7506-2021 Examen por Promoción 1C-2021

Román Vázquez Lareu

TOTAL POINTS

47.5 / 100

QUESTION 1

1 Pto 1 **15 / 20**

✓ **- 5 pts Idea correcta pero no aplica el algoritmo. Solo muestra el resultado final que aparece "magicamente".**

QUESTION 2

2 Pto 2 **12 / 15**

✓ **- 3 pts Para calcular la correlacion, toma el promedio de todos los elementos y no de los que tiene en comun con el otro vector.**

1 Es mejor ajustar con el promedio de los elementos que tiene en comun con el vector a comparar, y no con todos.

QUESTION 3

3 Pto 3 **12 / 20**

✓ **- 6 pts Solo plantea parte b.**

✓ **- 2 pts Faltan iteraciones parte a.**

QUESTION 4

4 Pto 4 **5 / 15**

✓ **- 5 pts Punto c para que realmente no funcione deberia generar todos ceros, lo que plantea podria funcionar luego con mas elementos.**

✓ **- 5 pts Mal punto b.**

2 Esto no esta nada bien.

QUESTION 5

5 Pto 5 **3.5 / 30**

✓ **- 7.5 pts a) No resuelto**

✓ **- 7.5 pts b) No resuelto**

✓ **- 7.5 pts d) No resuelto**

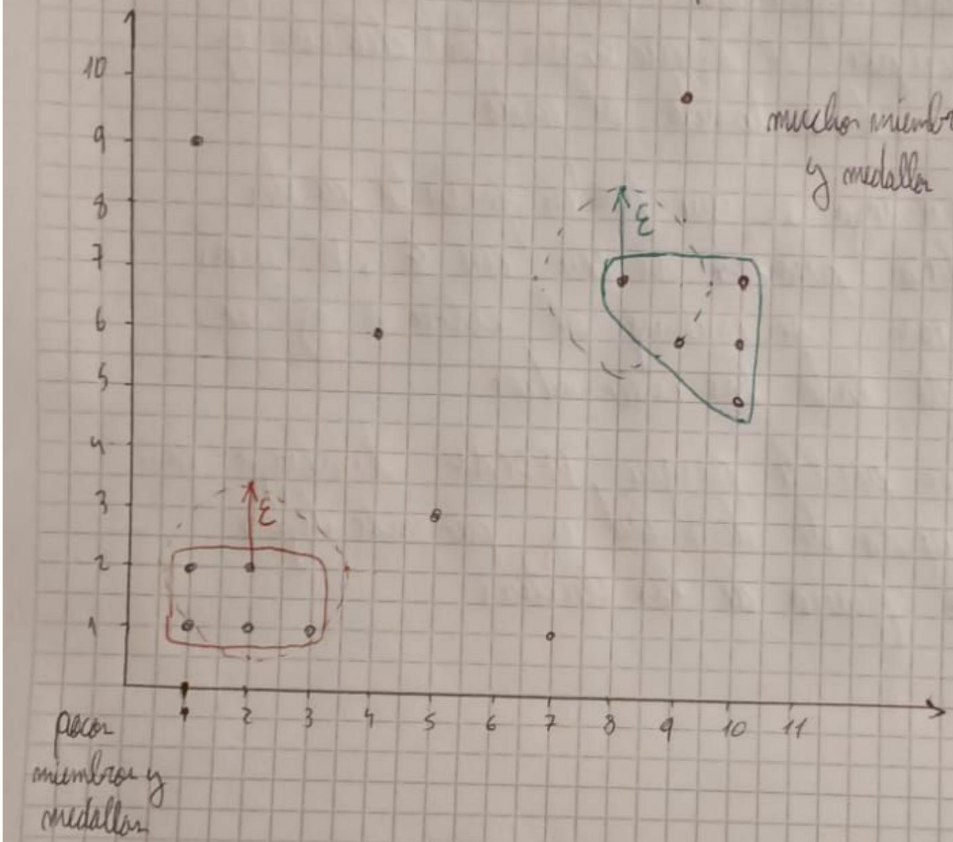
✓ **- 4 pts c) El arbol no es correcto**

3 Este split no fue entrenado en train, el split que le conviene al arbol es C2 y tendría accuracy 1

4 Este no es el MSE

5 Te faltan datasets y decir cúal es la regresión

1) Lo que interesa es hallar a los poderosos y a los débiles, por lo que los que se encuentran entre medio podrían ser considerados como ruido => planteo DBSCAN para eliminarlos



Con la solución actual los 2 grupos que quedan formados no aportan información ya que tienen presentes puntos que no pertenecen de acuerdo a lo que queremos.

DBSAN permite tener puntos que no son de ningún cluster, que es lo que queremos

Con un $\varepsilon = 1,5$ y un $K = 2$ se obtiene la configuración del gráfico. Si bien no logra incluir al punto extremo (9,10), lo que nos brinda esta situación a diferencia de la anterior son certezas.

Con total seguridad podemos afirmar que en esos grupos se encuentran los débiles y poderosos, descartando al resto.

Como desventaja es que pueden faltarle puntos.
Se podría probar si con un ε, K más permisivos se incluye al extremo y se dejan de lado los intermedios.

Esto es posible porque DBSCAN trabaja con el concepto de densidad y permite dejar puntos fuera de los clusters

1 Pto 1 **15 / 20**

✓ **- 5 pts** **Idea correcta pero no aplica el algoritmo. Solo muestra el resultado final que aparece "magicamente".**

2) CF user-user

Pearson para 2 usuarios más similares.

Estimar calificación del user "a" a item 3

	1	2	3	4	5	6	7	8
a	1			5	1		2	3
b	3	2		1	1		3	2
c	2		4		2		4	2
d	1	2			1			3
e		3	3		1	4		2
f	4		5	1			4	1

promedio

de elementos en común

$$\text{sim}(x, y) = \frac{\sum_{s \in S_{xy}} (r_{xs} - \bar{r}_x)(r_{ys} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{xs} - \bar{r}_x)^2} \sqrt{\sum_{s \in S_{xy}} (r_{ys} - \bar{r}_y)^2}}$$

vector de usuario "a"

vector de otro usuario que haya calificado item 3

c = [2 _ 4 _ 2 _ 4 2] → promedio = 2,8

e = [_ 3 3 _ 1 4 _ 2] → promedio = 2,6

f = [4 _ 5 1 _ _ 4 1] → promedio = 3

a = [1 _ _ 5 1 _ 2 3] → promedio = 2,4

resto los promedios por usuario

$c = [\underline{-0,8} \;-\; 1,2 \;-\; \underline{-0,8} \;-\; \underline{1,2} \;\underline{-0,8}]$

$e = [\;-\; 0,4 \;0,4 \;-\; \underline{-1,6} \;1,4 \;-\; \underline{-0,6}]$

$f = [\underline{1} \;-\; 2 \;\underline{-2} \;-\; -\; \underline{1} \;\underline{-2}]$

$a = [-1,4 \;-\; -\; 2,6 \;-1,4 \;-\; -0,4 \;0,6]$

°) entre a y c: $c = [-0,8 \;\; -0,8 \;\; 1,2 \;\; -0,8] \rightarrow [-0,4364 \;\; -0,4364 \;\; 0,6546 \;\; -0,4364]$

$a = [-1,4 \;\; -1,4 \;\; -0,4 \;\; 0,6] \rightarrow [-0,0265 \;\; -0,0265 \;\; -0,007 \;\; 0,0113]$

$sim(a,c) = [-0,4364 \;\; -0,4364 \;\; 0,6546 \;\; -0,4364] * [-0,0265 \;\; -0,0265 \;\; -0,007 \;\; 0,0113] = 0,0136$

°) a y e: $e = [-1,6 \;\; -0,6] \rightarrow [0,9363 \;\; 0,3511]$

$a = [-1,4 \;\; 0,6] \rightarrow [0,9191 \;\; 0,3939]$

$sim(a,e) = [0,9363 \;\; 0,3511] * [0,9191 \;\; 0,3939] =$

$= 0,9989$

°) a y f: $f = [1 \;\; -2 \;\; 1 \;\; -2] \rightarrow [0,3162 \;\; -0,6324 \;\; 0,3162 \;\; -0,6324]$

$a = [-1,4 \;\; 2,6 \;\; -0,4 \;\; 0,6] \rightarrow [-0,4606 \;\; 0,8553 \;\; -0,1316 \;\; 0,1974]$

$sim(a,f) = \underbrace{-0,8529}$

negativo => la distancia puede hacer diverger la aproximación más similares son e y c. Ahora estimo la calificación

$$r_{xi} = \frac{\sum_{Y \in N} s_{xy}\, r_{yi}}{\sum_{Y \in N} s_{xy}}$$

nota de c a item 3 — nota de e a item 3

$$r_{a,\text{item}3} = \frac{0{,}0136 \cdot 4 + 0{,}9989 \cdot 3}{0{,}0136 + 0{,}9989} = 3{,}0134$$

pondero la calificación por la similaridad y divido por la suma de similaridades

=> asumo las calificaciones deben ser enteras

$$r_{a,\text{item}3} = 3$$

2 Pto 2 **12 / 15**

✓ **- 3 pts Para calcular la correlacion, toma el promedio de todos los elementos y no de los que tiene en comun con el otro vector.**

1 Es mejor ajustar con el promedio de los elementos que tiene en comun con el vector a comparar, y no con todos.

3) $m = 8$ $\quad 1-\beta = 0,2$ $\quad (1-\beta)\frac{1}{m} = \frac{1}{40}$

$\beta = 0,8$

Con proba β reparte a nodos que linkea equitativamente

con proba $1-\beta$ asigna $1/m$ a todos

INICIALIZACIÓN $\quad$ 1era iter

$P0_1 = 1/8$ $\quad P0_2 = 0,8[0] + 1/40 = 1/40$

$P1_1 = 1/8$ $\quad P1_2 = 0,8[\frac{1}{2}P0_1 + P2_1] + 1/40 = 7/40$

$P2_1 = 1/8$ $\quad P2_2 = 0,8[\frac{1}{2}P0_1 + \frac{1}{4}P3_1] + 1/40 = 4/40$

$P3_1 = 1/8$ $\quad P3_2 = 0,8[P8_1] + 1/40 = 1/8$

$P4_1 = 1/8$ $\quad P4_2 = 0,8[P1_1 + \frac{1}{4}P3_1] + 1/40 = 3/20$

$P5_1 = 1/8$ $\quad P5_2 = 0,8[\frac{1}{2}P4_1 + \frac{1}{4}P3_1] + 1/40 = 1/10$

$P6_1 = 1/8$ $\quad P6_2 = 0,8[\frac{1}{4}P3_1] + 1/40 = 2/40$

$P8_1 = 1/8$ $\quad P8_2 = 0,8[P6_1 + \frac{1}{2}P4_1 + P5_1] + 1/40 = 11/40$

suman 1, bien las cuentas

2do iter

$P0_3 = 0,8[0] + 1/40 = 1/40$

$P1_3 = 0,8[\frac{1}{2}P0_2 + P2_2] + 1/40 = 23/200$

$P2_3 = 0,8[\frac{1}{2}P0_2 + \frac{1}{4}P3_2] + 1/40 = 3/50$

$P3_3 = 0,8[P8_2] + 1/40 = 49/200$

$P4_3 = 0,8[P1_2 + \frac{1}{4}P3_2] + 1/40 = 19/100$

$P5_3 = 0,8[\frac{1}{2}P4_2 + \frac{1}{4}P3_2] + 1/40 = 11/100$

$P6_3 = 0,8[\frac{1}{4}P3_2] + 1/40 = 1/20$

$P8_3 = 0,8[P6_2 + \frac{1}{2}P4_2 + P5_2] + 1/40 = 41/200$

suman 1, bien las cuentas

... iteraciones hasta converger

Por ej:

$P5_m = 0,8[\frac{1}{2}P4_{m-1} + \frac{1}{4}P3_{m-1}] + 1/40$

debo seguir iterando de esta manera hasta converger, es decir que los valores no se modifiquen.
En cada nueva iteración, utilizaré para el cálculo lo obtenido en la anterior

Si la Page 0 no es confiable, entonces no la voy a querer favorecer con la teletransportación.
Así, en primer lugar no le voy a sumar el $(1-\beta)\frac{1}{n}$
En segundo lugar, lo que no le doy a P0 se distribuirá entre el resto, por lo que la suma por teletransportación para el resto será de $(1-\beta)\cdot\frac{1}{n}$ con $n=7$ ya que no tengo en cuenta P0.

A modo de ejemplo.

$P0_m = 0{,}8\cdot[0]$

$P1_m = 0{,}8[\frac{1}{2}P0_{m-1} + P2_{m-1}] + (1-0{,}8)\cdot\frac{1}{7}$

$P2_m = 0{,}8[\frac{1}{2}P0_{m-1} + \frac{1}{4}P3_{m-1}] + (1-0{,}8)\cdot\frac{1}{7}$

$P3_m = 0{,}8[P8_{m-1}] + (1-0{,}8)\cdot\frac{1}{7}$

$P4_m = 0{,}8[P1_{m-1} + \frac{1}{4}P3_{m-1}] + (1-0{,}8)\frac{1}{7}$

$P5_m = 0{,}8[\frac{1}{2}P4_{m-1} + \frac{1}{4}P3_{m-1}] + (1-0{,}8)\frac{1}{7}$

$P6_m = 0{,}8[\frac{1}{4}P3_{m-1}] + (1-0{,}8)\frac{1}{7}$

$P8_m = 0{,}8[P6_{m-1} + \frac{1}{2}P4_{m-1} + P5_{m-1}] + (1-0{,}8)\frac{1}{7}$

Los valores de iniciación se conservan

3 Pto 3 **12 / 20**

- ✓ **- 6 pts** **Solo plantea parte b.**
- ✓ **- 2 pts** **Faltan iteraciones parte a.**

5) a)

4) $S = [3, 6, 6, 3, 3, 6, 5, 6]$

$h(x) = x \bmod 32$

$M^0(S)$ usamos Flajolet Martin con 1 estimador 5 bits considerando los ceros a derecha.

$M^0(S) = 3$ porque observé 3, 6 y 5

x	H(x)
3	$3_{10} = 00011_2$
5	$5_{10} = 00101_2$
6	$6_{10} = 00110_2$

FM aproxima con 2^r, por lo que r debería ser 1 o 2

$H(3) = 00011$ $r = 0$
$H(6) = 00110$ $r = 1$
$H(6) = 00110$ $r = 1$
$H(3) = 00011$ $r = 1$
$H(3) = 00011$ $r = 1$
$H(6) = 00110$ $r = 1$
$H(5) = 00101$ $r = 1$
$H(6) = 00110$ $r = 1$

$M^0(S) = 2^1 = 2$

para aproximar bien r debía ser 1 o 2 y dio bien
Hay que tener en cuenta que al recibir un número impar, nunca podrá detectarlo como nuevo elemento ya que termina en 1

$y = mx + b = x$

Esto no es bueno. Si bien dio bien la aproximación en este caso, la función no es buena ya que no cuenta como ~~[illegible]~~ nuevos a impares.

B. si planteo $3x + 1 \mod 32 = h(x)$

x	H(x)
3	$10_{10} = 01010_2$
5	$16_{10} = 10000_2$
6	$19_{10} = 10011_2$

si bien en este caso no aproximaría bien, de venir otros valores funcionaría mejor. Si quisiera que aproxime para este caso únicamente, con $x+1 \mod 32$ lograría tener un $r = 2$ aproximando a 4.

Esta función parece que funcionaría mejor ya que daría mejor respuesta a los impares. A su vez, si "a" fuera par, devolvería todos pares o impares, esta función lo evita. En términos binarios distribuye mejor los 0, 00, 000, etc.

C. si planteo $h(x) = 2x + 1 \mod 32$

x	H(x)
3	7
5	11
6	13

da siempre impar, el bit de la derecha será siempre 1, por lo que r no cambiaría su valor

4 Pto 4 5 / 15

✓ **- 5 pts Punto c para que realmente no funcione deberia generar todos ceros, lo que plantea podria funcionar luego con mas elementos.**

✓ **- 5 pts Mal punto b.**

2 Esto no esta nada bien.

3) c)

X · train set

3 clases ⇒ $\log_2 3 \to 2$ bits

binary encoding

A	⇒	0	0
B		0	1
C		1	0
D		1	1
		↑	↑
		C1	C2

X:

	label		C1	C2	
A	0	⇒	0	0	0
B	1		0	1	1
C	0		1	0	0
C	0		1	0	0

		real	prediction
~~A~~	0	0	0
0	0	0	0
1	1	0	1
0	1	1	1

$$\mathcal{E}(h) = \frac{1}{4} \mathbb{1}\{h(x_i) \neq y_i\} = \frac{1}{4} \cdot 1 = 0{,}25$$

⇒ accuracy = 0,75

a)

$h_\theta(\vec{x}) = \theta^t \cdot \vec{x}$

para la de entrenamiento

— valores de test

— valores de train

los valores d_1 y d_2 tienen que ser tales que $d_1^2 + d_2^2 = 0.5$

el error cuadrático es $J(\theta) = \Sigma(h_\theta(x) - y)^2$

5 Pto 5 **3.5 / 30**

- ✓ **- 7.5 pts a) No resuelto**
- ✓ **- 7.5 pts b) No resuelto**
- ✓ **- 7.5 pts d) No resuelto**
- ✓ **- 4 pts c) El arbol no es correcto**

3 Este split no fue entrenado en train, el split que le conviene al arbol es C2 y tendría accuracy 1

4 Este no es el MSE

5 Te faltan datasets y decir cúal es la regresión