R SDisc: Integrated methodology for the identification of homogeneous profiles in data distribution

F Colas

December 10, 2009

R SDisc is integrated set of tools and methods to identify homogeneous profiles/subtypes in data distribution by cluster analysis. It includes methods for data treatment and preprocessing, repeated cluster analysis, model selection, model reliability and reproducibility assessment, profiles characterization and validation by visual and table summaries. It applies particularly to the search for more homogeneous profiles in cohort studies.

This Vignette is an interactive documentation on the R SDisc package. The first part referred to as Hands on R SDisc, describes step by step with the help of several examples, how to carry an SDisc analysis. The second part referred to as About subtype discovery analysis with SDisc, presents different instances of research searching for more homogeneous patient profiles, an analysis use case, the rationale of the SDisc package, and the orientation of our ongoing developments around the SDisc package. In the last section of part 2, we point you to several important links with respect to subtype analyses and SDisc.

apply: clinical heterogeneity, complex diseases, patient profiles, complex interactions, phenotypes
infer: validate, evaluate, reproduce, κ, χ^2-association testing, odd ratios, rank, stability
analyse: cluster, mixture model, EM, hierarchical clustering, exploratory data analysis, data transform, repeat, characterize, compare, visualize

Contents

Introduction

The time and the expertise to perform robust subtyping inferences in data are often regarded as limiting factors for the range of analysis hypothesis considered. Indeed, not only competence in cluster analysis is required but also in exploratory data analysis, regression, statistical testing, computational statistics, classifier training and testing, data visualization and scientific programming. Identifying data subtypes is therefore greatly interdisciplinary. Hence, SDisc addresses an essential demand, originally emanating from clinical research, for an integrated scenario performing the different steps of a subtyping analysis.

With SDisc, analyzes also become more straightforward and therefore more accessible to many investigators. The well-defined data structures of the package greatly enhances the analysis reproducibility, whereas with the public release of the package, research teams from elsewhere can benefit of a tested scenario to perform their own analyzes. Additionally, more data analysis hypotheses than before are considered. For instance, adjusting the data preparation at an advanced stage is now possible and only requires new input settings for the scenario. The next calculation will update the graphics, the measurements and the statistics which, in turn, may enable to compare different data treatments at a *meta*-level.

The possible domains of application are in clinical research on complex pathologies like Osteoarthritis, Parkinson's disease and aggressive brain tumor diagnosis. For these pathologies, more homogeneous patient subtypes is expected to help to break down the existing clinical heterogeneity and thus further enhance the understanding of their underlying mechanisms. Hence, the discovered subtypes may help to advance the development of new treatment strategies.

Moreover, SDisc confronts particularly with clinical research requirements in terms of data analysis. It considers the validity aspect of the inference steps carried out in the course of a subtyping analysis, the accessibility facet to enable non-expert computer scientist to perform and/or reproduce analyzes independently and straightforwardly, as well as the availability aspect by the distribution of the generic solution as a documented open source R package.

1 Hands on R SDisc

```
> library(SDisc)

by using mclust, you accept the license agreement in the LICENSE file
and at http://www.stat.washington.edu/mclust/license.txt
```

1.1 Datasets

mixt3 is a matrix of three independent vectors of length 50, which follow the normal distribution having for parameters respectively $\mathcal{N}(0,1)$, $\mathcal{N}(3,5)$, $\mathcal{N}(-2,4)$.

```
> set.seed(6014)
> mixt3 <- matrix(c(rnorm(50), rnorm(50, mean = 3, sd = 5), rnorm(50, mean = -2,
      sd = 4)), 50, 3)
```

normdep is a matrix composed of five variables. The first variable is normal with $\mathcal{N}(0,1)$. The second variable represents the time element of which, the last variable depends upon. We add to this dependent variable an additional noise which we refer to as epsilon. To summary:

$$vDependent = 2 \times time + \epsilon \tag{1}$$

```
> set.seed(6015)
> epsilon <- runif(50)
> time <- sample(1:5, 50, replace = TRUE)
> vDependent <- 2 * time + epsilon
> normdep <- matrix(c(rnorm(50), time, epsilon, vDependent, vDependent), 50,
      5)
> colnames(normdep) <- c("vNormal", "time", "epsilon", "vDependentOrig", "vDependent")
```

The iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of three species of iris. The species are Iris setosa, versicolor, and virginica.

```
> library(datasets)
> help("iris")
```

The state data sets relate to the 50 states of the United States of America. The state.x77 matrix has 50 rows and 8 columns giving the population estimate as of July 1st of 1975, the Income per capita in 1974, the illiteracy in 1970 as a percent of the population, the life expectancy in years in the years 1969-71, the murder and non-negligent manslaughter rate per 100,000 population in 1976, the percent of high-school graduates in 1970 the mean number of days with minimum temperature below freezing in the years 1931-1960 in the capital or the large city, land area in square miles. Further, in complement to state.x77, we add the geo-localisation center from each state expressed in terms of longitude and latitude. The address of the geolocalisation database is http://www.maxmind.com/app/state_latlon

```
> state.loc <- read.csv("state.latlon.csv", row.names = 1)
> state <- data.frame(state.x77[, hclust(dist(t(state.x77)))$order], name = row.names(state.
      latitude = NA, longitude = NA)
> row.names(state) <- state.abb
> naRows <- row.names(state.loc)[(!row.names(state.loc) %in% row.names(state))]
```

```
> state <- rbind(state, matrix(NA, length(naRows), ncol(state), dimnames = list(naRows,
      colnames(state))))
> state[, c("latitude", "longitude")] <- state.loc[row.names(state), c("latitude",
      "longitude")]
```

Orchard sprays represents an experiment which wasconducted to assess the potency of various constituents of orchard sprays in repelling honeybees, using a Latin square design.

```
> help("OrchardSprays")

> osprays <- OrchardSprays
```

1.2 Configure and transform the data

```
> settingsMixt3 <- SDDataSettings(mixt3)

> settingsNormdep <- SDDataSettings(normdep)
> settingsNormdep[, "tFun"] <- c("mean sd", "", "", "", "lm(vDependent~time)")

> SDDataSettings(iris, latex = TRUE)
```

	oddGroup	inCAnalysis	tFun	vParGroup	vParY	vHeatmapY
Sepal.Length	Sepal.Length	TRUE	mean sd	varGroup1	1	1
Sepal.Width	Sepal.Width	TRUE	mean sd	varGroup1	2	2
Petal.Length	Petal.Length	TRUE	mean sd	varGroup1	3	3
Petal.Width	Petal.Width	TRUE	mean sd	varGroup1	4	4
Species	Species	TRUE	mean sd	varGroup1	5	5

Table 1: SDDataSettings

```
> SDDataSettings(iris, asCSV = TRUE)
> SDDataSettings(iris, asCSV = "irisSettings.csv")
> settingsIris <- SDDataSettings(iris)
> settingsIris["Species", ] <- c(NA, FALSE, NA, NA, NA, NA)

> settingsState <- SDDataSettings(state, asCSV = "stateSettings.csv")
> settingsState <- read.csv2("stateSettingsEdited.csv", row.names = 1)

> settingsOsprays <- SDDataSettings(osprays)
> settingsOsprays["treatment", ] <- NA

> dMixt3 <- SDData(mixt3, settings = settingsMixt3, prefix = "Mixt3")
> dNormdep <- SDData(normdep, settings = settingsNormdep, prefix = "Normdep")
> dState <- SDData(state, settings = settingsState, prefix = "state")
```

Yet, when calling SDisc, a call is immediately made to SDData. It results that an SDisc analysis holds a unique SDData container, i.e. the dataset. As such, the data of an SDisc analysis can be extracted with the SDData method. To illustrate this later, we do not process at this moment the **state** and **osprays** datasets with SDData.

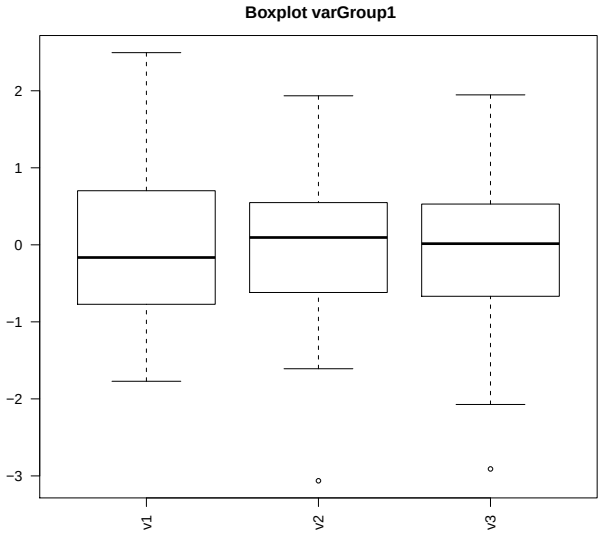


Figure 1: Mixt3, **boxplots** of the variables of the factor **varGroup1**.

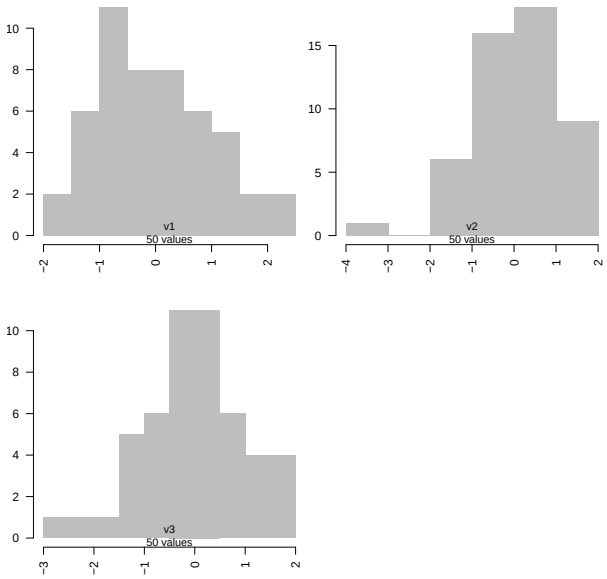


Figure 2: Mixt3, **histograms** of the variables of the factor **varGroup1**.

	v2	v3	v1
26	3.76	-2.74	-0.01
36	1.36	-9.46	-0.93
22	-3.77	-3.76	2.61

Table 2: Mixt3, extract of the **original** data matrix.

	v2	v3	v1
26	0.18	-0.14	0.10
36	-0.38	-1.80	-0.74
22	-1.58	-0.40	2.50

Table 3: Mixt3, extract of the **transformed** data matrix.

1.3 Explore and summary the data

```
> print(dMixt3, rseed = 6013, latex = TRUE)
```

```
> plot(dMixt3, latex = TRUE)
```

```
> summary(dMixt3, latex = TRUE)
```

	mean	sd
v1	-1.20e-01	1.09e+00
v2	3.01e+00	4.28e+00
v3	-2.15e+00	4.05e+00

Table 4: Mixt3 summary of the different data treatments operated on the data.

```
> print(dNormdep, rseed = 6013, latex = TRUE)
```

	epsilon	vDependent	time
26	0.40	10.40	5.00
36	0.24	8.24	4.00
22	0.54	10.54	5.00

Table 5: Normdep, extract of the **original** data matrix.

```
> plot(dNormdep, latex = TRUE)
```

```
> summary(dNormdep, q = "lm", latex = TRUE, sanitize = FALSE)
```

```
> summary(dNormdep, q = "mean|sd", latex = TRUE)
```

```
> naPattern(dState, latex = TRUE)
```

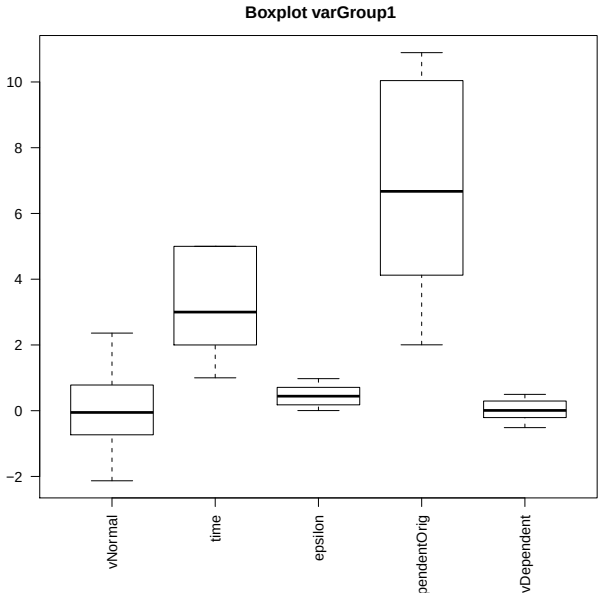


Figure 3: Normdep, **boxplots** of the variables of the factor **varGroup1**.

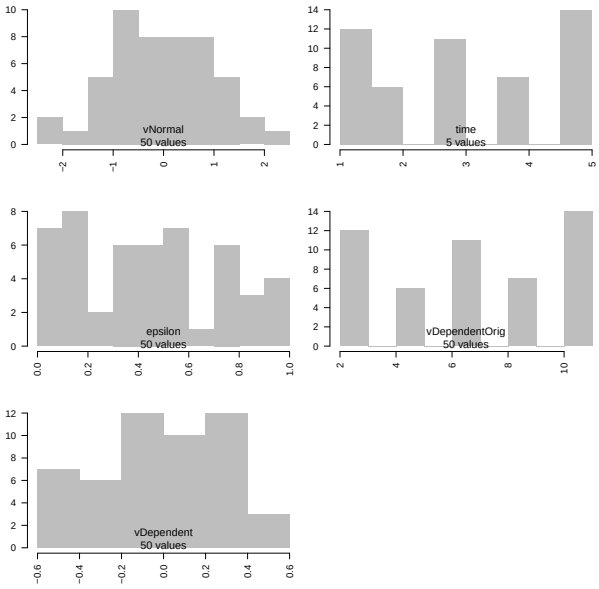


Figure 4: Normdep, **histograms** of the variables of the factor **varGroup1**.

	epsilon	vDependent	time
26	0.40	-0.00	5.00
36	0.24	0.18	4.00
22	0.54	-0.15	5.00

Table 6: Normdep, extract of the **transformed** data matrix.

	(Intercept) (SE; Pr(>|t|))	time (SE; Pr(>|t|))	R^2 (adj-R^2; N)
vDependent time	0.53 (0.09; 9.1e-07)	1.97 (0.03; 8.4e-51)	0.99 (0.99; 50)

Table 7: Normdep summary of the different data treatments operated on the data.

	mean	sd
vNormal	-1.82e-01	9.02e-01

Table 8: Normdep summary of the different data treatments operated on the data.

	isNA	isNotMissing	naRate
AS	9.00	2.00	81.82
DC	9.00	2.00	81.82
MP	9.00	2.00	81.82
PR	9.00	2.00	81.82
VI	9.00	2.00	81.82

Table 9: state, index of the cases presenting **missing values** along with the number of missings and non-missings; the cases with a missing value represent 9.09% of the available cases.

1.4 Predict new data

```
> set.seed(6016)
> epsilon <- runif(30)
> time <- sample(1:5, 30, replace = TRUE)
> vDependent <- 2 * time + epsilon
> mat <- matrix(c(rnorm(30), time, epsilon, vDependent, vDependent), 30, 5)
> colnames(mat) <- c("vNormal", "time", "epsilon", "vDependentOrig", "vDependent")
> dNormdepPredicted <- predict(dNormdep, newdata = mat, prefix = "NormdepPredicted")
> summary(dNormdepPredicted, q = "lm", latex = TRUE, sanitize = FALSE)
```

	(Intercept) (SE; Pr(>|t|))	time (SE; Pr(>|t|))	R^2 (adj-R^2; N)
vDependent time	0.53 (0.09; 9.1e-07)	1.97 (0.03; 8.4e-51)	0.99 (0.99; 50)

Table 10: NormdepPredicted summary of the different data treatments operated on the data.

```
> summary(dNormdepPredicted, q = "mean|sd", latex = TRUE)
```

	mean	sd
vNormal	-1.82e-01	9.02e-01

Table 11: NormdepPredicted summary of the different data treatments operated on the data.

1.5 Model repeatedly the data for clusters

```
> xNormdep <- SDisc(dNormdep)

 Prepare the data

 Modeling for clusters
EII,3,6013 VII,3,6013 EII,4,6013 VII,4,6013 EII,5,6013 VII,5,6013 EII,3,6014 VII,3,6014 EII,
 Collect BICs (likelihood) of the models

 Save modeling into Normdep/IMAGE.RData

 Save best models as CSV files

 Normdep/tables/MM-EII,5,6015.csv

 Normdep/tables/MM-EII,5,6013.csv

 Normdep/tables/MM-VII,5,6013.csv

 Normdep/tables/MM-VII,5,6014.csv

 Normdep/tables/MM-VII,4,6014.csv

> xState <- SDisc(state, settings = settingsState, prefix = "state", cFunSettings = list(mod
       "VII", "VEI", "VVI"), G = 3:5, rseed = 6013:6023))
```

```
 Prepare the data

 Modeling for clusters
EII,3,6013 VII,3,6013 VEI,3,6013 VVI,3,6013 EII,4,6013 VII,4,6013 VEI,4,6013 VVI,4,6013 EII,
 Collect BICs (likelihood) of the models

 Save modeling into state/IMAGE.RData

 Save best models as CSV files

 state/tables/MM-VVI,4,6022.csv

 state/tables/MM-VVI,4,6017.csv

 state/tables/MM-VII,4,6015.csv

 state/tables/MM-VII,4,6023.csv

 state/tables/MM-VII,4,6014.csv
```

```
> xOsprays <- SDisc(osprays, settings = settingsOsprays, prefix = "osprays",
       cFunSettings = list(modelName = c("EII", "VII", "VEI"), G = 3:6, rseed = 6013:6023))
```

```
 Prepare the data

 Modeling for clusters
EII,3,6013 VII,3,6013 VEI,3,6013 EII,4,6013 VII,4,6013 VEI,4,6013 EII,5,6013 VII,5,6013 VEI,
 Collect BICs (likelihood) of the models

 Save modeling into osprays/IMAGE.RData

 Save best models as CSV files

 osprays/tables/MM-VEI,3,6020.csv

 osprays/tables/MM-VEI,3,6014.csv

 osprays/tables/MM-VEI,3,6021.csv

 osprays/tables/MM-VEI,3,6017.csv

 osprays/tables/MM-VEI,3,6018.csv
```

1.6 Rank the models by their likelihood

```
> summary(bicTable(xNormdep), latex = TRUE)
```

```
> summary(bicTable(xState), latex = TRUE)
```

```
> print(bicTable(xState), modelName = "VII", G = 4, latex = TRUE)
```

```
> summary(bicTable(xOsprays), latex = TRUE)
```

	EII	VII
3	7.11 (7.11, 7.17)	4.85 (4.85, 4.85)
4	4.77 (4.77, 5.52)	2.48 (2.51, 4.01)
5	0.00 (0.00, 4.60)	0.80 (0.81, 3.43)

Table 12: Normdep, model EII,5,6015 shows the **highest BIC** score over: the repeated random starts, type of model and number of component.

	EII	VII	VEI	VVI
3	10.21 (10.21, 10.21)	3.98 (3.98, 7.76)	5.67 (5.67, 9.65)	3.09 (3.09, 10.97)
4	4.44 (4.44, 12.13)	1.58 (1.58, 7.53)	2.93 (2.93, 8.62)	0.00 (0.36, 9.78)
5	4.60 (4.66, 12.51)	2.99 (2.99, 3.74)	NA (4.04, 5.34)	2.16 (2.17, 9.58)

Table 13: state, model VVI,4,6022 shows the **highest BIC** score over: the repeated random starts, type of model and number of component.

	modelName	G	rseed	BIC	relativeBic
VII,4,6015	VII	4	6015	-980.58	1.58
VII,4,6023	VII	4	6023	-980.58	1.58
VII,4,6014	VII	4	6014	-980.58	1.58
VII,4,6018	VII	4	6018	-980.58	1.58
VII,4,6022	VII	4	6022	-980.58	1.58
VII,4,6020	VII	4	6020	-980.58	1.58
VII,4,6017	VII	4	6017	-980.58	1.58
VII,4,6016	VII	4	6016	-980.58	1.58
VII,4,6013	VII	4	6013	-980.58	1.58
VII,4,6019	VII	4	6019	-980.58	1.58
VII,4,6021	VII	4	6021	-1057.22	9.52

Table 14: state, models whose **relative BIC** score difference is **less than 5%**.

	EII	VII	VEI
3	1.54 (1.54, 1.55)	2.32 (2.32, 2.40)	0.00 (0.00, 1.83)
4	3.11 (3.11, 3.53)	3.77 (4.01, 5.25)	2.07 (2.07, 2.62)
5	4.50 (4.54, 5.05)	6.00 (6.00, 7.80)	4.55 (4.55, 5.40)
6	5.61 (5.61, 7.02)	7.58 (7.58, 8.71)	6.49 (6.49, 8.17)

Table 15: osprays, model VEI,3,6020 shows the **highest BIC** score over: the repeated random starts, type of model and number of component.

1.7 Compare the most likely models and assess ther stability

```
> print(xNormdep, latex = TRUE)
```

```
> print(xState, latex = TRUE)
```

```
> print(xState, m1 = 1, m2 = bestModel(xState, modelName = "VII", G = 4)[1],
       latex = TRUE)
```

	5	4	2	1	3
2	14				
5		12			
3			11		
1				6	
4					7

Table 16: Normdep, the **comparison of model** EII,5,6015 and EII,5,6013 exhibits a random index 100.0 (a $\kappa = 100.0$, and a relative degree of association $V = 100.0\%$ with $p_{\chi^2} = 0.0005$, $\chi^2 = 200.0$).

	4	1	2	3
3	17		1	
4		10		
1	4			10
2	2		4	2

Table 17: state, the **comparison of model** VVI,4,6022 and VVI,4,6017 exhibits a random index 82.8 (a $\kappa = 74.7$, and a relative degree of association $V = 77.8\%$ with $p_{\chi^2} = 0.0005$, $\chi^2 = 90.9$).

	2	4	1	3
1	22		9	4
2		10		1
3				4

Table 18: state, the **comparison of model** 1 and VII,4,6015 exhibits a random index 85.1 (a $\kappa = 78.7$, and a relative degree of association $V = 80.6\%$ with $p_{\chi^2} = 0.0005$, $\chi^2 = 65.0$).

```
> print(xOsprays, latex = TRUE)
```

	1	2	3
1	32		
2		4	
3			28

Table 19: osprays, the **comparison of model** VEI,3,6020 and VEI,3,6014 exhibits a random index 100.0 (a $\kappa = 100.0$, and a relative degree of association $V = 100.0\%$ with $p_{\chi^2} = 0.0005$, $\chi^2 = 128.0$).

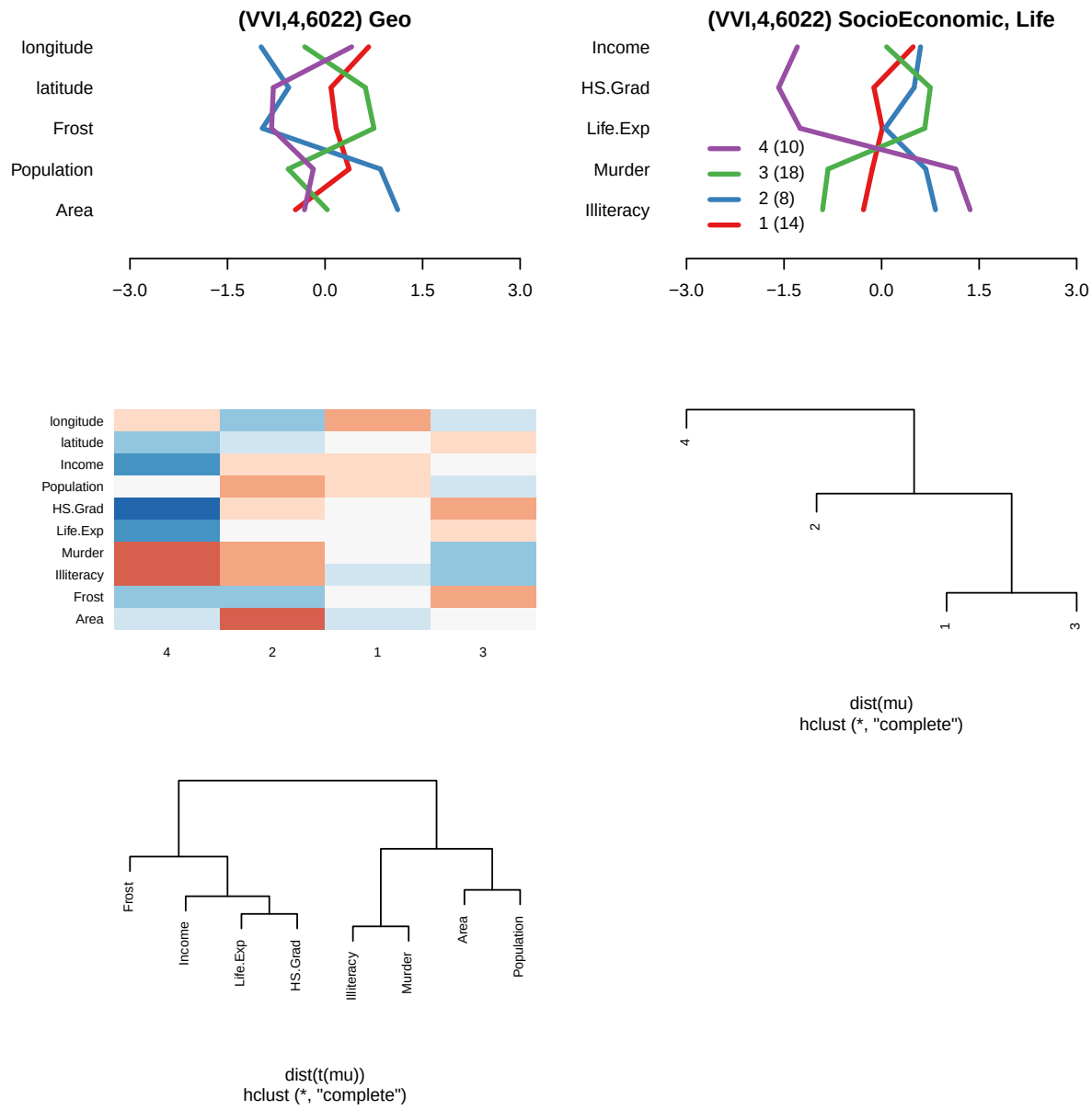


Figure 5: state, visual representation of **model** VVI,4,6022.

1.8 Exhibit the most characteristic features of each subtype

```
> plot(xState, latex = TRUE)
```

```
> plot(xOsprays, latex = TRUE)
```

```
> summary(xState, q = 1, latex = TRUE)
```

```
> summary(xOsprays, q = 1, latex = TRUE)
```

1.9 Validate the discovered subtypes

```
> summary(xOsprays, type = "chi2test", target = "treatment", latex = TRUE)
```

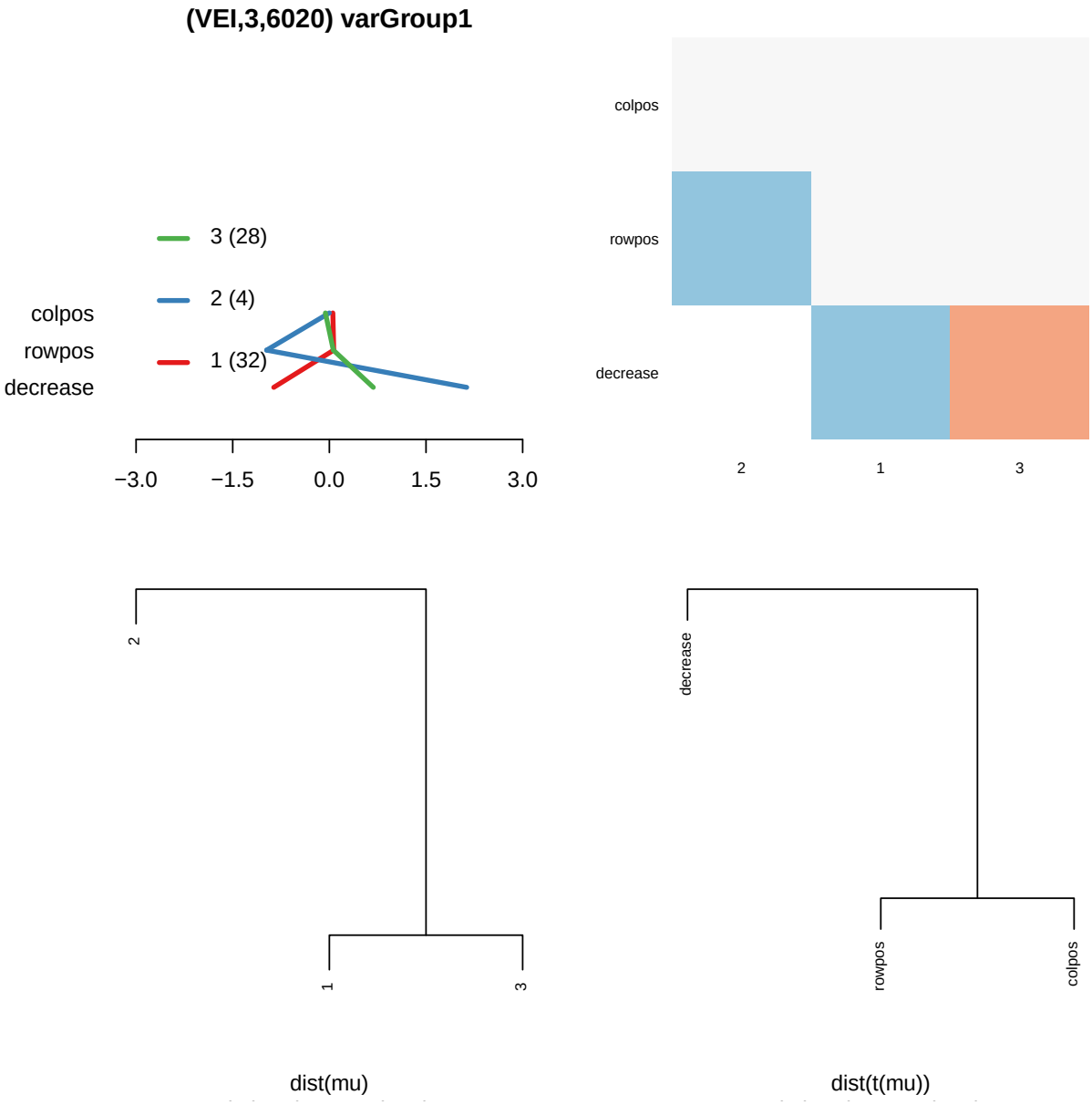


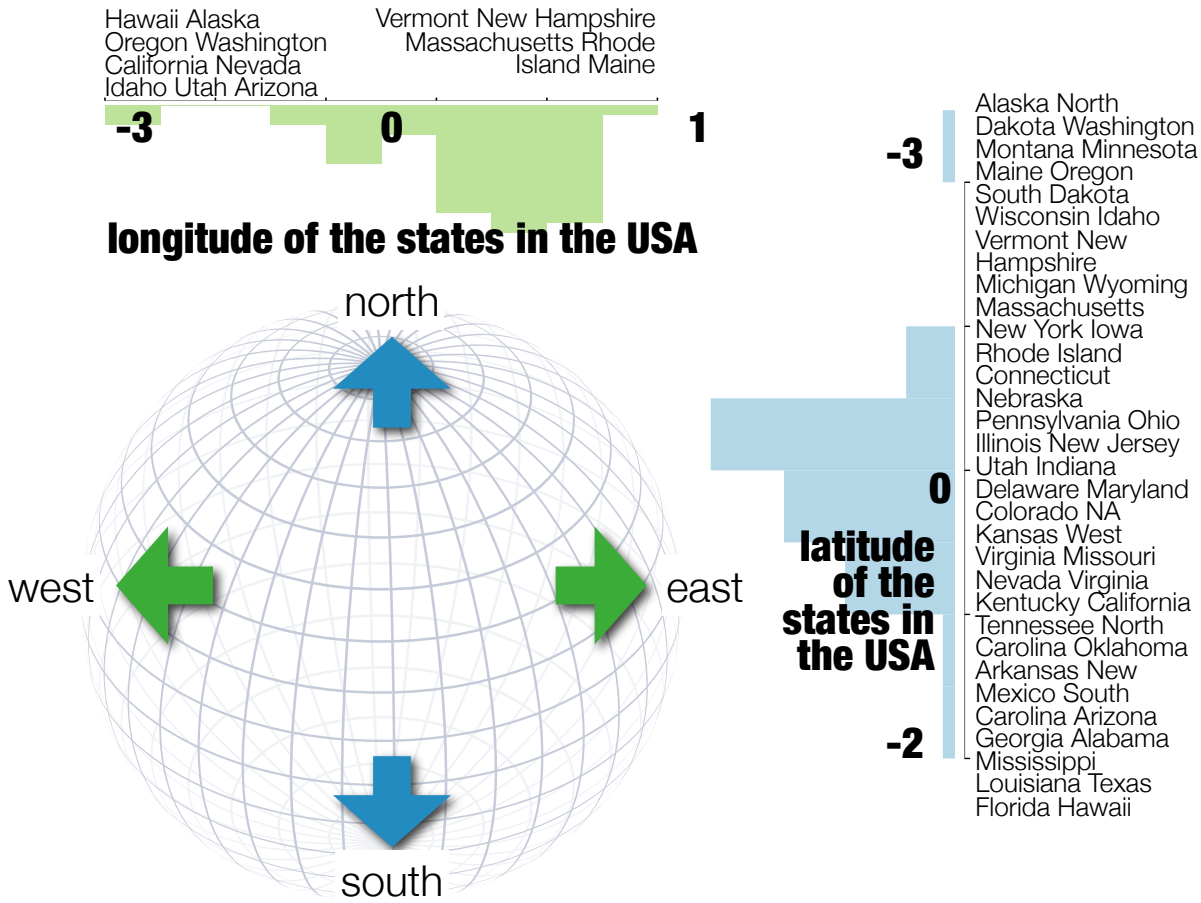
Figure 6: osprays, visual representation of **model** VEI,3,6020.

	1	2	3	4
Geo	1.77	4.01	1.09	-12.60
latitude	0.37	0.44	2.67	-12.21
Life	-0.32	1.29	-0.27	-0.29
longitude	2.32	0.40	-1.99	0.80
SocioEconomic	2.00	11.93	-0.09	-12.21

Table 20: state, (Bayesian) **oddratios** for the main factors in model VVI,4,6022.

	1	2	3
colpos	1.31	-1.35	-0.55
decrease	-11.10	9.12	10.95
rowpos	1.36	-10.51	-0.50

Table 21: osprays, (Bayesian) **oddratios** for the main factors in model VEI,3,6020.



1.10 Test the reproducibility of discovered subtypes on new data

- use predict.SDisc; TODO example

1.11 Install R SDisc

```
> install.packages("SDisc", dep = TRUE)
```

	1	2	3
1	2.000	-0.707	-1.871
2	2.000	-0.707	-1.871
3	1.500	-0.707	-1.336
4	1.000	-0.707	-0.802
5	-1.500	0.707	1.336
6	-1.500	0.707	1.336
7	-1.500	-0.707	1.871
8	-2.000	2.121	1.336

Table 22: For **treatment**: $p_{\chi^2} = 0.000$ ($\chi^2 = 48.3$) in model VEI,3,6020.

```
> library(SDisc)
```

```
R CMD INSTALL SDisc_1.18.tar.gz
```

2 About subtype discovery analysis with SDisc

In this second part of the SDisc vignette, we first present several application domains where the discovery of homogeneous subtypes is of interest, we then report a use case of a subtype analysis, we detail the rationale of the SDisc package, we outline the orientations of our current developments, and we provide several links that relate to SDisc and subtype discovery.

2.1 Instances of domains searching for homogeneous subtypes

In the following, we report the rationale of subtype discovery data analyzes by reviewing a number of domains facing this problem, in medical research (Mol LUMC, Neu LUMC, Psy LUMC, SOCO), in chemoinformatics (Pharma-IT) and in recycling (CIFASIS). For each application domain we motivate the research target.

Osteoarthritis (OA) Searching for subtypes in the distribution of OA may allow to study the spread of the disease across different sites and to show whether it is stochastic or follows a particular pattern. Such subtypes could contribute to elucidate the clinical heterogeneity of OA [**?**] and therefore enhance the identification of the disease pathways (genetics, pathophysiological mechanisms).

Parkinson's disease (PD) Among PD patients, there is marked heterogeneity in the clinical phenotype which differs in the presence, the severity, and the progression rate of the various features while differences are also observed in other clinical variables like age at onset [**?**]. This clinical heterogeneity may indicate the existence of subtypes, whose identification may advance our understanding of the underlying pathological mechanisms of PD and thus, advance the development of more focused treatment strategies.

Major depressive disorders (MDD) and anxiety disorders (ANX) According to the tripartite model, depression and anxiety symptoms are classified into three dimensions reflecting: a common factor of negative affect, and disorder/specific dimensions lack of positive affect (MDD) and somatic arousal (ANX) [**?**]. As there is substantial heterogeneity in these diagnostic categories, identifying more homogeneous subtypes of MDD/ANX based on symptom profiles could help to find prognostic factors, risk factors, and treatment strategies.

Glioblastoma and metastasis We attempt to find discriminative subtypes of aggressive brain tumors using long echo term spectroscopy data. In particular, we search for frequencies of the spectrum making the signals of these pathologies similar and, as a result, difficult to discriminate. Further, as the underlying heterogeneity of the glioblastoma pathology remains uncharacterized at large, subtypes of this brain tumor may enhance our understanding of the different forms of glioblastoma. Last, as effective patient care orientation depends on accurate medical diagnosis, new subtypes of these pathologies may provide a basis to improve their correct discrimination.

Additional analyzes The purpose of the Pharma-IT analysis is to identify subtypes in databases of molecules. As molecules are classified into a number of complex bioactivity classes, an automatic subtyping of the molecules, grouping them based on their similarity, may help to further understand those classes.

Second, with the CIFASIS, an automatic classifier is searched for capable to discriminate between different classes of plastics. In this analysis, the search for subtypes in the distribution of spectroscopy measurements is susceptible to report the most discriminative spectra frequencies, first, and second, to identify whether spectra subtypes exhibit a structure in correlation with the different classes of plastics.

2.2 A subtype analysis use case

The scenario illustrated by Figure 7, starts with a data preparation step where close collaboration with the domain experts is required to obtain a description of the data. These are written into a settings file that defines how to transform each variable, which variable to include in the cluster modeling, how to summarize variables graphically and statistically. To facilitate the task of writing that file, the package implements a function that generates default settings.

Next, a preliminary subtype discovery analysis is performed to test the flow of statistical inferences, and to commence the discussion with the research team. A graphic report of the data container is produced, which enables exploratory data analysis (EDA). It creates box plots, histograms, and several other variable-specific statistics. To characterize the mixture models, the scenario assembles a number of statistics and of graphics. This output enables to complete with the research team a first instructional walk over the whole inference process.

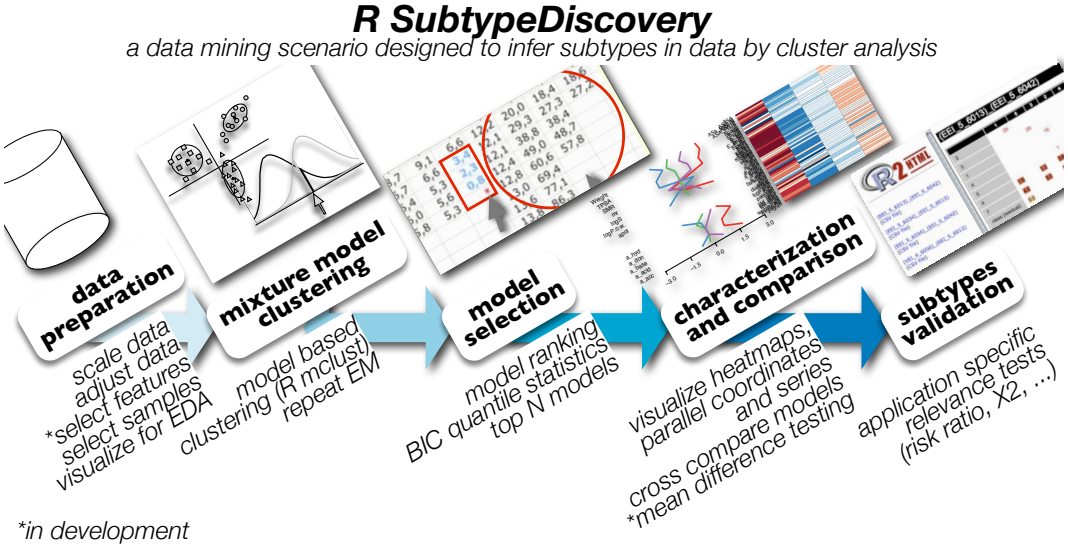


Figure 7: The data mining scenario consists in a sequence of five steps [Colas et al., 2008a]: the data preparation, the cluster modeling based on [**?**, **?**], the model selection, the characterization and comparison of the subtypes and the relevance evaluation. On top of each step, we illustrate some of the tables and graphics it can produces. For more details, see the vignette documentation [Colas, 2009b].

Subsequently, the subtype discovery can be adjusted given considerations over the number of samples, the number of dimensions, the calculation time, the evaluation of the significance of the subtypes by some statistical test (e.g. a χ^2 test of association or of goodness of fit, a risk ratio) or the posterior characterization of the subtypes. This adjustment may involve additional validation data, alternative data processing, filtering of outliers, re-organization of the graphics. Thus, it may require the preparation of a new

settings file and a new data container. The moment these considerations are fixed, a new analysis is performed.

In the succeeding, we present a résumé of the subtyping inference carried out on a cohort study of patients with PD.

> The clinical presentation of PD was described by 13 variables from which the variability explained by the disease duration was removed. Standard scores were taken and a model based cluster analysis was repeated from 50 different starting points, for 3, 4 and 5 clusters and for 5 differently parameterized Gaussian models. It resulted in 750 estimated models. Cluster average PD patterns were visualized using parallel coordinates and heat maps. The distributions of patients in the different cluster solutions were cross-compared in terms of association tables and of a χ^2-based coefficient of nominal association (Cramer's V). Finally, the consistency of the subtypes was evaluated for the reproducibility between the assessments of year one and two.

2.3 An R package to identify subtypes in data

The R platform for statistical computing [?] as well as the BioConductor project for the comprehension and the analysis of genomic data [?] are two projects that gained widespread exposure in the last years. This exposure is partly the result of the abundance of data sources in need of analysis and of a growing demand for analysis reproducibility.

For both projects, Figure 8 portrays the growing number of *new* submissions over the years. It shows the wide acceptance, and thus the relevance, of the R platform for statistical computing as a means to publish scientific programs. In parallel, the BioConductor initiative successfully attracted the creation of softwares in bioinformatics. Yet, for both projects the number of new submissions is reducing. A first hypothesis is that the field of bioinformatics and statistical computing is reaching maturity. A second one is that the total software production is reaching some limit. Or, else, new packages are no longer systematically added to those two repositories, of which SDisc would represent an illustrative example as it was initially submitted to the NBIC gforge.

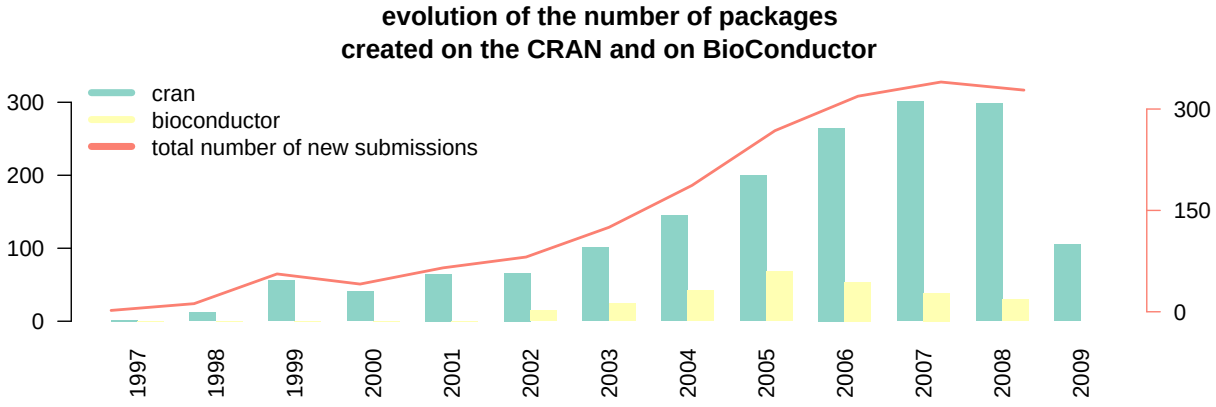


Figure 8: The number of *new* submissions attained 300 packages per year in 2007 and 2008 for the CRAN, and 68 for BioConductor. Yet, in 2008 and 2009, the number of new submissions is slowing down for both projects.

Thus, SDisc fits in the trend to make available and open source the software used to perform a data analysis. Further, as it was applied to very different application areas, the subtyping problem appears recurrent and thus, very general. Last, the variety of data types analyzed also demonstrates the scenario's flexibility.

2.4 Methodology and orientation of new SDisc developments

2.4.1 Research process for the development of new features in R SDisc

When applying the scenario to a growing number of application areas, we develop new methods and extend others to carry out subtype discovery analyzes on new data types and to report field-specific subtype validation methods. Consequently, in what follows, we describe our development methodology to extend the scenario's functionalities.

First, we implement a prototype of the new functionality using the real data of the new application. We update the prototype functionalities gradually, from a field-specific procedure to a more general one. Then, we re-design the procedure as a function, which enables its re-use in other contexts. Ultimately, we implement that procedure and the data structure in an object-oriented mode of programming which in turn, will improve its reliability and guarantee its generality. Later, as the new function stabilizes, or when another application area utilizes it, we include it into the development source code of the package. Periodically, we submit the development source code to the `subversion` system. Before each release of a new version that freezes the functionalities, we update the documentation.

2.4.2 SDisc research orientations

First, to work on the robustness and thus on the accuracy of the inferences made in the course of a subtype discovery analysis, we want to extend and systematize the use of state of the art computational statistics methods. Second, to enhance the scenario's accessibility to a public of non-scientific programmers, to make more straightforward the data analysis, and to guarantee their reproducibility, we want to improve the integration of the scenario. Further on, we describe both aspects.

Robustness In the following, we first discuss computational statistics methods for dimension reduction and second, for subtype characterization.

In problems where the target class is known, the χ^2 test of association can measure the discriminative potential of a dimension. In the case of subtype discovery, we regard χ^2 testing as a means to reduce the dimension of the problem, and thus, to focus the analysis to its most relevant dimensions. Yet, the likelihood to falsely report a dimension as discriminative increases with the number of tests performed. To tackle this problem, as presented in [**?**] (Chap. 5), the family-wise error rate must be controlled. Consequently, we estimate the tail probabilities for the proportion of false positives (TPPFP) [**?**] by resampling the original set of measurements and then, repeating the estimation of the χ^2 statistics (p-value). Our proposal is to implement both cut-off thresholding of the quantiles of p and dimension-ranking for a per class selection.

A t-test can assess the significance of a mean difference observed between a null distribution, composed of the original data, and the one of the subtypes. In the case of subtype discovery, we use t-testing to identify the most singular features of each subtype. Still, as we repeat t-testing over a large number of features, the likelihood increases with the number of tests to falsely report features as significant. To address this issue, we control the family-wise error rate by way of repeated t-testing and cut-off thresholding based on p-value's quantiles (TPPFP). However, the exactness of the t-test depends on the accuracy of the test statistics, i.e. the population mean and variance of the null distribution, and therefore on the normality of the data distribution. To elude the normality assumption and improve the robustness of the statistics, as in [**?**], we want to estimate the null distribution statistics by a resampling-based method and then perform TPPFP.

Integration To effectively integrate the software components of subtype discovery, we first describe source code factorization, and second, third-party software components incorporation. Next, we report various instruments to make the software accessible.

We are looking to further factorize the source code of the package by relying more systematically on object-oriented programming. Previously developed elements become re-usable, thus avoiding code functional redundancies, which is a typical source of programming errors and inconsistencies. Increasing the level of abstraction of the programming also enables to extend more easily the functionalities because it is no longer necessary to know the whole software to contribute new functionalities. Further, the software maintainability enhances because inner object routines are modifiable so long the fields interacting with external components are preserved.

Re-use of other research groups software represents, too, a means to extend the functionalities. For subtype discovery, we foresee the potent integration of four packages. First, `MLInterfaces` [**?**] that is an uniform interface to machine learning code for data in Bioconductor containers may enable to standardize the use of machine learning in subtype discovery [**?**]. Second, `MCRestimate` that calculates misclassification error rates by cross validation may complement effectively `MLInterfaces` for machine learning calculations [**?**]. Third, the `multtest` package implementing resampling-based multiple hypothesis testing represents the state of the art in terms of multiple testing software [**?**]. Last, `sweave` that enables to create and update reports after changes in the data or the analysis, can make uniform the software output [**?**]. Apart from these packages, we also want to take advantage of the generic R language mechanisms for plotting, printing and summarizing R objects.

Along with about 1800 other projects, we host SDisc on the Comprehensive R Archive Network (CRAN) [**?**]. We publish SDisc by means of a vignette [Colas, 2009b], a manual, the software source code and package binaries for Windows, Linux, and MacOSX. To guarantee reproducibility of the analyzes performed with previous versions of the software, we also make available older versions of the package; see archives.

2.5 Assistance, feature request, bug report and SDisc reviewing

- to ask for consultancy in subtype discovery, assistance in the use of SDisc or new features in SDisc, contact F Colas, Dr [Colas, 2009a, Colas et al., 2008a, Colas et al., 2008b]
- to submit a review about R SDisc, go to CRANtastic

List of Tables

List of Figures

References

[Colas, 2009a] Colas, F. (2009a). *Data Mining Scenarios for the Discovery of Subtypes and the Comparison of Algorithms*. PhD thesis, Leiden University.

[Colas, 2009b] Colas, F. (2009b). *R SubtypeDiscovery Vignette: a data mining scenario for the inference of subtypes by cluster analysis*. LIACS, Leiden University.

[Colas et al., 2008a] Colas, F., Meulenbelt, I., Houwing-Duistermaat, J. J., Kloppenburg, M., Watt, I., van Rooden, S. M., Visser, M., Marinus, J., Cannon, E. O., Bender, A., van Hilten, J. J., Slagboom, P. E., and Kok, J. N. (2008a). A scenario implementation in r for subtype discovery examplified on chemoinformatics data. In *ISoLA*, pages 669–683.

[Colas et al., 2008b] Colas, F., Meulenbelt, I., Houwing-Duistermaat, J. J., Kloppenburg, M., Watt, I., van Rooden, S. M., Visser, M., Marinus, J., van Hilten, J. J., Slagboom, P. E., and Kok, J. N. (2008b). Stability of clusters for different time adjustments in complex disease research. In *30th Annual International IEEE EMBS Conference (EMBC'08), Vancouver, British Columbia, Canada*.