5-regresion-logistica

October 17, 2023

1 5 Regresión logística

Francisco Mestizo Hernández A01731549

Comenzamos instalando las librerias que se utilizarán para la actividad

[1]:
```
install.packages('ISLR')
install.packages('vcd')
library(vcd)
library('ISLR')
library('tidyverse')
```

```
Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Loading required package: grid


Attaching package: ‘ISLR’


The following object is masked from ‘package:vcd’:

    Hitters


  Attaching core tidyverse packages                tidyverse
2.0.0
  dplyr     1.1.3      readr     2.1.4
  forcats   1.0.0      stringr   1.5.0
  ggplot2   3.4.3      tibble    3.2.1
  lubridate 1.9.3      tidyr     1.3.0
  purrr     1.0.2
  Conflicts
tidyverse_conflicts()
  dplyr::filter() masks stats::filter()
```

```
  dplyr::lag()    masks stats::lag()
  Use the conflicted package
(<http://conflicted.r-lib.org/>) to force all conflicts to
become errors
```

1.1 Los datos

Después cargamos lso datos. Podemos ver que se muestra al inicio la tabla con los datos que analizaremos, podemos ver estadísticos de cada variable, como su media, sus cuartiles y su rango.

Además, se puede ver la matriz de correlación para todas las variables. Claramente se puede ver que las variables no tienen correlación entre si ya que aparecen circulares. Excepto el tiempo y el volumen, esas dos variables sí se correlacionan y por eso hacemos una gráfica que muestra la relación en grande.

La gráfica nos dice que conforme avanza el tiempo hay una mayor venta de acciones y que al parecer crece parecido a una exponencial.

[2]:
```
head(Weekly)
glimpse(Weekly)
summary(Weekly)
pairs(Weekly)
cor(Weekly[, -9])
attach(Weekly)
plot(Volume)
```

A data.frame: 6 × 9

	Year <dbl>	Lag1 <dbl>	Lag2 <dbl>	Lag3 <dbl>	Lag4 <dbl>	Lag5 <dbl>	Volume <dbl>	Today <dbl>	Direction <fct>
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

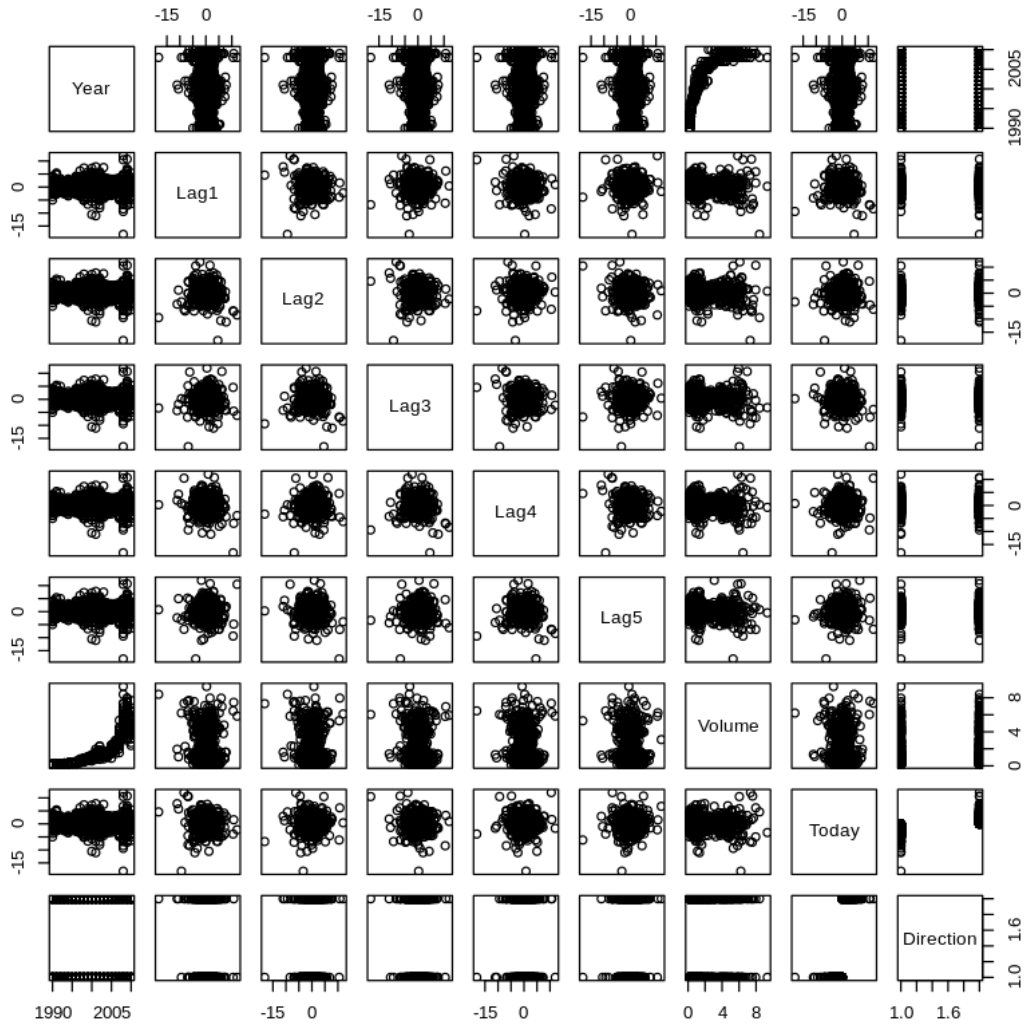
```
Rows: 1,089
Columns: 9
$ Year      <dbl> 1990, 1990, 1990, 1990, 1990, 1990, 1990,
1990, 1990, 1990, …
$ Lag1      <dbl> 0.816, -0.270, -2.576, 3.514, 0.712, 1.178,
-1.372, 0.807, 0…
$ Lag2      <dbl> 1.572, 0.816, -0.270, -2.576, 3.514, 0.712,
1.178, -1.372, 0…
$ Lag3      <dbl> -3.936, 1.572, 0.816, -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,
```

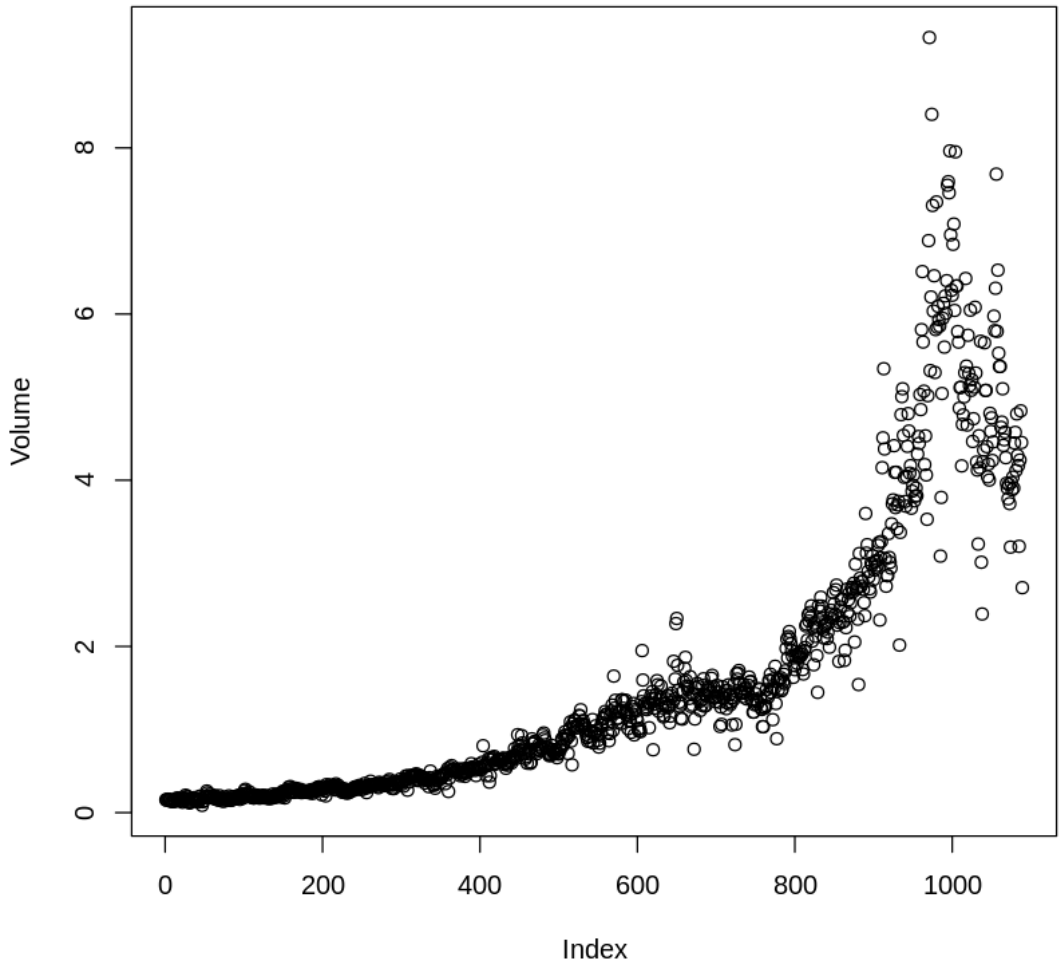
```
0.1537280, 0.154…
$ Today     <dbl> -0.270, -2.576, 3.514, 0.712, 1.178,
-1.372, 0.807, 0.041, 1…
$ Direction <fct> Down, Down, Up, Up, Up, Down, Up, Up, Up,
Down, Down, Up, Up…
```

```
      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  
```

A matrix: 8×8 of type dbl

	Year	Lag1	Lag2	Lag3	Lag4	La
Year	1.00000000	-0.032289274	-0.03339001	-0.03000649	-0.031127923	-0
Lag1	-0.03228927	1.000000000	-0.07485305	0.05863568	-0.071273876	-0
Lag2	-0.03339001	-0.074853051	1.00000000	-0.07572091	0.058381535	-0
Lag3	-0.03000649	0.058635682	-0.07572091	1.00000000	-0.075395865	0.
Lag4	-0.03112792	-0.071273876	0.05838153	-0.07539587	1.000000000	-0
Lag5	-0.03051910	-0.008183096	-0.07249948	0.06065717	-0.075675027	1.
Volume	0.84194162	-0.064951313	-0.08551314	-0.06928771	-0.061074617	-0
Today	-0.03245989	-0.075031842	0.05916672	-0.07124364	-0.007825873	0.


Year
Lag1
Lag2
Lag3
Lag4
Lag5
Volume
Today
Direction



1.2 El modelo

Ahora, creamos un modelo con todas las variable, así podremos saber cuales son las significativas.

[3]:
```
modelo.log.m <- glm(Direction ~ . -Today, data = Weekly, family = binomial)
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
```

Calculamos los intervalos de confianza para todas las betas del modelo que generamos. Como podemos ver en el modelo de arriba solamente Lag2 parece ser significativa.

[4]:
```
contrasts(as.factor(Direction))
confint(object = modelo.log.m, level = 0.95)
```

A matrix: 2 × 1 of type dbl

	Up
Down	0
Up	1

```
Waiting for profiling to be done…
```

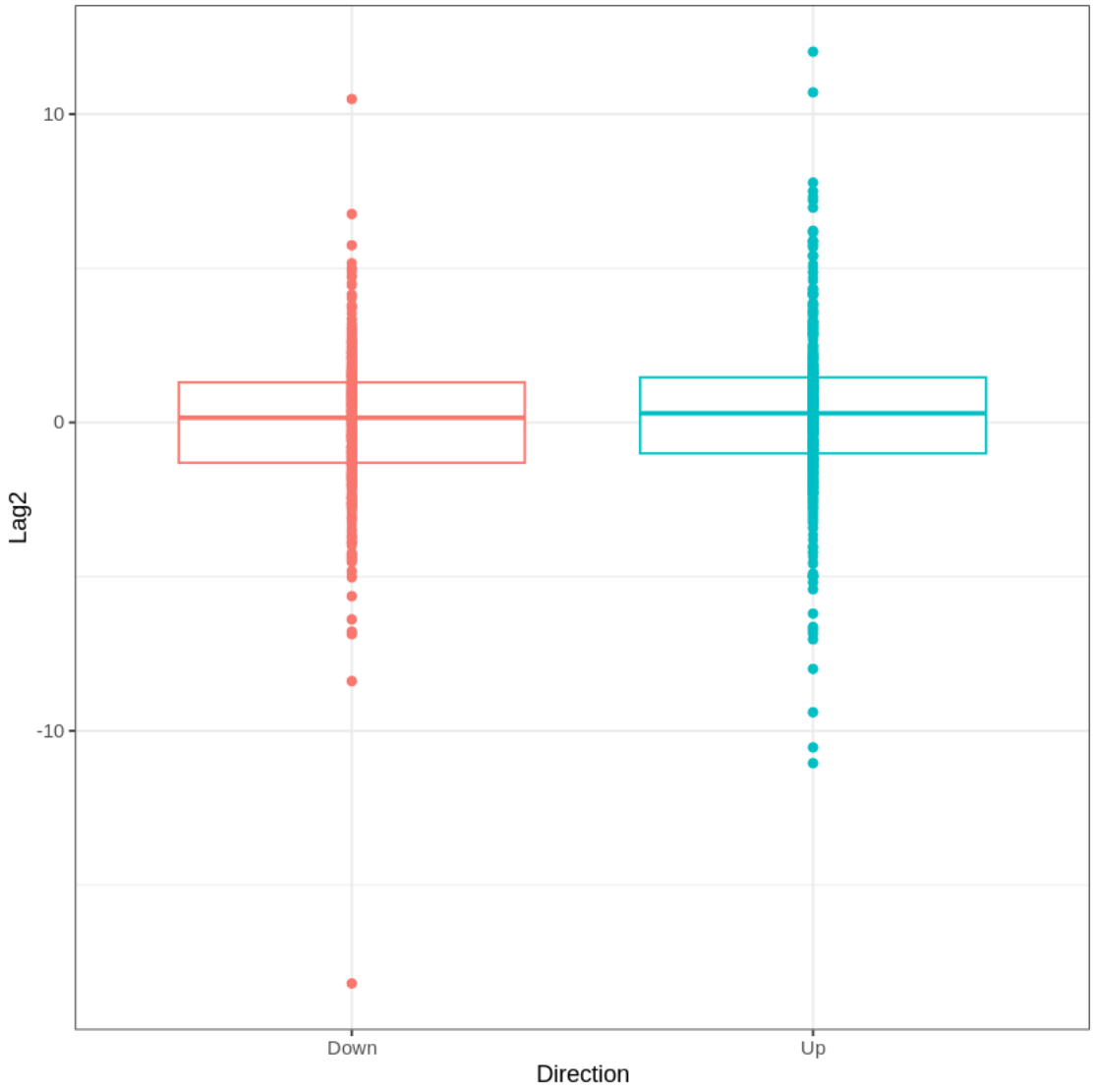
A matrix: 8 × 2 of type dbl

	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

Podemos ver que la variable significativa es Lag2. Aquí se muestra un boxplot de esta variable

[5]:
```
# Gráfico de las variables significativas (boxplot), ejemplo: 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")
```



Ahora, dividimos los datos de entrenamiento hasta el 2008 para hacer las pruebas y los datos posteriores (2009 y 2010) se usarán para hacer las predicciones con el modelo.

[6]:
```
# Training: observaciones desde 1990 hasta 2008
datos.entrenamiento <- (Year < 2009)
# Test: observaciones de 2009 y 2010
datos.test <- Weekly[!datos.entrenamiento, ]
# Verifica:
nrow(datos.entrenamiento) + nrow(datos.test)
# Ajuste del modelo logístico con variables significativas
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)

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
```

Ahora, podemos ver que solamente la variable de Lag2 parece ser significativa. Por esto es que podemos hacer otro modelo solamente con esta variable relacionada con la dirección.

El resultado que nos dará será la Beta 0 (0.203) y la beta 1 (0.058) que son los valores que van dentro de la formula de la regresión logística.

Con el modelo generado podemos ver las predicciones que realiza para los datos del 2009 y 2010

[7]:
```
# 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 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")
#El modelo devuelve las predicciones del logarítmo de Odds. La predicción se
#debe convertir en probabilidad. Eso se logra con el comando ‘predict’ y el
#‘type="response"’.
```

Aunque arriba se puede ver el resultado de las predicciones, es más facil verlo con la matriz generada aqui

[8]:
```
# Límites del intervalo de confianza (95%) de las predicciones
CI_inferior <- predicciones$fit - 1.96 * predicciones$se.fit
```

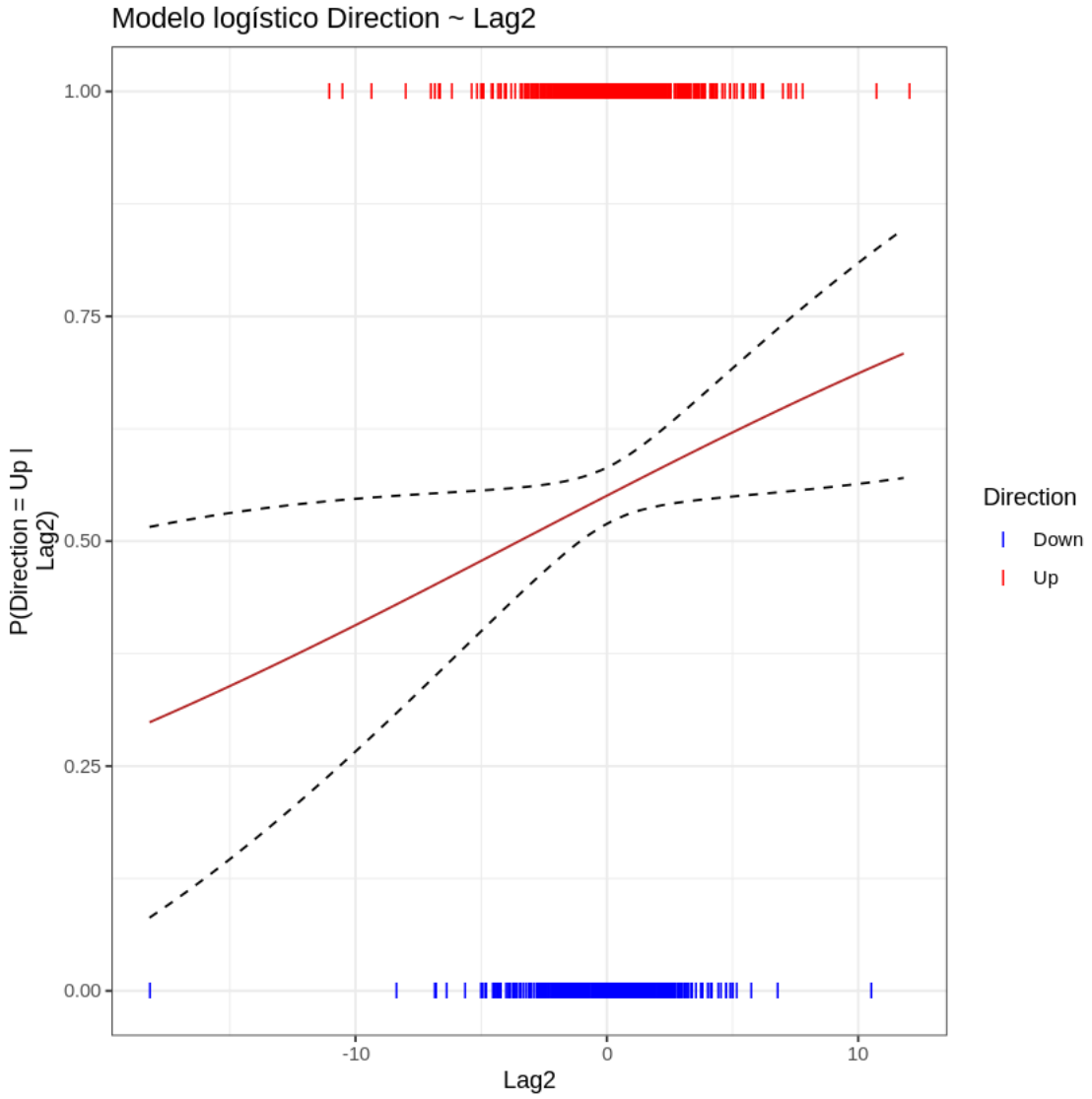
```
CI_superior <- predicciones$fit + 1.96 * predicciones$se.fit
# Matriz de datos con los nuevos puntos y sus predicciones
datos_curva <- data.frame(Lag2 = nuevos_puntos, probabilidad =
predicciones$fit, CI.inferior = CI_inferior, CI.superior = CI_superior)
datos_curva
```

A data.frame: 61 × 4

	Lag2 <dbl>	probabilidad <dbl>	CI.inferior <dbl>	CI.superior <dbl>
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
5	-16.195	0.3235302	0.12096662	0.5260938
6	-15.695	0.3299198	0.13146607	0.5283735
7	-15.195	0.3363729	0.14220145	0.5305443
8	-14.695	0.3428876	0.15316790	0.5326072
9	-14.195	0.3494620	0.16436005	0.5345640
10	-13.695	0.3560942	0.17577199	0.5364164
11	-13.195	0.3627820	0.18739729	0.5381668
12	-12.695	0.3695234	0.19922895	0.5398179
13	-12.195	0.3763161	0.21125942	0.5413727
14	-11.695	0.3831577	0.22348057	0.5428348
15	-11.195	0.3900459	0.23588368	0.5442082
16	-10.695	0.3969783	0.24845939	0.5454973
17	-10.195	0.4039523	0.26119770	0.5467069
18	-9.695	0.4109653	0.27408792	0.5478427
19	-9.195	0.4180147	0.28711859	0.5489108
20	-8.695	0.4250977	0.30027742	0.5499181
21	-8.195	0.4322117	0.31355115	0.5508722
22	-7.695	0.4393537	0.32692543	0.5517819
23	-7.195	0.4465208	0.34038459	0.5526571
24	-6.695	0.4537103	0.35391132	0.5535094
25	-6.195	0.4609192	0.36748624	0.5543521
26	-5.695	0.4681444	0.38108728	0.5552016
27	-5.195	0.4753830	0.39468875	0.5560773
28	-4.695	0.4826320	0.40826002	0.5570040
29	-4.195	0.4898883	0.42176344	0.5580132
30	-3.695	0.4971489	0.43515136	0.5591464
32	-2.695	0.5116706	0.4613097	0.5620314
33	-2.195	0.5189255	0.4738807	0.5639704
34	-1.695	0.5261725	0.4859140	0.5664311
35	-1.195	0.5334085	0.4971925	0.5696245
36	-0.695	0.5406305	0.5074442	0.5738168
37	-0.195	0.5478355	0.5163838	0.5792872
38	0.305	0.5550204	0.5238113	0.5862295
39	0.805	0.5621824	0.5297166	0.5946483
40	1.305	0.5693186	0.5342882	0.6043491
41	1.805	0.5764262	0.5378162	0.6150361
42	2.305	0.5835022	0.5405857	0.6264187
43	2.805	0.5905440	0.5428256	0.6382624
44	3.305	0.5975488	0.5447034	0.6503943
45	3.805	0.6045141	0.5463373	0.6626910
46	4.305	0.6114372	0.5478103	0.6750642
47	4.805	0.6183157	0.5491814	0.6874499
48	5.305	0.6251470	0.5504936	0.6998004
49	5.805	0.6319288	0.5517784	0.7120792
50	6.305	0.6386589	0.5530599	0.7242579
51	6.805	0.6453350	0.5543565	0.7363134

Finalmente, hacemos un gráfico del modelo direccion ~ lag2. El modelo es la linea roja y los intervalos de confianza estan representados por las lineas punteadas

[9]:
```
# Codificación 0,1 de la variable respuesta Direction
Weekly$Direction <- ifelse(Weekly$Direction == "Down", yes = 0, no = 1)
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()
```



1.3 Evaluación del modelo

Ahora que tenemos el modelo generado, le podemos hacer un anova con chi cuadrada para ver si es un buen modelo o no.

[10]:
```
#Chi cuadrada: Se evalúa la significancia del modelo con predictores con␣
 ↪respecto al
#modelo nulo ("Residual deviance" vs "Null deviance"). Si valor p es menor que␣
 ↪alfa será
#significativo.
anova(modelo.log.s, test ='Chisq')
```

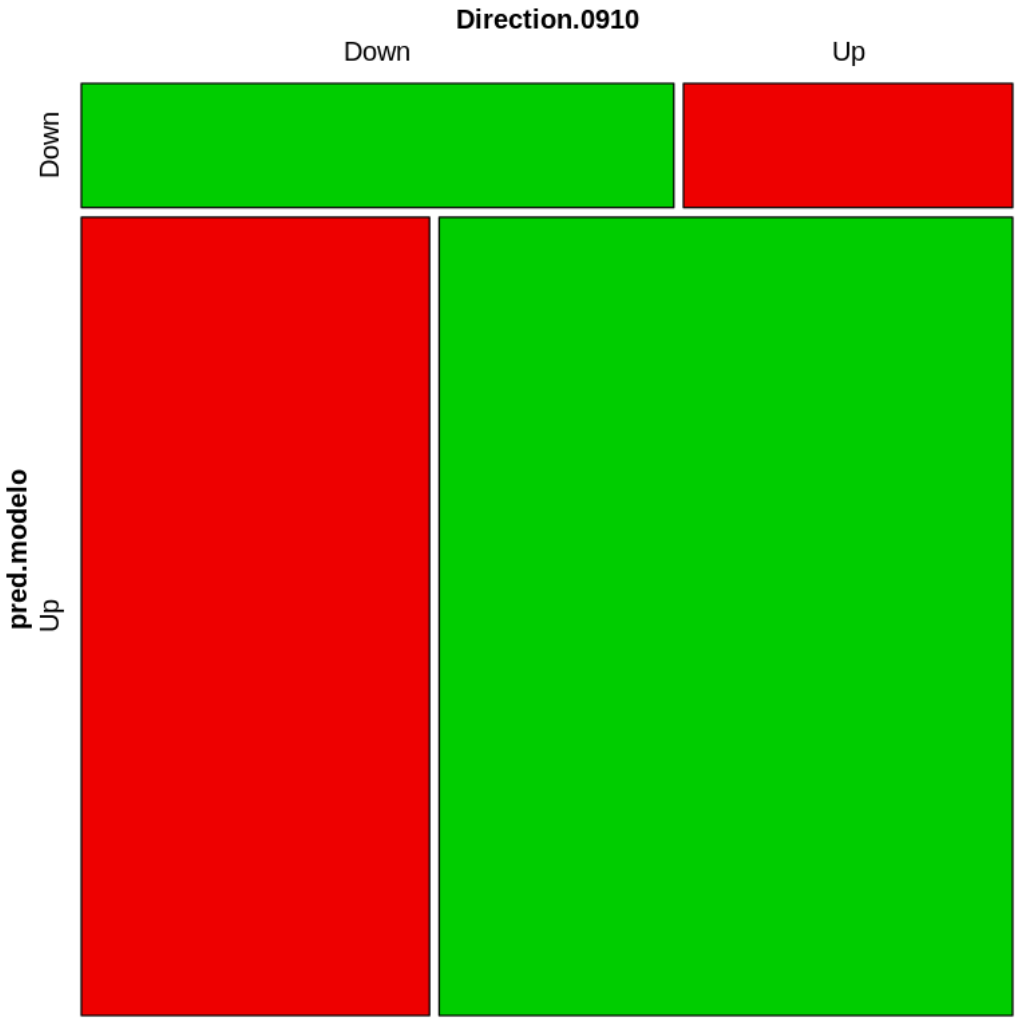
A anova: 2 × 5

	Df <int>	Deviance <dbl>	Resid. Df <int>	Resid. Dev <dbl>	Pr(>Chi) <dbl>
NULL	NA	NA	984	1354.710	NA
Lag2	1	4.166594	983	1350.543	0.04122861

Podemos ver que el modelo generado con Lag2 es bueno porque tiene un valor p menor a 0.5, por lo tanto es mas significativo. Por otro lado, el modelo con lag2 se explica mejor, por eso tiene un valor residual mas pequeño.

[11]:
```
# 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.confusión
mosaic(matriz.confusion, shade = T, colorize = T,
gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)))
mean(pred.modelo == Direction.0910)
```

0.625



Además, la matriz de confusión de arriba nos dice que tanto se equivoca el modelo y que tantos resultados correctos dio. Vemos que hay una gran área verde, lo que quiere decir que generalmente el modelo tiende a dar resultados correctos, aunque todavía se equivoca en pocas ocasiones.