Comparison and evaluation of clustering algorithms for tandem mass spectra

Vera Rieder, Karin U. Schork, Laura Kerschke, Bernhard Blank-Landeshammer, Albert Sickmann and Jörg Rahnenführer

Description

This code is designed to perform a cluster analysis of tandem mass spectra. We use the following algorithms for clustering:

- CAST
- DBSCAN
- hclust
- igraph
- N-Cluster

MS-Cluster and PRIDE Cluster are not part of this example because a shell with installations is needed to run the command line tools with R wrappers.

Evaluation of clustering is performed by these measures:

Measure	R function
Adjusted Rand Index	`adjustedRandIndex`
Retainment of identified peptides	`Ret_id_spec`
Proportion of spectra remaining	`Spec_rem`
Proportion clustered spectra	`Prop_clu_spec`
Proportion incorrectly clustered spectra	`Prop_inc_spec`
Purity	`Purity`

R code

In this section all needed functions are sourced.

```r
source(file.path("https://www.statistik.tu-dortmund.de", "fileadmin", "user_upload",
    "Lehrstuehle", "Genetik", "Forschung", "cluster-spectra.R"))
```

Read the data

In this section two vectors, d and annotation_d, are loaded. Numeric vector d is a distance vector for a subset of 2000 spectra (MS/MS spectra 10001-12000) of MS/MS run H1. Character vector annotation_d contains the corresponding peptide annotations. For the following analysis unannotated spectra are excluded.

```r
# data-cluster-spectra.RData contains distance vector d and
# annotation vector annotation_d for a subset of 2000 spectra
# of MS/MS run H1.

load(url(file.path("https://www.statistik.tu-dortmund.de", "fileadmin", "user_upload",
                 "Lehrstuehle", "Genetik", "Forschung","data-cluster-spectra.RData")))

# exclude unannotated spectra
```

```
na_id = which(is.na(annotation_d))
x = as.numeric(as.factor(annotation_d[-na_id]))
```

Cluster Analysis

Use functions to cluster 2000 spectra. Parameters correspond to optimal settings in Table 3.

```
set.seed(123)
cast_d = cast(S = 1 - d,
              thres = 0.9)

dbscan_d = DBSCAN(S = 1 - d,
                  eps = 0.05,
                  minPts = 2)

hclust_d = hclust2(S = 1 - d,
                   h = 0.1)

igraph_d = igraph(S = 1 - d,
                  csim = 0.95)

neighbor_d = neighbourClust(S = 1 - d,
                            c = 0.05)
```

Evaluation of clustering algorithms

Exemplary clustering results are similar to results in the manuscript. The ARI of all algorithms compared to the annotation is higher than 0.6. Further results are listed below.

```
ARI_tab = data.frame(CAST = adjustedRandIndex(cast_d[-na_id], x),
                     DBSCAN = adjustedRandIndex(dbscan_d[-na_id], x),
                     hclust = adjustedRandIndex(hclust_d[-na_id], x),
                     igraph = adjustedRandIndex(igraph_d[-na_id], x),
                     N_Cluster = adjustedRandIndex(neighbor_d[-na_id], x))

round(ARI_tab, 4)
```

```
##     CAST DBSCAN hclust igraph N_Cluster
## 1 0.6225 0.6135 0.6327 0.6135    0.6135
```

```
clu = cast_d[-na_id]

Prop_clu_spec(clu = clu)
```

```
## [1] 0.2421827
```

```
Prop_inc_spec(clu = clu, ann = x)
```

```
## [1] 0.01839362
```

```
Ret_id_spec(clu = clu, ann = x)
```

```
## [1] 0.9787903
```

```
Spec_rem(clu = clu)
```

```
## [1] 0.8571429
```

```
ids = unique(clu)
fre = sapply(ids, function(x) sum(clu == x))
ids1 = ids[fre == 1]
ids2 = ids[fre >= 2 & fre <= 3]
ids3 = ids[fre >= 4 & fre <= 7]
ids4 = ids[fre >= 8 & fre <= 15]

purity_ids1 = sapply(ids1, function(j) Purity(j = j,
                                              clu = clu,
                                              ann = annotation_d[-na_id]))

purity_ids2 = sapply(ids2, function(j) Purity(j = j,
                                              clu = clu,
                                              ann = annotation_d[-na_id]))

purity_ids3 = sapply(ids3, function(j) Purity(j = j,
                                              clu = clu,
                                              ann = annotation_d[-na_id]))

purity_ids4 = sapply(ids4, function(j) Purity(j = j,
                                              clu = clu,
                                              ann = annotation_d[-na_id]))

Ave_purity = data.frame("size_1" = mean(purity_ids1),
                        "size_2_to_3" = mean(purity_ids2),
                        "size_4_to_7" = mean(purity_ids3),
                        "size_8_to_15" = mean(purity_ids4))

round(Ave_purity, 4)
```

```
##   size_1 size_2_to_3 size_4_to_7 size_8_to_15
## 1      1      0.9589      0.7746            1
```

Technical reproducibility information

```
sessionInfo()
```

```
## R version 3.3.1 (2016-06-21)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## locale:
## [1] LC_COLLATE=German_Germany.1252  LC_CTYPE=German_Germany.1252
## [3] LC_MONETARY=German_Germany.1252 LC_NUMERIC=C
## [5] LC_TIME=German_Germany.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
```

```
## other attached packages:
## [1] stringr_1.2.0   igraph_1.0.1    clusteval_0.1   checkmate_1.8.2
## [5] BBmisc_1.11
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.10     codetools_0.2-15 mvtnorm_1.0-6    digest_0.6.12
##  [5] rprojroot_1.2    backports_1.0.5  formatR_1.5      magrittr_1.5
##  [9] evaluate_0.10    stringi_1.1.5    rmarkdown_1.5    tools_3.3.1
## [13] parallel_3.3.1   yaml_2.1.14      htmltools_0.3.6  knitr_1.15.1
```