INTRODUCCION AL ANALISIS MULTIVARIADO

ANALISIS DE AGRUPAMIENTOS (CLUSTER)

PRECALCULO

- Se analizan algunas variables que pueden estar relacionadas con la percepción hacia el curso de Precálculo en los estudiantes matriculados en las carreras de la Facultad de Ciencias Económicas de la Universidad de Costa Rica en el primer ciclo del 2019.
- La muestra está conformada por 107 estudiantes inscritos en alguna de las carreras de la Facultad de Ciencias Económicas (Estadística, Economía, Dirección de Empresas, Administración Aduanera y Contaduría Pública).
- Los estudiantes completaron un cuestionario en Google Forms.
- El cuestionario utilizado para la recolección de datos constaba de dos secciones: la primera incluía características sociodemográficas (edad, sexo, zona) y otras variables sobre el colegio de procedencia y experiencias educativas previas, y la segunda sección constaba de preguntas acerca de la percepción del curso de Precálculo, el tiempo de estudio que los alumnos le dedican a éste, y, por último, la influencia de las charlas del CASE en la motivación de los mismos.
- Las variables que se utilizaron para medir la percepción hacia el curso de precálculo fueron: 1) expectativa del curso, 2) evaluación (quices y exámenes), 3) comprensión de la explicación del profesor, 4) utilidad del material brindado en clase, 5) motivación por las charlas del CASE.
- Estas variables se midieron con una escala que varía de 1 a 10, donde en general 1 es muy poco y 10 es mucho.

METODOS JERARQUICOS

1. Considere solamente las variables de percepción, haga una base llamada base1 con solo esas variables)

- Calcule la distancia euclídea entre los primeros dos individuos usando la forma vectorial para el cálculo de esta distancia.
- Obtenga la distancia euclídea entre todos los pares de individuos usando: d = dist(base1)
- Extraiga la distancia entre el individuo 1 y el individuo 2 de la matriz de distancias. Para hacer esto primero convierta d en matriz y luego llame la posición [1,2].

2. Obtenga la matriz de covarianzas S.

- Calcule la distancia de Mahalanobis entre los primeros dos individuos usando la forma vectorial para el cálculo de esta distancia.
- Obtenga la matriz completa de distancias de Mahalanobis. Para obtener la distancia de Mahalanobis de R se usa la librería `biotools` . Hay que darle la matriz de covariancias S como un argumento de la función `D2.dist` : `dM=D2.dist(base1,S)`

- Extraiga la distancia entre el individuo 1 y el individuo 2 de la matriz de distancias de Mahalanobis.

3. Obtenga la distancia de Manhattan entre los primeros dos individuos usando la forma vectorial.

- Obtenga la matriz de distancias de Manhattan. En la función `dist` agregue `method = “manhattan”` .

4. Obtenga el dendograma usando: como distancia entre individuos, la distancia Euclídea y como distancia entre grupos, el vecino más cercano (single). Primero se hace el agrupamiento y se pone en un objeto llamado cs con la función `hclust` :

```
cs = hclust(d,method = "single")
```

- Luego se hace el dendograma con la función plot:

```
plot(cs,xlab="",main="Vecino más cercano",ylab="Distancia",sub="")
```

- Use ahora las otras distancias entre individuos pero siempre use el vecino más cercano.

5. Ahora cambie de distancia entre grupos, use el vecino más lejano (indique `method=”complete”` en `hclust` que es el default). Obtenga los dendogramas con las diferentes distancias entre individuos.

6. Use el salto promedio como distancia entre grupos (indique `method=”average”` en `hclust`). Obtenga los dendogramas con las diferentes distancias entre individuos.

7. Use la distancia de Ward para medir la distancia entre grupos (indique `method=”ward.D”`). Obtenga los dendogramas con las diferentes distancias entre individuos.

8. Escoja una distancia entre individuos y uno de los métodos de distancia entre grupos que haga un agrupamiento conveniente. Además de la forma del dendograma, use la correlación “cophenetic” para ver qué tanto el dendograma reproduce las distancias. Por ejemplo, `cor(d,cophenetic(cs))` , donde d es el objeto de distancias que se usó para el agrupamiento jerárquico obtenido en cs.

ALGORITMOS K-MEDIAS Y K-MEDOIDS

9. Ahora se va a aplicar el método de k-medias pero antes observe las variancias de las variables de percepción. Compare las variancias y decida si es necesario estandarizar las variables antes de hacer el análisis de conglomerados.

10. Escriba una función que calcule la SCDG a partir de una base de variables y una variable de grupos. La función debe empezar así:

```
scdg=function(base,grupo){
  k = length(unique(grupo))
  sc=0
  for(i in 1:k){
    mat = seleccione solo los datos del grupo k-ésimo
    centroide =  calcular el centroide de mat
    dif = dif + suma de las distancias euclídeas al cuadrado entre los elementos de mat y el cen
  troide
    }
  }
  return(sc)
}
```

- Inicie con una asignación en 3 grupos de forma aleatoria.

```
set.seed(10)
grupo=sample(1:3,20,replace=T)
```

- Aplique la función a la base para obtener la SCDG asociada a ese agrupamiento.
- Cambie el agrupamiento usando `set.seed(20)` y calcule nuevamente la SCDG.

11. Haga una base llamada “datos” que tenga solo las dos primeras variables y use la asignación de grupos obtenida con `set.seed(15)` :

- Calcule la SCDG
- Obtenga los centroides.

```
centros=matrix(nrow=3,ncol=k)
  for(j in 1:k){
    centros[,j]=tapply(datos[,j],grupo,mean)
  }
```

- Haga un gráfico con 3 colores para diferenciar los grupos y agregue los centroides.
- Calcule la distancia de cada punto a cada uno de los 3 centroides y redefina los grupos.

```
grupo1=c()
  for(i in 1:20){
    disti=c()
    for(j in 1:3){
      disti[j]=dist(rbind(centros[j,],datos[i,]) )
    }
    grupo1[i]=which(disti==min(disti))
  }
```

- Compare los grupos (grupo1 vs grupo).
- Si alguno de los puntos se mueve, repita el procedimiento, de lo contrario termine.
- Muestre la secuencia de gráficos y también muestre un gráfico de la evolución de la SCDG.

12. Continúe con todas las variables en base1. Obtenga las SCDG usando los siguientes pasos:

```
SCDG1 = c()
for (k in 1:6) SCDG1[k] = kmeans(base1,centers = k)$tot.withinss
```

13. Grafique los valores del SCDG contra el número de grupos. Use `type=”b”` para ver la forma de codo:

```
plot(1:6, SCDG1, type = "b", xlab = "Número de grupos",ylab = "SCDG")
```

- Observe cuántos grupos se recomiedan a partir de este gráfico.

- Obtenga el gráfico automático usando la función `fviz_nbclust` de la librería `factoextra` . Indique `method = "wss"` para que use el criterio de suma de cuadrados dentro de grupos, de la siguiente forma:

```
fviz_nbclust(base1, kmeans, method = "wss", diss = d)
```

14. Obtenga el agrupamiento con k-medias usando 3 grupos.

```
km=kmeans(datos,centers = 3)
```

15. Use la función `pam` de la librería `cluster` para obtener un agrupamiento con 3 clusters usando el método de k-medoides.

```
kmedoids=pam(base1,3)
```

- Use la variante de k-medoides para grandes conjuntos de datos. Como esta base no es tan grande se hará con 2 muestras, pero si la base fuera suficientemente grande se acostumbra a hacerlo con 50 muestras o más. Use la función `clara` de la librería `cluster` , donde se indica en `k` el número de grupos, en `metric` el tipo de distancia entre individuos, por ejemplo, `metric = "manhattan"` , también tiene la posibilidad de estandarizar los datos dentro de la función haciendo `stand=T` , se indica el número de muestras en `samples` , además debe indicarse `pamLike=T` para que use el algoritmo PAM.

16. Obtenga el agrupamiento por k-medias a partir de los centroides de los grupos creados por un método jerárquico. Use la función `hkmeans` de la librería `factoextra` , donde indique en `hc.metric` la distancia entre individuos, en `hc.method` el método aglomerativa y en `k` el número de grupos. Hágalo con la distancia euclídea y método Ward.

CREACION DE LOS GRUPOS O CLUSTERS

17. Obtenga una variable categórica con 3 grupos que indique el clúster al que se ha asignado a cada individuo. Hágalo para los agrupamientos usando siempre la distancia Euclídea entre individuos, y usando los tres tipos de distancia entre grupos (single, complete, average). El corte se hace de la siguiente manera:

```
clust.single = cutree(cs, k=3)
```

- Haga lo mismo para `clust.complete` , `clust.av` y `clust.ward`

18. Obtenga la variable categórica con 3 grupos para el método de k-medias. En este caso el corte se hace:

```
clust.km=km$cluster
```

- Obtenga los grupos que se obtienen por k-medoides, por clara y por k-medias jerárquico.

19. Compare los clusters obtenidos con los diferentes métodos haciendo tablas cruzadas. Identifique individuos que se mueven de un cluster a otro cuando se usan diferentes métodos.

- Use la correlación entre correlogramas con las funciones `dendlist` y `cor.dendlist` de la librería `dendextend` . Primero tiene que convertir los resultados del agrupamiento jerárquico en un objeto tipo dendograma con la función `as.dendogram` y ponerlos en una lista con `dendlist` . Luego esta lista de

dendogramas la pone en la función `cor.dendlist` , indicando `method="cophenetic"` .

- Obtenga la SCDG de los resultados por k-medias, k-medoides, clara y k-medias jerárquico.

CARACTERIZACION DE LOS GRUPOS

20. Se pueden caracterizar los grupos usando las variables originales. Compare los grupos obtenidos con alguno de los métodos. Use boxplots de las variables de percepción.

21. También se pueden usar componentes principales si estos explican un alto porcentaje de variabilidad. Haga una visualización de las agrupaciones resultantes de k-medias. Use la función `fviz_cluster` de la librería `factoextra` de la siguiente forma: También se pueden usar componentes principales si estos explican un alto porcentaje de variabilidad. Obtenga los puntajes de los primeros dos componentes principales basados en la matriz de correlaciones. Grafique estos puntajes y ponga colores según clúster obtenido con el vecino más cercano (por ejemplo) y etiquete los individuos según orden en la base1.

```
fviz_cluster(km, base1, show.clust.cent = TRUE,
             ellipse.type = "euclid", repel = TRUE) +
  labs(title = "Resultados K-medias") +
  theme_bw() +
  theme(legend.position = "none")
```

- Haga el gráfico usando 4 o 5 clusters con k-medias.
- Haga un gráfico de componentes principales para visualizar el agrupamiento por k-medoides.
- Para hacer el gráfico de componentes principales cuando se ha realizado el agrupamiento con un método jerárquico, debe usarse previamente la función `eclust` , donse se indica la distancia entre individuos en `hc_metric` , la distancia entre grupos en `hc_method` , el número de grupos en `k` . El resultado se usa luego para hacer el gráfico en `fviz_cluster` . Por ejemplo,

```
c.w2 = eclust(base1, k=3,"hclust", hc_metric = "euclidean",hc_method = "ward.D",nboot = 2)
```

22. Compare los grupos con otras variables que no se usaron en la creación de los grupos. Por ejemplo, compare los grupos según el sexo, la zona, la edad del estudiante, el gusto hacia la matemática, el tipo de colegio, el gusto por la carrera de ciencias económicas en la que se encuentra matriculado, así como la nota de admisión a la UCR.

METODOS DIFUSOS

23. La función `fanny` (fuzzy analysis) de la librería `cluster` permite aplicar el algoritmo c-means clustering (FCM). El parámetro `diss=T` se usa cuando se trabaja a partir de una matriz de distancias, de lo contrario se usa `diss=F` cuando se trabaja con los datos originales. Se debe indicar en `metric` la distancia entre individuos, en `k` el número de grupos y en `stand` si se quiere estandarizar los datos. Aplique este método con la distancia euclídea para 3 grupos.

- Observe que da un mensaje de advertencia y está relacionado a que el algoritmo no logra diferenciar muy claramente la pertenencia a cada grupo. Observe la pertenencia con `fcm$membership` , conde `fcm` es el nombre que le dimos al objeto resultante del agrupamiento.
- Para remediar este problema intente usando otra distancia como Manhattan.

- Ahora no da el problema anterior, pero indica que el algoritmo no converge. Aumente el número de iteraciones hasta que converja con `maxit`.
- Observe la pertenencia a los grupos.
- Note que hay individuos que tienen probabilidades de pertenencia muy similares en varios grupos. Esto es una indicación de que hay mucho puntos frontera (difusos). Para medir el nivel de difusión, calcule el índice de Dunn, valores normalizados próximos a 0 indican que la estructura tiene un alto nivel difuso y valores próximos a 1 lo contrario. Para esto use `fcm$coef`.
- A partir de la matriz de pertenencia, ´diseñe una forma de automatizar la búsqueda de individuos cuyas dos probabilidades de pertenencia no difieren más de una cierta cantidad, por ejemplo, 10%. Esto le permitirá encontrar aquellos individuos que están en la frontera.
- Obtenga los grupos que da este método y compárelos con los obtenidos con k-medias.
- Si no hay diferencias en los agrupamientos, qué ventaja tiene haber hecho el método difuso?

NUMERO DE GRUPOS

24. Evalúe el número de grupos usando el método de la silueta para k-medias. En el punto 13. se usó la función `fviz_nbclust` de la librería `factoextra`, con `method = "wss"`. Cambie el método ahora por `method = "silhouette"`.

- En el caso de clustering jerárquico, es necesario cortar el árbol para cada uno de los valores de k antes de calcular los coeficientes de silueta. Se corta el árbol con un número determinado de k y se aplica a esta poda la función `silhouette`, cuyo resultado se guara en un objeto llamado `s`, y la silueta promedio se obtiene con `summary(s)[[4]]`. Obtenga la silueta promedio para valores de k en el rango de 2 a 15 y haga un gráfico con los resultados.

25. Obtenga el estadístico GAP con la función `fviz_nbclust` pero cambie el método ahora por `method = "gap"`.

26. Use la función `NbClust` de la librería del mismo nombre para encontrar los diferentes indicadores que existen para determinar el número óptimo de clusters. Indique: `min.nc = 2`, `max.nc = 10`, `method = "kmeans"`, `index = "alllong"`. Haga el gráfico de los resultados con la función `fviz_nbclust`.

VALIDACION

27. El uso combinado de las funciones `eclust` y `fviz_silhouette` de la librería `factoextra` permiten obtener los coeficientes de silueta de forma sencilla. La función `eclust`, gracias a su argumento `FUNcluster`, facilita el uso de múltiples algoritmos de clustering mediante una misma función (internamente llama a las funciones kmeans, hclust, pam, clara, etc). Use las siguientes instrucciones:

```
km1 = eclust(base1, FUNcluster = "kmeans", k = 3, hc_metric = "euclidean", nstart = 50, graph =
   F)
fviz_silhouette(sil.obj = km1, print.summary = TRUE)
```

28. Use la función `cluster.stats` de la librería `fpc` para calcular el índice de Dunn. Indique el objeto de distancias y la variable de agrupamiento obtenida con alguno de los métodos, por ejemplo, k-medias. Use `$dunn` para extrae el índice de Dunn.

MAPA DE CALOR

29. Use la función `heatmap` para hacer un mapa de calor. Use la siguiente instrucción:

```
heatmap(base1, scale = "none",
        distfun = function(x){dist(x, method = "euclidean")},
        hclustfun = function(x){hclust(x, method = "ward.D")},
        cexRow = 0.7)
```