*Journal of Statistical Software*

*May 2021, Volume 98, Issue 3.* doi: 10.18637/jss.v098.i03

# Model-Based Clustering, Classification, and Discriminant Analysis Using the Generalized Hyperbolic Distribution: MixGHD R package

**Cristina Tortora**
San José State University

**Ryan P. Browne**
University of Waterloo

**Aisha ElSherbiny**
University of Chicago

**Brian C. Franczak**
MacEwan University

**Paul D. McNicholas**
McMaster University

**Abstract**

The **MixGHD** package for R performs model-based clustering, classification, and discriminant analysis using the generalized hyperbolic distribution (GHD). This approach is suitable for data that can be considered a realization of a (multivariate) continuous random variable. The GHD has the advantage of being flexible due to skewness, concentration, and index parameters; as such, clustering methods that use this distribution are capable of estimating clusters characterized by different shapes. The package provides five different models all based on the GHD, an efficient routine for discriminant analysis, and a function to measure cluster agreement. This paper is split into three parts: the first is devoted to the formulation of each method, extending them for classification and discriminant analysis applications, the second focuses on the algorithms, and the third shows the use of the package on real datasets.

*Keywords*: model-based clustering, classification, discriminant analysis, EM algorithm, generalized hyperbolic distribution.

## 1. Introduction

Broadly, classification refers to the process of assigning labels to sets of observations. In general, classification is unsupervised (also known as clustering), semi-supervised, or (fully) supervised. Generally speaking, the goal is the same, to group observations based on shared characteristics. Classifying, in fact, is a key instrument in data mining and data analysis.

Classification can serve the twofold aim of highlighting discriminating factors and grouping homogeneous collections of units in datasets. The latter point is extremely useful in many fields such as medicine, e.g., for identifying homogeneous groups of patients, or marketing, e.g., identifying homogeneous groups of customers. This main focus of this paper is cluster analysis but the described methods can be used for semi-supervised and supervised learning as well. Many cluster analysis techniques exist in the statistical and machine learning literature, in this paper we will focus on a non-hierarchical clustering technique known as model-based clustering (McNicholas 2016).

Of course, not all non-hierarchical clustering techniques are model-based and these are distinguished by not making any explicit assumptions on the distribution of the clusters. Typically, they group statistical units into $k$ clusters with respect to a distance measure. The most common method in this context is $k$-means clustering (MacQueen 1967). Several extensions of $k$-means for high-dimensional data clustering exist (e.g., Bock 1987; De Sarbo and Manrai 1992; Arabie and Hubert 1994; De Soete and Carroll 1994; Stute and Zhu 1995; Vichi and Kiers 2001; Vichi and Saporta 2009; Yamamoto and Hwang 2014). An alternative distance-based method is probabilistic distance (PD) clustering (Ben-Israel and Iyigun 2008), which assigns units to a cluster according to their probability of membership, under the constraint that the product of the probability and the distance of each point to any cluster center is a constant. Tortora, Gettler Summa, Marino, and Palumbo (2016a) propose a transformation of the method for high-dimensional data sets, Rainey, Tortora, and Palumbo (2019) and Tortora, McNicholas, and Palumbo (2020a) propose a new distance measure.

Model-based methods assume that a population is a convex linear combination of a finite number of (component) probability densities. Until recently, the component densities have typically been Gaussian distributed, and several parsimonious extensions of Gaussian mixtures for high-dimensional data have been proposed (e.g., Ghahramani and Hinton 1997; McLachlan, Peel, and Bean 2003; Bouveyron, Girard, and Schmid 2007; McNicholas and Murphy 2008, 2010; Baek, McLachlan, and Flack 2010; Montanari and Viroli 2011). Recently, the focus of the literature has been on mixtures of non-Gaussian distributions for high-dimensional datasets (e.g., Andrews and McNicholas 2011a,b; Steane, McNicholas, and Yada 2012; Lin, McNicholas, and Hsiu 2014; Murray, McNicholas, and Browne 2014b; Murray, Browne, and McNicholas 2014a; Lin, McLachlan, and Lee 2016; McNicholas, McNicholas, and Browne 2017; Tang, Browne, and McNicholas 2018; Kim and Browne 2019; Murray, Browne, and McNicholas 2020; Punzo, Blostein, and McNicholas 2020). Of particular interest is the generalized hyperbolic distribution (GHD) which can detect clusters with non-elliptical form because it contains skewness, concentration, and index parameters. These parameters allow the GHD to be much more flexible compared to most other distributions. Browne and McNicholas (2015) examine different representations of the GHD and outline a mixture of GHDs for clustering. Each component scale matrix has a number of free parameters that increases quadratically in the number of variables $p$. Tortora, McNicholas, and Browne (2016b) propose a parsimonious version of the model, the mixture of generalized hyperbolic factor analyzers, to extend the method for higher dimensional data sets. A multiple scaled extension of the method was proposed by Tortora, Franczak, Browne, and McNicholas (2019), where the authors added even more flexibility to the models letting the concentration and index parameters vary per dimension.

The volume of work on clustering and classification methodology has led to the release of new clustering software. A commonly used statistical software is R (R Core Team 2021),

and many of the previously cited methods have a corresponding R package. For example, $k$-means clustering is directly implemented in R through the **stats** package (R Core Team 2021), specifically with the `kmeans` function. Two packages that are worth mentioning, because they implement several techniques useful for cluster visualization and for the choice of the number of clusters together with some basic clustering methods, are **cluster** (Maechler, Rousseeuw, Struyf, Hubert, and Hornik 2021) and **fpc** (Hennig 2020). Some of the extensions of $k$-means for high-dimensional datasets can be found in the **clustrd** package (Markos, Iodice D'Enza, and Van de Velden 2019). PD-clustering and its extension are implemented in the package **FPDcluster** (Tortora, Vidales, Palumbo, and McNicholas 2020b). Among a large variety of packages available for model-based clustering is the widely used **mclust** package (Scrucca, Fop, Murphy, and Raftery 2016) and an analogue in parallel **pmclust** (Chen and Ostrouchov 2021). The two packages implement model-based clustering, classification, and density estimation using the Gaussian distribution. An alternative for model-based clustering using the Gaussian distribution is the **Rmixmod** package (Lebret, Iovleff, Langrognet, Biernacki, Celeux, and Govaert 2015), an R interface for the **MixMod** software (Biernacki, Celeux, Govaert, and Langrognet 2006). A third alternative is the package **mixture** (Pocuca, Browne, and McNicholas 2021), it carries out model-based clustering and classification using the 14 parsimonious Gaussian clustering models from Celeux and Govaert (1995). Several existing packages for clustering high-dimensional datasets use the Gaussian distribution, each implementing a different model. The **pgmm** package (McNicholas, ElSherbiny, McDaid, and Murphy 2019) implements the 12 parsimonious Gaussian mixture models for cluster analysis from McNicholas and Murphy (2008, 2010) and an associated classification model (see McNicholas 2010). **HDclassif** (Bergé, Bouveyron, and Girard 2012) and **FisherEM** (Bouveyron and Brunet 2020) implement the models described in Bouveyron *et al.* (2007) and Bouveyron and Brunet (2012), respectively.

The **EMMIXskew** package (Wang, Ng, and McLachlan 2018) implements model-based clustering using the normal, the Student-$t$, the skew normal, and the skew-$t$ distributions, while the **EMMIXuskew** package (Lee and McLachlan 2014b) implements model-based clustering using the unrestricted skew $t$ distribution given in Lee and McLachlan (2014a). The package **uskewFactors** (Murray, Browne, and McNicholas 2016) implements the mixtures of unrestricted skew-$t$ factor analyzers. An alternative to the common paradigms is proposed by Azzalini and Torelli (2007), who use a clustering method based on nonparametric density estimation. The corresponding package is **pdfclust** (Menardi and Azzalini 2014). For large and sparse data sets, mixtures of von Mises-Fisher distributions can be fit using the package **movMF** (Hornik and Grün 2014). The two packages **flexmix** (Leisch 2004; Grün and Leisch 2008) and **mixtools** (Benaglia, Chauveau, Hunter, and Young 2009) allow the user to choose different distributions. Specifically, **flexmix** is extremely flexible letting the user input the chosen distribution. For a list of R packages on cluster analysis and finite mixture models see Leisch and Grün (2021).

The aim of this paper is to describe the **MixGHD** package (Tortora, El-Sherbiny, Browne, Franczak, and McNicholas 2021) which implements five different methods based on the GHD. The package is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=MixGHD`. As mentioned before, the GHD is a very flexible distribution that has many other distributions as special or limiting cases. For these reasons, this package fills in the gap in the existing package landscape. Moreover, in this paper, the three methods proposed in Tortora *et al.* (2019) are extended to be used for discriminant

analysis and model-based classification.

This paper has the following structure. In Section 2, we introduce model-based classification. Sections 3 to 5 describe the five methods implemented in the **MixGHD** package, with some implementation details described in Section 6. Section 7 describes the **MixGHD** package with real data examples.

## 2. Model-based classification

The basic idea of model-based clustering is that a random vector $\mathbf{X}$ follows a (parametric) finite mixture distribution if, for all $\mathbf{x} \subset \mathbf{X}$, its density can be written as

$$f(\mathbf{x} \mid \boldsymbol{\vartheta}) = \sum_{g=1}^{G} \pi_g f_g(\mathbf{x} \mid \boldsymbol{\theta}_g),$$

where $G$ is the number of clusters, $\pi_g > 0$ is the $g$th mixing proportion such that $\sum_{g=1}^{G} \pi_g = 1$, $f_g(\mathbf{x} \mid \boldsymbol{\theta}_g)$ is the $g$th component density that we assume to be of the same type for all the components, i.e., $f_g(\mathbf{x} \mid \boldsymbol{\theta}_g) = f(\mathbf{x} \mid \boldsymbol{\theta}_g)$. Therefore, the model-based clustering likelihood function, for $\mathbf{x}_1, \ldots, \mathbf{x}_n$, can be written as

$$\mathcal{L}(\boldsymbol{\vartheta}) = \prod_{j=1}^{n} \sum_{g=1}^{G} \pi_g f(\mathbf{x}_j \mid \boldsymbol{\theta}_g). \tag{1}$$

In model-based classification, given $n$ $p$-dimensional vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$, $k$ of them have known labels and the model can be used to predict the other $n - k$ labels. Following McNicholas (2010), order the $n$ observations so that the first $k$ are labeled — this can be done without loss of generality. Let $G$ be the number of classes, $H \geq G$ be the number of fitted components, and $z_{ig}$ the component membership labels so that $z_{ig} = 1$ if $\mathbf{x}_i$ is in component $g$, and $z_{ig} = 0$ otherwise, for $i = 1, \ldots, k$ and $g = 1, \ldots, G$. The model-based classification likelihood is

$$\mathcal{L}(\boldsymbol{\vartheta}) = \prod_{i=1}^{k} \prod_{g=1}^{G} [\pi_g f(\mathbf{x}_i \mid \boldsymbol{\theta}_g)]^{z_{ig}} \prod_{j=k+1}^{n} \sum_{h=1}^{H} \pi_h f(\mathbf{x}_j \mid \boldsymbol{\theta}_h). \tag{2}$$

Note that $H \geq G$ in general, but it is typically assumed that $H = G$.

Discriminant analysis is a special case of classification in which $k = n$ and, therefore, we only use the first part of (2). Cluster analysis in (1) can be obtained setting $k = 0$, in which case we use only the second part of Equation (2); see McNicholas (2010) for details. In the following, we will consider the GHD density function. Extensive details on model-based clustering, classification, and discriminant analysis are given by McNicholas (2016).

## 3. Mixture of generalized hyperbolic distributions

A random $p$-dimensional variable $\mathbf{X}$ is distributed according to a GHD if its density can be represented as

$$f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}, \omega, \eta, \lambda) = \int_0^{\infty} \phi_p\left(\mathbf{x} \mid \boldsymbol{\mu} + v\boldsymbol{\alpha}, v\boldsymbol{\Sigma}\right) h\left(v \mid \omega, \eta, \lambda\right) dv, \tag{3}$$

where $\phi_p$ is a multivariate $p$ dimensional Gaussian distribution and $h\left(v \mid \omega, \eta, \lambda\right)$, called the weight function, is the density of a univariate generalized inverse Gaussian (GIG) distribution. Formally, the density of the GIG distribution is given by

$$h\left(v \mid \omega, \eta, \lambda\right) = \frac{(v/\eta)^{\lambda-1}}{2\nu K_\lambda(\omega)} \exp\left\{-\frac{\omega}{2}\left(\frac{v}{\eta} + \frac{\eta}{v}\right)\right\}, \tag{4}$$

where $\eta > 0$ is a scale parameter, $\omega > 0$ is a concentration parameter, $\lambda \in \mathbb{R}$ is an index parameter, and $K_\lambda$ is the modified Bessel function of the third kind with index $\lambda$.

Browne and McNicholas (2015) propose an identifiable representation of the GHD by setting $\eta = 1$, which gives

$$\begin{aligned} f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}, \omega, \lambda) &= \int_0^\infty \phi_p\left(\mathbf{x} \mid \boldsymbol{\mu} + v\boldsymbol{\alpha}, v\boldsymbol{\Sigma}\right) h\left(v \mid \omega, 1, \lambda\right) dv \\ &= \left[\frac{\omega + \delta(\mathbf{x}, \boldsymbol{\mu} \mid \boldsymbol{\Sigma})}{\omega + \boldsymbol{\alpha}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}}\right]^{\frac{\lambda-\frac{p}{2}}{2}} \frac{K_{\lambda-\frac{p}{2}}\left(\sqrt{[\omega + \boldsymbol{\alpha}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}][\omega + \delta(\mathbf{x}, \boldsymbol{\mu} \mid \boldsymbol{\Sigma})]}\right)}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}} K_\lambda(\omega) \exp\left\{-(\boldsymbol{\mu} - \mathbf{x})^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}\right\}}, \end{aligned} \tag{5}$$

where $\delta\left(\mathbf{x}, \boldsymbol{\mu} \mid \boldsymbol{\Sigma}\right) = \left(\mathbf{x} - \boldsymbol{\mu}\right)^\top\boldsymbol{\Sigma}^{-1}\left(\mathbf{x} - \boldsymbol{\mu}\right)$ is the squared Mahalanobis distance between $\mathbf{x}$ and location parameter $\boldsymbol{\mu} \in \mathbf{R}^p$, $\boldsymbol{\alpha} \in \mathbf{R}^p$ is a skewness parameter, $\boldsymbol{\Sigma}$ is a $p \times p$ positive defined scale matrix, and $K_\lambda$, $\omega$, and $\lambda$ are as defined for (4).

The random variable $\mathbf{X}$ can be generated via the relationship

$$\mathbf{X} = \boldsymbol{\mu} + V\boldsymbol{\alpha} + \sqrt{V}\mathbf{N}, \tag{6}$$

where $\mathbf{N} \sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Sigma})$ and $V \sim \text{GIG}\left(\omega, 1, \lambda\right)$, i.e., $V$ follows a GIG distribution with density as in (4). It follows that

$$\mathbf{X} \mid V = v \sim \mathcal{N}_p\left(\boldsymbol{\mu} + v\boldsymbol{\alpha}, v\boldsymbol{\Sigma}\right). \tag{7}$$

A finite mixture of GHDs (MGHD) has density

$$f_{\text{MGH}}(\mathbf{x} \mid \boldsymbol{\vartheta}) = \sum_{g=1}^{G} \pi_g f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g, \boldsymbol{\alpha}_g, \omega_g, \lambda_g),$$

where $f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g, \boldsymbol{\beta}_g, \omega_g, \lambda_g)$ is the density of the GHD given in (5) and, as before, $\pi_g$ is the $g$th mixing proportion.

## 4. Mixture of generalized hyperbolic factor analyzers

In the MGHD, the scale matrices $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G$ contain $Gp(p+1)/2$ free parameters, i.e., a number that is quadratic in $p$. When $p$ is large, the number of parameters to estimate becomes too big, so to overcome this issue Tortora *et al.* (2016b) propose the mixture of generalized hyperbolic factor analyzers (MGHFA). In a factor analyzers model (Ghahramani and Hinton 1997; McLachlan and Peel 2000), the random variable $\mathbf{X}$ can be represented as

$$\mathbf{X}_i = \boldsymbol{\mu}_g + \boldsymbol{\Lambda}_g\mathbf{U}_{ig} + \boldsymbol{\epsilon}_{ig} \tag{8}$$

with probability $\pi_g$, for $i = 1, \ldots, n$ and $g = 1, \ldots, G$. The matrix $\boldsymbol{\Lambda}_g$ is a $p \times q$ matrix of factor loadings. The factors $\mathbf{U}_{ig}$ are independently distributed $\mathbf{U}_{ig} \sim \mathcal{N}_q(\mathbf{0}, \mathbf{I}_q)$ with $q < p$,

independently of $\boldsymbol{\epsilon}_{ig} \sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Psi}_g)$, which are also independently distributed, where $\boldsymbol{\Psi}_g$ is a $p \times p$ diagonal matrix with positive diagonal entries. The marginal distribution of $\mathbf{X}_i$ from model (8) is $\mathcal{N}_p(\boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g^\top + \boldsymbol{\Psi}_g)$. Consider (6) and note that $\mathbf{N}$ can be decomposed as $\mathbf{N} = \boldsymbol{\Lambda}\mathbf{U} + \boldsymbol{\epsilon}$, where $\mathbf{U} \sim \mathcal{N}_q(\mathbf{0}, \mathbf{I}_q)$ and $\boldsymbol{\epsilon} \sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Psi})$, and $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}$ are a $p \times q$ factor loading matrix and a $p \times p$ diagonal matrix with positive entries, respectively. From (7), it follows that

$$\mathbf{X} \mid V = v \sim \mathcal{N}_p(\boldsymbol{\mu} + v\boldsymbol{\alpha}, v(\boldsymbol{\Lambda}\boldsymbol{\Lambda}^\top + \boldsymbol{\Psi})).$$

This leads to a mixture of generalized hyperbolic factor analyzers model with density

$$f_{\text{MGHFA}}(\mathbf{x} \mid \boldsymbol{\vartheta}) = \sum_{g=1}^{G} \pi_g f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g^\top + \boldsymbol{\Psi}_g, \boldsymbol{\alpha}_g, \lambda_g, \omega_g),$$

where $f_{\text{GH}}(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g^\top + \boldsymbol{\Psi}_g, \boldsymbol{\alpha}_g, \lambda_g, \omega_g)$ is the density of the GHD given in (5) and $\pi_g$ are the mixing proportions.

## 5. Extensions of the generalized hyperbolic distribution

The multiple scaled distributions are an extension of the distribution of the type in (3), where the weight function is the product of $p$ univariate functions (Forbes and Wraith 2014). This transformation can be obtained by letting $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Phi}\boldsymbol{\Gamma}^\top$ and adding $\boldsymbol{\Delta}_{\mathbf{v}} = \text{diag}\left(v_1^{-1}, \ldots, v_p^{-1}\right)$, so that the density function of a multiple scaled GHD (MSGHD) is

$$f_{\text{MSGH}}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \boldsymbol{\omega}, \boldsymbol{\lambda}) = \int_0^\infty \cdots \int_0^\infty \phi_p\left(\boldsymbol{\Gamma}^\top\mathbf{x} - \boldsymbol{\mu} - \boldsymbol{\Delta}_{\mathbf{v}}\boldsymbol{\alpha} \mid \mathbf{0}, \boldsymbol{\Delta}_{\mathbf{v}}\boldsymbol{\Phi}\right) \times h_{\mathbf{V}}(v_1, \ldots, v_p \mid \boldsymbol{\omega}, \boldsymbol{\lambda})\, dv, \tag{9}$$

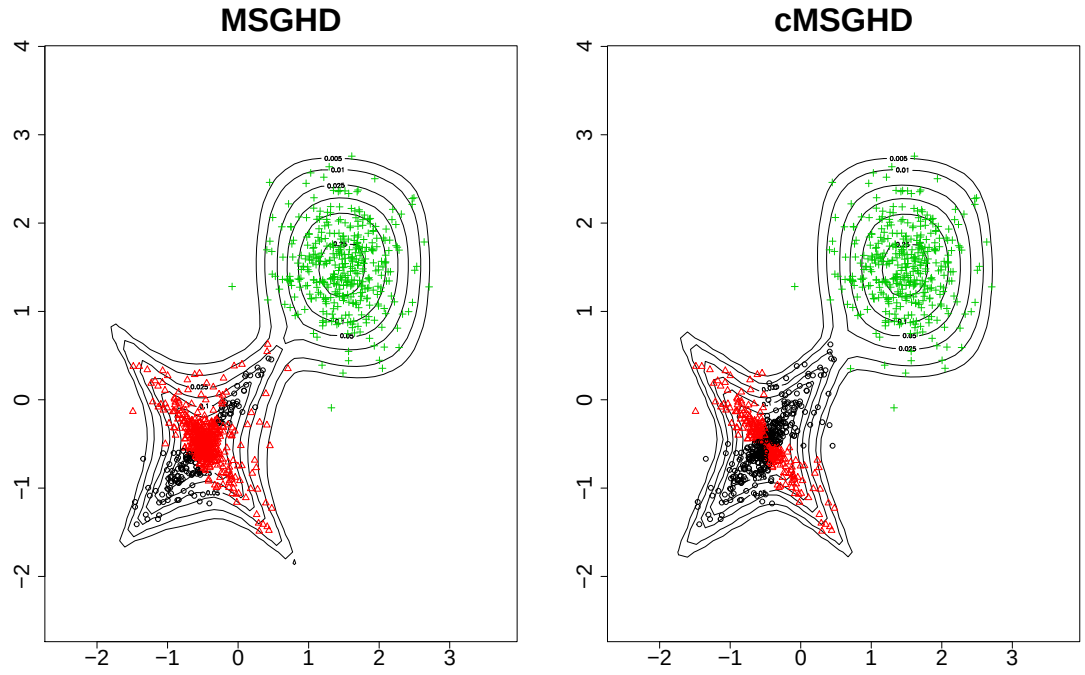
where $h_{\mathbf{v}}(v_1, \ldots, v_p \mid \boldsymbol{\theta}) = h(v_1 \mid \boldsymbol{\theta}_1) \times \cdots \times h(v_p \mid \boldsymbol{\theta}_p)$ is a $p$-dimensional density such that the random variables $V_1, \ldots, V_p$ are independent (Tortora *et al.* 2019). A finite mixture of multiple scaled GHDs (MMSGHDs) has density

$$f_{\text{MMSGH}}(\mathbf{x} \mid \boldsymbol{\vartheta}) = \sum_{g=1}^{G} \pi_g f_{\text{MSGH}}\left(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Gamma}_g, \boldsymbol{\Phi}_g, \boldsymbol{\alpha}_g, \boldsymbol{\omega}_g, \boldsymbol{\lambda}_g\right).$$

The MSGHD is not convex nor quasi-convex, and consequently there are situations in which the contour plots are not convex. In some situations, see Figure 1a and 1b, a convex contour plot can be more suitable. For this reason, Tortora *et al.* (2019) propose the convex MMSGHD (cMMSGHD). A convex contour plot can be ensured by adding a constraint to the index parameter $\lambda$, i.e., $\lambda \geq 1$, see Tortora *et al.* (2019) for details. The GHD cannot be obtained as a special or limiting case of the MSGHD and vice versa. For this reason, Tortora *et al.* (2019) propose the mixture of coalesced GHDs (MCGHD). A random variable $\mathbf{X}$ follows a CGHD if it can be modeled as follows

$$\mathbf{X} = \varpi\mathbf{R} + (1 - \varpi)\mathbf{S},$$

where $\varpi \in (0, 1)$, $\mathbf{S}$ is distributed according to a MSGHD $f_{\text{MSGH}}(\boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \boldsymbol{\omega}, \boldsymbol{\lambda})$, and $\mathbf{R} = \boldsymbol{\Gamma}\mathbf{Y}$ where $\mathbf{Y}$ is distributed according to a GHD $f_{\text{GH}}(\boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \omega_0, \lambda_0)$, where $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Phi}\boldsymbol{\Gamma}'$.




(a) Results obtained with MSGHDs. (b) Results obtained with cMSGHDs.

Figure 1: Cluster partitions obtained using the MMSGHDs and cMMSGHDs, respectively, on a three-component simulated dataset.

This implies that the density of a MCGHD is

$$f_{\text{CGH}}\left(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \boldsymbol{\omega}, \boldsymbol{\lambda}, \omega_0, \lambda_0, \varpi\right) = \varpi f_{\text{GH}}\left(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \omega_0, \lambda_0\right) + (1-\varpi) f_{\text{MSGH}}\left(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}, \boldsymbol{\alpha}, \boldsymbol{\omega}, \boldsymbol{\lambda}\right). \tag{10}$$

It is worth nothing that the main difference from the GHD is that the MSGHD parameters $\boldsymbol{\omega}_g$ and $\boldsymbol{\lambda}_g$ are $p$-dimensional vectors whereas the GHD parameters $\omega_{0g}$ and $\lambda_{0g}$ are unidimensional. The advantage of the MCGHD is that it includes both the GHD and the MSGHD as special cases.

# 6. Implementation

## 6.1. Overview

The most widely used technique for parameter estimation in a finite mixture model context is the expectation-maximization (EM) algorithm (Dempster, Laird, and Rubin 1977), which is an iterative technique for finding maximum likelihood estimates when data are incomplete or treated as incomplete. The EM algorithm iterates between two steps, an E-step and an M-step. On the E-step, the (conditional) expected value of the complete-data log-likelihood, $\mathcal{Q}$ say, is calculated. On the M-step, $\mathcal{Q}$ is maximized with respect to the parameter estimates.

The E-step and the M-step are iterated until convergence is reached; see Section 6.2 for details about stopping rules.

For the MGHFA, the parameters are estimated using an extension of the EM algorithm called the alternating expectation-conditional maximization (AECM) algorithm (Meng and Van Dyk 1997). Similar to the EM algorithm, it is based on the complete-data log-likelihood, but it allows for the specification of different complete-data at each stage of the algorithm and the M-step is replaced by a number of conditional maximization (CM) steps. For details on parameter estimation in the MGHFA, refer to Tortora *et al.* (2016b). For the MMSGHD, cMMSGHD and MCGHD, $\boldsymbol{\Gamma}$ cannot be found in closed form and an optimization routine is used. The result is that, in each M-step, the likelihood increases with respect to $\boldsymbol{\Gamma}$ but it is not maximized; accordingly, the algorithm is formally a generalized EM (GEM) algorithm. For details on parameter estimation, refer to Tortora *et al.* (2019).

## 6.2. Model selection, convergence, and evaluation

The five models require the choice of the number of components $G$ and the MGHFA requires the choice of the number of factors $q$. For both choices, the package offers four different criteria: the Akaike information criterion (AIC; Akaike 1974), the AIC3 (Bozdogan 1993), the Bayesian information criterion (BIC; Schwarz 1978), and the integrated completed likelihood (ICL; Biernacki, Celeux, and Govaert 2000). Write $l(\hat{\boldsymbol{\vartheta}})$ and $\hat{\boldsymbol{\vartheta}}$ to denote the maximized log-likelihood and the vector of parameters that maximizes the log-likelihood, respectively, and let $\rho$ denote the number of free parameters. When the algorithm converges, we compute $\hat{z}_{ig}$ as the a posteriori expected value of $z_{ig}$ and the maximum *a posteriori* (MAP) classification values using the final $\hat{z}_{ig}$; $\text{MAP}\{\hat{z}_{ig}\} = 1$ if $\max_h\{\hat{z}_{ih}\}$ occurs in component $h = g$, and $\text{MAP}\{\hat{z}_{ig}\} = 0$ otherwise. The various criteria are given as follows:

$$\text{AIC} = 2l(\hat{\boldsymbol{\vartheta}}) - 2\log n, \qquad \text{AIC3} = 2l(\hat{\boldsymbol{\vartheta}}) - 3\log n, \qquad \text{BIC} = 2l(\hat{\boldsymbol{\vartheta}}) - \rho\log n,$$

$$\text{ICL} \approx 2l(\hat{\boldsymbol{\vartheta}}) - \rho\log n + 2\sum_{i=1}^{n}\sum_{g=1}^{G}\text{MAP}\{\hat{z}_{ig}\}\log\hat{z}_{ig},$$

where $\sum_{i=1}^{n}\sum_{g=1}^{G}\text{MAP}\{\hat{z}_{ig}\}\log\hat{z}_{ig}$ is the estimated mean entropy.

The EM algorithm, the AECM algorithm, and the GEM algorithm used for the parameter estimation of the models are iterated until convergence is reached. The convergence is determined using a stopping rule based on the Aitken acceleration (Aitken 1926). Let $l^{(k)}$ be the value of the log-likelihood after $k$ iterations. The asymptotic maximum of the log-likelihood at iteration $k$ can be estimated using the Aitken acceleration via

$$a^{(k)} = \frac{l^{(k+1)} - l^{(k)}}{l^{(k)} - l^{(k-1)}}.$$

An asymptotic estimate of the log-likelihood at iteration $k+1$ is

$$l_{\infty}^{(k+1)} = l^{(k)} + \frac{1}{1-a^{(k)}}\left(l^{(k+1)} - l^{(k)}\right),$$

and we consider the algorithm to have converged if

$$l_{\infty}^{(k+1)} - l^{(k)} \in (0, \epsilon),$$

where $\epsilon$ is small (McNicholas, Murphy, McDaid, and Frost 2010).

The adjusted Rand index (ARI; Hubert and Arabie 1985), which compares predicted classifications with true classifications, can be used to evaluate the results. The ARI corrects the Rand index (Rand 1971) for chance; its expected value under random classification is 0 , and it takes a value 1 when there is perfect class agreement. Steinley (2004) gives guidelines for interpreting ARI values. For more pairwise agreement indices see the `cl_agreement` function in the **CLUE** package (Hornik 2005).

# 7. MixGHD R package

MixGHD is an R package developed in an object-oriented design using the standard S4 paradigm and C programming language. The package contains five functions for model-based clustering and classification: `MGHD`, `MGHFA`, `MSGHD`, `cMSGHD`, and `MCGHD`. The `DA` function is a routine for discriminant analysis, the `ARI` function computes the adjusted Rand index, and the `contourpl` function produce a contour plot. The package also contains the functions `rGHD`, `rMSGHD`, and `rMCGHD`, to pseudo-randomly generate numbers from the corresponding distributions, and the functions `dGHD`, `dMSGHD`, and `dMCGHD` to compute the density of the corresponding distributions. Table 1 shows the input arguments for the `MGHD`, `MGHFA`, `MSGHD`, `cMSGHD`, and `MCGHD` functions with a brief description.

## 7.1. Cluster analysis

To illustrate the use of the package, we use the `bankruptcy` dataset (Alman 1968) from the **MixGHD** package . The dataset contains the ratio of retained earnings (RE) to total assets as well as the ratio of earnings before interests and taxes (EBIT) to total assets of 66 American firms. Half of the selected firms had filed for bankruptcy.

```
R> library("MixGHD")
R> data("bankruptcy", package = "MixGHD")
R> res <- MCGHD(data = bankruptcy[, 2:3], G = 2:3, method = "kmedoids",
+    max.iter = 1000, modelSel = "BIC")

The best model (BIC) for the range of components used is G = 2.
The BIC for this model is -288.7835.

R> summary(res)

The number of components used for the model is G = 2.
BIC = -288.7835. AIC = -238.4214. AIC3 = -261.4214. ICL = -294.7374.
  Cluster N. of elements
1       1           36
2       2           30
```

The variables RE and EBIT are considered for cluster analysis. The BIC criterion is used to select between $G = 2$ or $G = 3$, the maximum number of iterations is 1000, and $k$-medoids is used as the starting criterion.

| Arguments | Description |
|---|---|
| `data` | An $n \times p$ matrix or data frame such that rows correspond to observations and columns correspond to variables. |
| `gpar0` | An optional list containing the initial parameters of the mixture model. If specified, it must have a list structure containing as many elements as the number of components $G$. Each element must include all the parameters for the selected model. |
| `G` | A numerical vector giving a range of values for the number of components/clusters; if not specified, $G = 2$. |
| `max.iter` | An optional numerical parameter giving the maximum number of iterations each EM algorithm is allowed to use; 100 by default. |
| `label` | An optional $n$ dimensional vector. If `label[i] = k`, then observation $i$ belongs to group $k$; If `label[i] = 0`, then observation $i$ is unlabeled; if `NULL`, then the data have no known groups. |
| `eps` | An optional number specifying the epsilon value for the convergence criteria used in the EM algorithms; see Section 6.2. |
| `method` | An optional string indicating the initialization criterion; if not specified $k$-means clustering is used. Alternative methods are hierarchical `"hierarchical"`, $k$-medoids `"kmedoids"`, random `"random"`, and model-based `"modelBased"` clustering. |
| `nr` | An optional number indicating the number of starting values when random is used, 10 by default. |
| `scale` | An optional logical value indicating whether or not the data should be scaled; true by default. |
| `modelSel` | An optional string indicating the model selection criterion; if not specified, the AIC is used. Alternative methods are the BIC, ICL, and AIC3. |
| `q` | Only when MGHFA is used, a numerical vector specifying the number of latent factors; $q = 2$ by default. |

Table 1: Arguments for the `MGHD`, `MGHFA`, `MSGHD`, `cMSGHD`, and `MCGHD` functions.

The function `summary` shows the value of the BIC, AIC, AIC3, and ICL and the number of elements in each cluster. The output is an S4 object of class '`MixGHD`' containing the following parameters:

- `index`: Value of the index used for model selection for each model, BIC in this case.
- `AIC`: Akaike information criterion.
- `AIC3`: Akaike information criterion 3.
- `BIC`: Bayesian information criterion.
- `ICL`: Integrated completed likelihood.
- `gpar`: A list of the model parameters in the rotated space.
- `loglik`: The log-likelihood values.

- `map`: A vector of integers indicating the maximum *a posteriori* classifications for the best model.
- `par`: A list of the model parameters.
- `z`: A matrix giving the raw values upon which map is based.

For each component, the estimated parameters are stored in the list `gpar`,

```
R> ls(res@gpar[[1]])

[1] "alpha" "cpl"   "cpl0"  "gam"   "mu"    "phi"   "wg"

R> ls(res@gpar[[2]])

[1] "alpha" "cpl"   "cpl0"  "gam"   "mu"    "phi"   "wg"
```

using the function `ARI` we can measure the accuracy of the classification, the vector map contains the membership for each unit.

```
R> ARI(res@map, bankruptcy[, 1])

[1] 0.8237573

R> table(res@map,bankruptcy[, 1])

     0  1
  1 33  3
  2  0 30
```

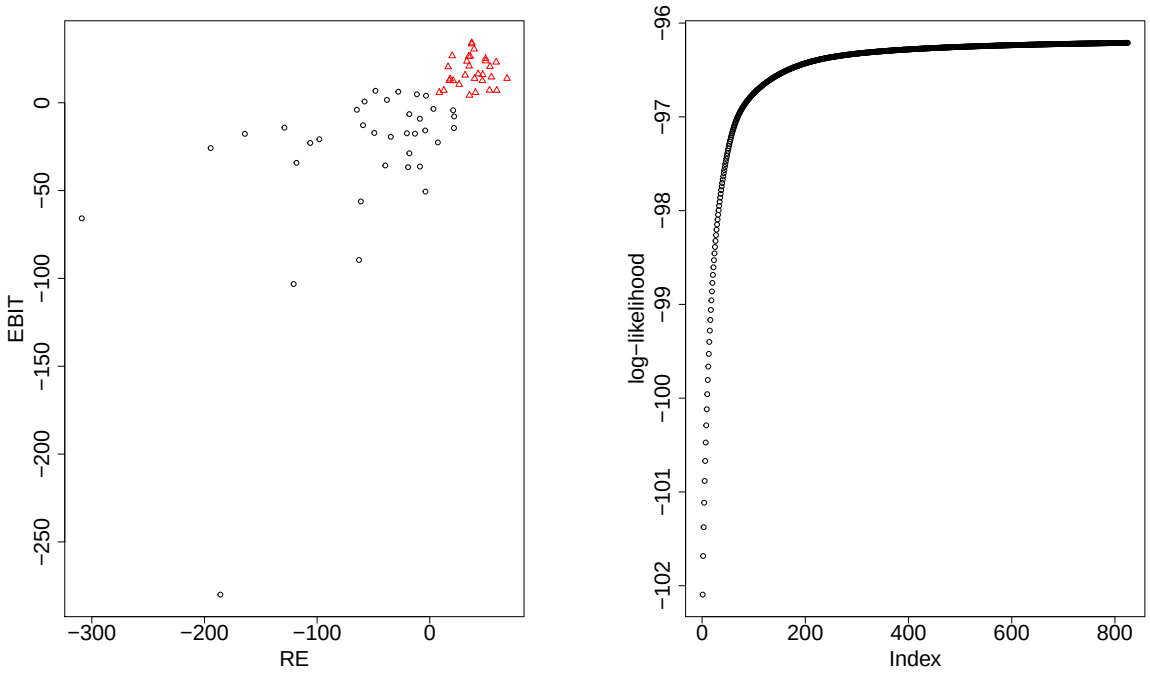
The MCGHD has good performance on the bankruptcy dataset, with an ARI of 0.824 and only three misclassifications. Figure 2a shows the obtained partition. The cluster represented by *o* is characterized by skewness in both directions, which makes it hard to be identified by less flexible clustering methods. Figure 2b shows the value of the log-likelihood at each iteration of the EM algorithm. For comparison, the MGHD is applied on the same dataset.

```
R> res1 <- MGHD(data = bankruptcy[,2:3], G = 2:3, method = "kmedoids",
+    max.iter = 1000, modelSel = "BIC")

R> ARI(res1@map, bankruptcy[, 1])

[1] 0.01863933
```

One of the clusters is characterized by two outliers in two different directions, and this characteristic affects the performance of the MGHD with an ARI close to zero. Figures 3a and3b show the contour plots obtained using the MGHD and the MMCGHD, respectively, which were obtained using the following commands:




(a) Clusters partition. (b) Log-likelihood at each iteration

Figure 2: Results obtained using MCGHD on the bankruptcy dataset.

```
R> plot(res1)
R> plot(res)
```

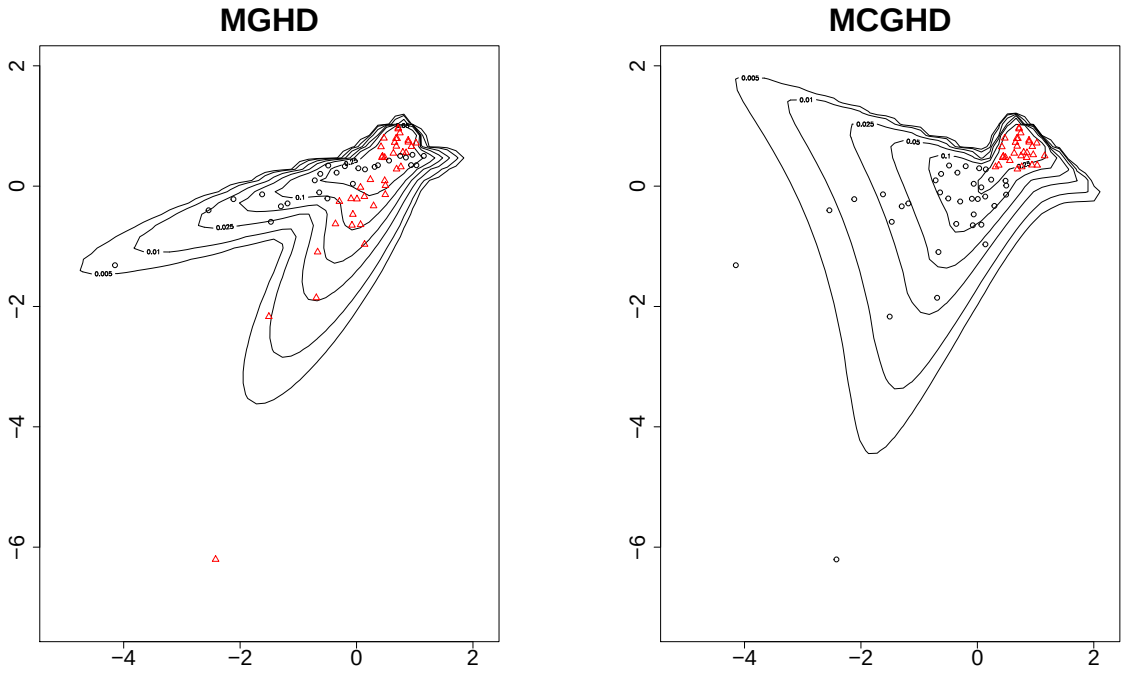
### 7.2. Data generation and density estimation

For the three density functions: GHD, MSGHD, and MCGHD, the package also contains functions to pseudo-randomly generate data (`rGHD`, `rMSGHD`, and `rMCGHD`) and for density estimates (`dGHD`, `dMSGHD`, and `dMCGHD`). The input of the functions are described in Table 2.

The output of the random generation functions are pseudo randomly generated $n \times p$ datasets, the output of the d functions are numerical vectors with the density values. The following examples show the use of the `rMCGHD` and `dMCGHD` function, the use of the other functions is analogues.

```
R> set.seed(12345)
R> data1 <- rCGHD(n = 600, p = 2)
R> set.seed(12345)
R> data2 <- rCGHD(n = 600, p = 2, alpha = c(2, -2), omegav = c(2, 2),
+    omega = 3, lambdav = c(0.7, 0.9))
R> densities <- dCGHD(data2, p = 2, alpha = c(2, -2), omegav = c(2, 2),
+    omega = 3, lambdav = c(0.7, 0.9))
R> head(densities, n = 3)

[1] 0.03646365 0.03328875 0.04613655
```




(a) Contour plot of the MGHD.

(b) Contour plot of the MMCGHD.

Figure 3: Contour plots for the bankruptcy data, with symbols denoting predicted classifications.

Figures 4a and 4b show the datasets obtained using the `rCGHD` function. commands.

### 7.3. Discriminant analysis

To easily perform discriminant analysis, the package contains a routine called `DA`. The `DA` function requires the input arguments in Table 1, with the exception of the data and labels that are substituted by the input parameters in Table 3.

Discriminant analysis requires the dataset to be divided into a training set and a test set, where $n_1$ and $n_2$ are the number of units in the training and test sets respectively. The input parameters change according to the chosen method. The outputs are:

- `model`: A list with the model parameters.
- `testMembership`: A vector of integers indicating the membership of the units in the test set.
- `ARItest` : A value indicating the adjusted Rand index for the test set.
- `ARItrain` : A value indicating the adjusted Rand index for the training set.

We applied the `DA` routine to the `sonar` dataset from the **MixGHD** R package. The data report the patterns obtained by bouncing sonar signals at various angles and under various conditions. There are 208 patterns in all: 111 obtained by bouncing sonar signals off a metal

| Arguments | Description |
|---|---|
| `data` | (Only for density estimates) A $n \times p$ dataset. |
| `n` | (Only for pseudo random number generation) number of observations to generate. |
| `p` | Number of variables. |
| `mu` | An optional $p$ dimensional numerical parameter giving the mean of the distribution; 0 by default. |
| `alpha` | An optional $p$ dimensional numerical parameter giving the skewness of the distribution; 0 by default. |
| `sigma` | An optional $p \times p$ symmetric scale matrix; identity matrix by default. |
| `omega` | An optional numerical parameter giving the concentration of the distribution; 1 by default. Only for the GHD and CGHD. |
| `lambda` | An optional numerical parameter giving the index of the distribution; 0.5 by default. Only for the GHD and CGHD. |
| `omegav` | An optional $p$ dimensional numerical parameter giving the concentration vector of the distribution; vector of 1s by default. Only for the MSGHD and CGHD. |
| `lambdav` | An optional $p$ dimensional numerical parameter giving the index vector of the distribution; vector of 0.5s by default. Only for the MSGHD and CGHD. |
| `gam` | An optional $p \times p$ $\boldsymbol{\Gamma}$ matrix. Only for the MSGHD and CGHD. |
| `phi` | An optional $p$ dimensional vector $\boldsymbol{\Phi}$. Only for the MSGHD and CGHD. |
| `wg` | An optional numerical parameter with the weight for the CGHD. |

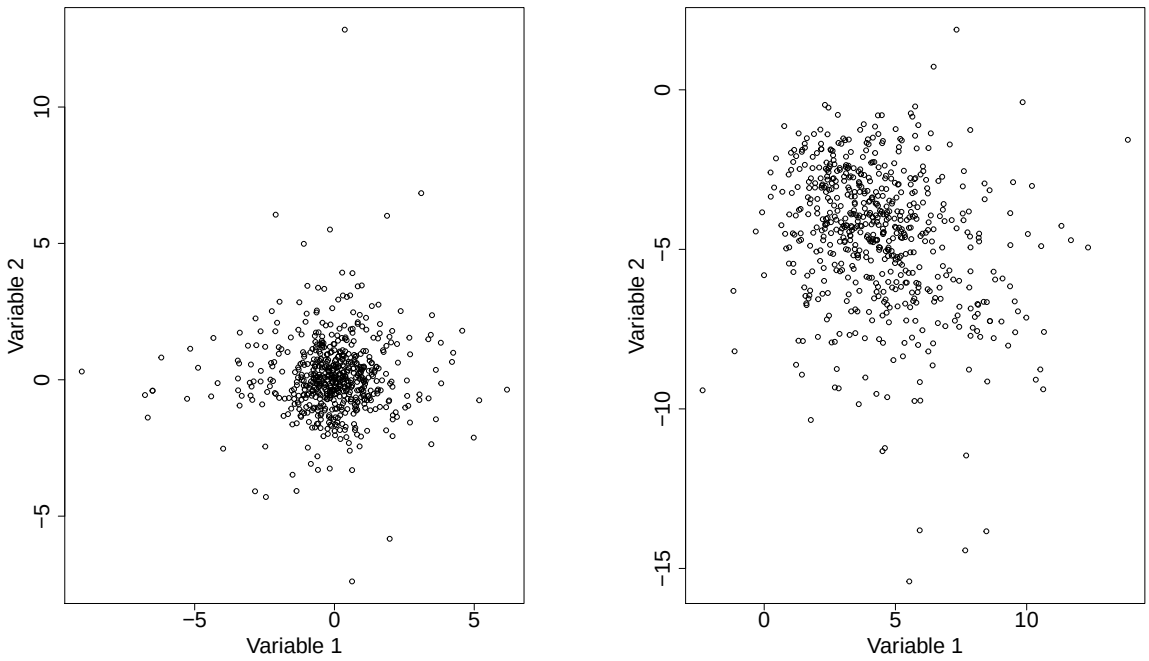
Table 2: Arguments for the `MGHD`, `MGHFA`, `MSGHD`, `cMSGHD`, and `MCGHD` functions.

| Arguments | Description |
|---|---|
| `train` | An $n_1 \times p$ matrix or data frame such that rows correspond to observations and columns correspond to variables of the training set. |
| `trainL` | An $n_1$ dimensional vector of membership for the units of the training set. If `trainL[i] = k`, then the observation $i$ belongs to group $k$. |
| `test` | An $n_2 \times p$ matrix or data frame such that rows correspond to observations and columns correspond to variables of the test set. |
| `testL` | An $n_2$ dimensional vector of membership for the units of the test set. If `testL[i] = k`, then the observation i belongs to group $k$. |
| `method` | An optional string indicating the method to be used for discriminant analysis; if not specified, `"GHD"` is used. Alternative methods are the `"MGHFA"`, `"MSGHD"`, `"cMSGHD"`, and `"MCGHD"`. |

Table 3: Additional arguments for the `DA` function.

cylinder and 97 obtained by bouncing signals off rocks. Each pattern is a set of 60 numbers (variables) taking values between 0 and 1.

```
R> data("sonar", package = "MixGHD")
R> lab <- as.numeric(factor(sonar[, 61]))
R> test <- sonar[c(1:29, 175:33), 1:60]
R> testL <- lab[c(1:29, 175:33)]
```




(a) Dataset 1. (b) Dataset 2.

Figure 4: Scatter plot of data generated using CGHDs.

```
R> train <- sonar[c(30:174), 1:60]
R> trainL <- lab[c(30:174)]
```

The command `lab <- as.numeric(factor(sonar[, 61]))` transforms the labels into a numerical vector. The data are divided into training and test sets, with 30% of the data in each cluster belonging to the test set.

```
R> set.seed(7)
R> modelDA <- DA(train, trainL, test, testL, max.iter = 200)

The best model (AIC) for the range of components used is G = 2.
The AIC for this model is -12460.25.

R> ls(modelDA)

[1] "ARItest"        "ARItrain"       "model"          "testMembership"

R> modelDA$ARItest

[1] 0.6605439

R> modelDA$ARItrain

[1] 1
```

As result of the DA routine, we obtain the ARI for the test and on the training sets, as well as the model and the membership for the test set. Model is an S4 object of class 'MixGHD'. For the test set of the sonar data, the ARI is 0.660 and, because no model was specified, the routine used the default model, i.e., MGHD.

### 7.4. Classification

The wine dataset, **pgmm** package (McNicholas *et al.* 2019), contains data on 27 chemical and physical properties of wine from the Piedmont region of Italy. There are three different types of wine: Barolo, Grignolino, and Barbera. To perform classification we assume that 25% of the memberships are unknown, the value 0 indicates unknown membership.

```
R> data("wine", package = "pgmm")
R> lab <- as.numeric(factor(wine[, 1]))
R> lab[seq(1, 178, 4)] <- 0
```

MGHD, MSGHD, cMSGHD, and MCGHD are used to classify the data, the parameter label contains the membership vector, the starting criterion used is $k$-medoids. To compute the ARI only the units with unknown membership are used.

```
R> resMGHD <- MGHD(wine[, 2:28], G = 3, label = lab, method = "kmedoids")

The best model (AIC) for the range of components used is G = 3.
The AIC for this model is -10121.01.

R> resMSGHD <- MSGHD(wine[, 2:28], G = 3, label = lab, method = "kmedoids")

The best model (AIC) for the range of components used is G = 3.
The AIC for this model is -11429.97.

R> rescMSGHD <- cMSGHD(wine[, 2:28], G = 3, label = lab, method = "kmedoids")

The best model (AIC) for the range of components used is G = 3.
The AIC for this model is -11439.12

R> resMCGHD <- MCGHD(wine[, 2:28], G = 3, label = lab, method = "kmedoids")

The best model (AIC) for the range of components used is G = 3.
The AIC for this model is -11350.15.

R> ARI(resMGHD@map[lab == 0], wine[lab == 0, 1])

[1] 1

R> ARI(resMSGHD@map[lab == 0], wine[lab == 0, 1])

[1] 0.9338421
```

```
R> ARI(rescMSGHD@map[lab == 0], wine[lab == 0, 1])

[1] 0.8627973

R> ARI(resMCGHD@map[lab == 0], wine[lab == 0, 1])

[1] 1
```

All the methods have good performances, however, MGHD and MCGHD outperform MSGHD and cMSGHD with an ARI equal to one.

### 7.5. Computational details

The package uses several R packages and functions. To implement the `Bessel` function the package **Bessel** (Maechler 2019) is used with exponentially scaled results to avoid underflow. The gradient is calculated using the function `grad`, from the package **numDeriv** (Gilbert and Varadhan 2019). To generate data the functions `rgig` and `rmvnorm` from the packages **ghyp** (Weibel, Luethi, and Breymann 2020) and **mvtnorm** (Genz, Bretz, Miwa, Mi, Leisch, Scheipl, and Hothorn 2020), respectively, are used. To reduce the computational time, the expectation step and the parameter updates of the functions `MGHD`, `MGFA`, `MSGHD`, `cMSGHD`, and `MCGHD`, are coded in C. The parameter initialization is done in R using the following functions: `kmeans` for $k$-means, `gpcm` for model-based (package **mixture**, Pocuca *et al.* 2021), `pam` for $k$-medoids (package **cluster**, Maechler *et al.* 2021), and `hclust` for hierarchical. The starting parameters are then passed to C where the appropriate algorithm for each function is used, see Section 6. The outputs from C are passed back to R where the indices discussed in Section 6.2 are computed. All the other functions are implemented entirely in R.

## 8. Conclusion

This paper illustrates the use of the **MixGHD** package for R. The package contains five main functions for model-based clustering, classification, and discriminant analysis based on the generalized hyperbolic distribution (GHD). The GHD is a very flexible distribution; other well-known distributions are special or limiting cases thereof. It can detect clusters characterized by a variety of shapes because it has skewness, concentration, and index parameters. The `MGHD` function performs clustering and classification using the GHD, the `MGHFA` function uses the mixture of generalized hyperbolic factor analyzers, useful for high-dimensional data. The other three functions: `MSGHD`, `cMSGHD`, and `MCGHD`, implement the three corresponding models that represent three recently proposed and more flexible variations of the MGHD. All of the models can be used with different starting techniques and several other options. The package also contains a `DA` routine for discriminant analysis, an `ARI` function that computes the adjusted Rand index, a `contourpl` function for contour plots and several functions for pseudo-random number generation and density estimation using the GHD, MSGHD, and MCGHD. The paper shows how to use the functions and to interpret the outputs on real datasets.

The current version of the package includes only one model for high-dimensional data, i.e., the MGHFA. Future research will focus on the extension of the MSGHD and MCGHD for high

dimensional data. Moreover, the GHD could also be used for model-based regression, in which the random response variables follow a generalized hyperbolic regression model given a set of explanatory variables.

## References

Aitken AC (1926). "On Bernoulli's Numerical Solution of Algebraic Equations." In *Proceedings of the Royal Society of Edimburgh*, volume 46, pp. 289–305. doi:10.1017/s0370164600022070.

Akaike H (1974). "A New Look at the Statistical Model Identification." *IEEE Transactions on Automatic Control*, **19**(6), 716–723. doi:10.1109/tac.1974.1100705.

Alman E (1968). "Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy." *The Journal of Finance*, **23**(4), 589–609. doi:10.2307/2978933.

Andrews JL, McNicholas PD (2011a). "Extending Mixtures of Multivariate $t$-Factor Analyzers." *Statistics and Computing*, **21**(3), 361–373. doi:10.1007/s11222-010-9175-2.

Andrews JL, McNicholas PD (2011b). "Mixtures of Modified $t$-Factor Analyzers for Model-Based Clustering, Classification, and Discriminant Analysis." *Journal of Statistical Planning and Inference*, **141**(4), 1479–1486. doi:10.1016/j.jspi.2010.10.014.

Arabie P, Hubert L (1994). "Cluster Analysis in Marketing Research." In RP Bagozzi (ed.), *Advanced Methods in Marketing Research*, pp. 160–189. Blackwell, Oxford.

Azzalini A, Torelli N (2007). "Clustering via Nonparametric Density Estimation." *Statistics and Computing*, **17**(1), 71–80. doi:10.1007/s11222-006-9010-y.

Baek J, McLachlan GJ, Flack L (2010). "Mixtures of Factor Analyzers with Common Factor Loadings: Applications to the Clustering and Visualization of High-Dimensional Data." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(7), 1298–1309. doi:10.1109/tpami.2009.149.

Ben-Israel A, Iyigun C (2008). "Probabilistic $D$-Clustering." *Journal of Classification*, **25**(1), 5–26. doi:10.1007/s00357-008-9002-z.

Benaglia T, Chauveau D, Hunter DR, Young D (2009). "**mixtools**: An R Package for Analyzing Finite Mixture Models." *Journal of Statistical Software*, **32**(6), 1–29. doi:10.18637/jss.v032.i06.

Bergé L, Bouveyron C, Girard S (2012). "**HDclassif**: An R Package for Model-Based Clustering and Discriminant Analysis of High-Dimensional Data." *Journal of Statistical Software*, **46**(6), 1–29. doi:10.18637/jss.v046.i06.

Biernacki C, Celeux G, Govaert G (2000). "Assessing a Mixture Model for Clustering with the Integrated Completed Likelihood." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(7), 719–725. doi:10.1109/34.865189.

Biernacki C, Celeux G, Govaert G, Langrognet F (2006). "Model-Based Cluster and Discriminant Analysis with the **MIXMOD** Software." *Computational Statistics & Data Analysis*, **51**(2), 587–600. doi:10.1016/j.csda.2005.12.015.

Bock HH (1987). "On the Interface between Cluster Analysis, Principal Component Analysis, and Multidimensional Scaling." In H Bozdogan, AK Gupta (eds.), *Multivariate Statistical Modeling and Data Analysis*, volume 8, pp. 17–34. Springer-Verlag, Dordrecht. doi:10.1007/978-94-009-3977-6_2.

Bouveyron C, Brunet C (2012). "Simultaneous Model-Based Clustering and Visualization in the Fisher Discriminative Subspace." *Statistics and Computing*, **22**(1), 301–324. doi:10.1007/s11222-011-9249-9.

Bouveyron C, Brunet C (2020). ***FisherEM**: The FisherEM Algorithm to Simultaneously Cluster and Visualize High-Dimensional Data.* R package version 1.6, URL https://CRAN.R-project.org/package=FisherEM.

Bouveyron C, Girard S, Schmid C (2007). "High-Dimensional Data Clustering." *Computational Statistics & Data Analysis*, **52**(1), 502–519. doi:10.1016/j.csda.2007.02.009.

Bozdogan H (1993). "Choosing the Number of Component Clusters in the Mixture-Model Using a New Informational Complexity Criterion of the Inverse-Fisher Information Matrix." In O Opitz, B Lausen, R Klar (eds.), *Information and Classification: Concepts, Methods and Applications*, pp. 40–54. Springer-Verlag, Berlin. doi:10.1007/978-3-642-50974-2_5.

Browne RP, McNicholas PD (2015). "A Mixture of Generalized Hyperbolic Distributions." *Canadian Journal of Statistics*, **43**(2), 176–198. doi:10.1002/cjs.11246.

Celeux G, Govaert G (1995). "Gaussian Parsimonious Clustering Models." *Pattern Recognition*, **28**(5), 781–793. doi:10.1016/0031-3203(94)00125-6.

Chen WC, Ostrouchov G (2021). ***pmclust**: Parallel Model-Based Clustering.* R package version 0.2-1, URL https://CRAN.R-project.org/package=pmclust.

De Sarbo WS, Manrai AK (1992). "A New Multidimensional Scaling Methodology for the Analysis of Asymmetric Proximity Data in Marketing Research." *Marketing Science*, **11**(1), 1–20. doi:10.1287/mksc.11.1.1.

De Soete G, Carroll JD (1994). "$k$-Means Clustering in a Low-Dimensional Euclidean Space." In E Diday, Y Lechevallier, M Schader, *et al.* (eds.), *New Approaches in Classification and Data Analysis*, pp. 212–219. Springer-Verlag, Berlin. doi:10.1007/978-3-642-51175-2_24.

Dempster AP, Laird NM, Rubin DB (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm." *Journal of the Royal Statistical Society B*, **39**(1), 1–38.

Forbes F, Wraith D (2014). "A New Family of Multivariate Heavy-Tailed Distributions with Variable Marginal Amounts of Tailweights: Application to Robust Clustering." *Statistics and Computing*, **24**(6), 971–984. doi:10.1007/s11222-013-9414-4.

Genz A, Bretz F, Miwa T, Mi X, Leisch F, Scheipl F, Hothorn T (2020). ***mvtnorm**: Multivariate Normal and t Distributions*. R package version 1.1-1, URL https://CRAN.R-project.org/package=mvtnorm.

Ghahramani Z, Hinton GE (1997). "The EM Algorithm for Mixtures of Factor Analyzers." *Technical report CRG-TR-96-1*, University of Toronto.

Gilbert P, Varadhan R (2019). ***numDeriv**: Accurate Numerical Derivatives*. R package version 2016.8-1.1, URL https://CRAN.R-project.org/package=numDeriv.

Grün B, Leisch F (2008). "**FlexMix** Version 2: Finite Mixtures with Concomitant Variables and Varying and Constant Parameters." *Journal of Statistical Software*, **28**(4), 1–35. doi:10.18637/jss.v028.i04.

Hennig C (2020). ***fpc**: Flexible Procedures for Clustering*. R package version 2.2-9, URL https://CRAN.R-project.org/package=fpc.

Hornik K (2005). "A CLUE for CLUster Ensembles." *Journal of Statistical Software*, **14**(12), 1–25. doi:10.18637/jss.v014.i12.

Hornik K, Grün B (2014). "**movMF**: An R Package for Fitting Mixtures of von Mises-Fisher Distributions." *Journal of Statistical Software*, **58**(10), 1–31. doi:10.18637/jss.v058.i10.

Hubert L, Arabie P (1985). "Comparing Partitions." *Journal of Classification*, **2**(1), 193–218. doi:10.1007/bf01908075.

Kim N, Browne RP (2019). "Subspace Clustering for the Finite Mixture of Generalized Hyperbolic Distributions." *Advances in Data Analysis and Classification*, **13**, 641–661. doi:10.1007/s11634-018-0333-2.

Lebret R, Iovleff S, Langrognet F, Biernacki C, Celeux G, Govaert G (2015). "**Rmixmod**: The R Package of the Model-Based Unsupervised, Supervised, and Semi-Supervised Classification Mixmod Library." *Journal of Statistical Software*, **67**(6), 1–29. doi:10.18637/jss.v067.i06.

Lee SX, McLachlan GJ (2014a). "Finite Mixtures of Multivariate Skew $t$-Distributions: Some Recent and New Results." *Statistics and Computing*, **24**(2), 181–202. doi:10.1007/s11222-012-9362-4.

Lee SX, McLachlan GJ (2014b). ***EMMIXuskew**: Fitting Unrestricted Multivariate Skew t Mixture Models*. R package version 0.11-6, URL https://CRAN.R-project.org/src/contrib/Archive/EMMIXuskew/.

Leisch F (2004). "**FlexMix**: A General Framework for Finite Mixture Models and Latent Class Regression in R." *Journal of Statistical Software*, **11**(8), 1–18. doi:10.18637/jss.v011.i08.

Leisch F, Grün B (2021). *CRAN Task View: Cluster Analysis & Finite Mixture Models*. Version 2021-05-11, URL https://CRAN.R-project.org/view=Cluster.

Lin TI, McLachlan GJ, Lee SX (2016). "Extending Mixtures of Factor Models Using the Restricted Multivariate Skew-Normal Distribution." *Journal of Multivariate Analysis*, **143**, 398–413. doi:10.1016/j.jmva.2015.09.025.

Lin TI, McNicholas PD, Hsiu JH (2014). "Capturing Patterns via Parsimonious $t$ Mixture Models." *Statistics and Probability Letters*, **88**, 80–87. doi:10.1016/j.spl.2014.01.015.

MacQueen J (1967). "Some Methods for Classification and Analysis of Multivariate Observations." In *Proceedings of the Fifth Berkeley Symposium*, volume 1, pp. 281–297.

Maechler M (2019). ***Bessel**: Computations and Approximations for Bessel Functions*. R package version 0.6-0, URL https://CRAN.R-project.org/package=Bessel.

Maechler M, Rousseeuw P, Struyf A, Hubert M, Hornik K (2021). ***cluster**: Cluster Analysis Basics and Extensions*. R package version 2.1.1, URL https://CRAN.R-project.org/package=cluster.

Markos A, Iodice D'Enza A, Van de Velden M (2019). "Beyond Tandem Analysis: Joint Dimension Reduction and Clustering in R." *Journal of Statistical Software*, **91**(10), 1–24. doi:10.18637/jss.v091.i10.

McLachlan GJ, Peel D (2000). "Mixtures of Factor Analyzers." In *Proceedings of the Seventh International Conference on Machine Learning*, pp. 599–606. Morgan Kaufmann, San Francisco.

McLachlan GJ, Peel D, Bean RW (2003). "Modelling High-Dimensional Data by Mixtures of Factor Analyzers." *Computational Statistics & Data Analysis*, **41**(3), 379–388. doi:10.1016/s0167-9473(02)00183-4.

McNicholas PD (2010). "Model-Based Classification Using Latent Gaussian Mixture Models." *Journal of Statistical Planning and Inference*, **140**(5), 1175–1181. doi:10.1016/j.jspi.2009.11.006.

McNicholas PD (2016). *Mixture Model-Based Classification*. Chapman & Hall/CRC, Boca Raton. doi:10.1201/9781315373577.

McNicholas PD, ElSherbiny A, McDaid AF, Murphy TB (2019). ***pgmm**: Parsimonious Gaussian Mixture Models*. R package version 1.2.4, URL https://CRAN.R-project.org/package=pgmm.

McNicholas PD, Murphy T (2008). "Parsimonious Gaussian Mixture Models." *Statistics and Computing*, **18**(3), 285–296. doi:10.1007/s11222-008-9056-0.

McNicholas PD, Murphy T (2010). "Model-Based Clustering of Microarray Expression Data via Latent Gaussian Mixture Models." *Bioinformatics*, **26**(21), 2705–2712. doi:10.1093/bioinformatics/btq498.

McNicholas PD, Murphy TB, McDaid AF, Frost D (2010). "Serial and Parallel Implementations of Model-Based Clustering via Parsimonious Gaussian Mixture Models." *Computational Statistics & Data Analysis*, **54**(2), 711–723. doi:10.1016/j.csda.2009.02.011.

McNicholas SM, McNicholas PD, Browne RP (2017). "A Mixture of Variance-Gamma Factor Analyzers." In SE Ahmed (ed.), *Big and Complex Data Analysis: Methodologies and Applications*, pp. 369–385. Springer-Verlag, Cham. doi:10.1007/978-3-319-41573-4_18.

Menardi G, Azzalini A (2014). "Clustering via Nonparametric Density Estimation: The R Package **pdfCluster**." *Journal of Statistical Software*, **11**, 1–26. doi:10.18637/jss.v057.i11.

Meng XL, Van Dyk D (1997). "The EM Algorithm – An Old Folk Song Sung to a Fast New Tune." *Journal of the Royal Statistical Society B*, **59**, 511–567. doi:10.1111/1467-9868.00082.

Montanari A, Viroli C (2011). "Maximum Likelihood Estimation of Mixtures of Factor Analyzers." *Computational Statistics & Data Analysis*, **55**(9), 2712–2723. doi:10.1016/j.csda.2011.04.001.

Murray PM, Browne RB, McNicholas PD (2014a). "Mixtures of Skew-*t* Factor Analyzers." *Computational Statistics & Data Analysis*, **77**, 326–335. doi:10.1016/j.csda.2014.03.012.

Murray PM, Browne RP, McNicholas PD (2016). ***uskewFactors**: Model-Based Clustering via Mixtures of Unrestricted Skew-t Factor Analyzer Models*. R package version 2.0, URL https://CRAN.R-project.org/package=uskewFactors.

Murray PM, Browne RP, McNicholas PD (2020). "Mixtures of Hidden Truncation Hyperbolic Factor Analyzers." *Journal of Classification*, **37**(2), 366–379. doi:10.1007/s00357-019-9309-y.

Murray PM, McNicholas PD, Browne RB (2014b). "A Mixture of Common Skew-*t* Factor Analyzers." *Stat*, **3**(1), 68–82. doi:10.1002/sta4.43.

Pocuca N, Browne RP, McNicholas PD (2021). ***mixture**: Mixture Models for Clustering and Classification*. R package version 2.0.3, URL https://CRAN.R-project.org/package=mixture.

Punzo A, Blostein M, McNicholas PD (2020). "High-Dimensional Unsupervised Classification via Parsimonious Contaminated Mixtures." *Pattern Recognition*, **98**, 107031. doi:10.1016/j.patcog.2019.107031.

Rainey C, Tortora C, Palumbo F (2019). "A Parametric Version of Probabilistic Distance Clustering." In *Studies in Classification, Data Analysis, and Knowledge Organization*, pp. 33–43. Springer-Verlag. doi:10.1007/978-3-030-21140-0_4.

Rand WM (1971). "Objective Criteria for the Evaluation of Clustering Methods." *Journal of the American Statistical Association*, **66**, 846–850. doi:10.1080/01621459.1971.10482356.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Schwarz G (1978). "Estimating the Dimension of a Model." *The Annals of Statistics*, **6**(2), 461–464.

Scrucca L, Fop M, Murphy TB, Raftery AE (2016). "**mclust** 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models." *The R Journal*, **8**(1), 289–317. doi:10.32614/RJ-2016-021.

Steane MA, McNicholas PD, Yada R (2012). "Model-Based Classification via Mixtures of Multivariate $t$-Factor Analyzers." *Communications in Statistics – Simulation and Computation*, **41**(4), 510–523. doi:10.1080/03610918.2011.595984.

Steinley D (2004). "Properties of the Hubert-Arable Adjusted Rand Index." *Psychological Methods*, **9**(3), 386. doi:10.1037/1082-989x.9.3.386.

Stute W, Zhu LX (1995). "Asymptotics of $k$-Means Clustering Based on Projection Pursuit." *Sankhyā: The Indian Journal of Statistics A*, **57**(3), 462–471.

Tang Y, Browne RP, McNicholas PD (2018). "Flexible Clustering of High-Dimensional Data via Mixtures of Joint Generalized Hyperbolic Distributions." *Stat*, **7**(1), e177. doi:10.1002/sta4.177.

Tortora C, El-Sherbiny A, Browne RP, Franczak BC, McNicholas PD (2021). ***MixGHD**: Model Based Clustering and Classification Using the Mixture of Generalized Hyperbolic Distributions*. R package version 2.3.5, URL https://CRAN.R-project.org/package=MixGHD.

Tortora C, Franczak BC, Browne RP, McNicholas PD (2019). "A Mixture of Coalesced Generalized Hyperbolic Distributions." *Journal of Classification*, **36**(1), 26–57. doi:10.1007/s00357-019-09319-3.

Tortora C, Gettler Summa M, Marino M, Palumbo F (2016a). "Factor Probabilistic Distance Clustering (FPDC): A New Clustering Method." *Advances in Data Analysis and Classification*, **10**, 441–464. doi:10.1007/s11634-015-0219-5.

Tortora C, McNicholas PD, Browne RP (2016b). "A Mixture of Generalized Hyperbolic Factor Analyzers." *Advances in Data Analysis and Classification*, **10**, 423–440. doi:10.1007/s11634-015-0204-z.

Tortora C, McNicholas PD, Palumbo F (2020a). "A Probabilistic Distance Clustering Algorithm Using Gaussian and Student-t Multivariate Density Distributions." *SN Computer Science*, **1**(2), 1–22. doi:10.1007/s42979-020-0067-z.

Tortora C, Vidales N, Palumbo F, McNicholas PD (2020b). ***FPDclustering**: PD-Clustering and Factor PD-Clustering*. R package version 1.4.1. URL https://CRAN.R-project.org/package=FPDclustering.

Vichi M, Kiers HAL (2001). "Factorial $k$-Means Analysis for Two-Way Data." *Computational Statistics & Data Analysis*, **37**(1), 49–64. doi:10.1016/s0167-9473(00)00064-5.

Vichi M, Saporta G (2009). "Clustering and Disjoint Principal Component Analysis." *Computational Statistics & Data Analysis*, **53**(8), 3194–3208. doi:10.1016/j.csda.2008.05.028.

Wang K, Ng A, McLachlan GJ (2018). ***EMMIXskew**: The EM Algorithm and Skew Mixture Distribution*. R package version 1.0.3, URL https://CRAN.R-project.org/src/contrib/Archive/EMMIXskew/.

Weibel M, Luethi D, Breymann W (2020). ***ghyp**: Generalized Hyperbolic Distribution and Its Special Cases*. R package version 1.6.1, URL https://CRAN.R-project.org/package=ghyp.

Yamamoto M, Hwang H (2014). "A General Formulation of Cluster Analysis with Dimension Reduction and Subspace Separation." *Behaviormetrika*, **41**, 115–129. doi:10.2333/bhmk.41.115.

**Affiliation:**

Cristina Tortora
Department of Mathematics and Statistics
San José State University
San José, California, United States of America, 95192
E-mail: cristina.tortora@sjsu.edu

*Journal of Statistical Software* http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
May 2021, Volume 98, Issue 3 *Submitted:* 2017-01-11
doi:10.18637/jss.v098.i03 *Accepted:* 2019-12-31