

rebmix: Finite Mixture Modeling, Clustering & Classification

Marko Nagode

April 7, 2018

Abstract

The **rebmix** package provides R functions for random univariate and multivariate finite mixture model generation, estimation, clustering and classification. Variables can be continuous, discrete, independent or dependent and may follow normal, lognormal, Weibull, gamma, binomial, Poisson, Dirac or von Mises parametric families.

1 Introduction

To cite the REBMIX algorithm please refer to (Nagode and Fajdiga, 2011a,b; Nagode, 2015, 2018). For theoretical backgrounds please upload also <http://doi.org/10.5963/JA00302001>.

2 What's new in version 2.10.1

Cholesky decomposition is now used to calculate logarithm of determinant and inverse of variance-covariance matrices instead of LU decomposition. Special attention is put in resolving numerical problems related to high dimensional datasets. Version 2.10.1 is further debugged version 2.10.0. Large K in combination with large dimension d can lead to histograms with numerous nonempty bins v . In order to restrain v , the well known RootN rule (Velleman, 1976) may intuitively be extended to multidimensions

$$v_{\max} = \frac{1+d}{d} n^{\frac{d}{1+d}}. \quad (1)$$

If $d = \infty$, then $v_{\max} = n$. If $d = 1$, then $v_{\max} = 2\sqrt{n}$. Minor debugging and function improvements are done in version 2.10.0, too. Acceleration rate is now progressively increasing. Each time the inner loop starts, counter I_2 (see Nagode, 2015, for details) is initiated and constant

$$A = \frac{1 - a_r}{a_r(D_l w_l - D_{\min})} \Big|_{I_2=1} \quad (2)$$

is calculated. Acceleration rate a_r at $I_2 = 1$ always equals the value stored in the input argument **ar**. Otherwise

$$a_r = \frac{1}{A(D_l w_l - D_{\min}) + 1} \Big|_{I_2>1}. \quad (3)$$

The Newton-Raphson root finding in C++ functions is improved in version 2.9.3, too. This affects only Weibull, gamma and von Mises parametric families. Circular von Mises parametric family is added and further debugging is done in version 2.9.2. Version 2.9.1 is further debugged version 2.8.4. The R code is extended and rewritten in S4 class system. The background C code is extended and rewritten as object-oriented C++ code, too. The package can easier be extended to other parametric families. Multivariate normal mixtures with unrestricted variance-covariance matrices are added. Clustering is added and classification is improved.

3 Examples

To illustrate the use of the REBMIX algorithm, univariate and multivariate datasets are considered. The **rebmix** is loaded and the prompt before starting new page is set to **TRUE**.

```
R> library("rebmix")
R> devAskNewPage(ask = TRUE)
```

3.1 Gamma datasets

Three gamma mixtures are considered (Wiper et al., 2001). The first has four well-separated components with means 2, 4, 6 and 8, respectively

$$\begin{array}{lll} \theta_1 = 1/100 & \beta_1 = 200 & n_1 = 100 \\ \theta_2 = 1/100 & \beta_2 = 400 & n_2 = 100 \\ \theta_3 = 1/100 & \beta_3 = 600 & n_3 = 100 \\ \theta_4 = 1/100 & \beta_4 = 800 & n_4 = 100. \end{array}$$

The second has equal means but different variances and weights

$$\begin{array}{lll} \theta_1 = 1/27 & \beta_1 = 9 & n_1 = 40 \\ \theta_2 = 1/270 & \beta_2 = 90 & n_2 = 360. \end{array}$$

The third is a mixture of a rather diffuse component with mean 6 and two lower weighted components with smaller variances and means of 2 and 10, respectively

$$\begin{array}{lll} \theta_1 = 1/20 & \beta_1 = 40 & n_1 = 80 \\ \theta_2 = 1 & \beta_2 = 6 & n_2 = 240 \\ \theta_3 = 1/20 & \beta_3 = 200 & n_3 = 80. \end{array}$$

3.1.1 Finite mixture generation

```
R> n <- c(100, 100, 100, 100)
R> Theta <- list(pdf1 = "gamma", theta1.1 = c(1/100, 1/100, 1/100,
+      1/100), theta2.1 = c(200, 400, 600, 800))
R> gamma1 <- RNGMIX(Dataset.name = "gamma1", n = n, Theta = Theta)
R> n <- c(40, 360)
R> Theta <- list(pdf1 = "gamma", theta1.1 = c(1/27, 1/270), theta2.1 = c(9,
+      90))
R> gamma2 <- RNGMIX(Dataset.name = "gamma2", n = n, Theta = Theta)
R> n <- c(80, 240, 80)
R> Theta <- list(pdf1 = "gamma", theta1.1 = c(1/20, 1, 1/20), theta2.1 = c(40,
+      6, 200))
R> gamma3 <- RNGMIX(Dataset.name = "gamma3", rseed = -4, n = n,
+      Theta = Theta)
```

3.1.2 Finite mixture estimation

```
R> gamma1est <- REBMIX(Dataset = gamma1@Dataset, Preprocessing = "Parzen window",
+      cmax = 8, Criterion = c("AIC", "BIC"), pdf = "gamma")
R> gamma2est <- REBMIX(Dataset = gamma2@Dataset, Preprocessing = "histogram",
+      cmax = 8, Criterion = "BIC", pdf = "gamma")
R> gamma3est <- REBMIX(Dataset = gamma3@Dataset, Preprocessing = "histogram",
+      cmax = 8, Criterion = "BIC", pdf = "gamma", K = 23:27)
```

3.1.3 Plot method

```
R> plot(gamma3est, pos = 1, what = c("den", "dis"), ncol = 2, npts = 1000)
```

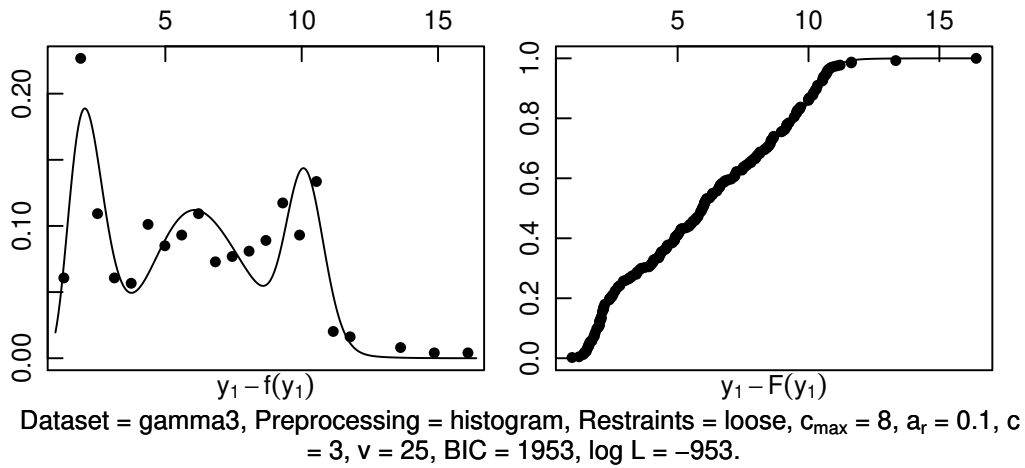


Figure 1: Gamma 3 dataset. Empirical density (circles) and predictive gamma mixture density in black solid line.

3.1.4 Summary and coef methods

```
R> summary(gamma2est)
```

	Dataset	Preprocessing	Criterion	c	v/k	IC	logL	M
1	gamma2	histogram	BIC	2	16	-1321	676	5

Maximum logL = 676 at pos = 1.

```
R> coef(gamma1est, pos = 2)
```

	comp1	comp2	comp3	comp4
w	0.25	0.25	0.25	0.25
	1			
theta1.1	0.01027			
theta1.2	0.00921			
theta1.3	0.00870			
theta1.4	0.01118			
	1			
theta2.1	195			
theta2.2	437			
theta2.3	918			
theta2.4	535			

3.1.5 Bootstrap methods

```
R> gamma3boot <- boot(x = gamma3est, pos = 1, Bootstrap = "p", B = 10)
R> gamma3boot
```

An object of class "REBMIX.boot"

Slot "c":

```
[1] 3 3 3 3 3 3 3 3 3 3
```

Slot "c.se":

```
[1] 0
```

Slot "c.cv":

```
[1] 0
```

Slot "c.mode":

```

[1] 3
Slot "c.prob":
[1] 1

R> summary(gamma3boot)

      comp1 comp2 comp3
w.cv 0.122   0.2 0.175
      1
theta1.1.cv 0.457
theta1.2.cv 0.840
theta1.3.cv 0.508
      1
theta2.1.cv 2.149
theta2.2.cv 0.697
theta2.3.cv 1.083
Mode probability = 1 at c = 3 components.

```

3.2 Poisson dataset

Dataset consists of $n = 600$ two dimensional observations obtained by generating data points separately from each of three Poisson distributions. The component dataset sizes and parameters, which are those studied in Ma et al. (2009), are displayed below

$$\begin{aligned}
\theta_1 &= (3, 2)^\top & n_1 &= 200 \\
\theta_2 &= (9, 10)^\top & n_2 &= 200 \\
\theta_3 &= (15, 16)^\top & n_3 &= 200
\end{aligned}$$

For the dataset Ma et al. (2009) conduct 100 experiments by selecting different initial values of the mixing proportions. In all the cases, the adaptive gradient BYY learning algorithm leads to the correct model selection, i.e., finally allocating the correct number of Poissons for the dataset. In the meantime, it also results in an estimate for each parameter in the original or true Poisson mixture which generated the dataset. As the dataset of Ma et al. (2009) can not exactly be reproduced, 10 datasets are generated with random seeds r_{seed} ranging from -1 to -10 .

3.2.1 Finite mixture generation

```

R> n <- c(200, 200, 200)
R> Theta <- list(pdf1 = rep("Poisson", 2), theta1.1 = c(3, 2), theta2.1 = c(NA,
+      NA), pdf2 = rep("Poisson", 2), theta1.2 = c(9, 10), theta2.2 = c(NA,
+      NA), pdf3 = rep("Poisson", 2), theta1.3 = c(15, 16), theta2.3 = c(NA,
+      NA))
R> poisson <- RNGMIX(Dataset.name = paste("Poisson_", 1:10, sep = ""),
+      n = n, Theta = Theta)

```

3.2.2 Finite mixture estimation

```

R> poissonest <- REBMIX(Dataset = poisson@Dataset, Preprocessing = "histogram",
+      cmax = 10, Criterion = "MDL5", pdf = rep("Poisson", 2), K = 1)

```

3.2.3 Plot method

3.2.4 Clustering

```
R> plot(poissonest, pos = 9, what = c("dens", "marg", "IC", "D",
+   "logL"), nrow = 2, ncol = 3, npts = 1000)
```

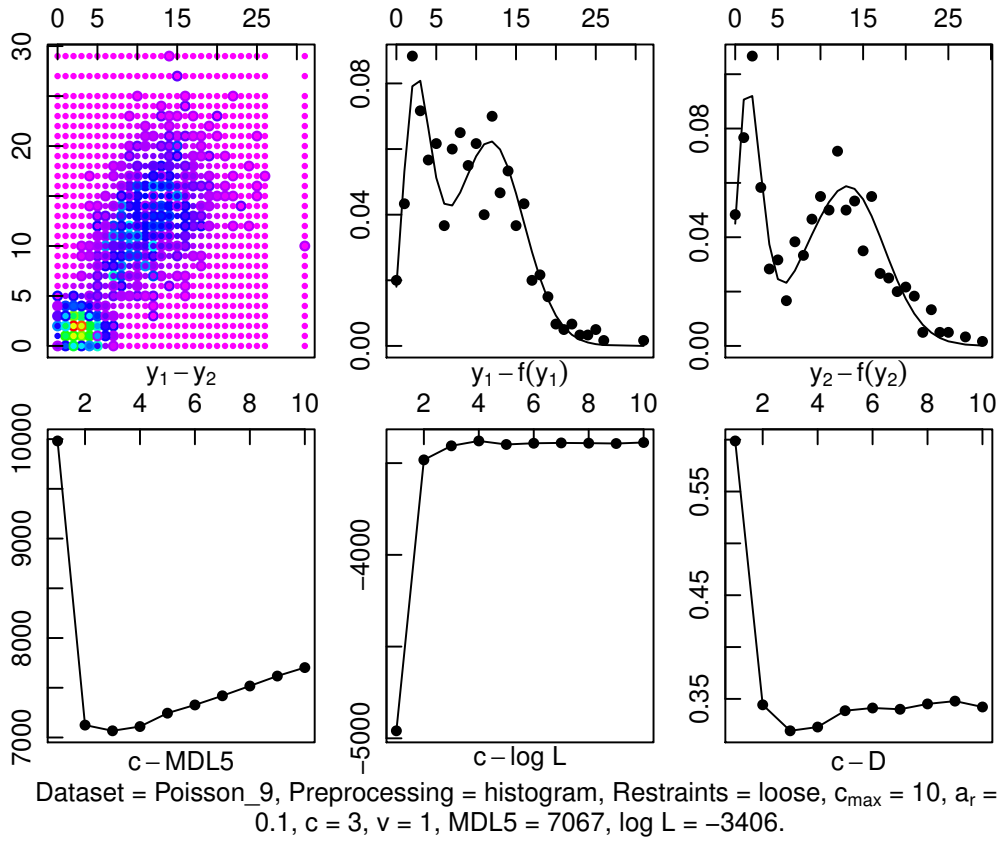


Figure 2: Poisson dataset. Empirical densities (coloured large circles), predictive multivariate Poisson-Poisson mixture density (coloured small circles), empirical densities (circles), predictive univariate marginal Poisson mixture densities and progress charts (solid line).

3.2.5 Summary and coef methods

```
R> summary(poissonest)
```

	Dataset	Preprocessing	Criterion	c	v/k	IC	logL	M
1	Poisson_1	histogram	MDL5	3	1	6992	-3368	8
2	Poisson_2	histogram	MDL5	2	1	7180	-3510	5
3	Poisson_3	histogram	MDL5	3	1	7030	-3387	8
4	Poisson_4	histogram	MDL5	3	1	7005	-3375	8
5	Poisson_5	histogram	MDL5	3	1	6992	-3368	8
6	Poisson_6	histogram	MDL5	3	1	7044	-3394	8
7	Poisson_7	histogram	MDL5	2	1	7239	-3539	5
8	Poisson_8	histogram	MDL5	3	1	7036	-3390	8
9	Poisson_9	histogram	MDL5	3	1	7067	-3406	8
10	Poisson_10	histogram	MDL5	3	1	7008	-3376	8

Maximum logL = -3368 at pos = 5.

```
R> coef(poissonest, pos = 9)
```

```
comp1 comp2 comp3
w 0.336 0.155 0.509
      1      2
theta1.1 2.93 2.01
```

```
R> poissonclu <- RCLRMIX(x = poissonest, pos = 9, Zt = poisson@Zt)
R> plot(poissonclu)
```

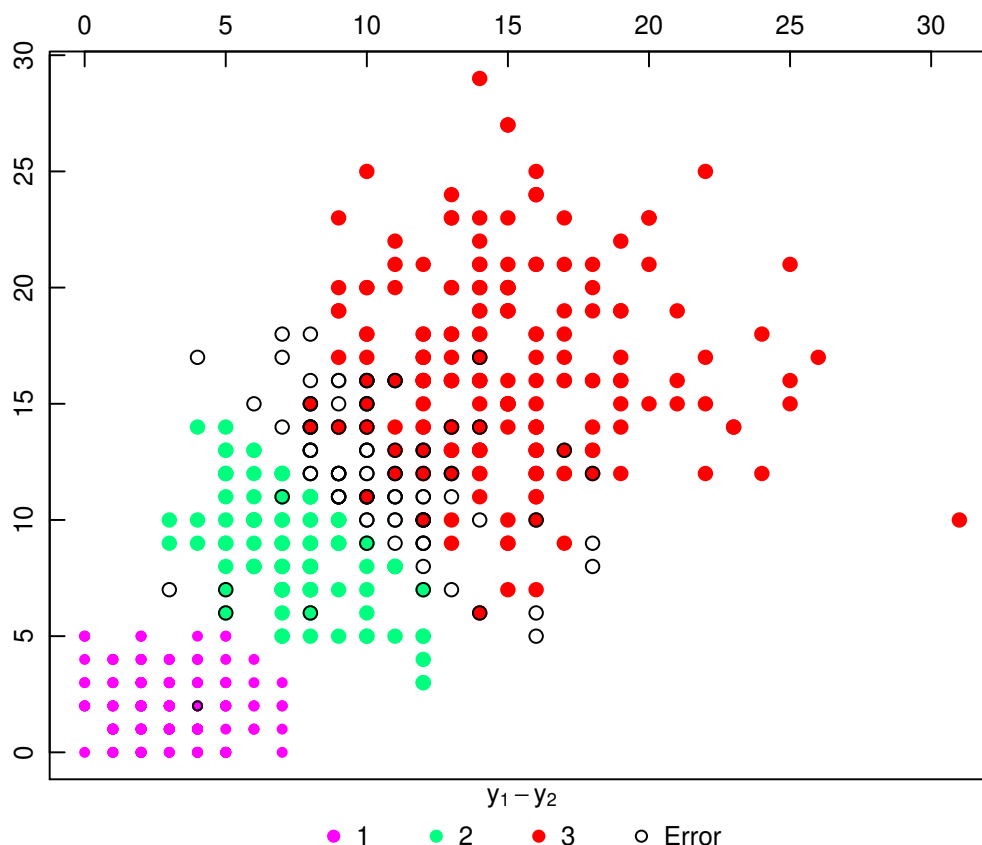


Figure 3: Poisson dataset. Predictive cluster membership (coloured circles), error (black circles).

```
theta1.2  7.75  8.68
theta1.3 13.05 14.42
      1  2
theta2.1  0  0
theta2.2  0  0
theta2.3  0  0
```

3.3 Multivariate normal wreath dataset

A wreath dataset (Fraley et al., 2005) consist of 1000 observations drawn from a 14-component normal mixture in which the covariances of the components have the same size and shape but differ in orientation.

```
R> data("wreath", package = "mclust")
```

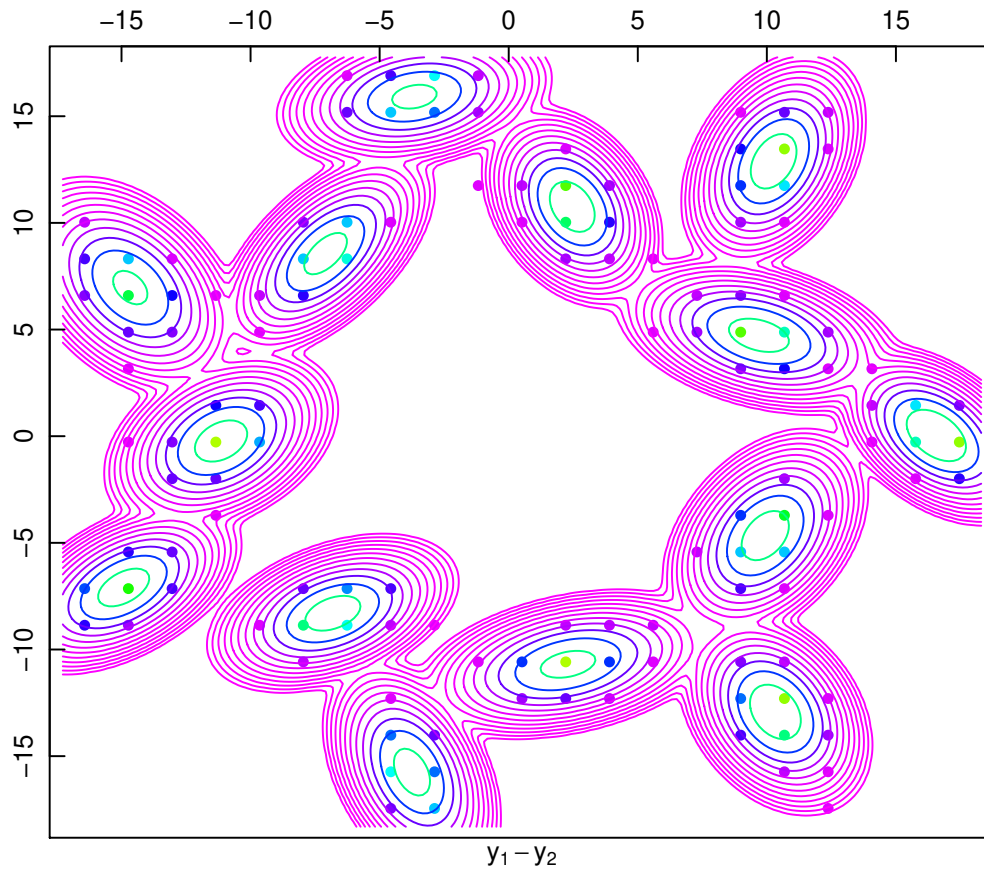
3.3.1 Finite mixture estimation

```
R> wreathest <- REBMIX(model = "REBMVNORM", Dataset = list(as.data.frame(wreath)),
+   Preprocessing = "histogram", cmax = 20, Criterion = "BIC")
```

3.3.2 Plot method

3.3.3 Clustering

```
R> plot(wreathest)
```



Dataset = dataset1, Preprocessing = histogram, Restraints = loose, $c_{\max} = 20$, $a_r = 0.1$, $c = 14$, $v = 21$, BIC = 11174, $\log L = -5300$.

Figure 4: Dataset `wreath`. Empirical densities (coloured circles), predictive multivariate normal mixture density (coloured lines).

3.3.4 Summary and coef methods

```
R> summary(wreathest)
```

	Dataset	Preprocessing	Criterion	c	v/k	IC	logL	M
1	dataset1	histogram	BIC	14	21	11174	-5300	83

Maximum logL = -5300 at pos = 1.

```
R> coef(wreathest)
```

	comp1	comp2	comp3	comp4	comp5	comp6	comp7	comp8	comp9	comp10	comp11	
w	0.0699	0.0753	0.0772	0.0803	0.0823	0.0702	0.0752	0.0652	0.0771	0.0701	0.0673	
	comp12	comp13	comp14									
w	0.0631	0.0611	0.0657									
	1	2										
theta1.1	2.30	-10.6940										
theta1.2	-11.14	-0.2258										
theta1.3	10.26	12.8882										
theta1.4	16.56	0.0341										
theta1.5	10.33	-12.9260										
theta1.6	2.46	10.7556										
theta1.7	9.69	4.7299										
theta1.8	-14.92	-7.0925										

```
R> wreathclu <- RCLRMIX(model = "RCLRMVNORM", x = wreathest)
R> plot(wreathclu, s = 14)
```

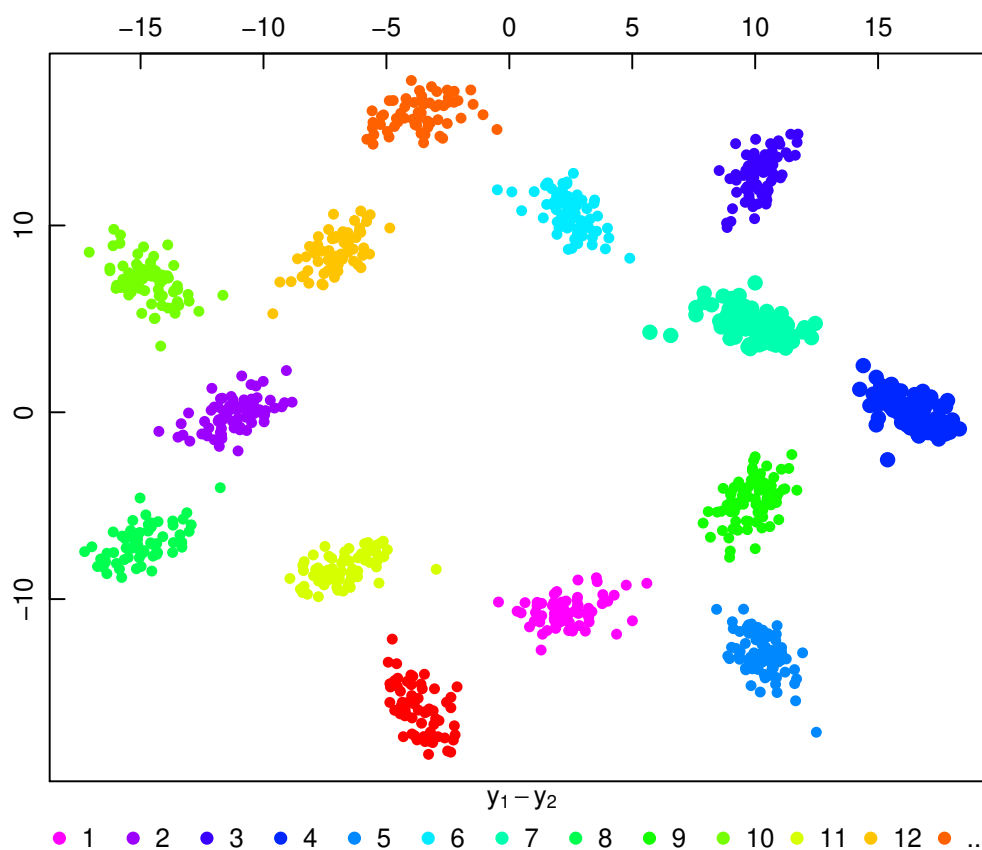


Figure 5: Dataset `wreath`. Predictive cluster membership (coloured circles).

```
theta1.9      9.94  -4.6601
theta1.10    -14.66   6.9672
theta1.11     -6.81  -8.3005
theta1.12     -7.09   8.5617
theta1.13     -3.65  15.9168
theta1.14     -3.75 -15.7497

      1-1    1-2    2-1    2-2
theta2.1  2.116  0.426  0.426  0.749
theta2.2  1.230  0.412  0.412  1.117
theta2.3  0.758  0.400  0.400  1.579
theta2.4  0.941 -0.443 -0.443  1.037
theta2.5  0.826 -0.343 -0.343  1.389
theta2.6  0.723 -0.328 -0.328  1.325
theta2.7  1.835 -0.437 -0.437  0.825
theta2.8  1.250  0.545  0.545  0.996
theta2.9  0.944  0.466  0.466  1.501
theta2.10 1.192 -0.569 -0.569  1.669
theta2.11 1.426  0.436  0.436  0.832
theta2.12 1.083  0.757  0.757  1.482
theta2.13 1.743  0.282  0.282  0.728
theta2.14 0.718 -0.445 -0.445  1.763
```

3.3.5 Summary method

```
R> summary(wreathclu)
```


Number of clusters	1	2	3	4	5
From cluster	3	2	13	5	4
To cluster	1	1	1	1	1
Entropy	2.03e-14	3.34e-03	8.28e-03	1.38e-02	2.28e-02
Entropy decrease	0.00334	0.00494	0.00552	0.00901	0.00751
Number of clusters	6	7	8	9	10
From cluster	9	14	10	11	12
To cluster	4	1	2	1	10
Entropy	3.03e-02	3.89e-02	5.83e-02	7.95e-02	1.20e-01
Entropy decrease	0.00858	0.01943	0.02119	0.04050	0.14527
Number of clusters	11	12	13		
From cluster	6	8	7		
To cluster	4	2	4		
Entropy	2.65e-01	5.90e-01	1.10e+00		
Entropy decrease	0.32504	0.51374	0.74344		

3.4 Multivariate normal ex4.1 dataset

A ex4.1 dataset (Baudry et al., 2010; Fraley et al., 2016) consist of 600 two dimensional observations.

```
R> data("Baudry_etal_2010_JCGS_examples", package = "mclust")
```

3.4.1 Finite mixture estimation

```
R> ex4.1est <- REBMIX(model = "REBMVNORM", Dataset = list(as.data.frame(ex4.1)),
+   Preprocessing = "Parzen window", cmax = 10, Criterion = "AIC")
```

3.4.2 Plot method

```
R> plot(ex4.1est, pos = 1, what = c("dens"), nrow = 1, ncol = 1)
```

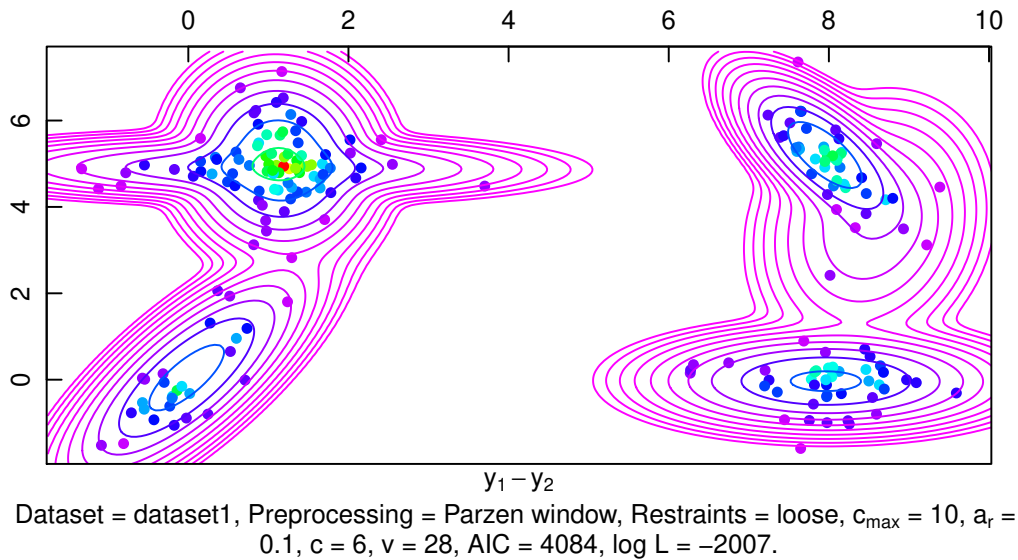


Figure 6: Dataset ex4.1. Empirical densities (coloured circles), predictive multivariate normal mixture density (coloured lines).

3.4.3 Clustering

```
R> ex4.1clu <- RCLRMIX(model = "RCLRMVNORM", x = ex4.1est)
R> plot(ex4.1clu)
```

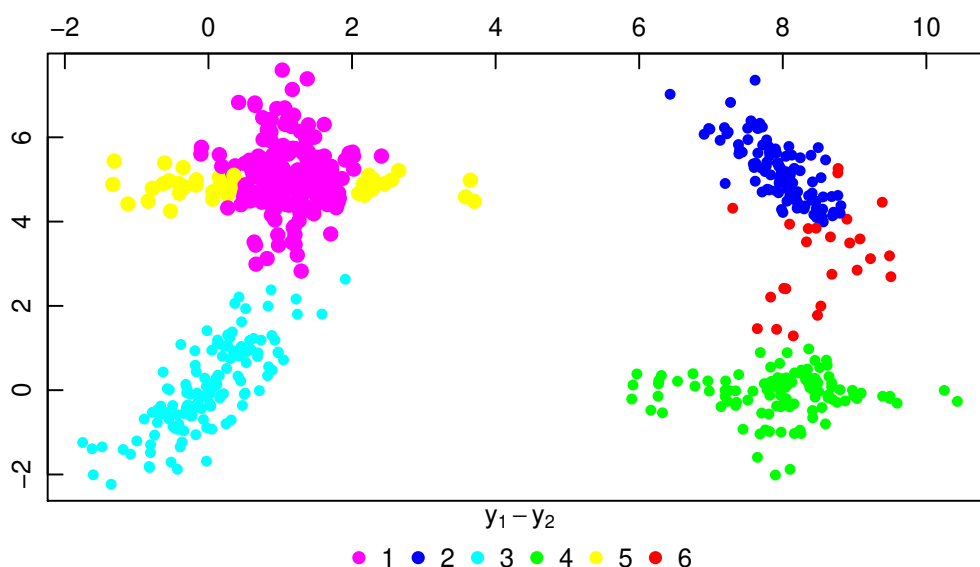


Figure 7: Dataset ex4.1. Predictive cluster membership (coloured circles).

3.4.4 Summary method

```
R> summary(ex4.1est)
```

```
Dataset Preprocessing Criterion c v/k IC logL M
1 dataset1 Parzen window AIC 6 28 4084 -2007 35
Maximum logL = -2007 at pos = 1.
```

3.5 Multivariate iris dataset

The well known set of iris data as collected originally by Anderson (1936) and first analysed by Fisher (1936) is considered here. It is available at Asuncion and Newman (2007) consisting of the measurements of the length and width of both sepals and petals of 50 plants for each of the three types of iris species setosa, versicolor and virginica. The iris dataset is loaded, split into three subsets for the three classes and the `Class` column is removed.

```
R> data("iris")
R> levels(iris[["Class"]])

[1] "iris-setosa"      "iris-versicolor" "iris-virginica"
```

```
R> set.seed(5)
R> Iris <- split(p = 0.75, Dataset = iris, class = 5)
```

3.5.1 Finite mixture estimation

```
R> irisest <- REBMIX(model = "REBMVNORM", Dataset = Iris@train,
+   Preprocessing = "Parzen window", cmax = 10, Criterion = "ICL-BIC")
```

3.5.2 Classification

```
R> iriscla <- RCLSMIX(model = "RCLSMVNORM", x = list(irisest), Dataset = Iris@test,
+   Zt = Iris@Zt)
```

3.5.3 Show and summary methods

```
R> iriscla
```

An object of class "RCLSMVNORM"

Slot "CM":

```
      1  2  3
1 13  0  0
2  0 13  0
3  0  1 12
```

Slot "Error":

```
[1] 0.0256
```

Slot "Precision":

```
[1] 1.000 1.000 0.923
```

Slot "Sensitivity":

```
[1] 1.000 0.929 1.000
```

Slot "Specificity":

```
[1] 1.000 1.040 0.963
```

Slot "Chunks":

```
[1] 1
```

```
R> summary(iriscla)
```

	Test	Predictive	Frequency
1	1	1	13
2	2	1	0
3	3	1	0
4	1	2	0
5	2	2	13
6	3	2	1
7	1	3	0
8	2	3	0
9	3	3	12

Error = 0.0256.

3.5.4 Plot method

3.6 Multivariate adult dataset

The `adult` dataset containing 48842 instances with 16 continuous, binary and discrete variables was extracted from the census bureau database Asuncion and Newman (2007). Extraction was done by Barry Becker from the 1994 census bureau database. The `adult` dataset is loaded, complete cases are extracted and levels are replaced with numbers.

```
R> data("adult")
```

```
R> adult <- adult[complete.cases(adult), ]
```

```
R> adult <- as.data.frame(data.matrix(adult))
```

Numbers of unique values for variables are determined and displayed.

```
R> cmax <- unlist(lapply(apply(adult[, c(-1, -16)], 2, unique),
+                   length))
R> cmax
```

```
R> plot(iriscla, nrow = 3, ncol = 2)
```

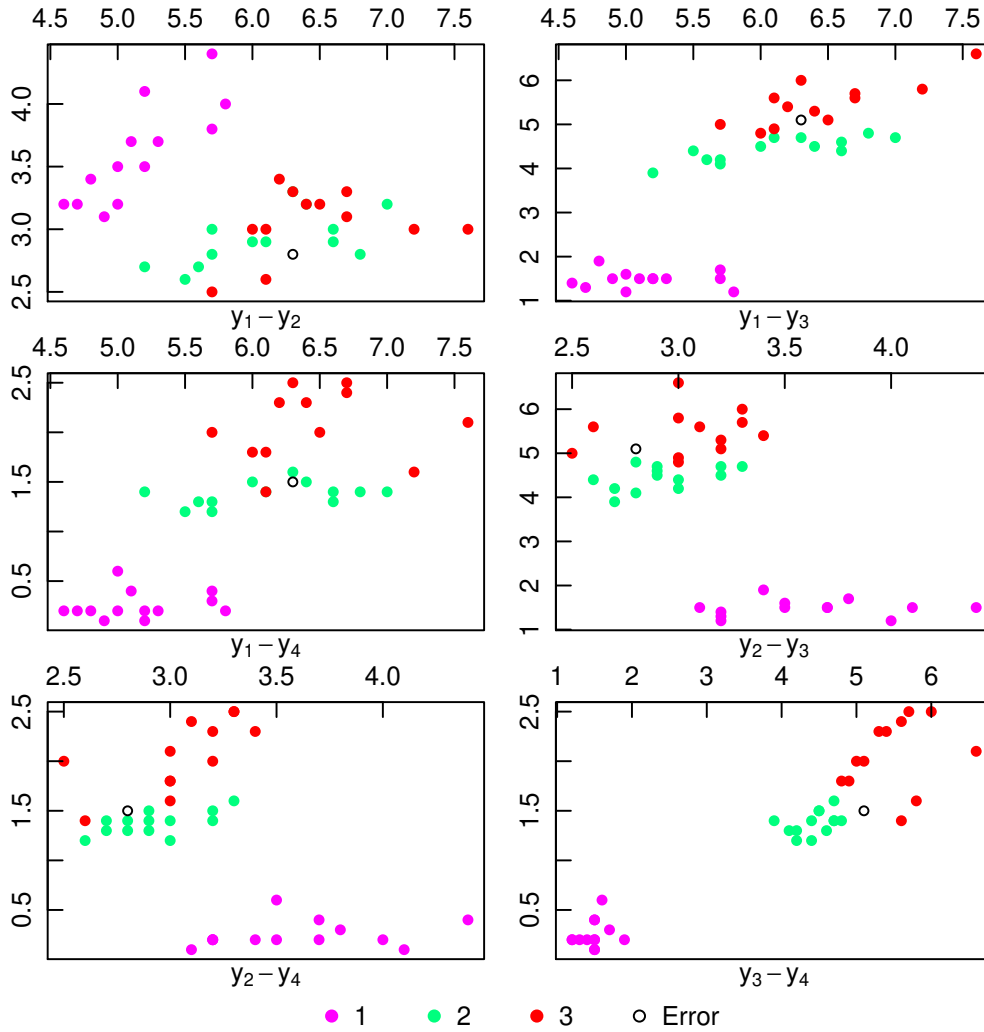


Figure 8: Dataset *iris*. Predictive class membership (coloured circles), error (black circles).

Age	Workclass	Fnlwgt	Education	Education.Num
74	7	26741	16	16
Marital.Status	Occupation	Relationship	Race	Sex
7	14	6	5	2
Capital.Gain	Capital.Loss	Hours.Per.Week	Native.Country	
121	97	96	41	

The dataset is split into train and test subsets for the two incomes and the **Type** and **Income** columns are removed.

```
R> Adult <- split(p = list(type = 1, train = 2, test = 1), Dataset = adult,
+               class = 16)
```

3.6.1 Finite mixture estimation

Number of components, component weights and component parameters are estimated assuming that the variables are independent for the set of chunks $y_{1j}, y_{2j}, \dots, y_{14j}$.

```
R> adulctest <- list()
R> for (i in 1:14) {
+   adulctest[[i]] <- REBMIX(Dataset = chunk(Adult, i)@train,
+   Preprocessing = "histogram", cmax = min(120, cmax[i]),
```

```
+      Criterion = "BIC", pdf = "Dirac", K = 1)
+ }
```

3.6.2 Classification

The class membership prediction is based upon the best first search algorithm.

```
R> adultcla <- BFSMIX(x = adultest, Dataset = Adult@test, Zt = Adult@Zt)
```

3.6.3 Show and summary methods

```
R> adultcla
```

An object of class "RCLSMIX"

Slot "CM":

```
      1      2
1 10649    711
2  1397   2303
```

Slot "Error":

```
[1] 0.14
```

Slot "Precision":

```
[1] 0.937 0.622
```

Slot "Sensitivity":

```
[1] 0.884 0.764
```

Slot "Specificity":

```
[1] 1.228 0.943
```

Slot "Chunks":

```
[1] 11 12  4  8  1
```

```
R> summary(adultcla)
```

```
Test Predictive Frequency
1      1              1    10649
2      2              1     1397
3      1              2      711
4      2              2     2303
Error = 0.14.
```

3.6.4 Plot method

4 Summary

The users of the `rebmix` package are kindly encouraged to inform the author about bugs and wishes.

5 Acknowledgement

The author thanks Branislav Panić for his contribution to the Velleman (1976) rule extension to multidimensions and for his other valuable suggestions on the package from version 2.9.0 on.

```
R> plot(adultcla, nrow = 5, ncol = 2)
```

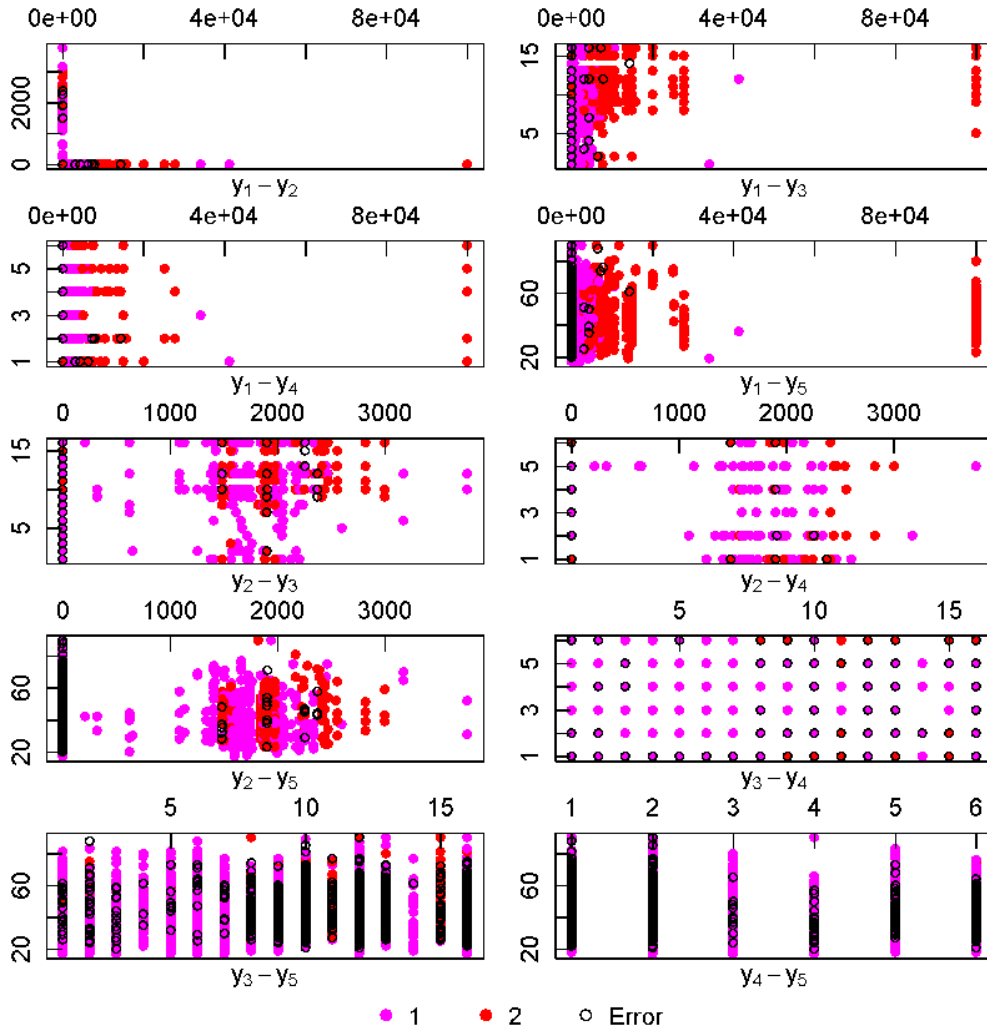


Figure 9: Dataset `adult`. Predictive class membership (coloured circles), error (black circles).

References

- E. Anderson. The species problem in iris. *Annals of the Missouri Botanical Garden*, 23(3):457–509, 1936. doi: 10.2307/2394164.
- A. Asuncion and D. J. Newman. Uci machine learning repository, 2007. URL <http://archive.ics.uci.edu/ml>.
- J. P. Baudry, A. E. Raftery, G. Celeux, K. Lo, and R. Gottardo. Combining mixture components for clustering. *Journal of Computational and Graphical Statistics*, 19(2):332–353, 2010. doi: 10.1198/jcgs.2010.08111.
- R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(3): 179–188, 1936. doi: 10.1111/j.1469-1809.1936.tb02137.x.
- C. Fraley, A. Raftery, and R. Wehrens. Incremental model-based clustering for large datasets with small clusters. *Journal of Computational and Graphical Statistics*, 14(3):529–546, 2005. doi: 10.1198/106186005X59603.
- C. Fraley, A. E. Raftery, L. Scrucca, T. B. Murphy, and M. Fop. *mclust: Normal Mixture Modelling for Model-Based Clustering, Classification, and Density Estimation*, 2016. URL <http://CRAN.R-project.org/package=mclust>. R package version 5.1.

- J. Ma, J. Liu, and Z. Ren. Parameter estimation of poisson mixture with automated model selection through byy harmony learning. *Pattern Recognition*, 42(11):2659–2670, 2009. doi: 10.1016/j.patcog.2009.03.029.
- M. Nagode. Finite mixture modeling via rebmix. *Journal of Algorithms and Optimization*, 3(2):14–28, 2015. doi: 10.5963/JAO0302001.
- M. Nagode. Multivariate normal mixture modeling, clustering and classification with the rebmix package. *ArXiv e-prints*, Jan. 2018.
- M. Nagode and M. Fajdiga. The rebmix algorithm for the univariate finite mixture estimation. *Communications in Statistics - Theory and Methods*, 40(5):876–892, 2011a. doi: 10.1080/03610920903480890.
- M. Nagode and M. Fajdiga. The rebmix algorithm for the multivariate finite mixture estimation. *Communications in Statistics - Theory and Methods*, 40(11):2022–2034, 2011b. doi: 10.1080/03610921003725788.
- P. F. Velleman. Interactive computing for exploratory data analysis i: Display algorithms. In *Proceedings of the Statistical Computing Section*, Washington, D.C., 1976. American Statistical Association.
- M. Wiper, D. R. Insua, and F. Ruggeri. Mixtures of gamma distributions with applications. *Journal of Computational and Graphical Statistics*, 10(3):440–454, 2001. URL <http://www.jstor.org/stable/1391098>.

Marko Nagode
 University of Ljubljana
 Faculty of Mechanical Engineering
 Aškerčeva 6
 1000 Ljubljana
 Slovenia
 Marko.Nagode@fs.uni-lj.si.