

goh

90

UNIVERSIDAD RAFAEL LANDÍVAR

FACULTAD DE INGENIERÍA

ANÁLISIS DE DATOS

SECCIÓN 1 VESPERTINA

MGTR. JAIRO OMAR SALAZAR CHÁVEZ

EXAMEN PARCIAL

2

Julio Anthony Engels Ruiz Coto 1284719

GUATEMALA DE LA ASUNCIÓN, ABRIL 26 DE 2023

CAMPUS CENTRAL

Serie 1 (30 puntos)

1. Desarrolle con sus palabras, ¿cuál es la diferencia entre aprendizaje supervisado y el no supervisado?

R// La diferencia es que el aprendizaje supervisado se orienta en predecir una variable especial que tiene relación con datos descriptivos, caso contrario el no supervisado se orienta en buscar información, estructuras o patrones en los datos en este aprendizaje no existe una variable especial que haya que predecir.

2. ¿En qué se basa el algoritmo de kmeans para determinar a qué clúster debe de pertenecer cada una de las observaciones?

R// El algoritmo es de aprendizaje no supervisado este se basa en elegir de manera aleatoria las coordenadas de los centroides (k) del conjunto de datos, los centroides son los puntos que marcan el centro de cada agrupación de datos, cada punto se agrupa con el centroide más cercano para eso se utiliza una medida de distancia llamada distancia euclidiana.

3. Desarrolle con sus palabras, ¿Qué acciones se pueden tomar si tengo datos incompletos en un set de datos?

R// Las acciones que se pueden tomar son determinar si la cantidad de datos faltante es mayor al 80% entonces se procede a eliminar los datos de dicha variable caso contrario, se rellenan los datos faltantes con el valor promedio del dato anterior y el dato posterior, otra alternativa rellenar con el valor promedio de los datos existentes,

4. Si tuviera un set de datos con variables categóricas, ¿qué acción tomaría para poder utilizar estos datos en el entrenamiento?

R// La acción que tomaría para este tipo de caso es, convertir esas variables categóricas en un formato numérico en la que pueda ser entendido por el modelo, una de las técnicas que se pueden aplicar es el one hot encoding este crea una nueva columna binaria para cada categoría donde 1 dato presente y 0 dato ausente. Así también el label encoding consiste en asignar a dichas categorías un valor numérico, es una manera simple, pero si no se tiene un orden puede causar problemas ya que el modelo puede entender una relación ordinal entre las categorías que no existe.

5. ¿por qué es importante “normalizar” las características numéricas para efectuar un entrenamiento?

R//Es recomendable hacerlo cuando los datos no tienen la misma escala, la normalización limita los valores entre $[0$ y $1]$ además el método de K-means es altamente sensible a los outliers.

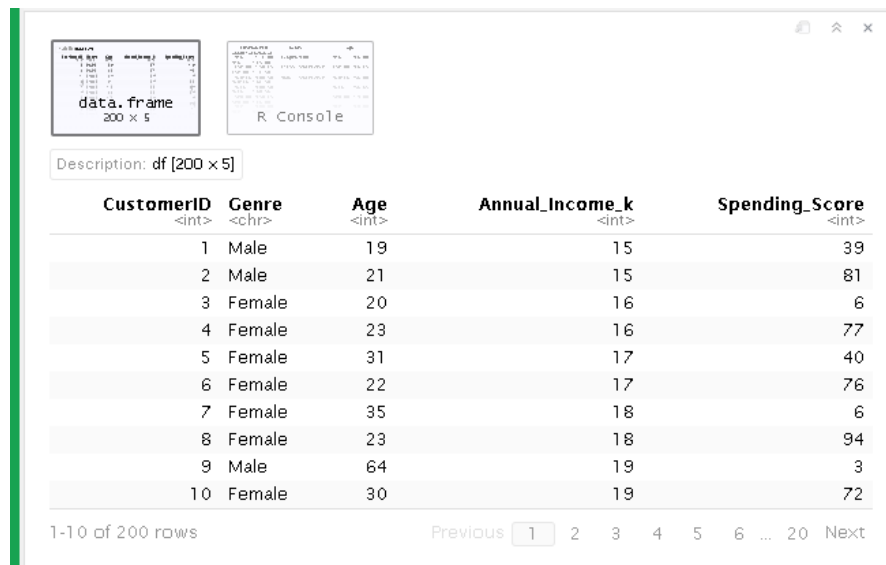
Serie 2 (70 puntos) En la siguiente serie deberá utilizar las fuentes de datos indicadas para analizar la información usando R.

Como primer paso importo el dataset a R y librerías a utilizar.

```

8 library(useful)
9 library(ggplot2)
10 library(dplyr)
11
12
16 dataset <- read.csv("Mall_Customers.csv")
17

```



Description: df [200 x 5]

CustomerID	Genre	Age	Annual_Income_k	Spending_Score
1	Male	19	15	39
2	Male	21	15	81
3	Female	20	16	6
4	Female	23	16	77
5	Female	31	17	40
6	Female	22	17	76
7	Female	35	18	6
8	Female	23	18	94
9	Male	64	19	3
10	Female	30	19	72

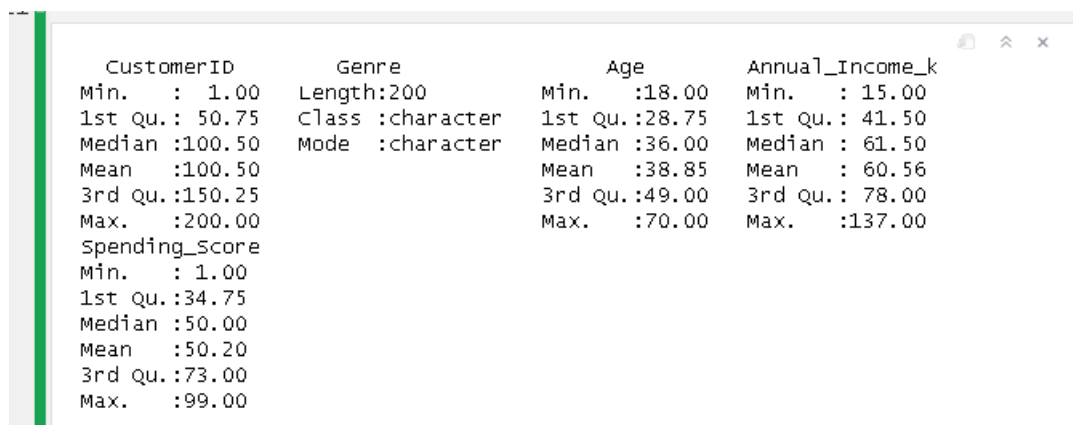
1-10 of 200 rows Previous 1 2 3 4 5 6 ... 20 Next

Luego genero la estadística general de los datos usando la función summary.

```

19 summary(dataset)
20

```



```

CustomerID      Genre      Age      Annual_Income_k
Min.   : 1.00   Length:200   Min.   :18.00   Min.   : 15.00
1st Qu.: 50.75   Class :character 1st Qu.:28.75   1st Qu.: 41.50
Median :100.50   Mode  :character Median :36.00   Median : 61.50
Mean   :100.50                Mean  :38.85   Mean   : 60.56
3rd Qu.:150.25                3rd Qu.:49.00   3rd Qu.: 78.00
Max.   :200.00                Max.   :70.00   Max.   :137.00

Spending_Score
Min.   : 1.00
1st Qu.:34.75
Median :50.00
Mean   :50.20
3rd Qu.:73.00
Max.   :99.00

```

Acá elimino la columna CustomerID porque es un identificador único para cada persona y no aporta una información útil para el agrupamiento, ya que en el clustering buscamos agrupar datos con características similares y no se relaciona con las otras características del conjunto de datos.

```
24 datasetmod <- subset(dataset, select = -CustomerID)
25
```

Luego aplico una codificación binaria para la columna Genre por que como tiene dos categorías Male, Female y se puede convertir en una variable binaria asignando 1 como Male y 0 para Female con esto facilito el análisis y el procedimiento de los datos por parte del algoritmo de aprendizaje, en este paso se usa la librería dplyr para el uso del if_else.

```
27 datasetmod$Genre <- if_else(datasetmod$Genre == "Male", 1, 0)
28 datasetmod
29
```

Description: df [200 x 4]

	Genre <dbl>	Age <int>	Annual_Income_k <int>	Spending_Score <int>
1	1	19	15	39
2	1	21	15	81
3	0	20	16	6
4	0	23	16	77
5	0	31	17	40
6	0	22	17	76
7	0	35	18	6
8	0	23	18	94
9	1	64	19	3
10	0	30	19	72

1-10 of 200 rows

Luego hago una verificación rápida si hay valores nulos, en este caso no presenta.

```
33 sum(is.na(datasetmod))
34
```

```
[1] 0
```

Ya con esto empiezo a entrenar el algoritmo de K-means.

```

38 datatrain <- kmeans(datasetmod, centers = 5)
39 datatrain
40
41 ...

```

K-means clustering with 5 clusters of sizes 28, 69, 53, 10, 40

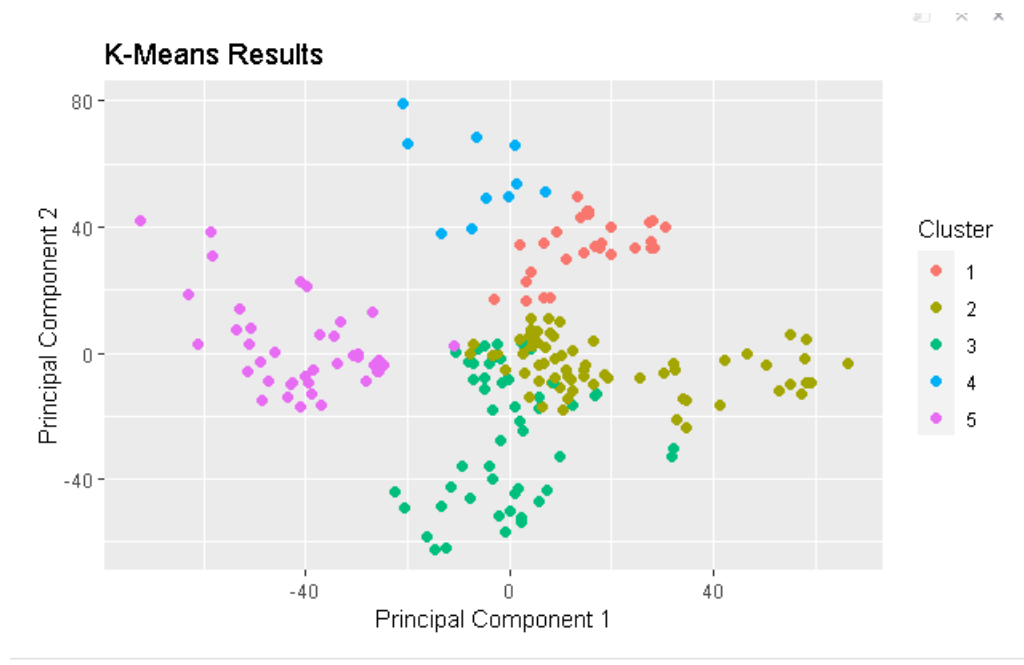
Cluster means:

	Genre	Age	Annual_Income_k	Spending_Score
1	0.6071429	40.17857	78.89286	17.42857
2	0.4057971	52.05797	46.42029	39.88406
3	0.4150943	25.05660	40.73585	62.62264
4	0.3000000	41.00000	109.70000	22.00000
5	0.4500000	32.87500	86.10000	81.52500

Clustering vector:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
3	3	2	3	3	3	2	3	2	3	2	3	2	3	2	3	2	3	
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	
2	3	2	3	2	3	2	3	2	3	2	3	3	3	2	3	3	2	
55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	
2	2	2	2	3	2	2	3	2	2	2	3	2	2	3	2	3	2	
73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	
2	2	2	3	2	2	3	2	2	3	2	2	3	2	2	3	3	2	
91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	
2	3	2	2	2	3	2	3	2	3	3	2	2	3	2	3	2	2	
109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	

Realizo el diagrama con centros y se ve la distribución de los clúster.



Luego aplico la regla del codo, el número de clúster con el error.

```
50 df <- data.frame(matrix(ncol = 2, nrow = 0))
51 colnames(df) <- c("K", "error")
52 for (i in 2:25) {
53   datatrain <- kmeans(x = datasetmod, centers = i)
54   df[i-1,] <- c(i, datatrain$tot.withinss)
55 }
56 df
57
58 ```
```

Description: df [24 × 2]

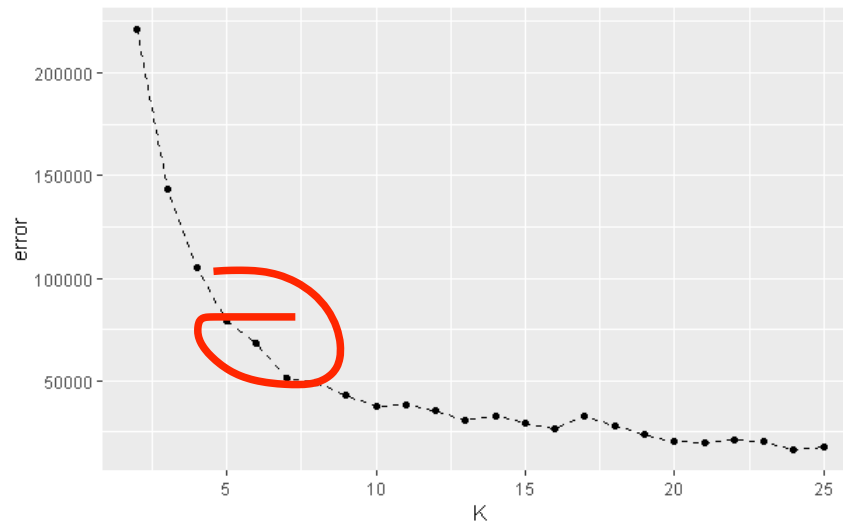
	K <dbl>	error <dbl>
1	2	221136.44
2	3	143391.59
3	4	105299.99
4	5	79309.99
5	6	68331.80
6	7	51130.69
7	8	49456.72
8	9	42755.45
9	10	37412.19
10	11	38621.94

1-10 of 24 rows

Previous 1 2 3 Next

Por último, se realiza la gráfica de la regla del codo y se determina que con un clúster de 5 es el valor óptimo para entrenar el algoritmo de clustering.

```
63 ggplot(data = df, aes(x = K, y = error)) +
64   geom_line(linetype = "dashed") +
65   geom_point()
66
67
68 ```
```



Y procedo a transformar la columna de annual_income_k aplicando la fórmula de estandarización.

```
72 datasetmod2 <- datasetmod
73 datasetmod2$Annual_Income_k <- (datasetmod2$Annual_Income_k -
  mean(datasetmod2$Annual_Income_k)) / sd(datasetmod2$Annual_Income_k)
74 datasetmod2
75
76
```

Description: df [200 x 4]

	Genre <dbl>	Age <int>	Annual_Income_k <dbl>	Spending_Score <int>
181	0	37	1.38741241	32
182	0	32	1.38741241	86
183	1	46	1.42548629	15
184	0	29	1.42548629	88
185	0	41	1.46356018	39
186	1	30	1.46356018	97
187	0	54	1.53970795	24
188	1	28	1.53970795	68
189	0	41	1.61585572	17
190	0	36	1.61585572	85

181-190 of 200 rows

Previous 1 ... 15 16 17 18 19 20 Next

Se repite los pasos para entrenar el algoritmo con datatrain2, income modificado.

```
80 datatrain2 <- kmeans(datasetmod, centers = 5)
81 datatrain2
82
```

K-means clustering with 5 clusters of sizes 28, 36, 95, 5, 36

Cluster means:

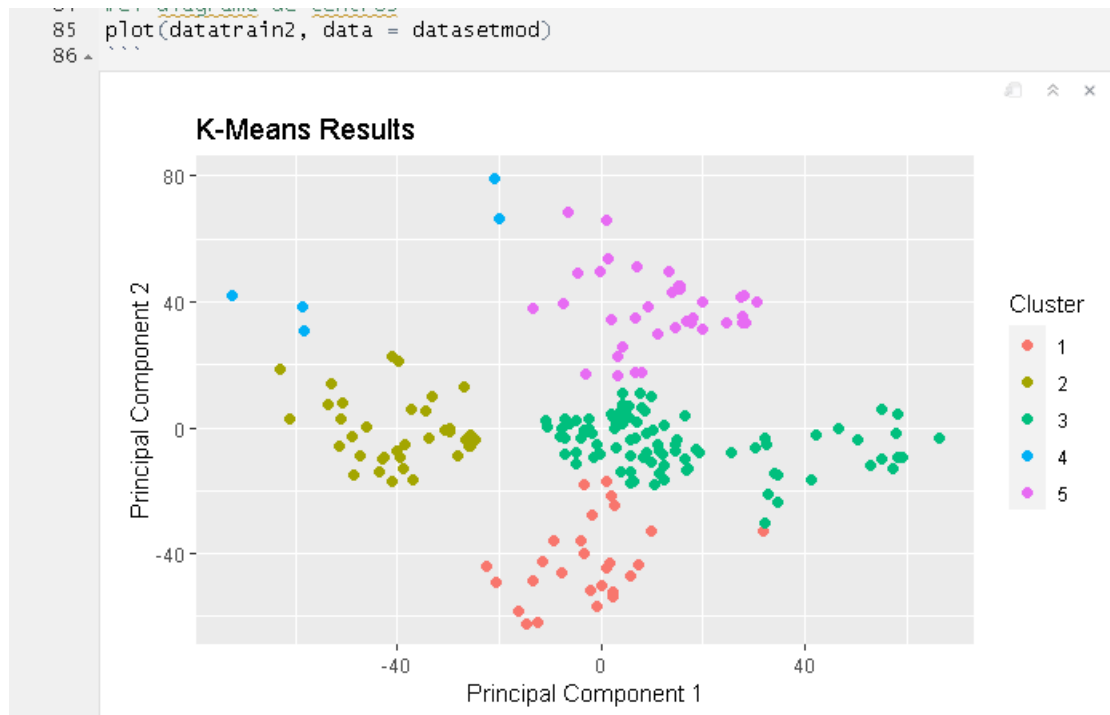
	Genre	Age	Annual_Income_k	Spending_Score
1	0.5000000	24.82143	28.71429	74.25000
2	0.4444444	32.72222	83.11111	82.41667
3	0.3789474	44.89474	48.70526	42.63158
4	0.6000000	34.80000	129.20000	56.40000
5	0.5277778	40.50000	84.52778	18.38889

Clustering vector:

```

1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
1  1  3  1  3  1  3  1  3  1  3  1  3  1  3  1  3  1
19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
3  1  3  1  3  1  3  1  3  1  3  1  3  1  3  1  3  1
37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
3  1  3  1  3  1  3  1  3  1  3  3  3  3  3  1  3  3
55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
3  3  3  3  3  3  3  1  3  3  3  1  3  3  1  3  3  3
73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90
3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3
91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108
3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  3
109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
3  3  3  3  3  3  3  3  3  3  3  3  3  3  3  2  5  2
127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144
5  2  5  2  5  2  5  2  5  2  5  2  5  2  5  2  5  2
145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
5  2  5  2  5  2  5  2  5  2  5  2  5  2  5  2  5  2
163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
```

El diagrama de centros con Income modificado, se puede ver como hay datos que tienen cierta relación por la forma de su agrupamiento cercano de los centroides con cada uno de los datos como es el clúster 1,3,5.



Luego aplico la regla del codo nuevamente, el número de clúster con el error.

```
90 df2 <- data.frame(matrix(ncol = 2, nrow = 0))
91 colnames(df2) <- c("K", "error")
92 for (i in 2:25) {
93   datatrain2 <- kmeans(x = datasetmod2, centers = i)
94   df2[i-1,] <- c(i, datatrain2$tot.withinss)
95 }
96 df2
97
```

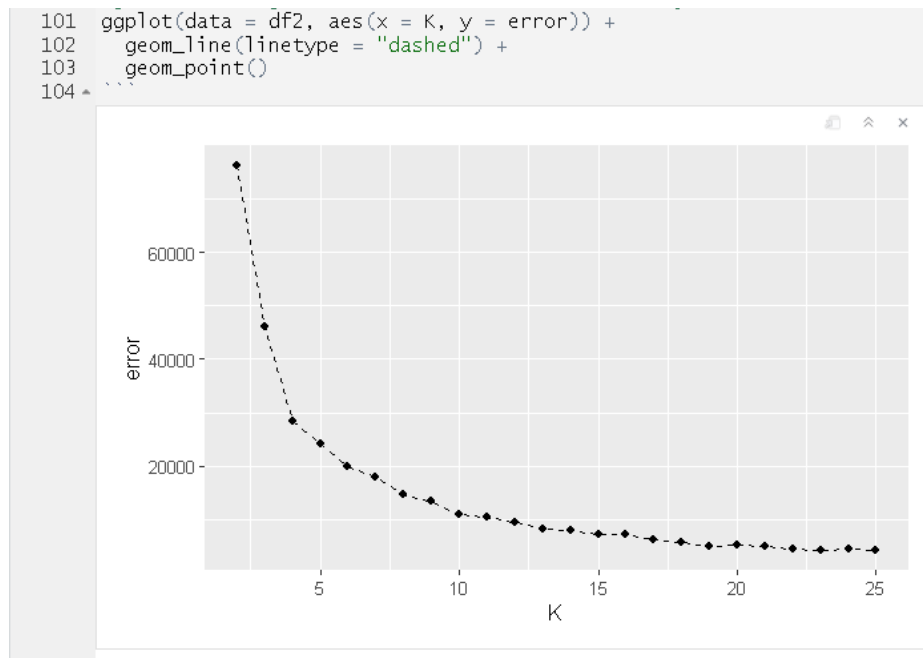
Description: df [24 × 2]

	K <dbl>	error <dbl>
1	2	76196.869
2	3	46078.924
3	4	28402.216
4	5	24045.534
5	6	19995.080
6	7	17300.911
7	8	13285.645
8	9	11835.534
9	10	12698.470
10	11	14358.350

1-10 of 24 rows

Previous 2 3 Next

Luego, se realiza la gráfica de la regla del codo y se determina como queda esta grafica con los valores estandarizados con el clúster igual a 5.



Por último, realizo la gráfica de resultados utilizando la función plot() de R, dicha grafica se observa la clasificación de los datos según la cantidad optima de clústers, usando la regla del codo las estrellas que se observan son los centros de los clústers y los círculos de colores las observaciones.

