KNN

SERGIO ADRIAN ORTIZ ORTEGA

2022-05-07

#K-vecinos proximos

##cargar el paquete y los datos

```
library(MASS)
Z<-as.data.frame(iris)
```

Definir la matriz de datos y la variable respuesta Con las clasificaciones

```
x<-Z[,1:4]
y<-Z[,5]
```

Se definen las variables y observaciones

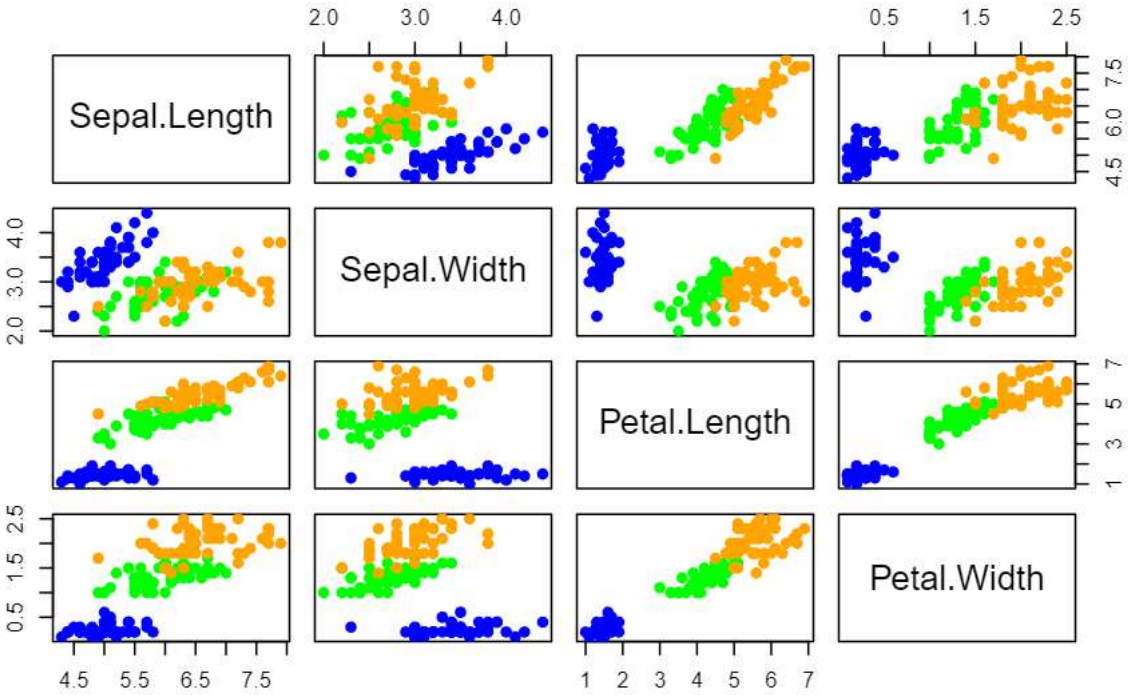
```
n<-nrow(x)
p<-ncol(x)
```

Grafico scatter plot Creacion de un vector de colores

```
col.iris<-c("blue","green","orange")[y]
col.iris
```

```
##   [1] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##   [9] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##  [17] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##  [25] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##  [33] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##  [41] "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"   "blue"
##  [49] "blue"   "blue"   "green"  "green"  "green"  "green"  "green"  "green"
##  [57] "green"  "green"  "green"  "green"  "green"  "green"  "green"  "green"
##  [65] "green"  "green"  "green"  "green"  "green"  "green"  "green"  "green"
##  [73] "green"  "green"  "green"  "green"  "green"  "green"  "green"  "green"
##  [81] "green"  "green"  "green"  "green"  "green"  "green"  "green"  "green"
##  [89] "green"  "green"  "green"  "green"  "green"  "green"  "green"  "green"
##  [97] "green"  "green"  "green"  "green"  "orange" "orange" "orange" "orange"
## [105] "orange" "orange" "orange" "orange" "orange" "orange" "orange" "orange"
## [113] "orange" "orange" "orange" "orange" "orange" "orange" "orange" "orange"
## [121] "orange" "orange" "orange" "orange" "orange" "orange" "orange" "orange"
## [129] "orange" "orange" "orange" "orange" "orange" "orange" "orange" "orange"
## [137] "orange" "orange" "orange" "orange" "orange" "orange" "orange" "orange"
## [145] "orange" "orange" "orange" "orange" "orange" "orange"
```

```
pairs(x, main="Data set Iris", pch=19,col=col.iris)
```



podemos ver la relaciones y como se comportan los datos entre ellos

#kNN ##cargar libreria

```
library(class)
```

Se fija una "semilla" para tener valores iguales

```
set.seed(1000)
```

##creacion de los ciclos para k=1 hasta k=20 Selecciona el valor de k que tenga el error mas bajo. InicializaciÃ³n de una lista vacia de tamaño 20

```
knn.class<-vector(mode="list",length=20)
knn.tables<-vector(mode="list", length=20)
```

Clasificaciones erroneas

```
knn.mis<-matrix(NA, nrow=20, ncol=1)
for(k in 1:20){
  knn.class[[k]]<-knn.cv(x,y,k=k)
  knn.tables[[k]]<-table(y,knn.class[[k]])
  # la suma de las clasificaciones menos las correctas
  knn.mis[k]<- n-sum(y==knn.class[[k]])
}
knn.mis
```

```
##       [,1]
##  [1,]    6
```

```
##  [2,]    7
##  [3,]    6
##  [4,]    6
##  [5,]    5
##  [6,]    4
##  [7,]    5
##  [8,]    5
##  [9,]    4
## [10,]    5
## [11,]    4
## [12,]    6
## [13,]    5
## [14,]    3
## [15,]    4
## [16,]    5
## [17,]    4
## [18,]    3
## [19,]    3
## [20,]    4
```

Numero optimo de k-vecinos

```
which(knn.mis==min(knn.mis))
```

```
## [1] 14 18 19
```

```
knn.tables[[10]]
```

```
##
## y            setosa versicolor virginica
##   setosa         50          0         0
##   versicolor      0         47         3
##   virginica       0          2        48
```

```
knn.tables[[18]]
```

```
##
## y            setosa versicolor virginica
##   setosa         50          0         0
##   versicolor      0         48         2
##   virginica       0          1        49
```

el mas eficiente es k=10 se señala el k mas eficiente

```
k.opt<-10
knn.cv.opt<-knn.class[[k.opt]]
knn.cv.opt
```

```
##   [1] setosa     setosa     setosa     setosa     setosa     setosa
##   [7] setosa     setosa     setosa     setosa     setosa     setosa
##  [13] setosa     setosa     setosa     setosa     setosa     setosa
##  [19] setosa     setosa     setosa     setosa     setosa     setosa
##  [25] setosa     setosa     setosa     setosa     setosa     setosa
##  [31] setosa     setosa     setosa     setosa     setosa     setosa
##  [37] setosa     setosa     setosa     setosa     setosa     setosa
##  [43] setosa     setosa     setosa     setosa     setosa     setosa
```

```
##  [49] setosa     setosa     versicolor versicolor versicolor versicolor
##  [55] versicolor versicolor versicolor versicolor versicolor versicolor
##  [61] versicolor versicolor versicolor versicolor versicolor versicolor
##  [67] versicolor versicolor versicolor versicolor virginica  versicolor
##  [73] versicolor versicolor versicolor versicolor versicolor virginica
##  [79] versicolor versicolor versicolor versicolor versicolor virginica
##  [85] versicolor versicolor versicolor versicolor versicolor versicolor
##  [91] versicolor versicolor versicolor versicolor versicolor versicolor
##  [97] versicolor versicolor versicolor versicolor virginica  virginica
## [103] virginica  virginica  virginica  virginica  versicolor virginica
## [109] virginica  virginica  virginica  virginica  virginica  virginica
## [115] virginica  virginica  virginica  virginica  virginica  virginica
## [121] virginica  virginica  virginica  virginica  virginica  virginica
## [127] virginica  virginica  virginica  virginica  virginica  virginica
## [133] virginica  virginica  virginica  virginica  virginica  virginica
## [139] versicolor virginica  virginica  virginica  virginica  virginica
## [145] virginica  virginica  virginica  virginica  virginica  virginica
## Levels: setosa versicolor virginica
```

##tabla de contingencia con las clasificaciones buenas y malas

```
knn.tables[[k.opt]]
```

```
##
## y            setosa versicolor virginica
##   setosa         50          0         0
##   versicolor      0         47         3
##   virginica       0          2        48
```

cantidad de observaciones mal clasificadas

```
knn.mis[k.opt]
```

```
## [1] 5
```

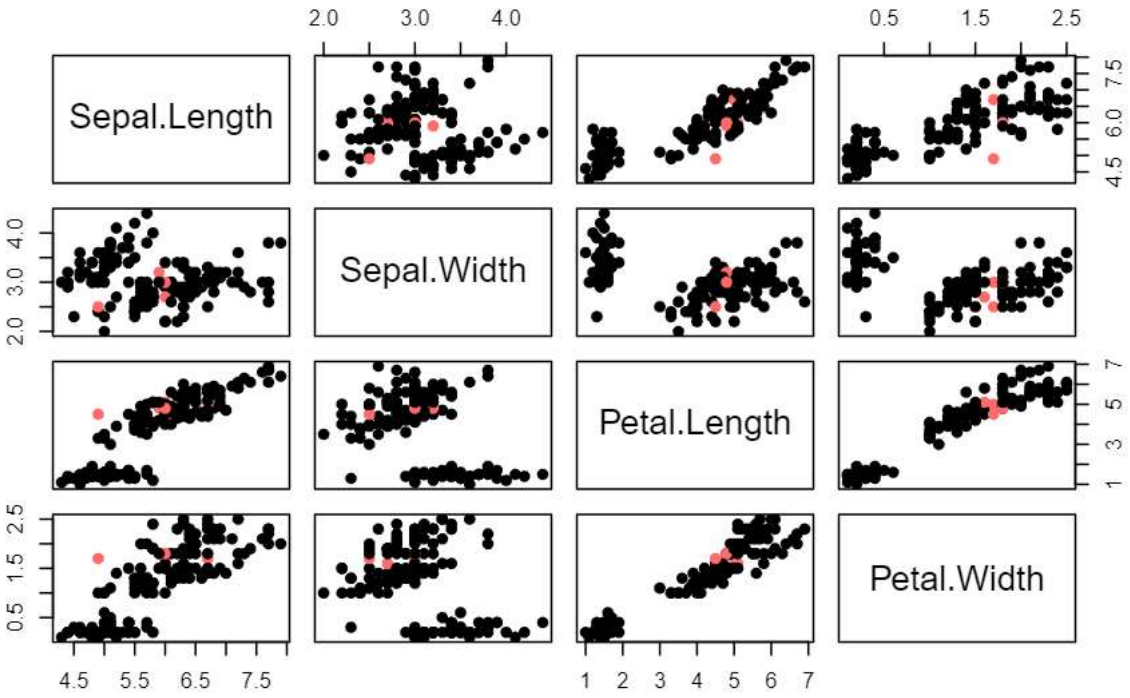
Error de clasificacion (MR)

```
knn.mis[k.opt]/n
```

```
## [1] 0.03333333
```

Grafico de clasificaciones correctas y erroneas

```
col.knn.iris<-c("indianred1","black")[1*(y==knn.cv.opt)+1]
pairs(x, main="Clasificacion kNN de Iris",
      pch=19, col=col.knn.iris)
```



#EJEMPLO DE PINGUINOS se hara lo mismo pero con la base de datos penguins y con la variable "especie"

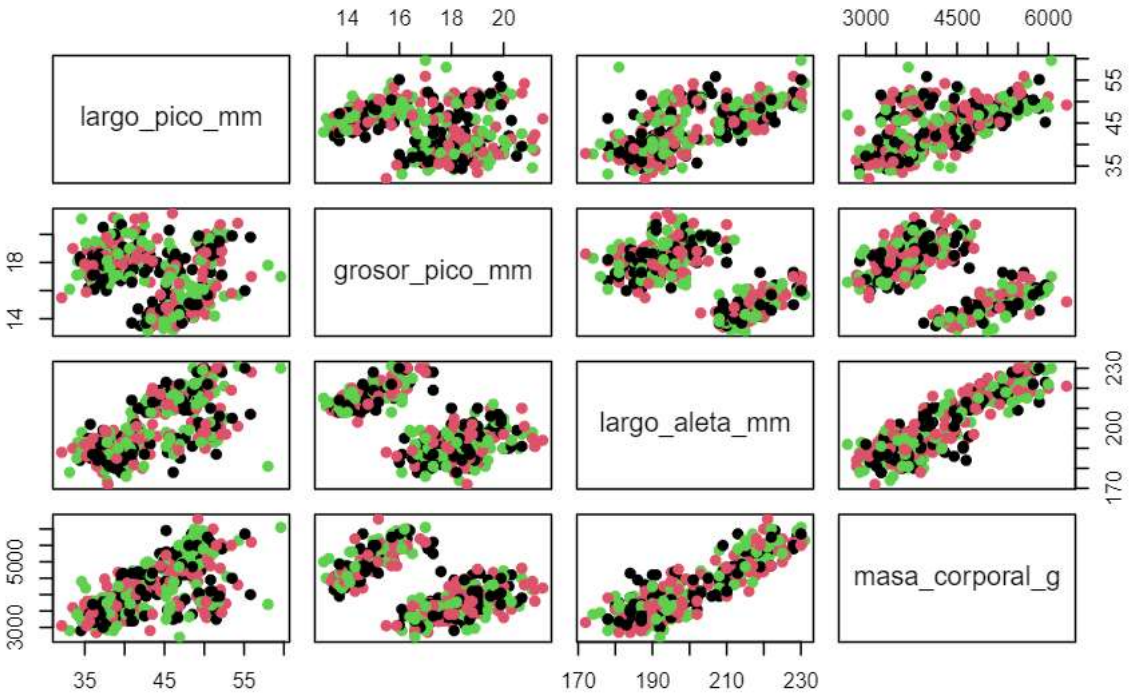
```
library(MASS)
library(readxl)
penguins <- read_excel("penguins.xlsx")
Z<-data.frame(penguins)
```

Definir la matriz de datos y la variable respuesta Con las clasificaciones

```
x<-Z[,2:5]
y<-Z[,1]
```

Se definen las variables y observaciones

```
n<-nrow(x)
p<-ncol(x)
pairs(x, main="Pinguinos", pch=19,col=1:3)
```



kNN

cargamos la libreria

```
library(class)
```

##Se fija una “semilla” para tener valores iguales

```
set.seed(1001)
```

##creacion de los ciclos para k=1 hasta k=20 Selecciona el valor de k que tenga el error mas bajo. InicializaciÃ³n de una lista vacia de tamaño 20

```
knn.class<-vector(mode="list",length=20)
knn.tables<-vector(mode="list", length=20)
```

##Clasificaciones erroneas

```
knn.mis<-matrix(NA, nrow=20, ncol=1)
for(k in 1:20){
  knn.class[[k]]<-knn.cv(x,y,k=k)
  knn.tables[[k]]<-table(y,knn.class[[k]])
  # la suma de las clasificaciones menos las correctas
  knn.mis[k]<- n-sum(y==knn.class[[k]])
}
knn.mis
```

```
##       [,1]
##  [1,]   44
##  [2,]   67
##  [3,]   70
##  [4,]   71
```

```
##  [5,]   71
##  [6,]   77
##  [7,]   78
##  [8,]   77
##  [9,]   75
## [10,]   74
## [11,]   71
## [12,]   72
## [13,]   74
## [14,]   74
## [15,]   80
## [16,]   85
## [17,]   88
## [18,]   88
## [19,]   83
## [20,]   83
```

Numero optimo de k-vecinos

```
which(knn.mis==min(knn.mis))
```

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

```
knn.tables[[10]]
```

```
##
## y           Adelie Chinstrap Gentoo
##   Adelie       136         5     11
##   Chinstrap     46        18      4
##   Gentoo         7         1    116
```

```
knn.tables[[18]]
```

```
##
## y           Adelie Chinstrap Gentoo
##   Adelie       136         3     13
##   Chinstrap     59         5      4
##   Gentoo         9         0    115
```

##el mas eficiente es k=10 se señala el k mas eficiente

```
k.opt<-10
knn.cv.opt<-knn.class[[k.opt]]
knn.cv.opt
```

```
##   [1] Adelie    Adelie    Chinstrap Adelie    Adelie    Chinstrap Adelie
##   [8] Gentoo    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [15] Gentoo    Adelie    Adelie    Gentoo    Adelie    Adelie    Adelie
##  [22] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [29] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [36] Adelie    Adelie    Adelie    Adelie    Gentoo    Adelie    Adelie
##  [43] Adelie    Gentoo    Adelie    Gentoo    Adelie    Adelie    Adelie
##  [50] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [57] Adelie    Adelie    Adelie    Adelie    Adelie    Gentoo    Adelie
##  [64] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [71] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [78] Adelie    Adelie    Adelie    Adelie    Gentoo    Adelie    Adelie
```

```
##  [85] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [92] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
##  [99] Adelie    Adelie    Adelie    Gentoo    Adelie    Adelie    Adelie
## [106] Adelie    Adelie    Adelie    Adelie    Gentoo    Adelie    Gentoo
## [113] Adelie    Adelie    Adelie    Chinstrap Adelie    Adelie    Adelie
## [120] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [127] Adelie    Adelie    Adelie    Chinstrap Adelie    Adelie    Adelie
## [134] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [141] Adelie    Adelie    Adelie    Adelie    Adelie    Chinstrap Adelie
## [148] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Gentoo
## [155] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [162] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Adelie    Gentoo
## [169] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [176] Gentoo    Gentoo    Gentoo    Chinstrap Gentoo    Gentoo    Gentoo
## [183] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [190] Gentoo    Gentoo    Gentoo    Adelie    Gentoo    Adelie    Gentoo
## [197] Gentoo    Gentoo    Adelie    Gentoo    Gentoo    Gentoo    Gentoo
## [204] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Adelie    Gentoo
## [211] Gentoo    Gentoo    Adelie    Gentoo    Gentoo    Gentoo    Gentoo
## [218] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [225] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [232] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [239] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [246] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [253] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [260] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [267] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [274] Gentoo    Gentoo    Gentoo    Adelie    Adelie    Chinstrap Adelie
## [281] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [288] Chinstrap Adelie    Chinstrap Adelie    Chinstrap Adelie    Adelie
## [295] Adelie    Gentoo    Adelie    Adelie    Adelie    Adelie    Adelie
## [302] Adelie    Adelie    Adelie    Adelie    Gentoo    Adelie    Adelie
## [309] Adelie    Chinstrap Adelie    Adelie    Chinstrap Gentoo    Adelie
## [316] Gentoo    Adelie    Chinstrap Adelie    Adelie    Chinstrap Adelie
## [323] Adelie    Adelie    Chinstrap Chinstrap Adelie    Adelie    Adelie
## [330] Chinstrap Adelie    Adelie    Adelie    Chinstrap Adelie    Adelie
## [337] Adelie    Chinstrap Chinstrap Chinstrap Adelie    Chinstrap Chinstrap
## [344] Chinstrap
## Levels: Adelie Chinstrap Gentoo
```

##tabla de contingencia con las clasificaciones buenas y malas

```
knn.tables[[k.opt]]
```

```
##
## y           Adelie Chinstrap Gentoo
##   Adelie       136         5     11
##   Chinstrap     46        18      4
##   Gentoo         7         1    116
```

##cantidad de observaciones mal clasificadas

```
knn.mis[k.opt]
```

```
## [1] 74
```

##Error de clasificacion (MR)

```
knn.mis[k.opt]/n
```

```
## [1] 0.2151163
```

##Grafico de clasificaciones correctas y erroneas

```
col.knn.iris<-c("indianred1","black")[1*(y==knn.cv.opt)+1]
pairs(x, main="Clasificacion KNN de pinguinos",
      pch=19, col=col.knn.iris)
```