library("tidyr")

library("ggplot2")

# for replicating same sequence

set.seed(123)

jobs\_data <- read.csv("C:/Users/ASHISH/OneDrive/Desktop/jobs\_in\_data.csv")

# Copy the dataset

jobs\_data\_copy <- jobs\_data

#check null values

sum(is.na(jobs\_data\_copy))

#### since there are no null values there is no need to do the drop operation

# remove null values

# jobs\_data\_copy <- drop\_na(jobs\_data\_copy)

jobs\_data\_copy$experience\_level <- as.numeric(as.factor(jobs\_data\_copy$experience\_level))

jobs\_data\_copy$employment\_type <- as.numeric(as.factor(jobs\_data\_copy$employment\_type))

jobs\_data\_copy$work\_setting <- as.numeric(as.factor(jobs\_data\_copy$work\_setting))

jobs\_data\_copy$company\_size <- as.numeric(as.factor(jobs\_data\_copy$company\_size))

# Calculating no of rows in the dataset

n <- nrow(jobs\_data\_copy)

"

Doing data sampling randomly. Since we have a lot of observations i.e 9400 approx.

processing these many will take a lot of time so sampling the data considering it

it represents the whole population may help us gain similar results.Here I am doing

sampling in such a way that the probability of selecting each sample is same as the other.

"

sampled\_rows <- sample(1:n, 170, replace = FALSE, prob = NULL)

jobs\_data\_copy <- jobs\_data\_copy[sampled\_rows,]

jobs\_data\_copy

# Selecting only certain columns

column\_data <- c("experience\_level","employment\_type","work\_setting","company\_size","salary")

clustering\_data <- jobs\_data\_copy[,column\_data]

# View(clustering\_data)

# Scaling the data

scaled\_data <- scale(clustering\_data)

# View(scaled\_data)

# Applying clustering

"

I am using clara cluster because all the other clustering methods formed were not informative,

clusters formed by pam gave me contradicting results on work\_setting and salary.

K-means cluster was not able to form better cluster.

Whereas clara formed better clusters when compared.

The number clusters formed where 3 because considering the no of clusters to be

formed 2 or greater than 4 did not give me good clusters,

that is they were separated well when the no of clusters assumed 3.

"

clustering\_jobs\_data <-cluster::clara(

x = scaled\_data,

k = 3

)

# pc\_d <- prcomp(scaled\_data)

# pc\_d$x

clustering\_jobs\_data

Call: cluster::clara(x = scaled\_data, k = 3)

Medoids:

experience\_level employment\_type work\_setting company\_size salary

195 -0.6822341 0.0766965 -0.7411217 0.1907096 -0.7934629

8508 0.5258888 0.0766965 1.1678281 0.1907096 0.4246162

5603 0.5258888 0.0766965 -0.7411217 0.1907096 -0.1167523

Objective function: 1.247353

Clustering vector: Named int [1:170] 1 2 2 2 3 3 2 3 3 3 1 2 1 3 1 3 3 2 ...

- attr(\*, "names")= chr [1:170] "2463" "2511" "8718" "2986" "1842" "9334" "3371" ...

Cluster sizes: 37 60 73

Best sample:

[1] 2463 2511 1842 2567 1614 6216 8780 3937 2907 8508 6134 6553 9198 5603 7816 4044 5027

[18] 6387 9175 2208 1029 8011 5884 6644 9260 2507 195 6098 1149 7741 8650 5015 6379 1313

[35] 185 564 3799 3581 6601 4713 618 9148 3625 5618 6815 6801

Available components:

[1] "sample" "medoids" "i.med" "clustering" "objective" "clusinfo"

[7] "diss" "call" "silinfo" "data"

pca\_result <- prcomp(scaled\_data, scale = TRUE)

# Extract PCA scores

pca\_scores <- as.data.frame(pca\_result$x[, 1:2]) # Extract the first two principal components

# Add cluster assignments to PCA scores

pca\_scores$cluster <- as.factor(clustering\_jobs\_data$clustering)

# Plot the clusters

ggplot(pca\_scores, aes(x = PC1, y = PC2, color = cluster)) +

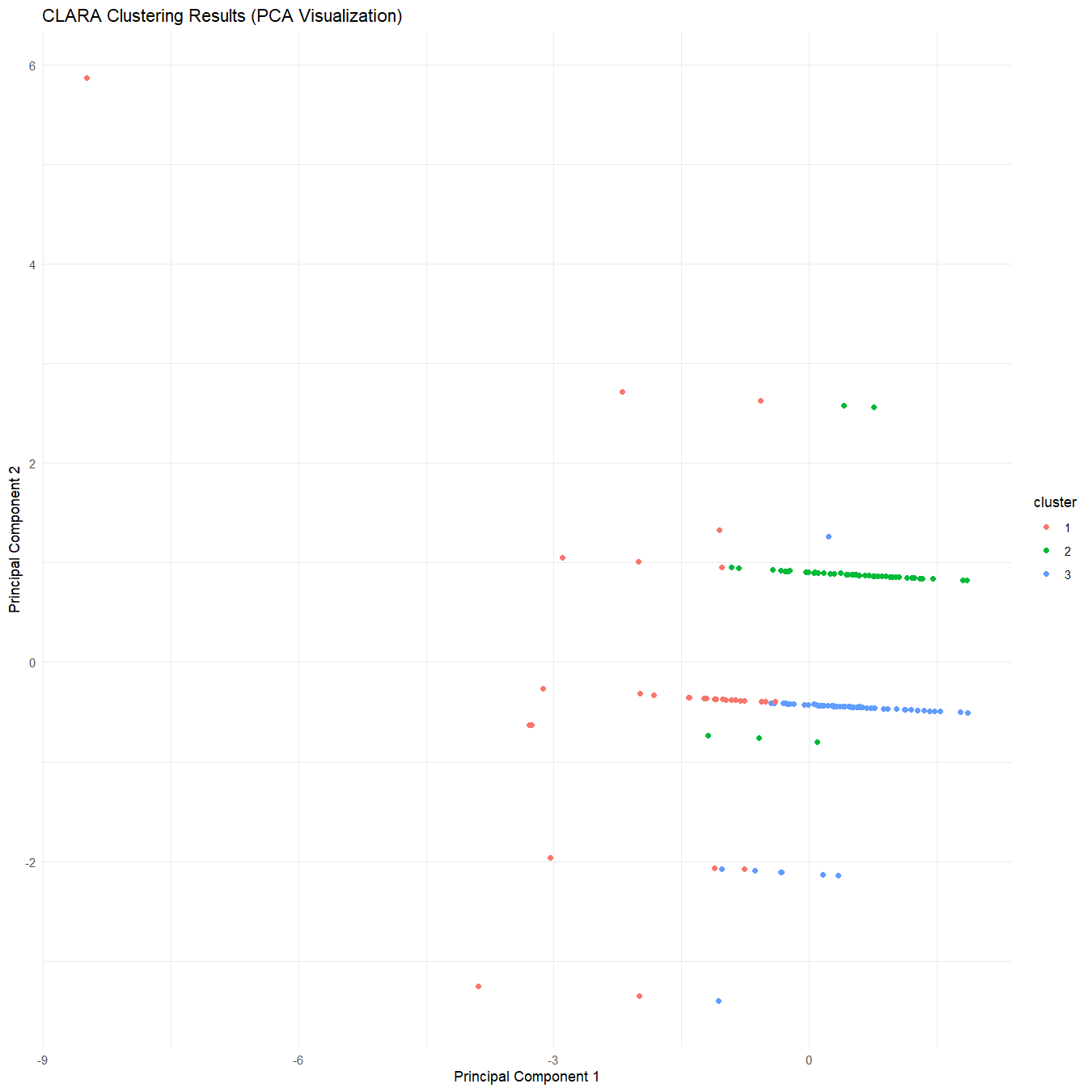
geom\_point() +

labs(title = "CLARA Clustering Results (PCA Visualization)",

x = "Principal Component 1",

y = "Principal Component 2") +

theme\_minimal()



dist\_M <- dist(x=scaled\_data)

"I choose hclust and diana for hierarchial clustering since the dedongrams

formed by these clusterings gave me a better visual interpretation of which

data points were unique i.e lie alone in the tree. hclust gave me good grouping

for ward.D methoda and diana cluster gave me better hierarchial for average metric."

hclust\_M <- hclust(

d = dist\_M,

method = "ward.D"

)

print(hclust)

Call:

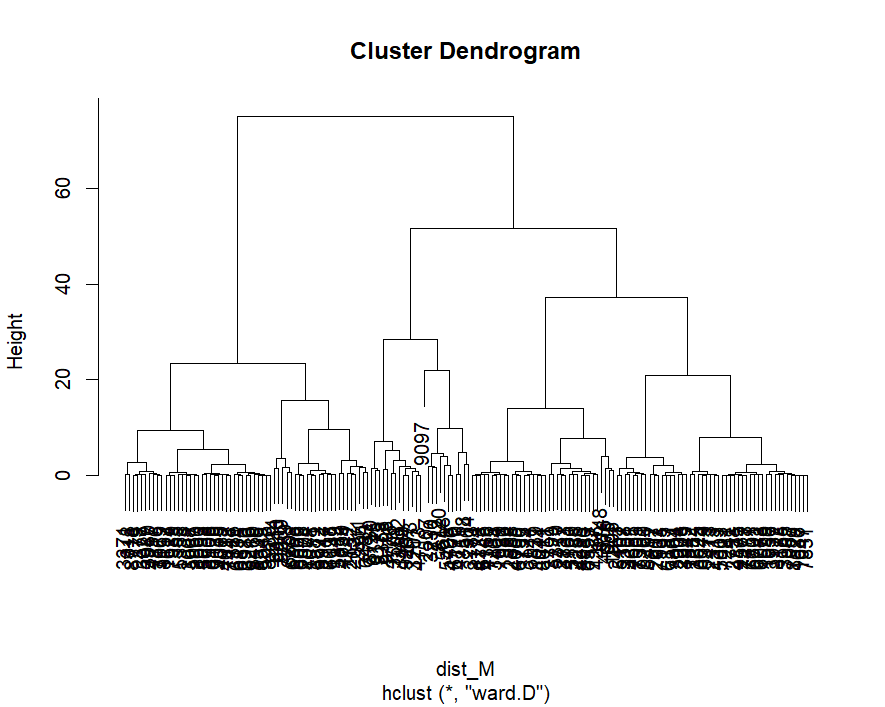
hclust(d = dist\_M, method = "ward.D")

Cluster method : ward.D

Distance : euclidean

Number of objects: 170

plot(x=hclust\_M)



hclust\_M <- cluster::diana(

dist\_M,

metric = "average"

)

print(hclust\_M)

Merge:

[,1] [,2]

[1,] -26 -150

[2,] -141 -170

[3,] -51 -88

[4,] -53 -167

[5,] -106 -151

[6,] -4 -32

[7,] -3 -152

[8,] -72 -90

[9,] -54 -110

[10,] -89 -114

[11,] -83 -140

[12,] -55 -99

[13,] -17 12

[14,] -65 -73

[15,] -23 14

[16,] -62 -78

[17,] -129 -136

[18,] -30 -38

[19,] -109 -138

[20,] -102 19

[21,] -76 20

[22,] -56 -70

[23,] -124 -154

[24,] 15 -160

[25,] -41 16

[26,] -132 -147

[27,] -67 -122

[28,] -10 -118

[29,] 5 -133

[30,] -50 -77

[31,] -16 -107

[32,] 25 -117

[33,] -98 -100

[34,] -18 -27

[35,] -1 -11

[36,] -64 11

[37,] -20 -68

[38,] -49 -111

[39,] -66 -131

[40,] 29 -144

[41,] 31 -142

[42,] 8 -161

[43,] 24 -24

[44,] -104 -135

[45,] -101 -168

[46,] -59 -156

[47,] -42 30

[48,] -92 -137

[49,] -19 -93

[50,] 34 -31

[51,] -116 -149

[52,] -86 33

[53,] -40 40

[54,] -35 27

[55,] -9 -153

[56,] 41 32

[57,] -15 -80

[58,] -71 -75

[59,] -43 -105

[60,] -44 -48

[61,] -5 -45

[62,] 4 -69

[63,] 39 2

[64,] 38 -164

[65,] -14 23

[66,] 44 17

[67,] -34 -91

[68,] -46 26

[69,] -7 -158

[70,] -85 -115

[71,] 47 3

[72,] 36 10

[73,] -36 48

[74,] -95 -134

[75,] 18 -155

[76,] 9 42

[77,] 56 -74

[78,] -47 51

[79,] -148 -157

[80,] 43 -28

[81,] -52 -126

[82,] 28 -146

[83,] 13 -58

[84,] 6 53

[85,] 55 -84

[86,] 7 58

[87,] 66 -163

[88,] 49 63

[89,] -81 -145

[90,] 60 -127

[91,] -37 71

[92,] -128 -130

[93,] 59 21

[94,] -25 70

[95,] 72 52

[96,] -8 79

[97,] 87 -139

[98,] 46 45

[99,] 80 73

[100,] 65 -120

[101,] 84 62

[102,] 77 78

[103,] 50 88

[104,] 86 54

[105,] -103 -121

[106,] 91 74

[107,] -113 92

[108,] -94 -112

[109,] 85 95

[110,] 90 81

[111,] 82 83

[112,] 94 93

[113,] 61 97

[114,] -39 -97

[115,] 104 -60

[116,] 100 76

[117,] 37 1

[118,] 35 98

[119,] 102 99

[120,] 103 68

[121,] -21 -169

[122,] 101 106

[123,] 96 111

[124,] 113 67

[125,] 110 64

[126,] 112 75

[127,] -96 -159

[128,] 109 116

[129,] -12 -79

[130,] 115 69

[131,] 117 -125

[132,] -29 -57

[133,] 123 89

[134,] -2 22

[135,] -63 127

[136,] 108 107

[137,] 124 119

[138,] 129 -33

[139,] -6 -82

[140,] 122 114

[141,] 120 125

[142,] -61 -119

[143,] -13 121

[144,] 118 126

[145,] 131 -165

[146,] 141 136

[147,] 142 -108

[148,] 57 -143

[149,] 130 140

[150,] 144 137

[151,] 145 105

[152,] 134 -166

[153,] 133 128

[154,] 139 143

[155,] 153 135

[156,] 138 151

[157,] 152 147

[158,] 157 -123

[159,] 149 146

[160,] 148 -87

[161,] 132 -162

[162,] 156 161

[163,] 150 155

[164,] 163 159

[165,] 164 154

[166,] 158 160

[167,] 165 162

[168,] 167 166

[169,] 168 -22

Order of objects:

[1] 2463 5107 3358 3201 4685 7005 3004 2208 3949 7391 9260 6387 7864 195 7933 217 2907

[18] 6491 1842 7284 6644 6379 2313 1313 3625 413 1599 1956 2567 7478 4875 41 5603 7281

[35] 2758 6078 6742 7757 6601 1047 7831 686 8529 7067 3995 3937 8172 185 4761 3129 2266

[52] 2757 3833 4256 2980 6553 4233 7127 151 3799 6746 4366 6911 3783 8174 4723 7735 3464

[69] 1029 986 1584 2888 4055 6790 5370 755 3124 4044 8011 539 4576 985 1905 8718 279

[86] 2378 5027 4237 7816 3069 3980 3371 618 2986 6216 8469 8536 8549 5015 8986 6134 6815

[103] 1386 4089 8508 6129 9039 4612 7448 4093 5658 294 6183 1614 7989 8157 555 5358 5967

[120] 8650 1333 3008 2504 1078 3581 6672 8944 1149 2117 6098 8566 6678 6868 5884 4650 2132

[137] 2037 7741 9334 6810 9209 7789 6801 9145 9175 8780 4469 9267 3207 4713 473 3462 4776

[154] 3501 8358 9198 9148 2511 5428 4706 5618 3230 712 2507 8720 6170 4715 564 8518 9097

Height:

[1] 1.353421e-02 5.464438e-01 3.383553e-02 2.283898e-01 2.537665e-02 1.536133e+00

[7] 1.776365e-01 8.458883e-02 4.364783e-01 5.836629e-02 1.742530e-01 0.000000e+00

[13] 0.000000e+00 0.000000e+00 8.509636e-01 0.000000e+00 8.458883e-02 2.159468e+00

[19] 6.767106e-02 4.567797e-01 2.537665e-02 7.612994e-02 0.000000e+00 1.370339e-01

[25] 2.216227e-01 7.367687e-01 7.697583e-02 1.471846e+00 1.015066e-02 2.030132e-02

[31] 4.736974e-02 5.075330e-03 0.000000e+00 1.015066e-02 9.473949e-02 2.639171e-01

[37] 1.002208e-01 4.229441e-02 5.752040e-01 0.000000e+00 0.000000e+00 4.060264e-03

[43] 2.097803e-02 1.015066e-01 2.368487e-01 8.458883e-02 3.383553e-02 4.403486e+00

[49] 2.106262e-01 1.015066e-01 7.012414e-01 8.458883e-03 1.218079e-01 3.806497e-01

[55] 0.000000e+00 0.000000e+00 1.268832e-01 1.260374e+00 1.556434e-01 2.371025e+00

[61] 4.736974e-02 1.319586e-01 3.493519e-01 1.691777e-02 0.000000e+00 8.458883e-02

[67] 0.000000e+00 1.801742e-01 4.314030e-02 1.099655e-02 9.414736e-01 7.612994e-02

[73] 2.706842e-03 2.537665e-01 5.126083e-01 0.000000e+00 8.966416e-02 0.000000e+00

[79] 2.030132e-02 2.695708e+00 1.328057e+00 8.898745e-01 4.511065e+00 0.000000e+00

[85] 1.353421e-01 5.075330e-02 2.994444e-01 4.567797e-02 8.458883e-03 5.075330e-01

[91] 1.184244e+00 7.951350e-02 2.078862e+00 0.000000e+00 1.285750e-01 4.398619e-02

[97] 0.000000e+00 8.881827e-03 1.903249e-02 2.537665e-01 0.000000e+00 6.767106e-02

[103] 6.767106e-01 1.691777e-01 3.383553e-02 9.304771e-03 8.458883e-02 0.000000e+00

[109] 3.383553e-01 8.458883e-02 1.474818e+00 5.075330e-01 3.752527e+00 1.268832e-02

[115] 3.383553e-02 2.706842e-01 3.383553e-02 1.522599e-01 1.860954e-02 6.767106e-02

[121] 0.000000e+00 5.885183e-01 7.760179e-02 7.156215e-03 1.488763e+00 5.887382e-02

[127] 1.637640e-01 3.600100e-01 1.116573e-01 7.714501e-01 1.691777e-02 6.767106e-02

[133] 1.865225e+00 3.383553e-01 1.353421e+00 3.383553e-01 1.691777e-01 5.503736e+00

[139] 1.474818e+00 2.677970e+00 1.524909e+00 5.921218e-01 6.852078e+00 1.015066e+00

[145] 1.474818e+00 2.865523e+00 1.691777e-02 5.362932e-01 0.000000e+00 1.187627e+00

[151] 1.694113e+00 2.317734e+00 3.011362e-01 4.314971e+00 1.234487e+00 4.314971e+00

[157] 7.670335e+00 1.310394e+00 1.691777e-05 2.362413e+00 3.094462e+00 1.499476e+00

[163] 1.945543e+00 3.677830e+00 6.626616e+00 5.075330e-02 2.042864e+00 3.845127e+00

[169] 1.441257e+01

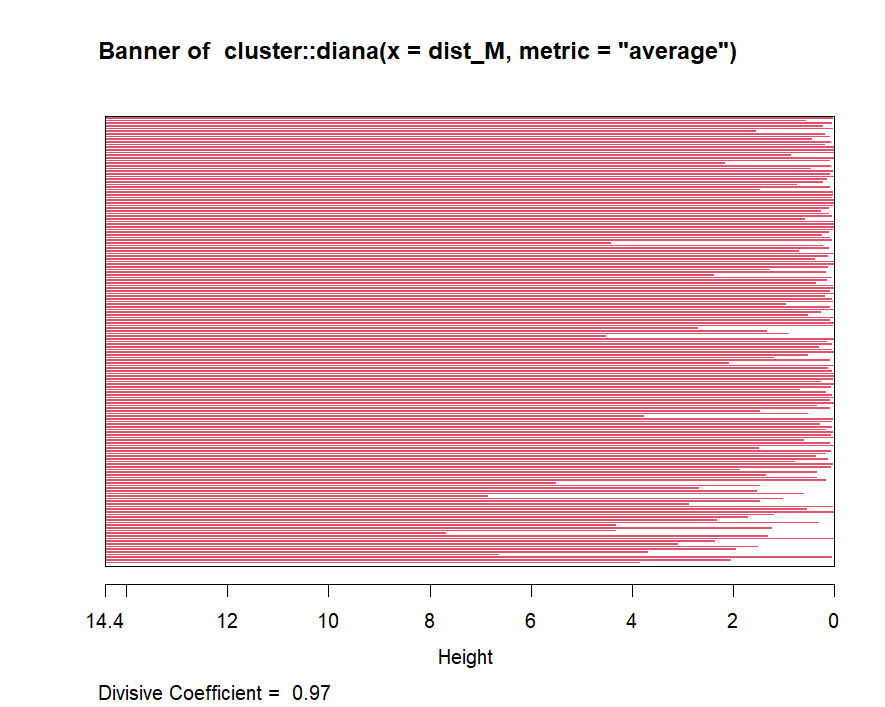
Divisive coefficient:

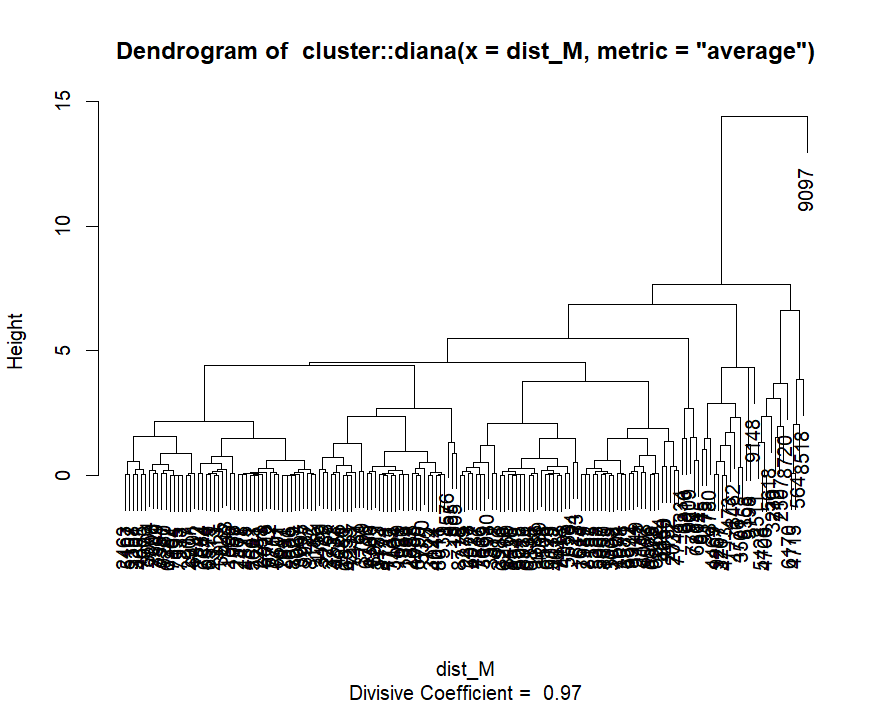
[1] 0.9740733

Available components:

[1] "order" "height" "dc" "merge" "diss" "call" "order.lab"

plot(x = hclust\_M)



\