
# Department of Computer Science

# California State University, Channel Islands

## COMP-546: Pattern Recognition

## Lab Report

Lesson 8 phys546 Clustering HW 8A

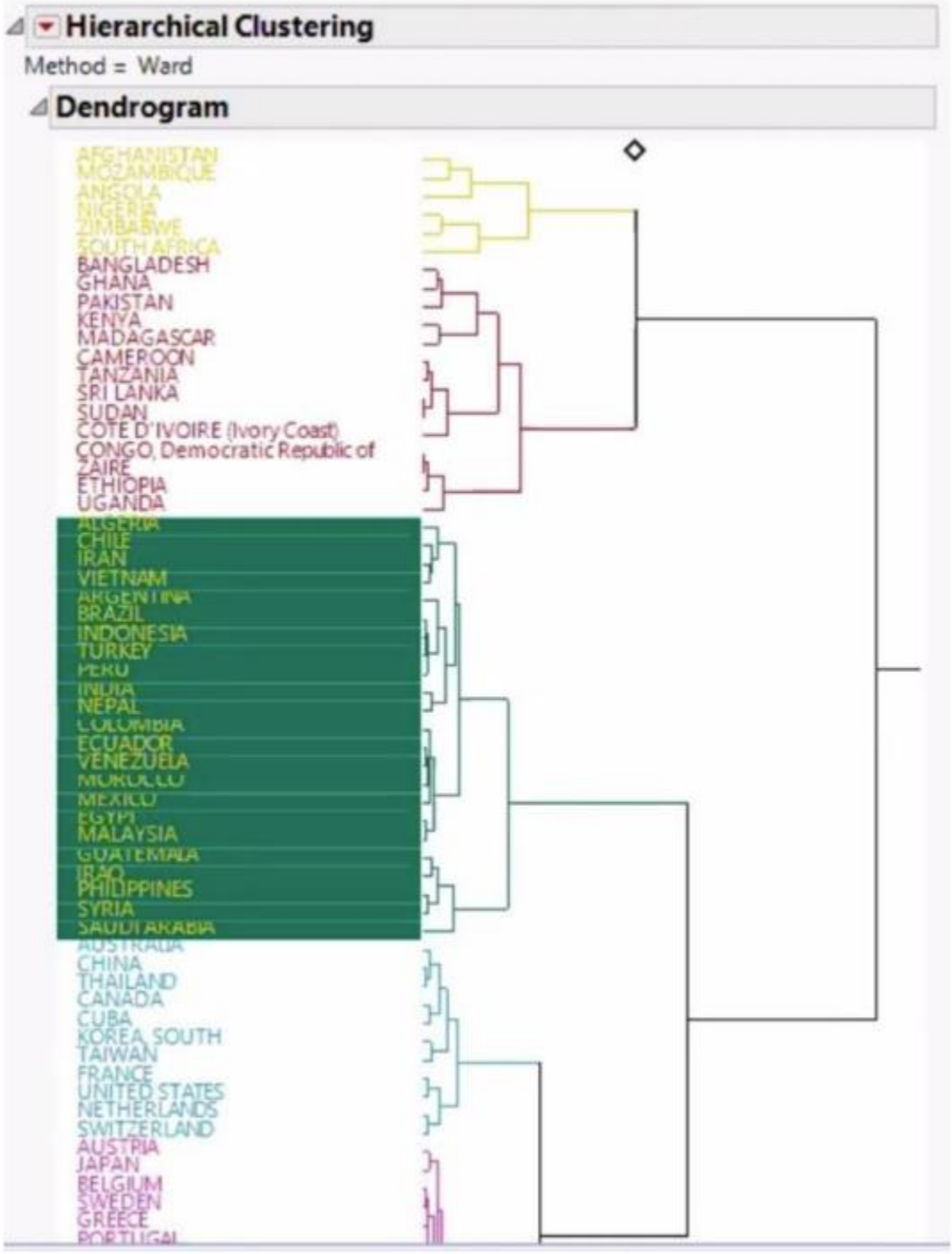
Student Name: Sandipta Subir Khare
Student Major: Masters in Computer Science

**Q1. The data set Birth Death.jmp contains mortality (i.e., birth and death) rates for several countries. Use cluster analysis to determine which countries share similar mortality characteristics. What do you notice that is similar among the countries that cluster together?**

Ans.

| | country | birth | death | Region |
|---|---|---|---|---|
| 1 | AFGHANISTAN | 45 | 19 | Asia |
| 2 | ALGERIA | 17 | 5 | Africa |
| 3 | ANGOLA | 44 | 24 | Africa |
| 4 | ARGENTINA | 18 | 7 | S America |
| 5 | AUSTRALIA | 12 | 7 | Pacific |
| 6 | AUSTRIA | 9 | 10 | Europe |
| 7 | BANGLADESH | 25 | 9 | Asia |
| 8 | BELGIUM | 10 | 10 | Europe |
| 9 | BRAZIL | 18 | 6 | S America |
| 10 | BULGARIA | 10 | 14 | Europe |
| 11 | BURMA | 17 | 9 | Asia |
| 12 | CAMEROON | 34 | 12 | Africa |
| 13 | CANADA | 10 | 8 | N America |
| 14 | CHILE | 15 | 6 | S America |
| 15 | CHINA | 14 | 7 | Asia |
| 16 | COLOMBIA | 20 | 6 | S America |
| 17 | CONGO, Democratic Republic of | 43 | 12 | Africa |
| 18 | COTE D'IVOIRE (Ivory Coast) | 32 | 11 | Africa |
| 19 | CUBA | 11 | 7 | N America |
| 20 | CZECH REPUBLIC | 9 | 11 | Europe |
| 21 | ECUADOR | 21 | 5 | S America |
| 22 | EGYPT | 22 | 5 | Africa |
| 23 | ETHIOPIA | 44 | 12 | Africa |
| 24 | FRANCE | 13 | 9 | Europe |
| 25 | GERMANY | 8 | 11 | Europe |
| 26 | GHANA | 29 | 9 | Africa |
| 27 | GREECE | 9 | 11 | Europe |
| 28 | GUATEMALA | 28 | 5 | S America |
| 29 | HUNGARY | 10 | 13 | Europe |
| 30 | INDIA | 22 | 6 | Asia |
| 31 | INDONESIA | 19 | 6 | Pacific |
| 32 | IRAN | 17 | 6 | Asia |
| 33 | IRAQ | 30 | 5 | Asia |
| 34 | ITALY | 8 | 11 | Europe |
| 35 | JAPAN | 8 | 10 | Asia |
| 36 | KAZAKHSTAN | 17 | 9 | Europe |
| 37 | KENYA | 37 | 10 | Africa |
| 38 | KOREA, NORTH | 15 | 11 | Asia |
| 39 | KOREA, SOUTH | 9 | 6 | Asia |
| 40 | MADAGASCAR | 38 | 8 | Africa |
| 41 | MALAYSIA | 22 | 5 | Asia |
| 42 | MEXICO | 20 | 5 | N America |
| 43 | MOROCCO | 21 | 5 | Africa |
| 44 | MOZAMBIQUE | 38 | 20 | Africa |
| 45 | NEPAL | 23 | 7 | Africa |

| | country | birth | death | Region |
|---|---|---|---|---|
| 46 | NETHERLANDS | 10 | 9 | Europe |
| 47 | NIGERIA | 37 | 17 | Africa |
| 48 | PAKISTAN | 28 | 8 | Asia |
| 49 | PERU | 19 | 6 | S America |
| 50 | PHILIPPINES | 26 | 5 | Pacific |
| 51 | POLAND | 10 | 10 | Europe |
| 52 | PORTUGAL | 10 | 11 | Europe |
| 53 | ROMANIA | 11 | 12 | Europe |
| 54 | RUSSIA | 11 | 16 | Europe |
| 55 | SAUDI ARABIA | 29 | 2 | Africa |
| 56 | SLOVAKIA | 11 | 10 | Europe |
| 57 | SOUTH AFRICA | 20 | 17 | Africa |
| 58 | SPAIN | 10 | 10 | Europe |
| 59 | SRI LANKA | 34 | 13 | Asia |
| 60 | SUDAN | 34 | 13 | Africa |
| 61 | SWEDEN | 10 | 10 | Europe |
| 62 | SWITZERLAND | 10 | 9 | Europe |
| 63 | SYRIA | 26 | 5 | Asia |
| 64 | TAIWAN | 9 | 7 | Asia |
| 65 | TANZANIA | 34 | 13 | Africa |
| 66 | THAILAND | 13 | 7 | Asia |
| 67 | TURKEY | 19 | 6 | Asia |
| 68 | UGANDA | 48 | 12 | Africa |
| 69 | UNITED KINGDOM | 11 | 10 | Europe |
| 70 | UNITED STATES | 14 | 8 | N America |
| 71 | VENEZUELA | 21 | 5 | S America |
| 72 | VIETNAM | 16 | 6 | Asia |
| 73 | ZAIRE | 43 | 12 | Africa |
| 74 | ZIMBABWE | 31 | 16 | Africa |


Hierarchical Clustering
Method = Ward
Dendrogram
AFGHANISTAN
MOZAMBIQUE
ANGOLA
NIGERIA
ZIMBABWE
SOUTH AFRICA
BANGLADESH
GHANA
PAKISTAN
KENYA
MADAGASCAR
CAMEROON
TANZANIA
SRI LANKA
SUDAN
COTE D'IVOIRE (Ivory Coast)
CONGO, Democratic Republic of
ZAIRE
ETHIOPIA
UGANDA
ALGERIA
CHILE
IRAN
VIETNAM
ARGENTINA
BRAZIL
INDONESIA
TURKEY
PERU
INDIA
NEPAL
COLOMBIA
ECUADOR
VENEZUELA
MOROCCO
MEXICO
EGYPT
MALAYSIA
GUATEMALA
IRAQ
PHILIPPINES
SYRIA
SAUDI ARABIA
AUSTRALIA
CHINA
THAILAND
CANADA
CUBA
KOREA, SOUTH
TAIWAN
FRANCE
UNITED STATES
NETHERLANDS
SWITZERLAND
AUSTRIA
JAPAN
BELGIUM
SWEDEN
GREECE
PORTUGAL

**Q2. Consider the data in teeth.jmp, which contains data on the numbers of teeth of different types in a variety of mammals. Do hierarchical clustering to obtain the relevant dendrogram vary the number of clusters? With reference to the scree plot, and your own opinion of which animals should be clustered together, how many clusters do you think is optimal?**

Ans.

```
data Teeth;
    title 'Mammals' 'Teeth';
    input mammal $ 1-16 (V1-V8) (1.);
    label V1 = 'Right Top Incisors'
          V2 = 'Right Bottom Incisors'
          V3 = 'Right Top Canines'
          V4 = 'Right Bottom Canines'
          V5 = 'Right Top Premolars'
          V6 = 'Right Bottom Premolar'
          V7 = 'Right Top Molars'
          V8 = 'Right Bottom Molars';
    datalines;
```

[ So we will be using 'THE TREE' Procedure to make the clusters.]

```
Brown Bat          23113333
Mole               32103333
Silver Hair Bat    23113233
Pigmy Bat          23113233
Hoary Bat          23111233
Red Bat            13112233
Pika               21002233
Rabbit             21003233
Beaver             11002133
Groundhog          11002133
Gray Squirrel      11001133
House Mouse        11000033
Porcupine          11001133
Wolf               33114423
Bear               33114423
Raccoon            33114432
```

| | |
|---|---|
| Marten | 33 11 44 12 |
| Weasel | 33 11 33 12 |
| Wolverine | 33 11 44 12 |
| Badger | 33 11 33 12 |
| River Otter | 33 11 43 12 |
| Sea Otter | 32 11 33 12 |
| Jaguar | 33 11 32 11 |
| Cougar | 33 11 32 11 |
| Fur Seal | 32 11 44 11 |
| Sea Lion | 32 11 33 22 |
| Grey Seal | 21 11 41 1 |
| Elephant Seal | 0 4 10 33 3 |
| Reindeer | 0 4 1 0 33 3 |
| Elk | 0 4 00 33 33 |
| Deer | 0 4 00 33 33 |
| Moose | 0 4 00 33 33 |

options pagesize = 60 linesize = 110;

proc cluster method = average std pseudo noeigen
outtree = tree;
id mammal;
var V1 - V8;
run;

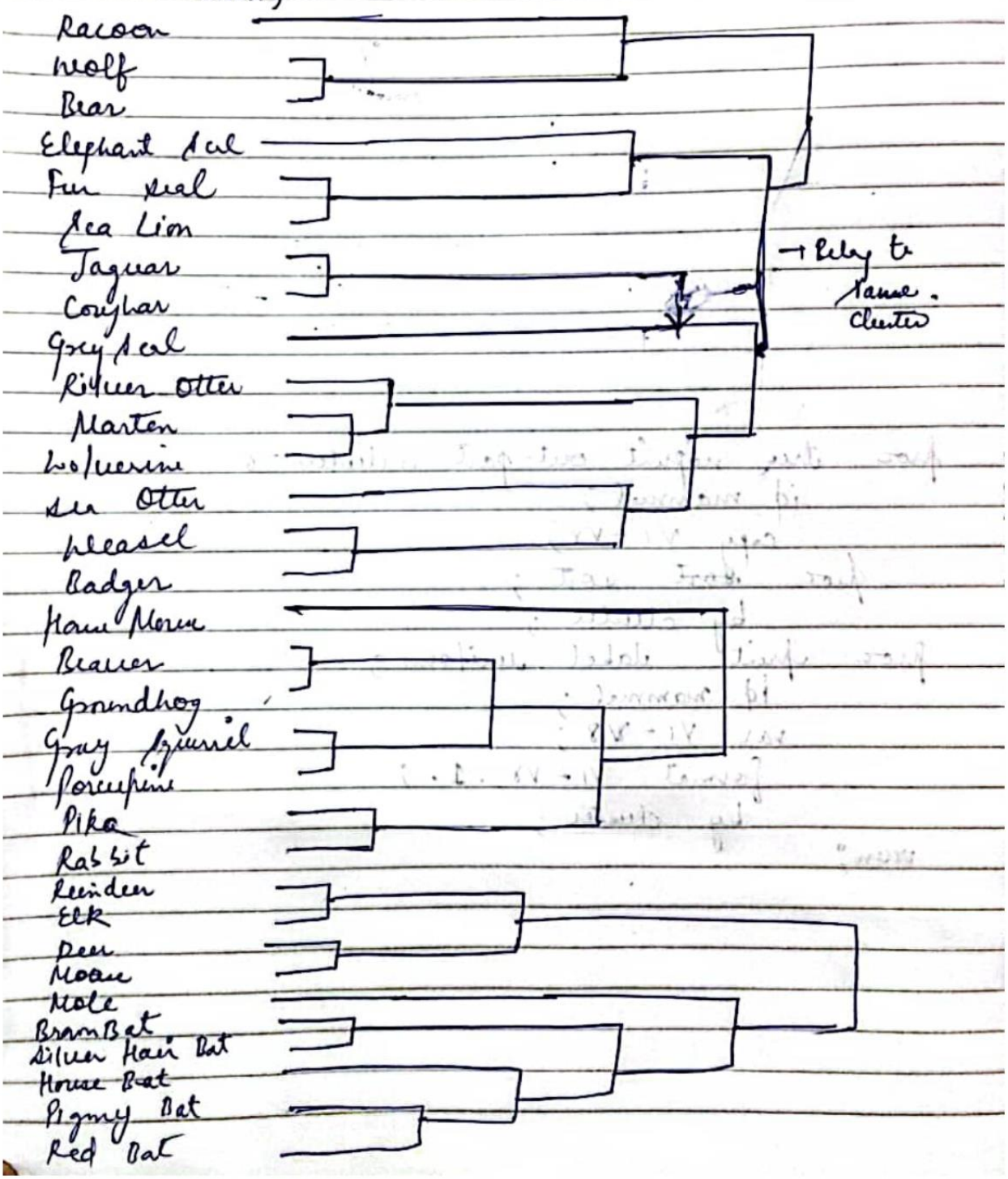
proc tree graphics horizontal;
run;

## CLUSTERS :-

| | --- Clusters Found --- | | Freq | PAF | PAT$_2$ |
|---|---|---|---|---|---|
| 31 | Beaver | Groundhog | 2 | – | – |
| 30 | Gray Squirrel | Porcupine | 2 | – | – |
| 29 | Wolf | Bear | 2 | – | |
| 28 | Marten | Wolverine | 2 | – | – |
| 27 | Weasel | Badger | 2 | – | – |
| 26 | Jaguar | Cougar | 2 | – | – |
| 25 | Fur Seal | Sea Lion | 2 | – | – |
| 24 | Reindeer | Elk | 2 | – | – |
| 23 | Deer | Moose | 2 | – | – |
| 22 | Pigmy Sd | Red Bat | 2 | 291 | – |
| 21 | CL28 | River Otter | 3 | 139 | – |
| 20 | CL31 | CL30, | 4 | 82.2 | – |
| 19 | Lion Bat | Silver Haired Bat | 2 | 76.7 | – |
| 18 | Pika | Rabbit | 2 | 77.2 | – |
| 17 | CL27 | Sea Otter | 3 | 67.4 | – |
| 16 | CL22 | Hoary Bat | 3 | 62.9 | 1.7 |
| 15 | CL21 | CL17 | 6 | 47.4 | 6.8 |
| 14 | CL25 | Elephant Seal | 3 | 45.0 | – |
| 13 | CL19 | CL16 | 5 | 40.8 | 3.5 |
| 12 | CL15 | Gray Seal | 7 | 39.9 | 2.8 |
| 11 | CL29 | Raccoon | 3 | 39.0 | – |
| 10 | CL18 | CL20 | 6 | 34.5 | 10.7 |
| 9 | CL12 | CL26 | 9 | 30.0 | 7.1 |
| 8 | CL24 | CL23 | 4 | 28.7 | – |
| 7 → | CL9 | CL14 | 12 | 25.7 | 7.0 |
| 6 → | CL16 | House Mouse | 1 | 23.3 | 4.1 |
| 5 → | CL10 | CL7 | 15 | 26.8 | 6.9 |
| 4 → | CL13 | Mole | 6 | 31.9 | 7.2 |
| 3 → | CL4 | CL8 | 10 | 31.0 | 12.7 |
| 2 → | CL3 | CL6 | 17 | 27.1 | 16.1 |
| 1 → | CL2 | CL5 | 32 | – | 27.1 |

We can also make cluster tree with sort & height:-

proc tree sort height = n horizontal;
run;



```
proc tree noprint out=part nclusters=6
     id mammal;
        copy V1-V8;
proc sort sort;
        by cluster;
proc print label uniform;
     id mammal;
     var V1-V8;
      format V1-V8 .1.;
       by cluster;
run;
```

**Q3. Consider the problem of clustering nine major cities in the United States. The distance between them (in miles) is given below:**

| | BOS | NYC | DC | MIA | CHI | SEA | SF | LA | DEN |
|---|---|---|---|---|---|---|---|---|---|
| BOS | 0 | 206 | 429 | 1,504 | 963 | 2,976 | 3,095 | 2,979 | 1,949 |
| NYC | 206 | 0 | 233 | 1,308 | 802 | 2,815 | 2,934 | 2,786 | 1,771 |
| DC | 429 | 233 | 0 | 1,075 | 671 | 2,684 | 2,799 | 2,631 | 1,616 |
| MIA | 1,504 | 1,308 | 1,075 | 0 | 1,329 | 3,273 | 3,053 | 2,687 | 2,037 |
| CHI | 963 | 802 | 671 | 1,329 | 0 | 2,013 | 2,142 | 2,054 | 996 |

(continued)

(continued)

| | BOS | NYC | DC | MIA | CHI | SEA | SF | LA | DEN |
|---|---|---|---|---|---|---|---|---|---|
| SEA | 2,976 | 2,815 | 2,684 | 3,273 | 2,013 | 0 | 808 | 1,131 | 1,307 |
| SF | 3,095 | 2,934 | 2,799 | 3,053 | 2,142 | 808 | 0 | 379 | 1,235 |
| LA | 2,979 | 2,786 | 2,631 | 2,687 | 2,054 | 1,131 | 379 | 0 | 1,059 |
| DEN | 1,949 | 1,771 | 1,616 | 2,037 | 996 | 1,307 | 1,235 | 1,059 | 0 |

(i) Use single-linkage and (ii) complete-linkage.

Ans.

ii) single linkage:-

step(i):- find the minelement

206 → (NYC, BOS)


| | NYC, BOS | DC | MIA | CHI | SEA | SF | LA | DEN |
|---|---|---|---|---|---|---|---|---|
| NYC, BOS | 0 | | | | | | | |
| DC | 233 | 0 | | | | | | |
| MIA | 1308 | 1075 | 0 | | | | | |
| CHI | 802 | 671 | 1329 | 0 | | | | |
| SEA | 2815 | 2684 | 3273 | 2013 | 0 | | | |
| SF | 2934 | 2799 | 3053 | 2142 | 808 | 0 | | |
| LA | 2786 | 2631 | 2687 | 2054 | 1131 | 379 | 0 | |
| DEN | 1771 | 1616 | 2037 | 996 | 1307 | 1235 | 1059 | 0 |

To update matrix.

→ Min[ dis (NYC, BOS), DC]

⇒ Min[ dis (NYC, DC), (BOS, DC)]

⇒ Min[ 233, 429] = 233

→ Min [ dis ( NYC, MIA), ( BOS, MIA)]

⇒ Min[ 1308, 1504] = 1308

→ Min [ dis ( NYC, CHI), ( BOS, CHI)].

⇒ Min[ 802, 963] = 802

→ Min[ dis ( NYC, SEA), (BOS, SEA)]

⇒ Min[ 2815, 2976] = 2815

→ Min [ dis ( NYC, SF), (BOS, SF)]

⇒ Min [ 2934, 3053] = 2934

→ Min [ dis ( NYC, LA), (BOS, LA)]

⇒ Min [ 2786, 2979] = 2786

→ Min [ dis (NYC, DEN), (BOS, DEN)]

⇒ Min [ 1771, 1949] = 1771

Step(2):- find minelement

233 → ( DC, (NYC, BOS) )

| | NYC, BOS, DC | MIA | CHI | SEA | SF | LA | DEN |
|---|---|---|---|---|---|---|---|
| NYC, BOS, DC | 0 | | | | | | |
| MIA | 1075 | 0 | | | | | |
| CHI | 671 | 1329 | 0 | | | | |
| SEA | 2684 | 3273 | 2013 | 0 | | | |
| SF | 2799 | 3053 | 2142 | 808 | 0 | | |
| LA | 2631 | 2687 | 2054 | 1131 | 379 | 0 | |
| DEN | 1616 | 2037 | 996 | 1307 | 1235 | 1059 | 0 |

→ Min [ dis (DC, (NYC, BOS)), MIA)]

⇒ Min [ dis ( (DC, MIA), ((NYC, BOS), MIA))]

⇒ Min [ 1075, 1308] = 1075

→ min [ dis ( DC, CHI), ((NYC, BOS), CHI)]

⇒ min[ 671, 802] = 671

→ min[ dis ( DC, SEA), ((NYC, BOS), SEA)]

⇒ min[ 2684, 2815] = 2684

→ min [ dis ( DC, SF), ((NYC, BOS), SF)]

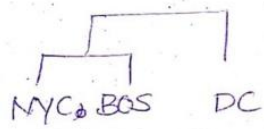
⇒ min[ 2799, 2934] = 2799

→ min [ dis ( DC, LA), ((NYC, BOS), LA)]

⇒ min[ 2631, 2786] = 2631

→ min [ dis (DC, DEN), ((NYC, BOS), DEN)]

⇒ min[ 1616, 1771] = 1616




Step(3) :- find min element

379 → (LA, SF).

| | NYC, BOS, DC | MIA | CHI | SEA | SF, LA | DEN |
|---|---|---|---|---|---|---|
| NYC, BOS, DC | 0 | | | | | |
| MIA | 1075 | 0 | | | | |
| CHI | 671 | 1329 | 0 | | | |
| SEA | 2684 | 3273 | 2013 | 0 | | |
| SF, LA | 2631 | 2687 | 2054 | 808 | 0 | |
| DEN | 1616 | 2037 | 996 | 1307 | 1059 | 0 |

→ Min[ dis (SF,LA), (NYC,BOS,DC)]

⇒ Min[ dis ( SF, (NYC,BOS,DC)), (LA, (NYC,BOS,DC))]

⇒ Min[ 2799, 2631] = 2631.

→ Min[ dis (SF,LA), MIA]

⇒ Min[ dis (SF,MIA), (LA,MIA)]

⇒ Min[ 3653, 2687] = 2687

→ Min[ dis (SF,CHI), (LA,CHI)]

⇒ Min[ 2142, 2054] = 2054

→ Min[ dis [SF,SEA]), (LA,SEA)]

⇒ Min[ 808, 1131] = 808

→ Min[ dis [SF, DEN), (LA,DEN)]

⇒ Min[ 1235, 1059] = 1059


Step(4):— find min element

671 → (CHI, (NYC, BOS, DC))

| | NYC, BOS, DC, CHI | MIA | SEA | SF, LA | DEN |
|---|---|---|---|---|---|
| NYC, BOS, DC, CHI | 0 | | | | |
| MIA | 1075 | 0 | | | |
| SEA | 2013 | 3278 | 0 | | |
| SF, LA | 2054 | 2687 | 808 | 0 | |
| DEN | 996 | 2037 | 1307 | 1059 | 0 |

→ min [ dis ( (CHI, (NYC, BOS, DC)), MIA)]

⇒ min [ dis ( (CHI, MIA), ((NYC, BOS, DC), MIA)]

⇒ min [ 1329, 1075 ] = 1075

→ min [ dis ( (CHI, SEA), ((NYC, BOS, DC), SEA)]

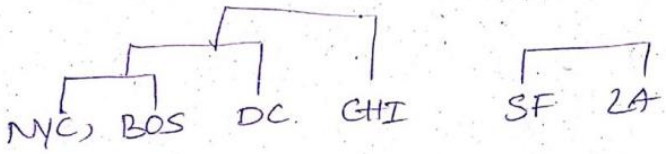
⇒ min [ 2013, 2654 ] = 2013

→ min [ dis ( (CHI, (SF, LA)), (NYC, BOS, DC), (SF, LA)]

⇒ min [ 2054, 2631 ] = 2054

→ min [ dis ( (CHI, DEN), (NYC, BOS, DC), (DEN)]

⇒ min [ 996, 1616 ] = 996




Step(5) → find min element

808 → ((SF, LA), SEA)

| | NYC, BOS, DC, CHI | MIA | SEA, SF, LA | DEN |
|---|---|---|---|---|
| NYC, BOS, DC, CHI | 0 | | | |
| MIA | 1075 | 0 | | |
| SEA, SF, LA | 2013 | 2687 | 0 | |
| DEN | 996 | 2037 | 1059 | 0 |

→ Min[ dis((SF,LA),SEA), (NYC,BOS,DC,CHI)]

⇒ Min[ dis ((SF,LA),(NYC,BOS,DC,CHI)),(SEA, (NYC,BOS,DC,CHI)]

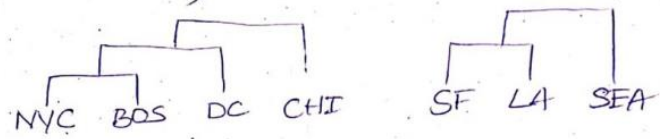
⇒ Min[ 2054, 2013] = 2013

→ Min[ dis ((SF,LA), MIA), (SEA,MIA)]

⇒ Min[ 2687, 3273] = 2687

→ Min[ dis ((SF,LA), DEN), (SEA,DEN)]

⇒ Min[ ~~dis~~ 1059, 807] = 1059




Step(6):- find min element

996 → (DEN, (NYC,BOS,DC,CHI))

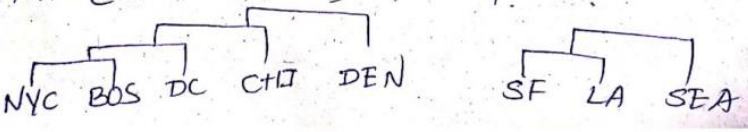
| | N,B,D,C, DEN | MIA | SEA,SF,LA |
|---|---|---|---|
| N,B,DC, DEN | 0 | | |
| MIA | 1075 | 0 | |
| SEA,SF,LA | 1059 | 1059 | 0 |

→ Min[ dis ((DEN, MIA), ((NYC,B,D,CHI), MIA)]

⇒ Min[ 2037, 1075] = 1075

→ Min[ dis( (DEN,(SEA,SF,LA)), (N,B,D,C), (SF,SAN)]

⇒ Min[ 1059, 2013] = 1059



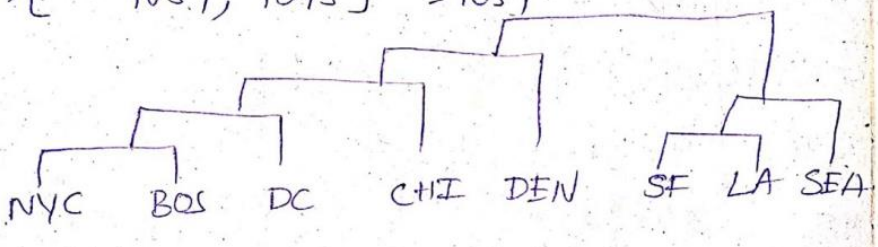
step(4):- find min element

1059 → ((SEA, SF, LA), (NB, D, C, DEN))

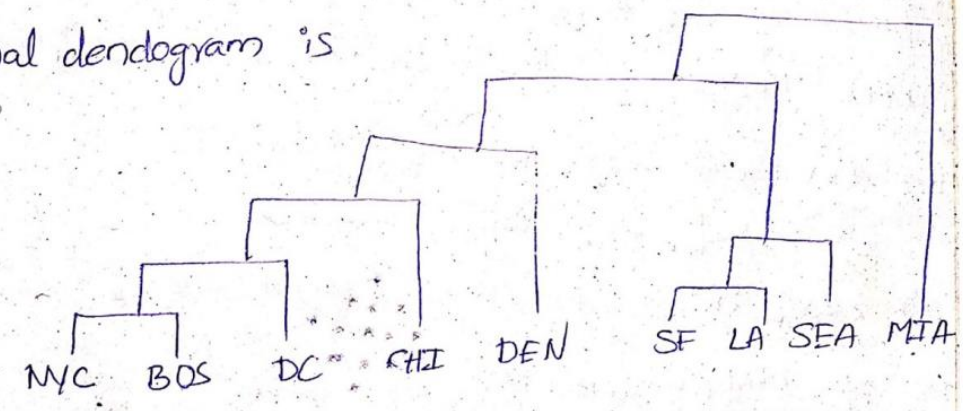
| | NB, D, C, DEN, SEA, SF, LA | MIA |
|---|---|---|
| NB, D, C, DEN, SEA, SF, LA | 0 | |
| MIA | 1059 | 0 |

⇒ min [ dis ( (SEA, SF, LA), MIA), ((N, B, D, C, DEN), MIA)]

⇒ Min [ 1059, 1075] = 1059




final dendogram is



(ii) complete linkage :-

same as single linkage but take max element :

Step(1) :- find max element

3273 → (SEA, MIA).

SEA MIA

| | BOS | NYC | DC | SEA, MIA | CHI | SF | LA | DEN |
|---|---|---|---|---|---|---|---|---|
| BOS | 0 | | | | | | | |
| NYC | 206 | 0 | | | | | | |
| DC | 429 | 233 | 0 | | | | | |
| SEA, MIA | 2976 | 2815 | 2684 | 0 | | | | |
| CHI | 963 | 802 | 671 | 2013 | 0 | | | |
| SF | 3095 | 2934 | 2799 | 3053 | 2142 | 0 | | |
| LA | 2979 | 2786 | 2631 | 2687 | 2054 | 379 | 0 | |
| DEN | 1949 | 1771 | 1616 | 2037 | 996 | 1235 1059 | | 0 |

→ Max [ dis ( SEA, BOS), (MIA, BOS)]

⇒ max [ 2976, 1504] = 2976

→ Max [ dis ( SEA, NYC), (MIA, NYC)]

⇒ Max [ 2815, 1308] = 2815

→ Max [ dis( SEA, DC), (MIA, DC)]

⇒ Max [ 2684, 1075] = 2684

→ Max [ dis ( SEA, SF), (MIA, SF)]

⇒ max [ 808, 3053] = 3053

→ Max [ dis ( SEA, CHI), (MIA, CHI)]

⇒ Max [ 2013, 1329] = 2013

→ Max[ dis( SEA, LA), (MIA, LA)]

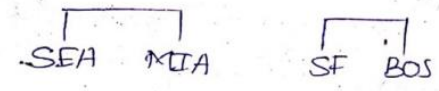
⇒ Max[ 1131, 2687] = 2687

→ Max( dis( SEA, DEN), (MIA, DEN)]

⇒ Max[ 1307, 2037] = 2037.

STEP2:- find max element.

3095 → (SF, BOS)




| | SF, BOS | NYC | DC | SEA, MIA | CHI | LA | DEN |
|---|---|---|---|---|---|---|---|
| SF, BOS | 0 | | | | | | |
| NYC | 2934 | 0 | | | | | |
| DC | 2795 | 233 | 0 | | | | |
| SEA, MIA | 3053 | 2815 | 2684 | 0 | | | |
| CHI | 2142 | 802 | 671 | 2013 | 0 | | |
| LA | 2979 | 2786 | 2631 | 2687 | 2054 | 0 | |
| DEN | 1949 | 1771 | 1616 | 2037 | 996 | 1059 | 0 |

→ Max [dis( SF, NYC), (BOS, NYC)]

⇒ Max[ 2934, 206] = 2934

→ Max[ dis( SF, DC), (BOS, DC)]

⇒ Max( 2799, 429] = 2799

→ Max( dis( SF, (SEA, MIA)), (BOS, (SEA, MIA))]

⇒ Max[ 3053, 2976] = 3053

→ Max[ dis( SF, CHI), (BOS, CHI)]

⇒ Max( 2142, 963] = 2142.

→ Max [ dis ( SF, LA), (BOS, LA)]

⇒ Max [ 379, 2979] = 2979

→ Max [ dis ( SF, DEN), (BOS, DEN)]

⇒ Max [ 1235, 1949] = 1949

Step(3):- find max element

3013 → (SEA, MIA), (SF, BOS)

| | SF, BOS, SEA, MIA | NYC | DC | CHI | LA | DEN |
|---|---|---|---|---|---|---|
| SF, BOS, SEA, MIA | 0 | | | | | |
| NYC | 2934 | 0 | | | | |
| DC | 2799 | 233 | 0 | | | |
| CHI | 2142 | 802 | 671 | 0 | | |
| LA | 2979 | 2786 | 2631 | 2054 | 0 | |
| DEN | 2037 | 1771 | 1616 | 996 | 1059 | 0 |

→ Max [ dis ( (SEA, MIA), NYC), ((SF, BOS), NYC)]

⇒ Max [ 2815, 2934] = 2934

→ Max [ dis ( (SEA, MIA), DC) ((SF, BOS), DC)]

⇒ Max [ 2684, 2799] = 2799

→ Max [ dis ( (SEA, MIA), CHI), ((SF, BOS), CHI)]

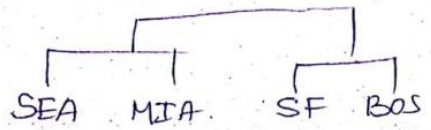
⇒ Max [ 2013, 2142] = 2142

→ Max [ dis ( (SEA, MIA), LA), ((SF, BOS), LA)]

⇒ Max [ 2687, 2979] = 2979

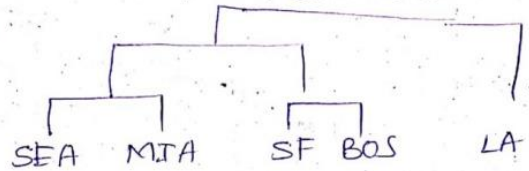
→ Max [ dis((SEA, MIA), DEN) (SF, BOS), DEN)]

⇒ Max[ 2037, 1949] = 2037




step 4):- find maxelement

2979 → (LA, (SEA, MIA, SF, BOS))




| | SF, BOS, SEA, MIA, LA | NYC | DC | CHI | DEN |
|---|---|---|---|---|---|
| SF, BOS, SEA, MIA, LA | 0 | | | | |
| NYC | 2934 | 0 | | | |
| DC | 2799 | 233 | 0 | | |
| CHI | 2142 | 802 | 671 | 0 | |
| DEN | 2037 | 1771 | 1616 | 996 | 0 |

→ Max [dis ( (LA, NYC), ((SEA, MIA, SF, BOS), NYC)]

⇒ Max [ 2786, 2934] = 2934

→ Max [ dis ( (LA, DC), ((SEA, MIA, SF, BOS), DC)]

⇒ Max [ 2631, 2799] = 2799

→ Max [ dis ( LA, CHI), ((SEA, MIA, SF, BOS), CHI)]

⇒ Max [ 2054, 2142] = 2142

→ Max [dis ((LA, DEN), ((SF, BOS, SEA, MIA), DEN)] 12

⇒ Max [ 1059, 2037] = 2037

Step(R5):- find max element

2934 → (NYC, (SF, BOS, SEA, MIA, LA))


| | SF, BOS, SEA, MIA, LA, NYC | DC | CHI | DEN |
|---|---|---|---|---|
| SF, BOS, SEA, MIA, LA, NYC | 0 | | | |
| DC | 2799 | 0 | | |
| CHI | 2142 | 671 | 0 | |
| DEN | 2037 | 1616 | 996 | 0 |

→ Max [ dis ( NYC, DC), [(SF, BOS, SEA, MIA, LA), DC]]

⇒ Max [ 233, 2799] = 2799

→ Max [ dis ( NYC, CHI), ( SF, BOS, SEA, MIA, LA), CHI)]

⇒ Max [ 802, 2142] = 2142

→ Max [ dis ( NYC, DEN), (SF, BOS, SEA, MIA, LA), DEN]

⇒ Max [ 1771, 2037] = 2037.

step(6):- find max element.

2799 → ⊗('DC, (SF, BOS, SEA, MIA, LA, NYC))

| | SF, BOS, SEA, MIA, LA, NYC, DC | CHI | DEN |
|---|---|---|---|
| SF, BOS, SEA, MIA, LA, NYC, DC | 0 | | |
| CHI | 2142 | 0 | |
| DEN | 2037 | 996 | 0 |

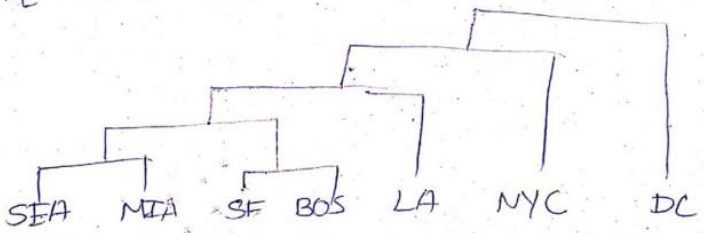
→ Max [ dis ( DC, CHI), ((S, B, S, M, L, N), CHI) ]

⇒ Max [ 671, 2142] = 2142

→ Max [ dis ( DC, DEN), ((S, B, S, M, L, N), DEN)]

⇒ Max [ 1616, 2037] = 2037

step(7):-




step(7):- find maxelement

2142 → (CHI, (S, M, S, B, L, N, D))

| | S, M, S, B, L, N, D, CHI | DEN |
|---|---|---|
| S, M, S, B, L, N, D, CHI | 0 | |
| DEN | 2037 | 0 |