**Topic – 5**

WEATHER

Name :- Akash Mandal

Roll No. :- 16MA20007

Course :- CS40003

Problem Statement –

a) The data pertaining to weather from National Data Centres across the major cities in

India. The data are in PDF form, which can be easily converted to CSV (Comma

Separate Value) format or XLS (Excel Worksheet) format according to your

requirement.

b) Store the data using data cube model.

c) Apply the operation(s) (e.g., slice, dice, roll up drill down) to extract a particular data

(e.g., the data about North-East region) from the data cube you have obtained.

Reference –

WEATHER.pdf (Included in the project folder for reference)

(Converted to WEATHER.xlsx for analysis)

Observation of data –

The database WEATHER.xlsx reveals the following basic structure-

1. 1308 observations of 7 variables.
2. It is a database about mean rainfall and temperature in different months over 1901-2000 in different cities of India.
3. The categorical variables that can be grouped are Station Name and Month, over the measurable data, which is taken to be Mean Rainfall in mm.
4. Thus we consider the above two as dimensions for our two dimensional data cube modelling.

Data cube Model –

1. There are only two categorical variables so we will implement a 2-d cube. And we choose Mean Rainfall as out numeric measurement.

Code Snippet

> q1 = c('January', 'February', 'March')

> q2 = c('April', 'May', 'June')

> q3 = c('July', 'August', 'September')

> WEATHER$Quarter <- ifelse(is.element(WEATHER$Month, q1), 'Q1', ifelse(is.element(WEATHER$Month, q2), 'Q2', ifelse(is.element(WEATHER$Month, q3), 'Q3', 'Q4')))

> cube <- tapply(WEATHER$`Mean Rainfall in mm`, WEATHER[ ,c("Station Name", "Month")], FUN = function(x){return(sum(x))})

Code Snippet

> print(cube)

Month

Station Name April August December February January July June March May November October September

Abu 2.6 600.3 2.4 4.4 5.3 573.2 101.6 6.5 16.4 7.9 19.4 214.2

Agartala (A) 199.7 298.7 10.6 21.5 27.5 363.1 393.4 60.7 329.9 46.0 162.5 232.4

Agra 6.3 243.2 6.1 17.6 13.2 203.3 55.7 9.3 11.3 4.3 24.8 129.7

Ahmedabad 1.9 213.8 1.7 1.2 2.1 309.8 97.4 1.1 9.1 6.1 13.5 126.6

Aijal/Aizwal 167.7 320.6 15.3 23.4 13.4 320.4 406.1 73.4 289.0 43.2 183.7 305.2

Ajmer 4.0 157.5 3.8 6.0 7.3 181.5 58.1 5.0 15.7 4.0 13.1 73.0

Akola (A) 4.1 196.6 12.1 8.1 10.4 217.2 144.9 10.0 9.8 18.7 47.7 122.7

Allahabad 5.5 296.1 7.0 17.3 17.9 280.9 88.8 9.6 8.8 9.3 36.6 185.0

Ambikapur 13.0 389.8 6.5 22.5 22.6 428.3 211.2 18.4 19.4 9.7 56.8 227.2

Amini Divi 22.6 214.0 38.5 2.9 15.9 314.5 371.1 3.8 131.1 88.6 139.2 149.9

Amritsar (Rajasansi) 19.3 174.5 14.6 29.2 28.3 224.7 51.7 34.8 19.6 5.7 21.3 94.6

Anantpur 16.9 75.3 11.2 1.3 0.9 62.7 55.8 4.7 55.0 37.8 104.7 133.2

Slice and Dice-

Dice involves selecting a few variables of each dimension and displaying only that portion of the cube. In our case we can choose a combination of cities and months, to demonstrate the rainfall in different cities for different months.

Slice is a special case, in which we choose only one dimension of the cube, and display all other measurements for that dimension.In our case it will involve choosing a city and displaying data for that city over 12 months, or choosing a month and displaying data of all cities for that month.

Code Snippet

> dice <- function(cube, col = c(), row = c()){

+ if(length(row) & length(col))

+ print(cube[row, col])

+ else if(length(row))

+ print(cube[row, ])

+ else if(length(col))

+ print(cube[, col])

+ else

+ print(cube)

+ }

>

> slice <- function(cube, col='', row=''){

+ if(col != '')

+ dice(cube, col = col)

+ else

+ dice(cube, row = row)

+ }

Code Snippet

> slice(cube, row = 'Abu')

April August December February January July June March

2.6 600.3 2.4 4.4 5.3 573.2 101.6 6.5

May November October September

16.4 7.9 19.4 214.2

> slice(cube, col = 'July')

Abu Agartala (A) Agra

573.2 363.1 203.3

Ahmedabad Aijal/Aizwal Ajmer

309.8 320.4 181.5

Akola (A) Allahabad Ambikapur

217.2 280.9 428.3

Amini Divi Amritsar (Rajasansi) Anantpur

314.5 224.7 62.7

Androth Aurangabad Balasore

486.1 157.9 297.9

Bangalore Bareilly Baroda (A)

109.7 307.0 274.6

Belgaum Samra Bhagalpur Bhatinda

268.9 264.5 133.1

Bhopal (Bairagarh) Bhubaneshwar (A) Bhuj (Rudramata)

354.1 326.2 125.7

Chandigarh Chennai (Minambakkam) Cherrapunji

276.3 99.6 2622.0

Coimbatore (Pilamedu) Cooch Behar Darjeeling

......

Raipur Rajkot (A) Ranchi (A)

369.1 242.5 337.9

Sambalpur Shillong Shimla

459.0 394.5 419.0

Silchar Solapur Sri Niketan

531.3 129.9 317.2

Srinagar Surat Thiruvananthapuram

54.3 440.8 218.6

Tirupathy Tura Udaipur (Dabok)

113.0 628.3 175.1

Uthagamandalam Varanasi (Babatpur) Vijayawada

181.1 309.9 173.0

Vishakhapatnam

142.1

> dice(cube, col = c('June', 'July', 'August'), row = c('Gangtok', 'Pondicherry (A)'))

Month

Station Name June July August

Gangtok 611.3 628.6 563.2

Pondicherry (A) 79.6 61.0 148.2

Roll Up and Drill Down-

Roll up involves aggregating variables of a dimension into higher levels of hierarchy, eg months into quarters, quarters into years, cities into states, states into countries, and so on. In our data, we are suggested to group cities region-wise, but due to the large number of cities, and absence of data in the table, it is not feasible to manually classify the cities into regions. On the other hand, the other dimension, months, can be aggregated into quarters, which is what we have implemented.

Similarly, drill down involves breaking down the higher levels of hierarchy into lower levels, ie. more of details.

Code Snippet

> rolledup = 0

> rollup <- function(cube){

+ if(rolledup == 1)

+ {

+ print("Rolled up Already")

+ return(cube)

+ }

+ rolledup <<- 1

+ return(tapply(WEATHER$`Mean Rainfall in mm`, WEATHER[ ,c("Station Name", "Quarter")], FUN = function(x){return(sum(x))}))

+ }

>

> drilldown <- function(cube){

+ if(rolledup == 0)

+ {

+ print("Drilled down already")

+ return(cube)

+ }

+ rolledup <<- 0

+ return(tapply(WEATHER$`Mean Rainfall in mm`, WEATHER[ ,c("Station Name", "Month")], FUN = function(x){return(sum(x))}))

+ }

Code Snippet

> cube <- drilldown(cube)

[1] "Drilled down already"

> cube <- rollup(cube)

> print(cube)

Quarter

Station Name Q1 Q2 Q3 Q4

Abu 16.2 120.6 1387.7 29.7

Agartala (A) 109.7 923.0 894.2 219.1

Agra 40.1 73.3 576.2 35.2

Ahmedabad 4.4 108.4 650.2 21.3

Aijal/Aizwal 110.2 862.8 946.2 242.2

Ajmer 18.3 77.8 412.0 20.9

Akola (A) 28.5 158.8 536.5 78.5

Allahabad 44.8 103.1 762.0 52.9

. . .

Thiruvananthapuram 81.2 664.0 514.3 537.1

Tirupathy 44.6 129.3 350.7 506.7

Tura 73.1 1162.1 1464.5 241.6

Udaipur (Dabok) 22.4 97.3 459.8 32.0

Uthagamandalam 56.5 359.6 439.7 381.9

Varanasi (Babatpur) 40.5 111.5 829.5 52.3

Vijayawada 15.4 106.4 499.1 115.3

Vishakhapatnam 43.6 218.9 471.3 368.9

> slice(cube, col = 'Q2')

Abu Agartala (A) Agra

120.6 923.0 73.3

Ahmedabad Aijal/Aizwal Ajmer

108.4 862.8 77.8

Akola (A) Allahabad Ambikapur

158.8 103.1 243.6

. . . .

195.2 221.4 664.0

Tirupathy Tura Udaipur (Dabok)

129.3 1162.1 97.3

Uthagamandalam Varanasi (Babatpur) Vijayawada

359.6 111.5 106.4

Vishakhapatnam

218.9

> dice(cube, col = c('Q1', 'Q3'), row = c('Abu', 'Raipur', 'Ludhiana', 'Karnal'))

Quarter

Station Name Q1 Q3

Abu 16.2 1387.7

Raipur 52.3 951.1

Ludhiana 93.4 482.9

Karnal 63.0 509.6

> cube <- rollup(cube)

[1] "Rolled up Already"

> cube <- drilldown(cube)

> print(cube)

Month

Station Name April August December

Abu 2.6 600.3 2.4

Agartala (A) 199.7 298.7 10.6

…

New Delhi (SFD) 10.7 206.5 8.8

Palakkad (Palghat) 72.9 317.6 27.6…

Month

Station Name February January July

Abu 4.4 5.3 573.2

Agartala (A) 21.5 27.5 363.1

…

New Delhi (SFD) 21.0 20.9 198.4

Palakkad (Palghat) 6.7 3.5 522.6…

Month

Station Name June March May

Abu 101.6 6.5 16.4

Agartala (A) 393.4 60.7 329.

…

New Delhi (SFD) 66.3 14.5 14.1

Palakkad (Palghat) 408.2 20.4 151.7…

Month

Station Name November October September

Abu 7.9 19.4 214.2

Agartala (A) 46.0 162.5 232.4

…

New Delhi (SFD) 3.9 20.8 130.3

Palakkad (Palghat) 111.1 215.2 253.9…

Conclusion-

Here we have attempted a very basic implementation of a data cube. While this implementation may be restricted by my limited knowledge of R, it aims to portray the basics of the data cube model, and the simple operations to be performed on it.