PROYECTO FINAL

SERGIO ADRIAN ORTIZ ORTEGA

30-05-2022

CAPITULO 1 INTRODUCCION

Pokémon es una franquicia de medios que originalmente comenzó como un videojuego RPG, pero debido a su popularidad ha logrado expandirse a otros medios de entretenimiento como series de televisión, películas, juegos de cartas, ropa, entre otros, convirtiéndose en una marca que es reconocida en el mercado mundial.

Este conjunto de datos fue descargado por la página Kaggle y que incluye 721 Pokémon, incluido su número, nombre, primer y segundo tipo y estadísticas básicas: HP, Ataque, Defensa, Ataque especial, Defensa especial y Velocidad.

K-means es un algoritmo de clasificación no supervisada (clusterización) que agrupa objetos en k grupos basándose en sus características. El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o cluster. Se suele usar la distancia cuadrática.

Los algoritmos de clustering son considerados de aprendizaje no supervisado. Este tipo de algoritmos de aprendizaje no supervisado busca patrones en los datos sin tener una predicción específica como objetivo (no hay variable dependiente). En lugar de tener una salida, los datos solo tienen una entrada que serían las múltiples variables que describen los datos.

El objetivo principal de este trabajo es demostrar lo aprendido de la experiencia educativa Estadística Multivariada usando una base de datos que no se haya utilizado antes para un trabajo anterior y utilizando el método de agrupamiento de partición k-medias, se analizaron los datos.

CAPITULO 2 TRATAMIENTO DE LA MATRIZ

Ha esta base de datos se obtuvo en Kaggle y la base de datos fue revisada y modificada antes del análisis, cheque que no hubiera datos faltantes, las dimensiones son de 151 a 13, también vi que los tipos de variables de la base son de tipo carácter y numérico, elimine datos de manera manual a partir de la fila 151 para tener solo los de la primera generación de pokemons y también elimine las mega evoluciones.

CAPITULO 3 METODOLOGIA

Primero que se hizo para este trabajo fue el buscar una base de datos en Kaggle y descargar la base actual, después se le hizo un chequeo rápido a la base para ver si tenia datos faltantes y ver que variables tengo, ya con un chequeo rápido decidí eliminar datos a partir del Pokémon 151 para tener solo los de la primera generación y me quede con los datos de las estadísticas de los Pokémon junto con su nombre, tipo y si es legendario o no, luego pase la base de datos a R-cloud para empezar a analizar pude notar que las variables eran de tipos que no me funcionaria para el análisis así que primero las modifique para que sean de tipo factor y numérico, y hacer una nueva matriz de datos, estime los clouster para ver cual seria la mejor forma de agruparlos y ya con eso hacer las graficas de k-means y vi que se dividían muy bien los dos grupos y después de eso con el grafico de Silhouette pude notar que el segundo clúster es el que mejor comportamiento tenia ya que es el que mayor porcentaje tiene.

CAPITULO 4 RESULTADOS

Cargar las LIBRERIAS

```
library(tidyverse)
    library(cluster)
    library(factoextra)
    library(NbClust)
    library(readxl)
    library(dplyr)
    library(ggplot2)
    library(tools)
    library(MVN)
    library(cluster)
    library(carData)
    library(car)
    library(rJava)
    library(xlsx)
    library(plotly)
    library(fpc)
```

Reconocimiento de la matriz de datos

```
Pokemon <- read_csv("Pokemon.csv")
```

```
## Rows: 151 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Name, Type_1, Type_2
## dbl (9): Pokedex, Total, HP, Attack, Defense, Sp_Atk, Sp_Def, Speed, Generation
## lgl (1): Legendary
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Exploracion de matriz

```
dim(Pokemon)
```

```
## [1] 151  13
```

```
str(Pokemon)
```

```
## spec_tbl_df [151 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Pokedex   : num [1:151] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Name      : chr [1:151] "Bulbasaur" "Ivysaur" "Venusaur" "Charmander" ...
##  $ Type_1    : chr [1:151] "Grass" "Grass" "Grass" "Fire" ...
##  $ Type_2    : chr [1:151] "Poison" "Poison" "Poison" NA ...
##  $ Total     : num [1:151] 318 405 525 309 405 534 314 405 530 195 ...
##  $ HP        : num [1:151] 45 60 80 39 58 78 44 59 79 45 ...
##  $ Attack    : num [1:151] 49 62 82 52 64 84 48 63 83 30 ...
##  $ Defense   : num [1:151] 49 63 83 43 58 78 65 80 100 35 ...
##  $ Sp_Atk    : num [1:151] 65 80 100 60 80 109 50 65 85 20 ...
##  $ Sp_Def    : num [1:151] 65 80 100 50 65 85 64 80 105 20 ...
```

```
##  $ Speed     : num [1:151] 45 60 80 65 80 100 43 58 78 45 ...
##  $ Generation: num [1:151] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Legendary : logi [1:151] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Pokedex = col_double(),
##   ..   Name = col_character(),
##   ..   Type_1 = col_character(),
##   ..   Type_2 = col_character(),
##   ..   Total = col_double(),
##   ..   HP = col_double(),
##   ..   Attack = col_double(),
##   ..   Defense = col_double(),
##   ..   Sp_Atk = col_double(),
##   ..   Sp_Def = col_double(),
##   ..   Speed = col_double(),
##   ..   Generation = col_double(),
##   ..   Legendary = col_logical()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
colnames(Pokemon)
```

```
##  [1] "Pokedex"    "Name"       "Type_1"     "Type_2"     "Total"
##  [6] "HP"         "Attack"     "Defense"    "Sp_Atk"     "Sp_Def"
## [11] "Speed"      "Generation" "Legendary"
```

##cambiar los tipos de las variables

```
nombre<-factor(Pokemon$Name)
Tipo<-factor(Pokemon$Type_1,
             levels= c("Grass","Fire","Water","Bug","Normal","Poison","Electric",
                       "Ground","Fairy","Fighting","Flying","Psychic","Rock",
                       "Steel","Ice","Dragon","Dark","Ghost"))
Generacion<-factor(Pokemon$Generation)
legendario<-factor(Pokemon$Legendary)

HP<-as.numeric(Pokemon$HP,strict = TRUE)
Attack<-as.numeric(Pokemon$Attack,strict = TRUE)
Defense<-as.numeric(Pokemon$Defense,strict = TRUE)
Sp_Atk<-as.numeric(Pokemon$Sp_Atk,strict = TRUE)
Sp_Def<-as.numeric(Pokemon$Sp_Def,strict = TRUE)
Speed<-as.numeric(Pokemon$Speed,strict = TRUE)

X<-data.frame(nombre,HP,Attack,Defense,Sp_Atk,Sp_Def,Speed)
BASE=X
```

volver factor variable cualitativa

```
BASE$nombre=as.factor(BASE$nombre)
```

Volvemos "Estado" al marco de los datos

```
DATOS = data.frame(BASE,row.names=BASE$nombre)
```

Eliminacion de la variables

```
DATOS[, 1] <- NULL
```

Separacion de filas y columnas.

```
n<-dim(DATOS)[1]
p<-dim(DATOS)[2]
```

Estandarizacion univariante.

```
X.s<-scale(DATOS)
```

Escalar la base de datos

```
datos.scale = scale (DATOS)
```

Matrix de distancia

```
Mdistancia = get_dist(datos.scale,method = "manhattan")
```

Estimar el numero de cluster

```
#numCluster = NbClust(data=DATOS, method = "median", distance = "euclidean", diss=NULL,  min.nc=2, max.
```

- Among all indices:

- 14 proposed 2 as the best number of clusters
- 1 proposed 3 as the best number of clusters
- 2 proposed 4 as the best number of clusters
- 9 proposed 5 as the best number of clusters
- 1 proposed 6 as the best number of clusters

```
***** Conclusion *****
```

- According to the majority rule, the best number of clusters is 2

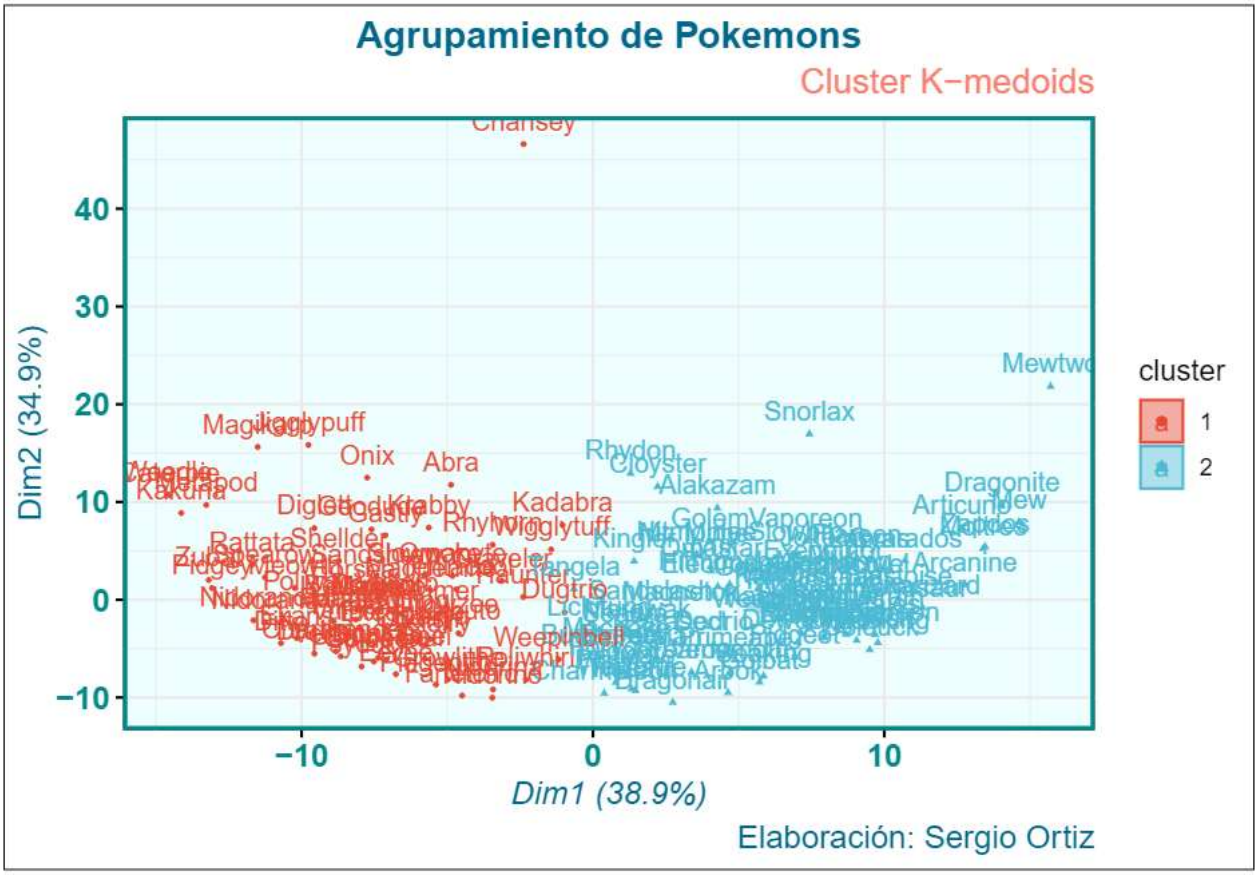
En este resultados me dicen que mi mejor numero de clouster seria 2 (este codigo es muy pesado y cuando intento hacer que el codigo corra en markdown R se cae incluso con rcloud no lo corre asi que pongo el resultado de ese codigo junto con una pequeña interpretasion)

Clusters K-medoids

```
set.seed(10)
#usando una semilla para que siempre den el mismo resultados
method_Cluster = eclust(Mdistancia,FUNcluster = "kmeans", k = 2, nstart = 25, graph = F)
```

Graficar clusters Kmeans

```
Kmedoids =   fviz_cluster(method_Cluster, data = Mdistancia,  main = "Agrupamiento de Pokemons",
     repel=F,star.plot=F,ellipse = T, ellipse.type="euclid" ,ellipse.level = 0.95,ellipse.alpha=.45,
     palette="npg",ggtheme = theme_minimal(),show.clust.cent=T,pointsize = .8,labelsize = 11,font.ticks
     font.x = c(12, "italic", "deepskyblue4"),font.y = c(12, "plain", "deepskyblue4"))+
theme(panel.background = element_rect(fill = "azure")) +
theme(plot.background = element_rect(fill = "white"))+
theme(panel.border = element_rect(colour = "darkcyan", fill=NA, size=1.5))+
     theme(plot.title = element_text(size= 14,  vjust=.75, color="deepskyblue4", lineheight=1,face="bol
     theme(plot.caption = element_text(size = 12,color = "deepskyblue4",hjust=1))+
     theme(plot.subtitle= element_text(size = 13,color = "salmon",hjust=1,face="plain"))+
labs(subtitle = "Cluster K-medoids", caption = "Elaboración: Sergio Ortiz")
Kmedoids
```



Gracias a la grafica podemos notar como se comportan los datos y se dividen muy bien.

3.- Algoritmo k-medias (2 grupos) cantidad de subconjuntos aleatorios que se escogen para realizar los calculos de algoritmo.

```
Kmeans.3<-kmeans(DATOS, 2, nstart=25)
```

Centroides

```
Kmeans.3$centers
```

```
##          HP   Attack  Defense   Sp_Atk   Sp_Def    Speed
## 1 76.48276 85.54023 76.50575 81.35632 80.40230 80.13793
## 2 47.53125 54.89062 56.96875 47.81250 46.46875 53.70312
```

Cluster de Pertenencia

Kmeans.3$cluster

Bulbasaur 2 Ivysaur 1 Venusaur 1 Charmander 2
Charmeleon 1 Charizard 1 Squirtle 2 Wartortle 1 Blastoise 1 Caterpie 2

SCDG

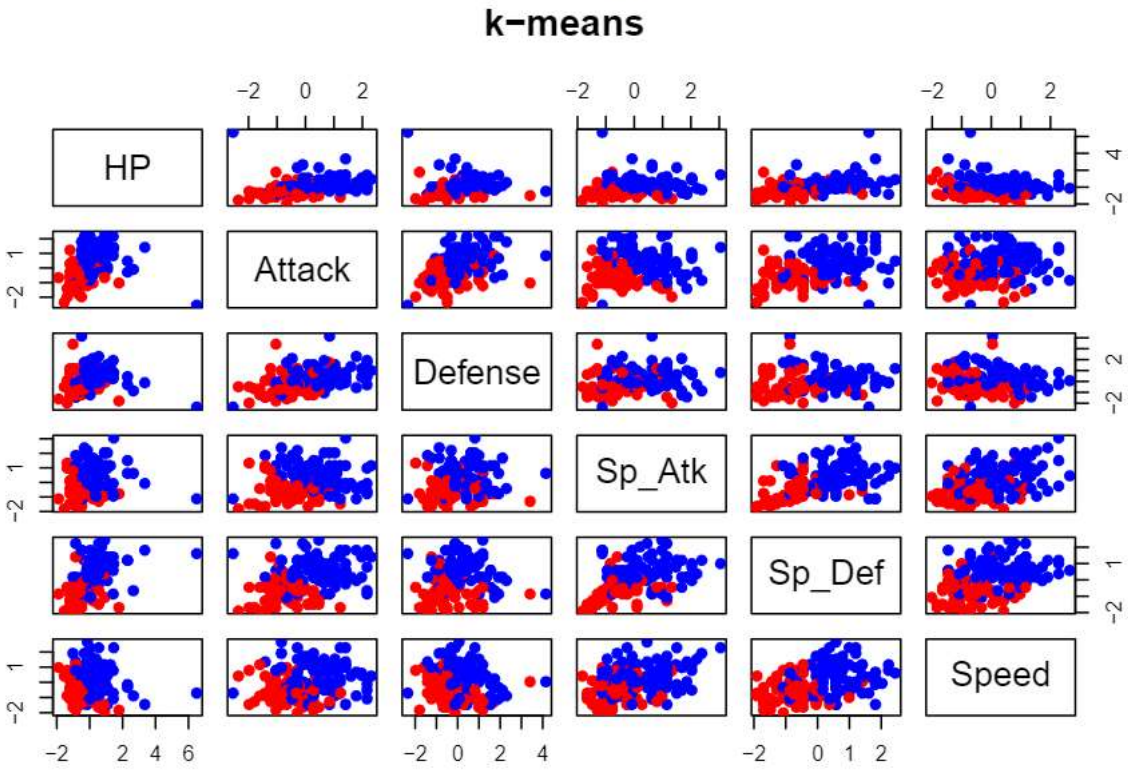
```
SCDG<-sum(Kmeans.3$withinss)
```

Clusters

```
cl.kmeans<-Kmeans.3$cluster
```

Scatter plot con la division de grupos obtenidos (se utiliza la matriz de datos centrados).

```
col.cluster<-c("blue", "red")[cl.kmeans]
pairs(X.s, col=col.cluster, main="k-means", pch=19)
```



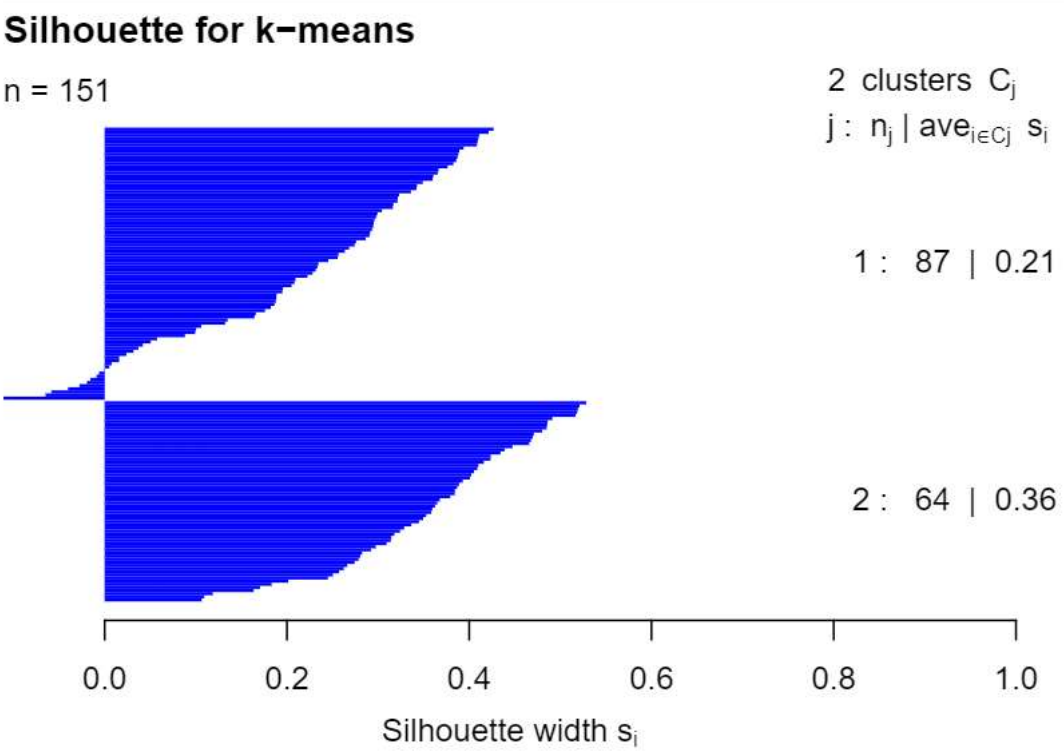
En esta grafica podemos ver el comportamiento de los datos y se ven que se comportan bien.

Silhouette

Representacion grafica de la eficacia de clasificacion de una observacion dentro de un grupo.

Generacion de los calculos y Generacion del grafico

```
dist.Euc<-dist(X.s, method = "euclidean")
Sil.kmeans<-silhouette(cl.kmeans, dist.Euc)
plot(Sil.kmeans, main="Silhouette for k-means",
col="blue")
```



Average silhouette width : 0.28

CAPITULO 5 CONCLUSION

Con los resultados que obtuvimos podemos decir que la cantidad de clusters para una mejor k-means para esta base de datos de pokedex es de 2 clusters y en Silhouette podemos ver que no es muy grande los porcentajes, pero el segundo cluster es mas alto de los dos, los datos se amoldan muy bien con dos clouster y en las graficas podemos notar que se separan por pokemons debiles y pokemons fuertes.

REFERENCIAS

es, Q. (2016). ¿Qué es Pokémon? Descubre el fenómeno Pokémon en este artículo dedicado. Nintendo of Europe GmbH. https://www.nintendo.es/Noticias/2016/agosto/-Que-es-Pokemon-Descubre-el-fenomeno-Pokemon-en-este-articulo-dedicado-1128960.html

de, C. (2003, October 18). franquicia de medios japonesa. Wikipedia.org; Wikimedia Foundation, Inc. https://es.wikipedia.org/wiki/Pok%C3%A9mon

Barradas, A. (2016). Pokemon with stats. Kaggle.com. https://www.kaggle.com/datasets/abcsds/pokemon

RPubs - Introducción a los Modelos de Agrupamiento en R. (2018, June 23). Rpubs.com. https://rpubs.com/rdelgado/399475#:~:text=modelos%20de%20agrupamiento.-,Agrupamiento%20por%20K%2DMedios%20(K%2DMeans%20Clust

Duk2. (2019, January 8). K-Means: Agrupamiento con Minería de datos [Introducción]. Retrieved June 4, 2022, from ESTRATEGIAS DE TRADING website: https://estrategiastrading.com/k-means/

kmeans. (2022). Retrieved June 4, 2022, from Unioviedo.es website: https://www.unioviedo.es/compnum/laboratorios_py/kmeans/kmeans.html

Juan Gabriel Gomila. (2018). 28 - La técnica de k-means en RStudio [YouTube Video]. In YouTube. https://www.youtube.com/watch?v=b-LtNvGXcLo&ab_channel=JuanGabrielGomila