# cgmanalyserDTW

12/12/2019

Code ▼

```
m1 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID01.csv')
m2 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID02.csv')
m3 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID03.csv')
m4 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID11.csv')
m5 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID12.csv')
m6 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID13.csv')
m7 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID21.csv')
m8 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID22.csv')
m9 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID23.csv')
m10 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID29.csv')
m11 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID30.csv')
m12 = read.csv('~/Desktop/NCSA/CGManalyzer-datasets/ID31.csv')
training_frame = rbind.data.frame(
                            m1$glucoseValue,
                            m2$glucoseValue,
                            m3$glucoseValue,
                            m4$glucoseValue,
                            m5$glucoseValue,
                            m6$glucoseValue,
                            m7$qlucoseValue,
                            m8$glucoseValue,
                            m9$glucoseValue,
                            m10$glucoseValue,
                            m11$glucoseValue,
                            m12$glucoseValue
                            )
training frame
```

c.00000000001 <dbl></dbl>	<b>c.0000000000</b> <dbl></dbl>
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0

Hide

```
require(dtwclust)
```

```
Loading required package: dtwclust
Loading required package: proxy
Attaching package: 'proxy'
The following objects are masked from 'package:stats':
    as.dist, dist
The following object is masked from 'package:base':
    as.matrix
Loading required package: dtw
Loaded dtw v1.21-3. See ?dtw for help, citation("dtw") for use in publication.
Registered S3 method overwritten by 'dplyr':
 method
                   from
 print.rowwise df
Registered S3 methods overwritten by 'htmltools':
 method
                       from
 print.html
                       tools:rstudio
 print.shiny.tag
                       tools:rstudio
 print.shiny.tag.list tools:rstudio
dtwclust:
Setting random number generator to L'Ecuyer-CMRG (see RNGkind()).
To read the included vignettes type: browseVignettes("dtwclust").
See news(package = "dtwclust") after package updates.
```

Hide

```
require(mcclust)
```

```
Loading required package: mcclust
Loading required package: lpSolve
```

```
clust.hier <- tsclust(training_frame, type = "h", k = 4L, distance = "dtw2", trace=TRUE,
control = hierarchical_control(method = "ward.D"))</pre>
```

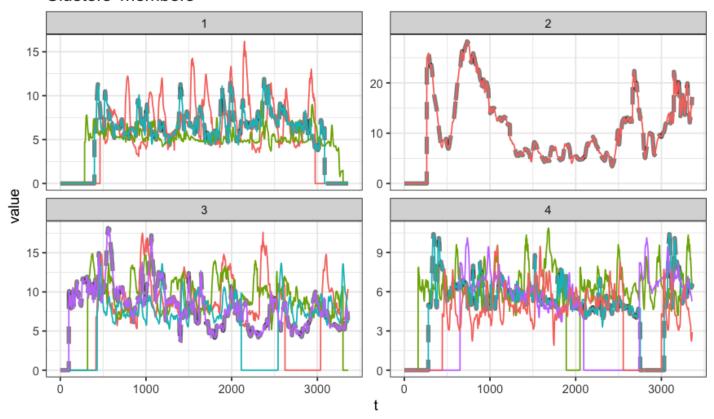
```
Calculating distance matrix...
Performing hierarchical clustering...
Extracting centroids...
```

Elapsed time is 105.74 seconds.

Hide

plot(clust.hier, type="sc")

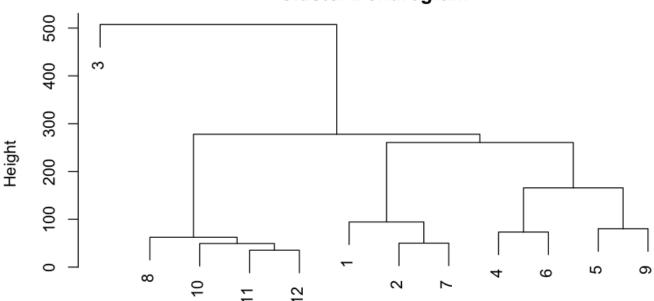
### Clusters' members



Hide

plot(clust.hier)





stats::as.dist(distmat) stats::hclust (\*, "ward.D")

```
Hide
```

```
t(cbind(training_frame[,0], cluster = clust.hier@cluster))
```

```
1 2 3 4 5 6 7 8 9 10 11 12
cluster 1 1 2 3 3 3 1 4 3 4 4 4
```

Hide

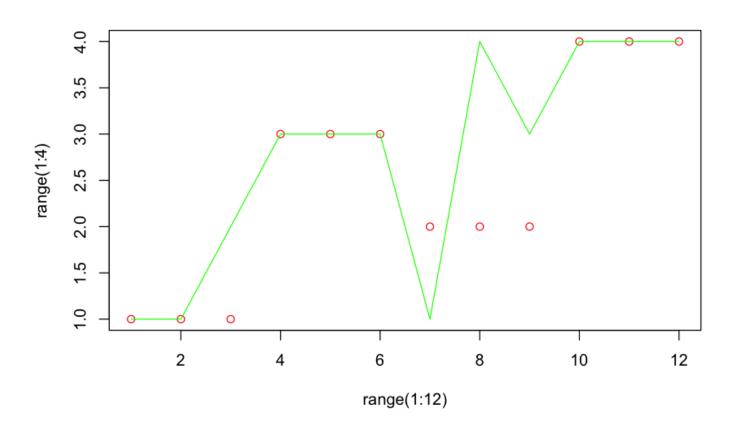
```
l_hier <- clust.hier@cluster</pre>
```

```
m_hier <- c(1,1,1,3,3,3,2,2,2,4,4,4)
```

Hide

```
plot(range(1:12),range(1:4), type='n')
points(m_hier, col='red')
```

```
lines(l hier, col='green')
```



Hide

predict(clust.hier,newdata=unlist(m3\$glucoseValue))

[1] 2

Hide

predict(clust.hier,newdata=unlist(m6\$glucoseValue))

[1] 3

Hide

predict(clust.hier,newdata=unlist(m9\$glucoseValue))

[1] 3

Hide

predict(clust.hier,newdata=unlist(m9\$glucoseValue))

[1] 4

```
clust.par <- tsclust(training_frame, type = "partitional", k = 4L, distance = "dtw2", tr
ace=TRUE)
```

```
Precomputing distance matrix...

Iteration 1: Changes / Distsum = 12 / 611.5245

Iteration 2: Changes / Distsum = 1 / 545.6121

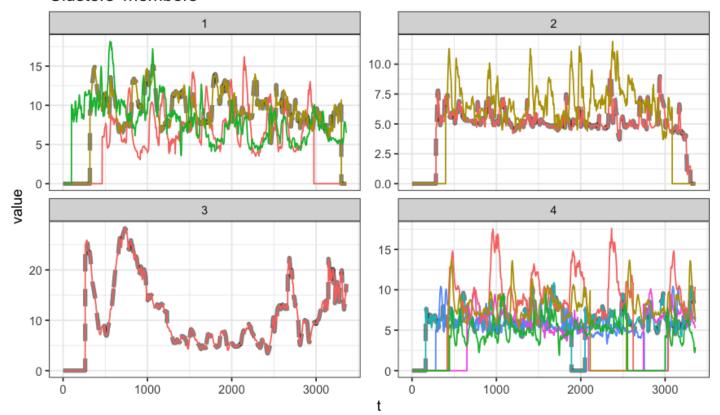
Iteration 3: Changes / Distsum = 0 / 545.6121

Elapsed time is 62.168 seconds.
```

Hide

```
plot(clust.par, type="sc")
```

#### Clusters' members



Hide

t(cbind(training\_frame[,0], cluster = clust.par@cluster))

```
[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] cluster 1 2 3 4 1 4 2 4 1 4 4 4
```

```
l_par <- clust.par@cluster

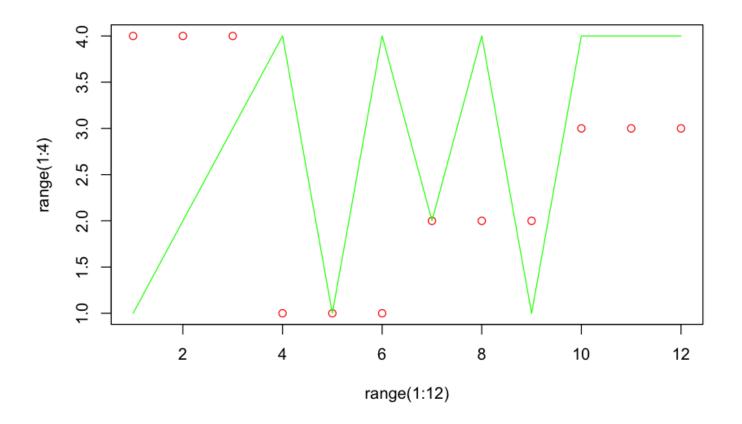
m_par <- c(4,4,4,1,1,1,2,2,2,3,3,3)</pre>
```

Hide

```
plot(range(1:12),range(1:4), type='n')
points(m_par, col='red')
```

Hide

lines(l\_par, col='green')



Hide

predict(clust.par,newdata=unlist(m3\$glucoseValue))

[1] 3

Hide

predict(clust.par,newdata=unlist(m6\$glucoseValue))

[1] 1

```
Hide
```

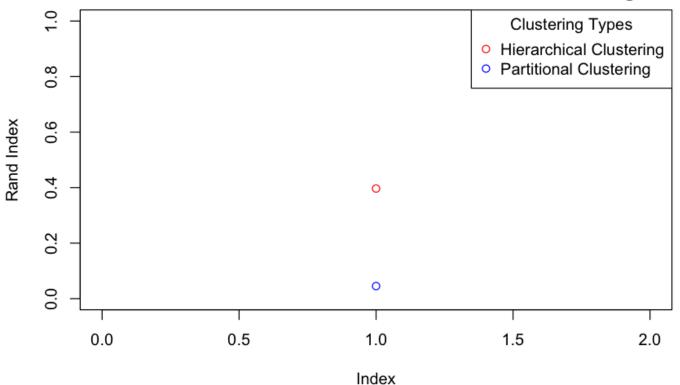
```
predict(clust.par,newdata=unlist(m9$glucoseValue))
[1] 1
                                                                                         Hide
predict(clust.par,newdata=unlist(m12$glucoseValue))
[1] 2
                                                                                         Hide
#predicted_labels
#test_labels
index_hier=arandi(l_hier,m_hier)
unadjusted_hier=arandi(l_hier,m_hier,adjust=FALSE)
index_hier
[1] 0.3966245
                                                                                         Hide
unadjusted_hier
[1] 0.8030303
                                                                                         Hide
index_par=arandi(l_par,m_par)
unadjusted_par=arandi(l_par,m_par,adjust=FALSE)
index_par
[1] 0.04528302
                                                                                         Hide
unadjusted_par
[1] 0.6515152
                                                                                         Hide
colors = c('red','blue')
types = c('Hierarchical Clustering','Partitional Clustering')
```

```
plot(NULL, NULL, type='n', ylim=c(0,1), xlim=c(0,2), xlab='Index', ylab='Rand Index', main
='Random Index of Hierarchical and Partitional Clustering')
legend("topright", legend = types, col = colors, title = 'Clustering Types', pch=1)
```

par(new=TRUE)
points(index\_hier, col='red')

points(index\_par, col='blue')

## Random Index of Hierarchical and Partitional Clustering



Hide

Hide

Hide

NA NA

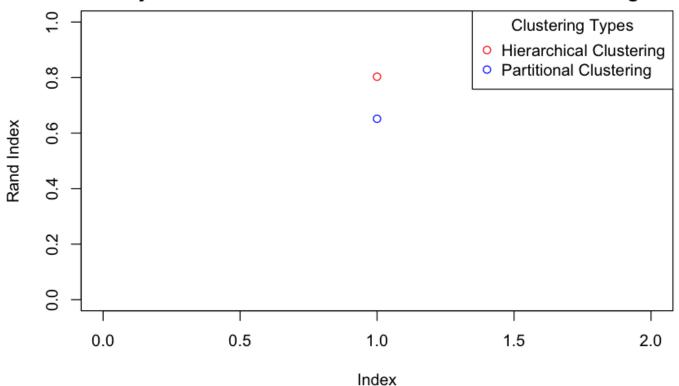
```
plot(NULL,NULL, type='n',ylim=c(0,1), xlim=c(0,2), xlab='Index', ylab='Rand Index', main
='Adjusted Index of Hierarchical and Partitional Clustering')
legend("topright",legend = types, col = colors, title = 'Clustering Types', pch=1)
```

par(new=TRUE)

points(unadjusted\_hier, col='red')

points(unadjusted\_par,col='blue')

## Adjusted Index of Hierarchical and Partitional Clustering



Hide

Hide

Hide

NA NA