Comp 9318 assignment 1

Z5196480

Huiyao zuo

## Q1

(1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Location | Time | Item | Quantity |
| 0 | Sydney | 2005 | PS2 | 1400 |
| 1 | Sydney | 2005 | ALL | 1400 |
| 2 | Sydney | 2006 | PS2 | 1500 |
| 3 | Sydney | 2006 | Wii | 500 |
| 4 | Sydney | 2006 | ALL | 2000 |
| 5 | Sydney | ALL | PS2 | 2900 |
| 6 | Sydney | ALL | Wii | 500 |
| 7 | Sydney | ALL | ALL | 3400 |
| 8 | Melbourne | 2005 | XBox 360 | 1700 |
| 9 | Melbourne | ALL | XBox 360 | 1700 |
| 10 | Melbourne | 2005 | ALL | 1700 |
| 11 | Melbourne | ALL | ALL | 1700 |
| 12 | ALL | 2005 | PS2 | 1400 |
| 13 | ALL | 2005 | XBox 360 | 1700 |
| 14 | ALL | 2005 | ALL | 3100 |
| 15 | ALL | 2006 | PS2 | 1500 |
| 16 | ALL | 2006 | Wii | 500 |
| 17 | ALL | 2006 | ALL | 2000 |
| 18 | ALL | ALL | PS2 | 2900 |
| 19 | ALL | ALL | Wii | 500 |
| 20 | ALL | ALL | XBox 360 | 1700 |
| 21 | ALL | ALL | ALL | 5100 |

(2)

SELECT Location, Time, Item, SUM(Quantity)

From Sales

Group by Location, Time, Item

UNION ALL

SELECT Location, Time, ALL, SUM(Quantity)

From Sales

Group by Location, Time

UNION ALL

SELECT Location, ALL, Item, SUM(Quantity)

From Sales

Group by Location, Item

UNION ALL

SELECT ALL, Time, Item, SUM(Quantity)

From Sales

Group by Time, Item

UNION ALL

SELECT Location, ALL, ALL SUM(Quantity)

From Sales

Group by Location

UNION ALL

SELECT ALL, Time, ALL, SUM(Quantity)

From Sales

Group by Time

UNION ALL

SELECT ALL, ALL, Item, SUM(Quantity)

From Sales

Group by Item

UNION ALL

SELECT ALL, ALL, ALL, SUM(Quantity)

From Sales;

(3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Location | Time | Item | Quantity |
| 0 | Sydney | 2006 | ALL | 2000 |
| 1 | Sydney | ALL | PS2 | 2900 |
| 2 | Sydney | ALL | ALL | 3400 |
| 3 | ALL | 2005 | ALL | 3100 |
| 4 | ALL | 2006 | ALL | 2000 |
| 5 | ALL | ALL | PS2 | 2900 |
| 6 | ALL | ALL | ALL | 5100 |

(4)

Index = ltem(index) + 4\*Time(index) + 4\*3\* Location(index)

|  |  |
| --- | --- |
| index | Quantity |
| 17 | 1400 |
| 16 | 1400 |
| 21 | 1500 |
| 23 | 500 |
| 20 | 2000 |
| 13 | 2900 |
| 15 | 500 |
| 12 | 3400 |
| 30 | 1700 |
| 26 | 1700 |
| 28 | 1700 |
| 24 | 1700 |
| 5 | 1400 |
| 6 | 1700 |
| 4 | 3100 |
| 9 | 1500 |
| 11 | 500 |
| 8 | 2000 |
| 1 | 2900 |
| 3 | 500 |
| 2 | 1700 |
| 0 | 5100 |

## Q2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | P1 | P2 | P3 | P4 | P5 |
| P1 | 1.00 | 0.1 | 0.41 | 0.55 | 0.35 |
| P2 | 0.1 | 1.00 | 0.64 | 0.47 | 0.98 |
| P3 | 0.41 | 0.64 | 1.00 | 0.44 | 0.85 |
| P4 | 0.55 | 0.47 | 0.44 | 1.00 | 0.76 |
| P5 | 0.35 | 0.98 | 0.85 | 0.76 | 1.00 |

Max = p25 = 0.98

Update the table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | P1 | P25 | P3 | P4 |
| P1 | 1.00 | 0.476 | 0.41 | 0.55 |
| P25 | 0.476 | 1.00 | 0.823 | 0.736 |
| P3 | 0.41 | 0.823 | 1.00 | 0.44 |
| P4 | 0.55 | 0.736 | 0.44 | 1.00 |

Max = p253 = 0.823

Update the table

|  |  |  |  |
| --- | --- | --- | --- |
|  | P1 | P253 | P4 |
| P1 | 1.00 | 0.555 | 0.55 |
| P253 | 0.555 | 1.00 | 0.69 |
| P4 | 0.55 | 0.69 | 1.00 |

Max = p2534 = 0.69

Last dot is p1,then link p1

图片包含 游戏机, 截图

描述已自动生成

## Q3

(1)

Initialize k centers C = [c1, c2, . . . , ck];  
canStop ← false;  
while canStop = false do

Initialize k empty clusters G = [g1, g2, . . . , gk]; for each data point p ∈ D do

cx ← NearestCenter(p, C);

gcx .append(p);

count = 0

for each group g ∈ G do

origal\_ci = gci

ci ← ComputeCenter(g);

if ci == origal\_ci ;

count += 1

if count == k ;

canStop ← Ture

return G;

according the change of the center to compare the group have change or not ,if all the groups has no change means the group will not change and we get the answer.

(2)

The cost of k clusters would not increase, because in each iteration ,the cost is decrease or not change ,so the total cost is decrease or not change.

In the first iteration, each dot will get into a new group whose center is closer to the dot than the dot’s group before, so every dot will get a smaller  , so in this iteration total cost is decrease or not change.

In the second iteration, all the dot in the group will computer a new center, the new center will balance the distance with all the dot, because the new center will have the smallest total distance with all the dot than rest of dot to be the center. so the total in the group will decrease or not change.

So the cost of k cluster at the end in the iteration will never increase.

(3)

because after each iteration, the total cost is decrease, and the cost would not decrease infinitly, because the distance between dots will not change. So it will always converges to a local minima. it would not change between two situation, because it will always choose the smaller cost one, and if two situation is the same, it will chose one and this will not influence the total cost.

When the iteration would not change the dot in the groups, it converges to the minima Which is the best group split suit the original input condition.