KNN para el análisis de variables cualitativas con R

Contents

1 Introducción

Se quiere predecir la especie de dos pingüinos con las siguientes características:

```
xp = data.frame(
  long_pico = c(39.8,46.1),
  prof_pico = c(18.4,15.5),
  long_aleta = c(192,202),
  peso = c(3250,4000),
  isla = factor(c("Dream","Biscoe"), levels = c("Biscoe","Dream","Torgersen")),
  genero = factor(c("hembra","macho"))
)
xp
```

```
##   long_pico prof_pico long_aleta peso   isla genero
## 1      39.8      18.4        192 3250  Dream hembra
## 2      46.1      15.5        202 4000 Biscoe  macho
```

utilizando los datos del archivo pinguinos.csv y el algoritmo KNN.

2 Lectura y preparación de los datos

En primer lugar se leen los datos y se crean los factores correspondientes

```
## se leen los datos
d = read.csv("datos/pinguinos.csv", sep = ";")
```

```
# es conveniente convertir a factores las variables cualitativas
d$especie = factor(d$especie)
d$isla = factor(d$isla)
d$genero = factor(d$genero)
```

- Se normalizan los regresores numéricos:

```
# se lee la funcion que normaliza
source("funciones/knn_funciones.R")
```

```
# se normalizan y se guardan en un data.frame
d1 = data.frame(
```

```
  long_pico = knn_normaliza(d$long_pico),
  prof_pico = knn_normaliza(d$prof_pico),
  long_aleta = knn_normaliza(d$long_aleta),
  peso = knn_normaliza(d$peso)
)
```

- Se definen las variables auxiliares. En este caso se deciden utilizar una variable menos que el número de niveles:

```
#
d1$isla_Bis = ifelse(d$isla == "Biscoe", 1, 0)
d1$isla_Dre = ifelse(d$isla == "Dream", 1, 0)
#
d1$genero_H = ifelse(d$genero == "hembra", 1, 0)
```

- Se preparan los valores que se quieren predecir:

```
# variables cuantitativas normalizadas
xp1 = data.frame(
  long_pico = knn_normaliza(xp$long_pico, min(d$long_pico), max(d$long_pico)),
  prof_pico = knn_normaliza(xp$prof_pico, min(d$prof_pico), max(d$prof_pico)),
  long_aleta = knn_normaliza(xp$long_aleta, min(d$long_aleta), max(d$long_aleta)),
  peso = knn_normaliza(xp$peso, min(d$peso), max(d$peso))
)
# variables auxiliares
xp1$isla_Bis = ifelse(xp$isla == "Biscoe", 1, 0)
xp1$isla_Dre = ifelse(xp$isla == "Dream", 1, 0)
#
xp1$genero_H = ifelse(xp$genero == "hembra", 1, 0)
```

3 Algoritmo KNN con R

Para aplicar el algoritmo se va a utilizar el paquete FNN de R:

```
# se utiliza la funcion knn.reg del paquete FNN
yp = FNN::knn(d1, test = xp1, cl = d$especie, k = 3)
yp
```

```
## [1] Adelie Gentoo
## attr(,"nn.index")
##      [,1] [,2] [,3]
## [1,]   28   84   79
## [2,]  189  170  110
## attr(,"nn.dist")
##           [,1]      [,2]      [,3]
## [1,] 0.1000516 0.1180674 0.1201057
## [2,] 0.3456152 0.3673515 0.3703687
## Levels: Adelie Gentoo
```

- Se observa que se devuelve la predicción: Adelie, Gentoo
- attr(,"nn.index") devuelve los k puntos más cercanos.
- attr(,"nn.dist") devuelve la distancia de los k puntos más cercanos.

```
# la especie de los k puntos más cercanos es
(p1 = attr(yp,"nn.index")[1,])
```

```
## [1] 28 84 79
```

```
d$especie[p1]
```

```
## [1] Adelie Adelie Adelie
## Levels: Adelie Chinstrap Gentoo
```

Luego la predicción es Adelie.

3.1 Cálculo del error de predicción: matriz de confusión

```
# se crean los datos de entrenamiento y los datos test (80%-20%)
set.seed(123)
n = nrow(d)
pos_train = sample(1:n,round(0.8*n), replace = F)
train_x = d1[pos_train,]
test_x = d1[-pos_train,]
train_y = d$especie[pos_train]
test_y = d$especie[-pos_train]
```

```
# se utiliza la funcion knn.reg del paquete FNN
yp = FNN::knn(train_x, test = test_x, cl = train_y, k = 1)
```

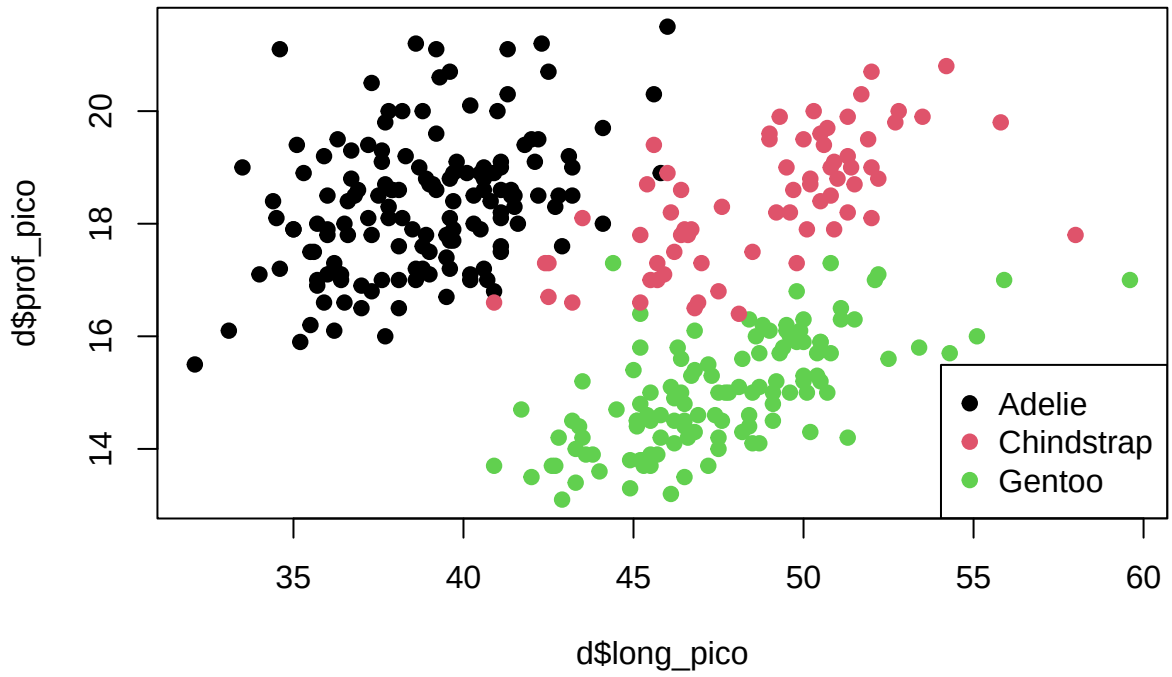
El método más sencillo para calcular el error de predicción es la **matriz de confusión**:

```
# matriz de confusion
(t = table(test_y, yp))
```

```
##            yp
## test_y      Adelie Chinstrap Gentoo
##   Adelie        28         0      0
##   Chinstrap      0        15      0
##   Gentoo         0         0     24
```

Los valores de la diagonal de la matriz están bien predichos, los de fuera de la diagonal son errores de predicción. En este caso la predicción es perfecta. Esto ocurre porque la longitud del pico y la profundidad del pico caracterizan muy bien la especie de los pingüinos:

```
plot(d$long_pico, d$prof_pico, col = d$especie, pch = 19)
legend("bottomright", legend = c("Adelie","Chindstrap","Gentoo"), pch = 19, col = 1:3)
```



Luego K = 1 es suficiente para realizar la predicción final:

```
# se utilizan todos los datos
(yp = FNN::knn(d1, test = xp1, cl = d$especie, k = 1))
```

```
## [1] Adelie Gentoo
## attr(,"nn.index")
##      [,1]
## [1,]   28
## [2,]  189
## attr(,"nn.dist")
##           [,1]
## [1,] 0.1000516
## [2,] 0.3456152
## Levels: Adelie Gentoo
```