Regresion Logistica 2020 reto

Andrés Villarreal González

2023-10-18

Leyendo los datos y librerias

```r
library(nnet)
library(tidyverse)
```

```
## ── 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()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force
all conflicts to become errors
```

```r
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
data <- read.csv("data_2020_RegLog.csv")

head(data)
```

```
##      PRS RH    SR  TOUT WSR WDR O3Concentracion
## 1 707.4 17 0.188 34.77 7.0  59               3
## 2 705.1 21 0.183 34.53 7.4  86               3
## 3 710.7 40 0.182 30.38 7.1  95               3
## 4 708.1 20 0.177 32.72 5.8 158               3
## 5 708.5 49 0.175 30.76 7.9  64               3
## 6 709.5 51 0.175 29.63 7.7  82               3
```

Dividir datos en entrenamiento y prueba

```
set.seed(123)

trainIndex <- createDataPartition(data$O3Concentracion, p = 0.8, list =
FALSE)

train_data <- data[trainIndex, ]
test_data <- data[-trainIndex, ]
```

Modelo de regresion logistica multinomial

```
model <- multinom(O3Concentracion ~ ., data = train_data)

## # weights:  24 (14 variable)
## initial  value 23056.576102
## iter  10 value 7125.538911
## iter  20 value 4729.345862
## iter  30 value 4476.012822
## iter  40 value 4409.160107
## iter  50 value 4335.656236
## final  value 4334.807740
## converged

summary(model)

## Call:
## multinom(formula = O3Concentracion ~ ., data = train_data)
##
## Coefficients:
##   (Intercept)         PRS          RH         SR      TOUT        WSR
## 2    4.792634 -0.01153557 -0.04447838 -0.1986845 0.1088022 0.09700283
## 3   17.447508 -0.03738808 -0.05544725 -1.7216879 0.2670784 0.03169910
##            WDR
## 2 -0.004790449
## 3 -0.010733776
##
## Std. Errors:
##    (Intercept)          PRS          RH          SR        TOUT
WSR
## 2 5.019627e-04 0.0002920562 0.002018622 0.137175769 0.005936102
0.005934777
## 3 3.925946e-05 0.0016513575 0.009031635 0.003569803 0.032597018
0.023017900
```

```
##             WDR
## 2 0.0004481321
## 3 0.0020268434
##
## Residual Deviance: 8669.615
## AIC: 8697.615
```

Predicción y precision del modelo

```
predicted_probs <- predict(model, newdata = test_data, type = "probs")
predicted_class <- predict(model, newdata = test_data, type = "class")

predicted_class <- factor(predicted_class, levels = c(1, 2, 3))
true_class <- factor(test_data$O3Concentracion, levels = c(1, 2, 3))

confusionMatrix(predicted_class, true_class)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction    1    2    3
##          1 4847  333   20
##          2   27   16    3
##          3    0    0    0
##
## Overall Statistics
##
##                Accuracy : 0.927
##                  95% CI : (0.9196, 0.9339)
##     No Information Rate : 0.9291
##     P-Value [Acc > NIR] : 0.7335
##
##                   Kappa : 0.0697
##
##  Mcnemar's Test P-Value : <2e-16
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity           0.99446 0.045845 0.000000
## Specificity           0.05108 0.993874 1.000000
## Pos Pred Value        0.93212 0.347826      NaN
## Neg Pred Value        0.41304 0.935962 0.995616
## Prevalence            0.92909 0.066527 0.004384
## Detection Rate        0.92394 0.003050 0.000000
## Detection Prevalence  0.99123 0.008769 0.000000
## Balanced Accuracy     0.52277 0.519860 0.500000
```

Valores p de los coeficientes

```
coefficients <- coef(model)
std_errors <- summary(model)$standard.errors
```

```
t_statistics <- coefficients / std_errors
df <- nrow(train_data) - length(coefficients)
p_values <- 2 * (1 - pt(abs(t_statistics), df))
p_values

##   (Intercept) PRS           RH       SR          TOUT       WSR
WDR
## 2           0   0 0.000000e+00 0.147522 0.000000e+00 0.0000000
0.000000e+00
## 3           0   0 8.441385e-10 0.000000 2.220446e-16 0.1684805
1.196845e-07
```

Significancia de variables independientes mediante la prueba de Wald

```
library(nnet)
library(car)

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some

Anova(model, type="II")

## Analysis of Deviance Table (Type II tests)
##
## Response: O3Concentracion
##      LR Chisq Df Pr(>Chisq)
## PRS     16.76  2  0.0002291 ***
## RH     510.14  2  < 2.2e-16 ***
## SR      13.65  2  0.0010889 **
## TOUT   493.53  2  < 2.2e-16 ***
## WSR    169.40  2  < 2.2e-16 ***
## WDR    138.06  2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```