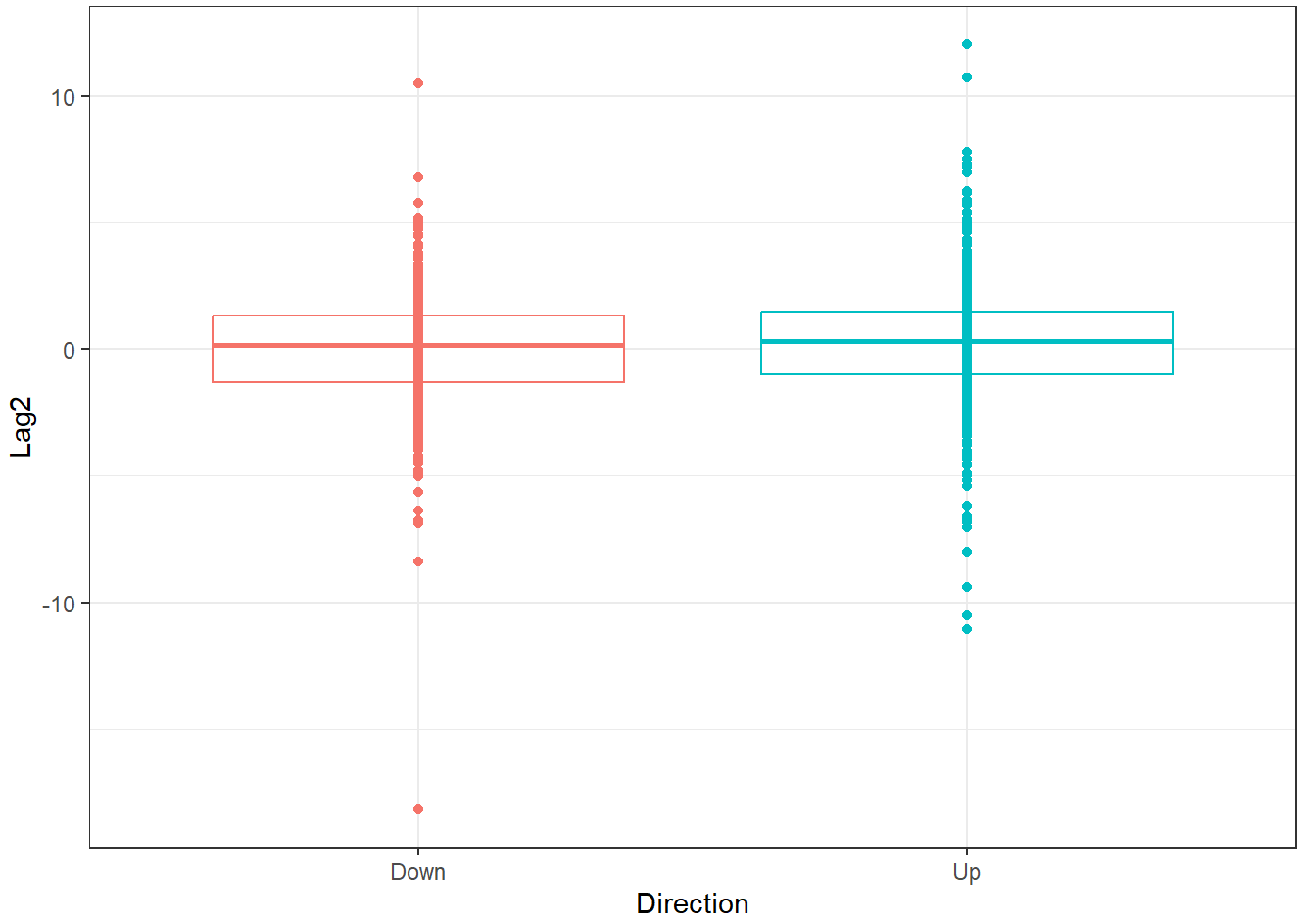
```
## Waiting for profiling to be done...
```

```
##                    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
```

```
# Interpretación de los coeficientes en términos de odds

exp(coef(modelo.log.m))
```

```
##  (Intercept)         Year         Lag1         Lag2         Lag3         Lag4
## 3.027468e+07 9.915361e-01 9.601291e-01 1.061251e+00 9.846412e-01 9.730534e-01
##         Lag5       Volume
## 9.860757e-01 1.003262e+00
```

```
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")
```



3. Divide la base de datos en un conjunto de entrenamiento (datos desde 1990 hasta 2008) y de prueba (2009 y 2010). Ajusta el modelo encontrado.

```
# División de los datos

train_data <- subset(Weekly, Year >= 1990 & Year <= 2008)
test_data <- subset(Weekly, Year >= 2009 & Year <= 2010)

# Ajustar el modelo en el conjunto de entrenamiento

modelo.log.train <- glm(Direction ~ . - Today, data = train_data, family = binomial)

cat("--- MODELO LOGÍSITICO --- \n")
```

```
## --- MODELO LOGÍSITICO ---
```

```
summary(modelo.log.train)
```

```
##
## Call:
## glm(formula = Direction ~ . - Today, family = binomial, data = train_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.779438  42.446904   0.065   0.9478
## Year        -0.001227   0.021282  -0.058   0.9540
## Lag1        -0.062163   0.029466  -2.110   0.0349 *
## Lag2         0.044903   0.030066   1.493   0.1353
## Lag3        -0.015305   0.029595  -0.517   0.6050
## Lag4        -0.030967   0.029342  -1.055   0.2913
## Lag5        -0.037599   0.029353  -1.281   0.2002
## Volume      -0.085115   0.096432  -0.883   0.3774
## ---
## 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: 1342.3  on 977  degrees of freedom
## AIC: 1358.3
##
## Number of Fisher Scoring iterations: 4
```

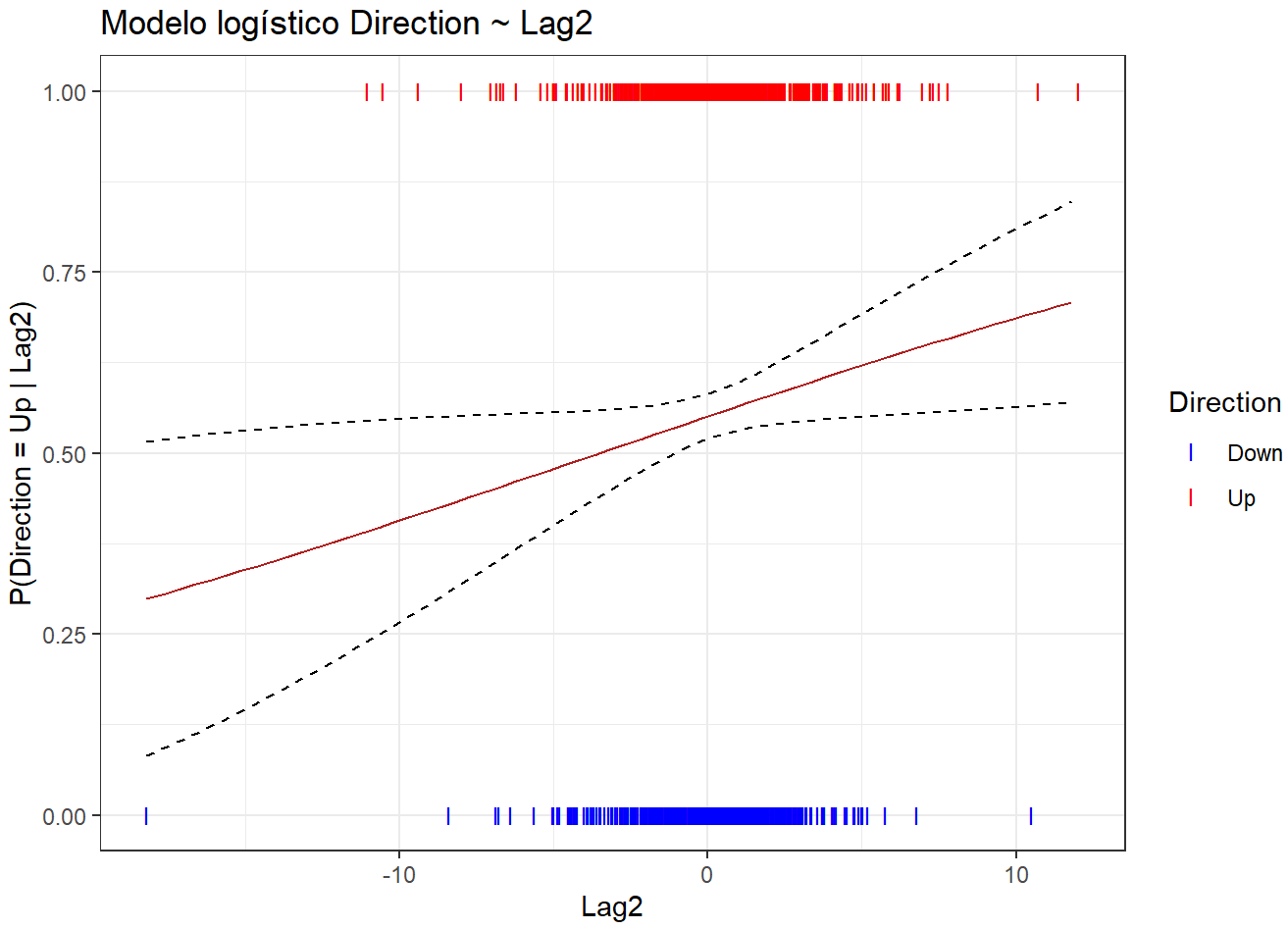
4. Formula el modelo logístico sólo con las variables significativas en la base de entrenamiento.

```
# M odelo logístico sólo con las variables significativas

modelo.log.s <- glm(Direction ~ Lag2, data = train_data, family = binomial)

# Resumen

summary(modelo.log.s)
```

```
##
## Call:
## glm(formula = Direction ~ Lag2, family = binomial, data = train_data)
##
## 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
```

5. Representa gráficamente el modelo:

```
# 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 con el comando predict(), se calcula la probabilidad
# de que la variable respuesta pertenezca al nivel de referencia (en este caso "Up")

predicciones <- predict(modelo.log.s, newdata = data.frame(Lag2 = nuevos_puntos),
                        se.fit = TRUE, type = "response")

# 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)

# Codificación 0,1 de la variable respuesta Direction

Weekly$Direction <- ifelse(Weekly$Direction == "Down", yes = 0, no = 1)

# Gráfico con ggplot2

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()
```

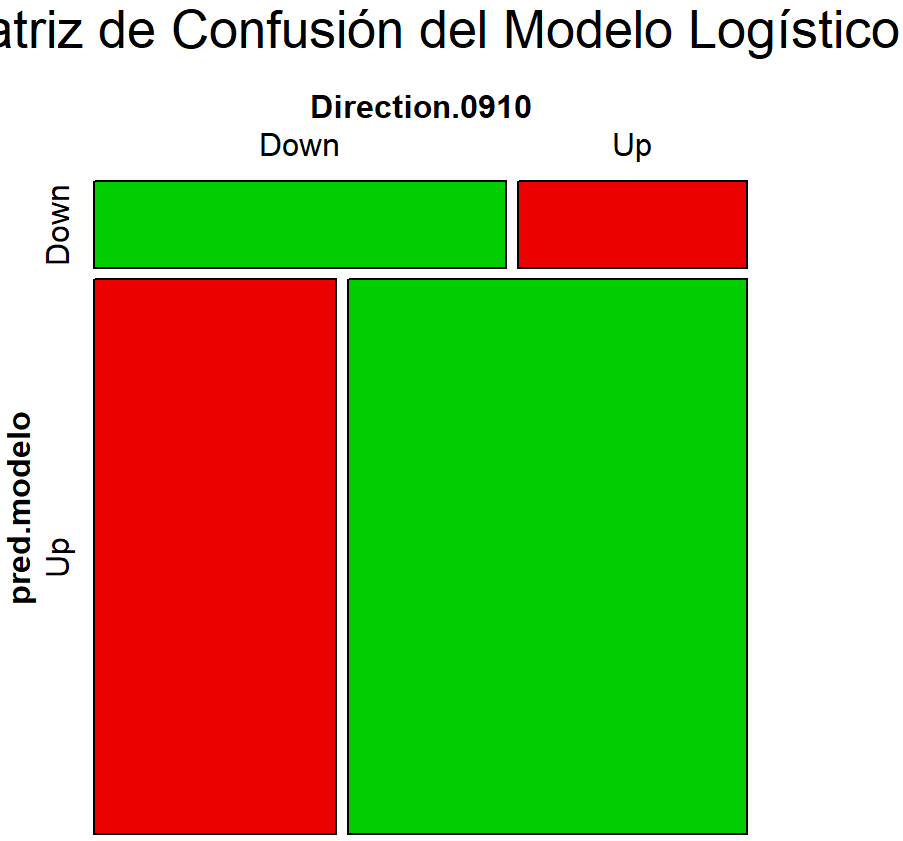


6. Evalúa el modelo con las pruebas de verificación correspondientes (Prueba de chi cuadrada, matriz de confusión).

```
# Cálculo de la probabilidad predicha por el modelo con los datos de prueba

prob.modelo <- predict(modelo.log.s, newdata = test_data, type = "response")

# Vector de elementos “Down”

pred.modelo <- rep("Down", length(prob.modelo))

# Sustitución de “Down” por “Up” si la probabilidad es mayor a 0.5

pred.modelo[prob.modelo > 0.5] <- "Up"

# Extraemos la variable Direction en el conjunto de prueba

Direction.0910 <- test_data$Direction

# Matriz de confusión

matriz.confusion <- table(pred.modelo, Direction.0910)
print(matriz.confusion)
```

```
##            Direction.0910
## pred.modelo Down Up
##        Down    9  5
##        Up     34 56
```

```
# Visualización con gráfico de mosaico

mosaic(matriz.confusion, shade = TRUE, colorize = TRUE,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)),
       main = "Matriz de Confusión del Modelo Logístico")
```




```
# Cálculo de la precisión

precision <- mean(pred.modelo == Direction.0910)
print(paste("Precisión del modelo:", round(precision * 100, 2), "%"))
```

```
## [1] "Precisión del modelo: 62.5 %"
```

7. Escribe (ecuación), grafica el modelo significativo e interprétalo en el contexto del problema. Añade posibles es buen modelo, en qué no lo es, cuánto cambia)

La ecuación es:

$$\text{logit}(P(\text{Direction} = \text{Up})) = \beta_0 + \beta_1 \times \text{Lag2}$$

Donde:

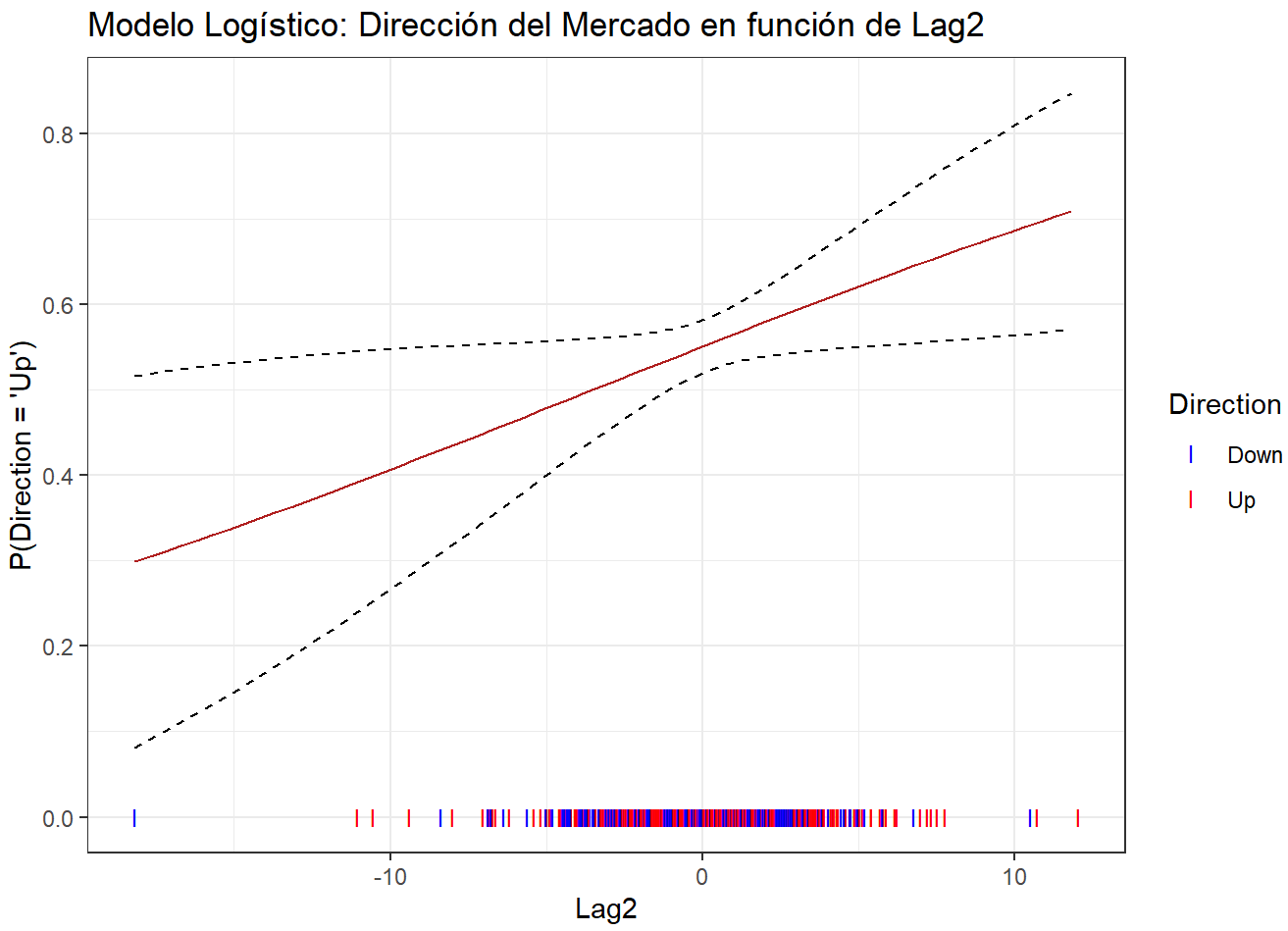
- β_0 es el intercepto estimado.
- β_1 es el coeficiente de la variable Lag2.

```
# Coeficientes del modelo final

coeficientes <- coef(modelo.log.s)
cat("La ecuación del modelo es: logit(P(Direction = 'Up')) =",
    round(coeficientes[1], 2), "+", round(coeficientes[2], 2), "* Lag2\n")
```

```
## La ecuación del modelo es: logit(P(Direction = 'Up')) = 0.2 + 0.06 * Lag2
```

Ahora para la gráfica del modelo

```
# Vector de nuevos valores para Lag2

nuevos_puntos <- seq(from = min(Weekly$Lag2), to = max(Weekly$Lag2), by = 0.5)

# Predicción de probabilidad con intervalo de confianza

predicciones <- predict(modelo.log.s, newdata = data.frame(Lag2 = nuevos_puntos),
                        se.fit = TRUE, type = "response")

# Cálculo del intervalo de confianza

CI_inferior <- predicciones$fit - 1.96 * predicciones$se.fit
CI_superior <- predicciones$fit + 1.96 * predicciones$se.fit

datos_curva <- data.frame(Lag2 = nuevos_puntos, probabilidad = predicciones$fit,
                          CI_inferior = CI_inferior, CI_superior = CI_superior)

ggplot(Weekly, aes(x = Lag2, y = as.numeric(Direction == "Up"))) +
  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: Dirección del Mercado en función de Lag2",
       y = "P(Direction = 'Up')", 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()
```



Con base en esto se puede decir que un coeficiente positivo para Lag2 sugiere que a medida que Lag2 aumenta, la probabilidad de que el mercado suba también aumenta. Igualmente, en términos de odds, cada incremento en Lag2 multiplica los odds de “Up” por e^{β_1}. Este modelo muestra una relación significativa entre Lag2 y la dirección del mercado, aunque su precisión depende del contexto y de la influencia de otros factores. En el caso de la matriz de confusión, esta muestra que el modelo tiene más predicciones correctas de “Up” que de “Down”, lo que nos dice que podría estar más inclinado a predecir correctamente cuando la dirección es “Up”.