Journal of Statistical Software
August 2013, Volume 54, Issue 12. *http://www.jstatsoft.org/*

mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions

Marcos Oliveira Prates
Universidade Federal de Minas Gerais

Celso Rômulo Barbosa Cabral
Universidade Federal do Amazonas

Víctor Hugo Lachos
Universidade Estadual de Campinas

Abstract

We present the R package **mixsmsn**, which implements routines for maximum likelihood estimation (via an expectation maximization EM-type algorithm) in finite mixture models with components belonging to the class of scale mixtures of the skew-normal distribution, which we call the FMSMSN models. Both univariate and multivariate responses are considered. It is possible to fix the number of components of the mixture to be fitted, but there exists an option that transfers this responsibility to an automated procedure, through the analysis of several models choice criteria. Plotting routines to generate histograms, plug-in densities and contour plots using the fitted models output are also available. The precision of the EM estimates can be evaluated through their estimated standard deviations, which can be obtained by the provision of an approximation of the associated information matrix for each particular model in the FMSMSN family. A function to generate artificial samples from several elements of the family is also supplied. Finally, two real data sets are analyzed in order to show the usefulness of the package.

Keywords: skew-normal distribution, finite mixtures, EM algorithm, scale mixtures, culstering.

1. Introduction

In this paper we present the R (R Core Team 2013) package **mixsmsn**, a powerful tool to fit finite mixtures of distributions, which are densities of the form

$$g(x|\mathbf{\Theta}) = \sum_{i=1}^{g} p_i f(x|\boldsymbol{\theta}_i), \tag{1}$$

where $p_i \geq 0$, $i = 1, \ldots, g$, with $\sum_{i=1}^{g} p_i = 1$, are called *mixing weights*, the density $f(\cdot|\boldsymbol{\theta}_i)$ is the i-th *component of the mixture*, which is indexed by the (possibly multivariate) parameter $\boldsymbol{\theta}_i$, $i = 1, \ldots, n$ and $\boldsymbol{\Theta} = ((p_1, \ldots, p_g)^\top, \boldsymbol{\theta}_1^\top, \ldots, \boldsymbol{\theta}_g^\top)^\top$.

Mixture models have been widely applied in several scientific areas as a tool for modeling population heterogeneity, allowing posterior unsupervised classification of the observations, for example. Also, because of its extreme flexibility, this class of models is an excellent alternative to approximate complicated probability densities, presenting multimodality, skewness and heavy tails. The theme has received considerable attention in the statistical and applied literature, with highly recommended texts available. To cite only a few, we have the books of McLachlan and Peel (2000), Frühwirth-Schnatter (2006), Schlattmann (2010) and Mengersen, Robert, and Titterington (2011), and the special editions of the journal *Computational Statistics & Data Analysis* (Böhning and Seidel 2003; Böhning, Seidel, Alfó, Garel, Patilea, and Walther 2007). Besides this, at this moment, another special issue is being prepared in this journal, edited by B. Böhning, C. Hennig, G. McLachlan and P. McNicholas.

A considerable portion of the literature in mixture models refers to symmetric components, like normal or Student-t. Inference in these cases has been extensively studied, as we can see in the references cited above. After Azzalini and his colleagues (Azzalini 1985; Azzalini and Valle 1996; Azzalini and Capitanio 1999) work, there was a rapid dissemination of the theory and applications of asymmetric distributions. Other proposals extending the normal or the Student$-t$ distributions are very popular also, see Sahu, Dey, and Branco (2003), for example. In the last 5 years, there are some effort to generalize mixture models results, in order to obtain a greater degree of flexibility, introducing components that accommodate asymmetry and/or heavy tails. Notice that if the true distribution of at least one component has one of these characteristics, then the model fit using a symmetric distribution can result in undesirable results, like overestimation of the number of components of the mixture, for example. Some works that replace the normal assumption in mixture models by more flexible distributions are Lin, Lee, and Hsieh (2007a), Lin, Lee, and Yen (2007c), Cabral, Bolfarine, and Pereira (2008), Lin (2009), Castillo and Daoudi (2009), Karlis and Santourian (2009), Lin (2010), Basso, Lachos, Cabral, and Ghosh (2010), Frühwirth-Schnatter and Pyne (2010), Vrbik and McNicholas (2012), Ho, Pyne, and Lin (2012), Cabral, Lachos, and Prates (2012) and Lee and McLachlan (2013).

We can find several R packages for finite mixture models like, for instance, **flexmix** (Leisch 2004; Grün and Leisch 2008), **mixAK** (Komárek 2009), **mixreg** (Turner 2011), **bayesmix** (Grün 2011), **mclust** (Fraley, Raftery, Murphy, and Scrucca 2012), **mixtools** (Benaglia, Chauveau, Hunter, and Young 2009) and **EMCluster** (Chen, Maitra, and Melnykov 2013). But none of them deals with the issue of skewed or heavy-tailed components. Regarding R packages for modeling data presenting skewness and/or outliers (but not unobserved heterogeneity), we can cite the package **sn** (Azzalini 2011), which provides functions related to the skew-normal (SN) and the skew-Student-t distributions and the package **nlsm** (Garay, Prates, and Lachos 2012), which deals with estimation in univariate non-linear regression models with observational errors belonging to the class of scale mixtures of the skew-normal distribution (SMSN, Lachos, Ghosh, and Arellano-Valle 2010).

In this work, we assume that the components of the mixture belong to the class of SMSN distributions. It is a rich class of flexible distributions, including versions of classical symmetric distributions (like normal, Student-t, etc.), accommodating simultaneously skewness and robustness to discrepant observations. Both the univariate (Basso *et al.* 2010) and the multivariate (Cabral *et al.* 2012) cases are considered. The estimation procedure is maximum likelihood via an EM-type algorithm. The available component distributions are: normal, Student-t, skew-normal, skew-contaminated normal, skew-slash and skew-Student-t.

In our proposal, the user can pass values to the arguments of the functions with great flexibility. Specifically, the user can specify its own set of starting values for initialization of the algorithm and also specify the number of components to be fitted, although there are automated options to do theses tasks. In particular, the choice of the number of components is made through the analysis of several classical models selection criteria (like AIC and BIC) and the starting values are defined through a combination of the k-means method and the method of moments. In addition, there are functions to generate histograms and contour plots of the data, to generate artificial observations from finite mixtures of SMSN distributions and to obtain an approximated information matrix for each subfamily considered. Also, the unsupervised clustering of the observations, which is an important issue related to modeling using finite mixtures, is provided.

The library **mixsmsn** has been used recently with great success in several applications. See, for example, Basso *et al.* (2010) and Cabral *et al.* (2012). But we believe that exists a vast collection of possible applications, which includes, for instance, image processing, signal processing and analysis of microarray data.

The remainder of the paper is organized as follows. In Section 2 we give a short introduction to the theory of the finite mixtures of SMSN distributions and estimation via EM-type algorithm; in Section 3 we present the two real data sets that will be used to illustrate the usefulness of the package; in Section 4 we introduce the functions useful to fit mixtures and to generate random samples from the available mixture distributions; in Section 5 we proceed full analyses of real data sets.

2. Finite mixtures of scale mixtures of skew-normals

2.1. The skew-normal distribution

A skew-normal distribution is a distribution that extends the normal one by the introduction of an additional parameter (or maybe more than one) regulating skewness. Some versions, extensions and unification of the skew-normal distribution are carefully surveyed in works like Azzalini (2005) and Arellano-Valle and Azzalini (2006).

For our purposes, we say that a $p \times 1$ random vector $\mathbf{Y}$ follows a skew-normal distribution with $p \times 1$ location vector $\boldsymbol{\mu}$, $p \times p$ positive definite dispersion matrix $\boldsymbol{\Sigma}$ and $p \times 1$ skewness parameter vector $\boldsymbol{\lambda}$, and we write $\mathbf{Y} \sim \mathrm{SN}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\lambda})$, if its density is given by

$$\mathrm{SN}_p(\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\lambda}) = 2\phi_p(\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\Phi(\boldsymbol{\lambda}^\top \boldsymbol{\Sigma}^{-1/2}(\mathbf{y} - \boldsymbol{\mu})),$$

where $\boldsymbol{\lambda}^\top$ denotes the transpose of $\boldsymbol{\lambda}$, $\boldsymbol{\Sigma}^{-1/2}$ is the square root of $\boldsymbol{\Sigma}^{-1}$, that is, $\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Sigma}^{-1/2} = \boldsymbol{\Sigma}^{-1}$ (this square root is unique see, for example, Theorem 3.5 in Zhang 2011), $\phi_p(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ stands for the density of the p–variate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, $\mathrm{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ say, and $\Phi(\cdot)$ represents the distribution function of the standard univariate normal distribution. We drop some indices when there is no possibility of confusion: $\mathrm{N}(0, 1)$ and $\phi(\cdot)$ will denote the univariate standard normal distribution and its respective density, for instance. It is important to note that the case $\boldsymbol{\lambda} = \mathbf{0}_p$ corresponds to the usual $p-$variate normal distribution, where $\mathbf{0}_p$ is the null vector of dimension $p \times 1$.

2.2. Univariate finite mixtures of scale mixtures of skew-normals

First we will present the definition of the family of scale mixtures of the skew-normal distribution (SMSN), given by Branco and Dey (2001). Here, we consider the case $p = 1$ and drop some indices, writing $Y \sim \mathrm{SN}(\mu, \sigma^2, \lambda)$, for example.

Definition 1 *The distribution of the random variable Y belongs to the univariate SMSN family when $Y = \mu + U^{-1/2}Z$, where $\mu \in \mathbb{R}$ is a location parameter, $Z \sim \mathrm{SN}(0, \sigma^2, \lambda)$ and U is a positive random variable, independent of Z, with distribution function $H(\cdot|\boldsymbol{\nu})$.*

In the definition above $\sigma^2 > 0$ and $\lambda \in \mathbb{R}$ are scale and shape parameters, respectively, and $H(\cdot|\boldsymbol{\nu})$ is known as *the mixing scale distribution*, indexed by the (possibly multivariate) parameter $\boldsymbol{\nu}$. The marginal density of Y is

$$\mathrm{SMSN}(y|\mu, \sigma^2, \lambda, \boldsymbol{\nu}) = 2\int_0^\infty \phi(y|\mu, u^{-1}\sigma^2)\Phi(u^{\frac{1}{2}}\lambda\sigma^{-1}(y-\mu))dH(u|\boldsymbol{v}).$$

See Basso *et al.* (2010) for details like moments and a fundamental stochastic representation for the SMSN family.

For each choice of $H(\cdot|\boldsymbol{\nu})$ in Definition 1 we obtain a different member of the family. These are some examples:

- The univariate normal distribution: this is the case when $U = 1$ and $\lambda = 0$;
- The univariate skew-normal distribution: this is the case when $U = 1$;
- The univariate skew-Student-t distribution: this is the case when $U \sim \mathrm{Gamma}(\nu/2, \nu/2)$, with $\nu > 0$ and $\mathrm{Gamma}(a, b)$ denoting the gamma distribution with mean a/b;
- The univariate skew-slash distribution: this is the case when $U \sim \mathrm{Beta}(\nu, 1)$, $\nu > 0$;
- The univariate skew-contaminated normal distribution: this is the case when U is a discrete random variable taking state ν_2 with probability ν_1 and state 1 with probability $1 - \nu_1$, where ν_1 and ν_2 are in the interval $(0, 1)$.

A finite mixture of SMSN distributions model (FMSMSN) is a density defined as in (1), where the i-th component of the mixture is a SMSN distribution with parameters μ_i, σ_i^2, λ_i and $\boldsymbol{\nu}_i$. Concerning the parameters indexing the mixing distributions, we assume that $\boldsymbol{\nu}_1 = \ldots = \boldsymbol{\nu}_g = \boldsymbol{\nu}$. For the particular cases presented above we will use the notations FMNOR, FMSN, FMST, FMSSL and FMSCN, respectively.

2.3. Multivariate finite mixtures of scale mixtures of skew-normals

It is straightforward to extend the definition of scale mixtures of the SN distribution to the multivariate case.

Definition 2 *The $p-$dimensional random vector $\boldsymbol{Y}$ belongs to the SMSN family when $\boldsymbol{Y} = \boldsymbol{\mu} + U^{-1/2}\boldsymbol{Z}$, where $\boldsymbol{\mu} : p \times 1$ is a location vector parameter, $\boldsymbol{Z} \sim \mathrm{SN}_p(\mathbf{0}, \boldsymbol{\Sigma}, \boldsymbol{\lambda})$ and U is a positive random variable, independent of $\boldsymbol{Z}$, with distribution function $H(\cdot|\boldsymbol{\nu})$.*

In the definition above $\boldsymbol{\Sigma}$ is a $p \times p$ positive definite scale matrix, $\boldsymbol{\lambda}$ is the $p \times 1$ shape vector and $H(\cdot|\boldsymbol{\nu})$ is the mixing distribution, exactly as before. Thus, the marginal density of $\boldsymbol{Y}$ is

$$\mathrm{SMSN}_p(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = 2\int_0^\infty \phi_p(\boldsymbol{y}|\boldsymbol{\mu}, u^{-1}\boldsymbol{\Sigma})\Phi(u^{\frac{1}{2}}\boldsymbol{\lambda}^\top\boldsymbol{\Sigma}^{-\frac{1}{2}}(\boldsymbol{y}-\boldsymbol{\mu}))dH(u|\boldsymbol{v}). \quad (2)$$

For more details see Cabral *et al.* (2012).

As in the univariate SMSN family, if the random variable U is chosen to follow one the distributions presented in Section 2.2 and Definition 2 is applied, we have versions of the normal, skew-normal, skew-Student-t, skew-slash and skew-contaminated normal as specific members of the multivariate SMSN family. Also, extending the ideas presented for the univariate FMSMSN case, we can define multivariate FMSMSN distributions by considering that the i-th component of the mixture is a SMSN density with parameters $\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$, $\boldsymbol{\lambda}_i$ and $\boldsymbol{\nu}$.

2.4. Maximum likelihood estimation via an EM-type algorithm

In Basso *et al.* (2010) and Cabral *et al.* (2012) the algorithms used to implement the estimation routine of the package **mixsmsn** are presented in details, the former paper deals with the univariate case while the latter deals with the multivariate case. It is worth to mention that the algorithm is very general, encompassing all members of the SMSN family. However, because of computational effort reasons, only the distributions presented in sections 2.2 and 2.3 are considered. Excepting for the skew-slash case, the updating expressions for the location, scale and skewness parameters are written in a closed form. This is an advantage over competitors, like the algorithm presented by Lin (2010), where in the E-step Monte Carlo integration is needed and the moments of the truncated multivariate normal distribution have to be computed (in fact, this author considered the skew-normal of Sahu *et al.* 2003).

Another interesting feature of the package is that, considering the parametrization

$$\boldsymbol{\Delta}_i = \boldsymbol{\Sigma}_i^{1/2}\boldsymbol{\delta}_i, \ \ \boldsymbol{\delta}_i = \frac{\boldsymbol{\lambda}_i}{\sqrt{1+\ \boldsymbol{\lambda}_i^\top\ \boldsymbol{\lambda}_i}}, \ \ \boldsymbol{\Gamma}_i = \boldsymbol{\Sigma}_i - \boldsymbol{\Delta}_i\boldsymbol{\Delta}_i^\top, \qquad i = 1, \ldots, g,$$

we have that a more parsimonious model is achieved by supposing $\boldsymbol{\Gamma}_1 = \ldots = \boldsymbol{\Gamma}_g = \boldsymbol{\Gamma}$, which can be seen as an extension of the normal mixture model with restricted variance-covariance components.

2.5. The observed information matrix

The package **mixsmsn** provides an approximation of the asymptotic covariance matrix of the vector of EM estimates, using a method suggested by Basford, Greenway, Mclachlan, and Peel (1997). In Basso *et al.* (2010) and Cabral *et al.* (2012) we can find expressions for all cases considered in Section 2.2 and Section 2.3 and an extensive simulation study evaluating the quality of estimates of the standard deviations obtained through this method.

3. Data Sets

In this section we present two data sets: the body mass index (BMI), which will be useful to illustrate the applicability of the package for the univariate case, and the Old Faithful geyser, for a multivariate illustration.

3.1. Body mass index

The body mass index data set was collected for men aged between 18 to 80 years old. It came from the National Health and Nutrition Examination Survey, made by the National Center for Health Statistics (NCHS) of the Center for Disease Control (CDC) in the USA. With the increase of chronic diseases around the USA, attention was attracted to the obesity problem in the past few years. It is known that people with obesity have higher chances of developing chronic diseases. To quantify overweight and obesity, the BMI, which is defined as the ratio of body weight in kilograms and body height in meters squared, was selected as the standard measure, where people with high BMI (> 25) are considered to have overweight and people with BMI > 30 are considered to be obese.

Lin, Lee, and Hsieh (2007b) considered the reports made in years 1999-2000 and years 2001-2002. In their paper they only considered participants who have weight within [39.50 kg, 70.00 kg] and [95.01 kg, 196.80 kg], allowing them to explore a mixture pattern. From the original 4579 participants, they used a total of 2123, where the first group had 1069 participants and the second 1054. In their analysis, Lin *et al.* (2007b) fitted the models FMNOR, FMT (that is, with Student t components), FMSN and FMST.

3.2. Old Faithful geyser

Park geologists have been collecting data of geyser eruptions over different USA parks. The Yellowstone National Park was created in 1872 and was the first America's national park. Inside the Yellowstone National Park are Old Faithful and a collection of the world's most extraordinary geysers and hot springs.

Azzalini and Bowman (1990) presented an analysis of data from the Old Faithful geyser. It consists of 272 pairs of measurements, referring to the time interval between the starts of successive eruptions and the duration of the subsequent eruption. Scientists have been analyzing the Old Faithful data since 1978 (e.g., Denby and Pregibon 1987; Silverman 1985). Background information on the Old Faithful geyser is provided by Rinehart (1969).

4. mixsmsn

In this section we will show how to fit and analyze data using univariate and multivariate FMSMSN distributions through the package **mixsmsn**.

4.1. Univariate finite mixtures of scale mixtures of skew-normals

The function `smsn.mix()` is responsible for the implementation of the inferential procedures to fit the univariate FMSMSN distributions presented in Section 2.2. It has the form

```
smsn.mix(y, nu, mu = NULL, sigma2 = NULL, shape = NULL, pii = NULL,
  g = NULL, get.init = TRUE, criteria = TRUE, group = FALSE,
  family = "Skew.normal", error = 0.00001, iter.max = 100,
  calc.im = TRUE, obs.prob = FALSE, kmeans.param = NULL)
```

where `y` is the vector of responses and `nu` is the initial value for the mixing distribution parameter (it must be bidimensional for the FMSCN case, with coordinates constrained to the interval (0,1)). For the FMNOR and FMSN models, any value can be passed to this argument, since it will be ignored. The argument `g` is the number of mixture components to be fitted; `get.init` is a `TRUE`/`FALSE` variable to choose if the initial values should be created, if its value is `FALSE` it is necessary to specify: `pii` is the g-dimensional vector of initial values for the weights (`pii` is constrained to sum 1), `mu` must be a g-dimensional vector with the the i-th coordinate being the starting value for the location parameter of the i-th component of the mixture, `sigma2` and `shape` are also g-dimensional vectors, following the same pattern, with starting points for scale and shape parameters, respectively, $i = 1, \ldots, g$; `criteria` is a `TRUE`/`FALSE` variable to choose if the criteria (AIC, DIC, ECD and ICL) should be calculated or not; `group` is a `TRUE`/`FALSE` variable to choose if an unsupervised clustering of the observations should be performed, if its value is `TRUE` then each subject in the sample is allocated to one and only one of g groups (cluster, class). We allocate the subject i to group j^*, where $j^* = \arg\max\{\hat{z}_{ij},\ \ j = 1, \ldots, g\}$ and $\hat{z}_{ij}$ is the estimated posterior probability

$$\hat{z}_{ij} = \frac{\hat{p}_j \mathrm{SMSN}(y_i|\hat{\mu}_j, \hat{\sigma}_j^2, \hat{\lambda}_j, \hat{\nu})}{\sum_{k=1}^{g} \hat{p}_k \mathrm{SMSN}(y_i|\hat{\mu}_k, \hat{\sigma}_k^2, \hat{\lambda}_k, \hat{\nu})},$$

where the notation $\hat{\mu}_j$ indicates the estimate of μ_j and so on. If `obs.prob = TRUE`, then a matrix with these probabilities is provided; `family` sets the component distribution family of the mixture to be fitted (`Normal`, `t`, `Skew.normal`, `Skew.t`, `Skew.slash` and `Skew.cn`); `error` is the stopping criterion for the EM algorithm; `iter.max` is the maximum number of iterations for the EM algorithm when it does not achieve convergence; `calc.im` is a `TRUE`/`FALSE` variable to choose if the information matrix must be provided and the standard errors reported. If `get.init = TRUE`, then the initial values for the EM algorithm are obtained using a combination of the R function `kmeans` and the method of moments. Sees details in Basso *et al.* (2010). If `kmeans.param = NULL`, the the default values of the function `kmeans` are used, otherwise the user must pass a list with alternative parameters values, for example `kmeans.param = list(iter.max = 20, n.start = 2, algorithm = "Forgy")`.

Another important function in the **mixsmsn** package is `rmix()`. With `rmix()` it is possible to generate data from any of the SMSN distributions presented in Section 2.2. The function `rmix()` has the form

```
rmix(n, pii, family, arg)
```

where `n` is the number of observations to be generated, `pii` and `family` are as above and `arg` is a list with each entry containing a vector with the necessary parameters of the distribution specified in `family`.

For the family of univariate FMSMSN distributions we also have the following functions: `mix.hist()`, `mix.dens()`, `mix.lines()`, `mix.print()`, `smsn.search()` and `im.smsn()`. The function `mix.hist()` is equivalent to the R function `hist()` and plots a histogram of the data with the plug-in (fitted) density superimposed. The function `mix.dens()` plot the estimated density (or log-density) of the fitted model. The function `mix.lines()` is similar to the R function `lines()` and allow to add estimated densities curves in the plot generated by the `mix.dens()` function. The `mix.print()` function is equivalent to the R function `print()` and prints some basic information of the output. The `im.smsn()` function provides the approximated information matrix of the FMSMSN parameters.

The `smsn.search()` is responsible to search for the best model under a pre-specified criterion (AIC, BIC, EDC or ICL) and from a specified range of number of mixture components (g) to be considered. Denoting the vector with all parameters to be estimated by $\mathbf{\Theta}$ and its EM estimator by $\hat{\mathbf{\Theta}}$, we have that the AIC, BIC and EDC are of the form $-2\ell(\hat{\mathbf{\Theta}}) + \gamma c_n$ where $\ell(\hat{\mathbf{\Theta}})$ is the actual log-likelihood, γ is the number of free parameters that have to be estimated under the model and the penalty term c_n is a convenient sequence of positive numbers. The ICL is defined as $-2\ell^{\star}(\hat{\mathbf{\Theta}}) + \gamma \log(n)$ where $\ell^{\star}(\hat{\mathbf{\Theta}})$ is the integrated log-likelihood. For further details see Basso *et al.* (2010) and Cabral *et al.* (2012).

We will show the usage of the functions presented above in Section 5.

4.2. Multivariate finite mixtures of scale mixtures of skew-normals

The function `smsn.mmix()` is responsible for the implementation of the EM-type algorithm for the multivariate FMSMSN models presented in Section 2.3. It has the form

```
smsn.mmix(y, nu=1, mu = NULL, Sigma = NULL, shape = NULL, pii = NULL,
  g = NULL, get.init = TRUE, criteria = TRUE, group = FALSE,
  family = "Skew.normal", error = 0.0001, iter.max = 100, uni.Gama = FALSE,
  calc.im=FALSE, obs.prob = FALSE, kmeans.param = NULL)
```

where the parameters are multivariate versions of the ones presented for the function `smsn.mix()`. The parameter `uni.Gama` is introduced here and is a `TRUE`/`FALSE` variable to choose the option $\mathbf{\Gamma}_1 = \ldots = \mathbf{\Gamma}_g$, see Section 2.4.

As presented in the univariate case, the `rmmix()` is the equivalent data generator for the multivariate FMSMSN distributions. However, in this case, the parameter `arg` is a list of g lists with each one containing the necessary parameters of the selected `family`.

In addition, we have the following functions: `rmmix()`, `mix.contour()`, `smsn.search()` and `imm.smsn()`. The `mix.contour()` function plots the contour of the fitted output when the dimension of the analysis is 2. The `imm.smsn()` function provides an approximated information matrix of the FMSMSN parameters. Finally, the `smsn.search()` function searches for the best model under one of the possible criterion presented in Section 2.2, for a pre-specified range of values of g. Section 5 has an illustration of the usage of the functions above.

5. Examples continued

In this section we revisit the examples presented in Section 3 in order to illustrate the usefulness of the **mixsmsn** package, both in the univariate and in the multivariate cases.

5.1. Body mass index data

The BMI data presented in Section 3.1 was incorporated in the package **mixsmsn** and will be used to illustrate the inferential methods using the univariate FMSMSN models available in the package.

The initial step is to load the BMI data

```
R> library("mixsmsn")
R> data("bmi")
```

Once the data is loaded we are able to obtain an overview of the response variable, by constructing a histogram

```
R> hist(bmi$bmi, breaks = 40, main = "Histogram of BMI", xlab = "bmi")
```

(see Figure 1) where we can visualize that the data has a bimodality with some right skewness for each mode, and it seems to be reasonable to fit to this data FMSMSN models with two components. In order to do so, we will rely on the `smsn.mix()` function.

```
R> par(mfrow = c(2, 2))
R> Snorm.analysis <- smsn.mix(bmi$bmi, nu = 3, g = 2, get.init = TRUE,
+    criteria = TRUE, group = TRUE, family = "Skew.normal", calc.im = FALSE)
R> mix.hist(bmi$bmi, Snorm.analysis)
R> St.analysis <- smsn.mix(bmi$bmi, nu = 3, g = 2, get.init = TRUE,
+    criteria = TRUE, group = TRUE, family = "Skew.t", calc.im = FALSE)
```

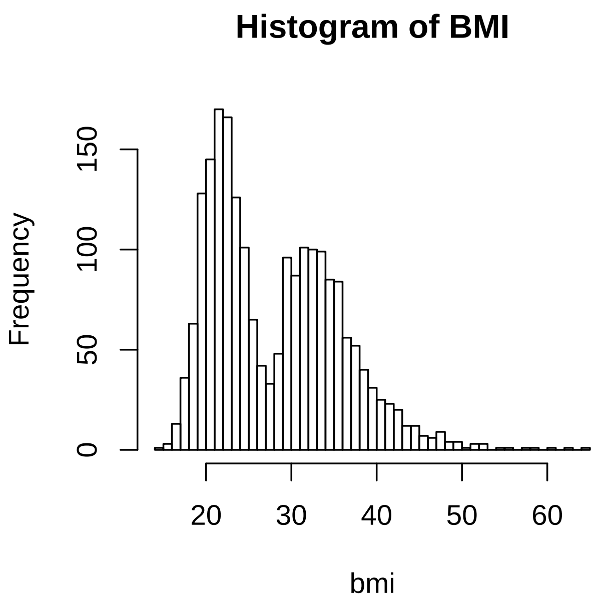

Figure 1: BMI histogram.

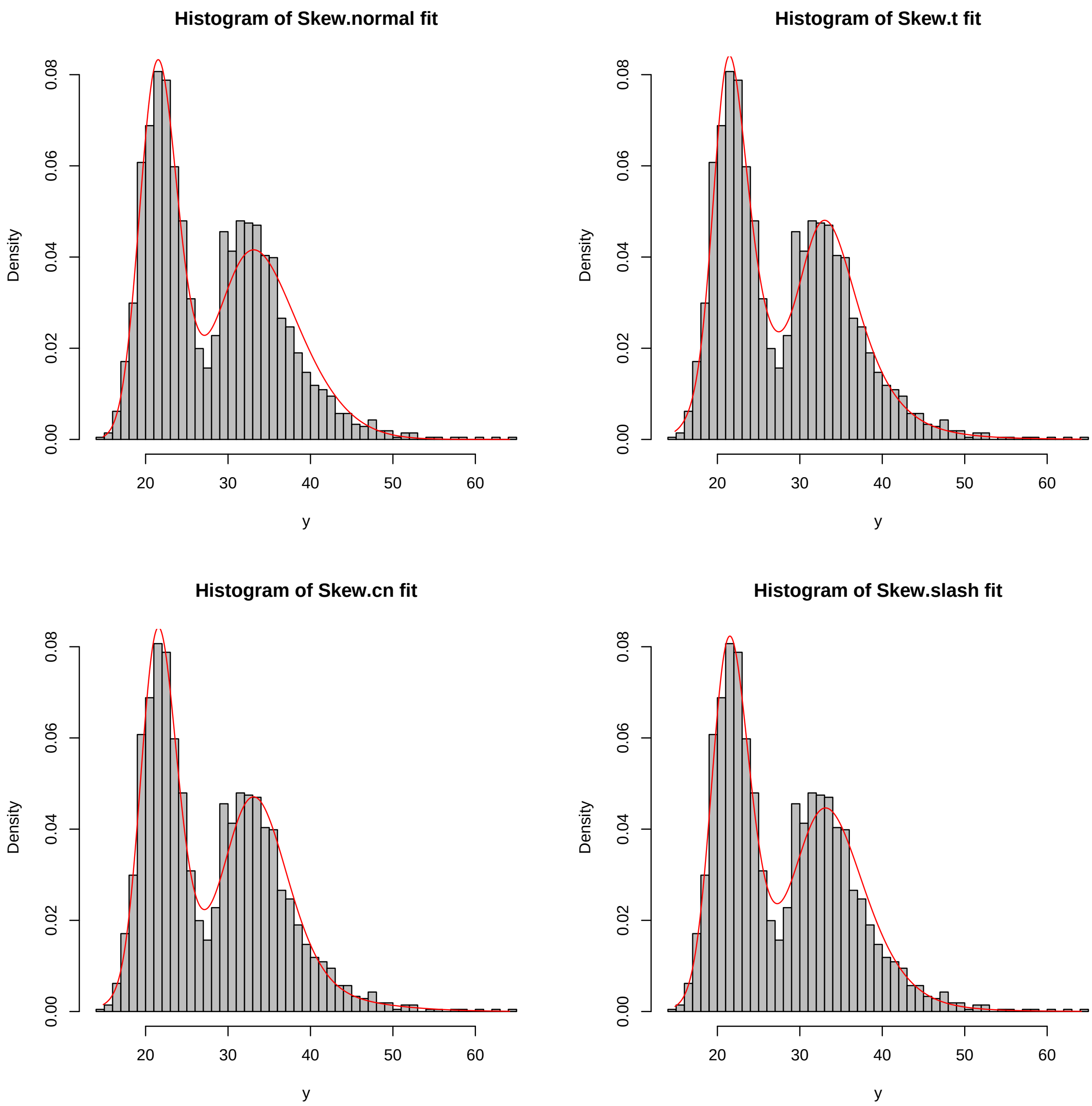

Figure 2: Fitted FMSMSN distributions with the BMI response, from left to right: FMSN, FMST, FMSCN and FMSSL.

```
R> mix.hist(bmi$bmi, St.analysis)
R> Scn.analysis <- smsn.mix(bmi$bmi, nu = c(0.3, 0.3), g = 2, get.init = TRUE,
+    criteria = TRUE, group = TRUE, family = "Skew.cn", calc.im = FALSE)
R> mix.hist(bmi$bmi, Scn.analysis)
R> Sslash.analysis <- smsn.mix(bmi$bmi, nu = 3, g = 2, get.init = TRUE,
+    criteria = TRUE, group = TRUE, family = "Skew.slash", calc.im = FALSE)
R> mix.hist(bmi$bmi, Sslash.analysis)
```

Once the models are fitted, Figure 2 shows the plots generated by the function `mix.hist()` for the mixtures of skew-normal, skew-Student t, skew-contaminated normal and skew-slash

Model	AIC	BIC	EDC	ICL
FMSN	13972.95	13821.22	13828.86	13992.45
FMST	13754.06	13782.33	13789.76	14016.90
FMSCN	*13748.51*	*13776.77*	*13784.41*	14007.47
FMSSL	13751.98	13780.24	13787.88	*13987.77*

Table 1: Models selection criteria for the BMI data set (all models with two components).

distributions, respectively, with the respective plug-in densities superimposed. This figure shows some evidence that the FMSN model presents the worst performance. With the help of the `mix.print()` function we can obtain the estimates of the parameters and the values of the models selection criteria.

```
R> mix.print(Snorm.analysis)

 Number of observations:  2107

       group 1 group 2
mu      29.207  19.907
sigma2  53.588   8.931
shape    1.920   1.215

 AIC:  13792.95

 BIC:  13821.22

 EDC:  13828.86

 ICL:  13992.45

 EM iterations:  11
```

Table 1 presents the values of the models selection criteria for each mixture model. From it we can see that the FMSCN model presents the best fit according to AIC, BIC and EDC criterions, while FMSSL performs better under the ICL criteria. After this we can use the `im.smsn()` function to obtain the approximated information matrix of the parameters and further the respective estimated standard deviations.

```
R> bmi.im <- im.smsn(bmi$bmi, Sslash.analysis)
R> sdev <- sqrt(diag(solve(bmi.im$IM)))
R> sdev

       mu1     sigma1     shape1         p1        mu2     sigma2     shape2
0.60863704 5.84197907 0.71105460 0.02879698 0.43553030 2.43468500 0.52871330
        nu
0.42393606
```

Now we introduce the function `smsn.search()`, fitting normal mixtures with number of components varying from 1 to 5. By default, the value passed to the argument `criteria` is `"BIC"` and the alternatives are `"AIC"`, `"EDC"` or `"ICL"`. In this case, for illustration purposes, we will use the `"AIC"` criterion. As before, in order to obtain starting points for the estimation algorithm, we can pass alternative values for the argument `kmeans.param`, altering the default values of the R function `kmeans`. See Section 4.1 for details.

```
R> bmi.analysis <- smsn.search(bmi$bmi, nu = 3, g.min = 1, g.max = 5,
+    family = "Normal", criteria = "aic")
R> bmi.analysis$criteria

     g=1      g=2      g=3      g=4      g=5
14472.38 13835.42 13745.19 13746.49 13749.53

R> mix.print(bmi.analysis$best.model)

 Number of observations:  2107

       group 1 group 2 group 3
mu      21.743  32.504  39.081
sigma2   5.227  11.232  46.242

 AIC:  13745.19

 BIC:  13790.41

 EDC:  13802.63

 ICL:  14283.64

 EM iterations:  8
```

Then, between the normal mixture models considered, the best fit occurs when we have three components – although, as commented before, the data has a clear bimodal nature. That is, we need more normal than skew-slash components to accommodate the asymmetric and/or heavy tailed behaviour of this data, showing the flexibility of the latter model.

5.2. Old Faithful geyser data

To illustrate the applicability of the package in the multivariate case, we consider the Old Faithful data mentioned before. Now we will impose starting values for the parameters of the SMSN distributions under consideration, instead of generate them, as we proceeded in the previous example. This is done by fixing `FALSE` for the argument `get.init`. First, we load the data.

```
R> data("faithful")
```

After this, we pass the initial values to **mixsmsn** using the following codes

```
R> mu1 <- c(5, 77)
R> Sigma1 <- matrix(c(0.18, 0.60, 0.60, 41), 2, 2)
R> shape1 <- c(0.69, 0.64)
R> mu2 <- c(2, 52)
R> Sigma2 <- matrix(c(0.15, 1.15, 1.15, 40), 2, 2)
R> shape2 <- c(4.3, 2.7)
R> pii<-c(0.65, 0.35)
R> mu <- list(mu1, mu2)
R> Sigma <- list(Sigma1, Sigma2)
R> shape <- list(shape1, shape2)
```

Once the initial values are fixed, we run the package in order to fit the multivariate SMSN models cited in Section 2.3

```
R> par(mfrow = c(2, 2))
R> Norm.analysis <- smsn.mmix(faithful, nu = 3, mu = mu, Sigma = Sigma,
+    shape = shape, pii = pii, g = 2, get.init = FALSE, group = TRUE,
+    family = "Normal", calc.im = FALSE)
R> mix.contour(faithful, Norm.analysis, x.min = 1, x.max = 1, y.min = 15,
+    y.max = 10, levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))
R> Snorm.analysis <- smsn.mmix(faithful, nu = 3, mu = mu, Sigma = Sigma,
+    shape = shape, pii = pii, g = 2, get.init = FALSE, group = TRUE,
+    family = "Skew.normal", calc.im = FALSE)
R> mix.contour(faithful, Snorm.analysis, x.min = 1, x.max = 1, y.min = 15,
+    y.max = 10, levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))
R> St.analysis <- smsn.mmix(faithful, nu = 3, mu = mu, Sigma = Sigma,
+    shape = shape, pii = pii, g = 2, get.init = FALSE, group = TRUE,
+    family = "Skew.t", calc.im = FALSE)
R> mix.contour(faithful, St.analysis, x.min = 1, x.max = 1, y.min = 15,
+    y.max = 10, levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))
R> Scn.analysis <- smsn.mmix(faithful, nu = c(0.3, 0.3), mu = mu,
+    Sigma = Sigma, shape = shape, pii = pii, g = 2, get.init = FALSE,
+    group = TRUE, family = "Skew.cn", calc.im = FALSE)
R> mix.contour(faithful, Scn.analysis, x.min = 1, x.max = 1, y.min = 15,
+    y.max = 10, levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))
```

Table 2 presents the models choice criteria. From it we can see that the FMSCN model has the best performance. Also, from this table we can see that the FMST and the FMSCN models have very close results, providing a better fit than the FMSN or FMNOR ones. Figure 3 presents the contours of the fitted models. In this figure, the points are distinguished by different colors through information provided by the commands `Norm.analysis$group`, `Snorm.analysis$group` and so on. They yield a vector of the same length of the data vector, and its i-th coordinate (with value 1 or 2, in this case) represents the group where the i-th subject will be allocated. Observe that, although the function `mix.contour()` is applicable only for bivariate data, the multivariate functions `smsn.mmix()` and `imm.smsn()` work for any $p \geq 2$.

Having chosen the FMSCN model, we will show how to use the function `imm.smsn()` to obtain

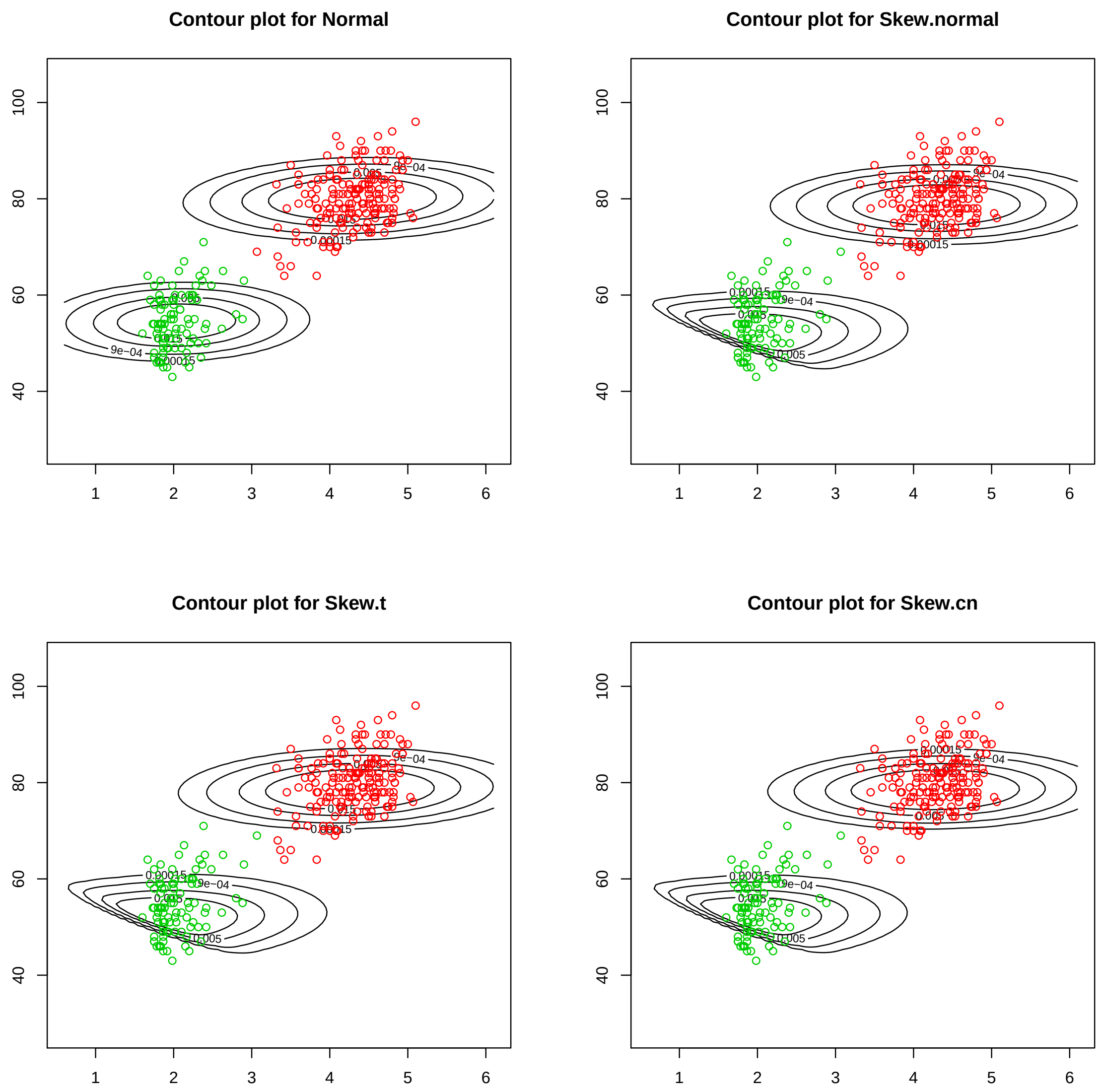

Figure 3: Contours of the fitted FMSMSN distributions for the Old Faithful data, from left to right: FMNOR, FMSN, FMST and FMSCN.

Model	AIC	BIC	EDC	ICL
FMNOR	2292.53	2350.22	2313.31	2350.72
FMSN	2265.55	2323.24	2286.32	2325.18
FMST	2265.09	2322.78	2285.86	2324.87
FMSCN	*2264.98*	*2322.68*	*2285.76*	*2324.56*

Table 2: Models selection criteria for the Old Faithful data (all models with two components).

the approximated information matrix and the respective standard deviation estimates of the EM estimates.

```
R> faithful.im <- imm.smsn(faithful, Scn.analysis)
R> sdev <- sqrt(diag(solve(faithful.im$IM)))
R> sdev

       mu1_1         mu1_2      shape1_1      shape1_2     Sigma1_11     Sigma1_12
5.765085e-01 6.159679e+00 2.166146e+00 1.460854e+00 2.602896e+00 1.177035e+00
   Sigma1_22         mu2_1         mu2_2      shape2_1      shape2_2     Sigma2_11
4.411384e+01 3.230842e-02 1.240486e+00 2.324035e+00 2.695586e+00 2.304114e+00
   Sigma2_12     Sigma2_22          pii1           nu1           nu2
1.558676e+00 4.047053e+01 2.991640e-02 1.161834e+03 1.299269e+03
```

The function `smsn.search()` presented in Section 5.1 can also be used for multivariate data. We present an example of its usage with the normal distribution.

```
R> faithful.analysis <- smsn.search(faithful, nu = 3, g.min = 1, g.max = 5,
+    family = "Normal")
R> faithful.analysis$criteria

     g=1      g=2      g=3      g=4      g=5
2624.440 2350.222 2374.135 2412.447 2444.987
```

We can see that, for the normal distribution, the best fit is achieved using two components.

5.3. Simulating data with mixsmsn

The **mixsmsn** package provides the functions `rmix()` and the `rmmix()` to generate data sets for the univariate and multivariate FMSMSN distributions, respectively. These tools allow researchers to create simulated data sets with several different parameters setups, improving the undertanding of the phenomena under study. We start presenting how to generate data for univariate FMSMSN models using the function `rmix()`. To use the random data generator the user must specify the parameters values.

```
R> mu1 <- 5
R> mu2 <- 20
R> mu3 <- 35
R> sigma2.1 <- 9
R> sigma2.2 <- 16
R> sigma2.3 <- 9
R> lambda1 <- 5
R> lambda2 <- -3
R> lambda3 <- -6
R> nu <- 5
R> pii <- c(0.5, 0.2, 0.3)
```

Having done this, the user must organize these values in order to call properly the function `rmix()`. Here, we will generate a sample of size $n = 5000$ from a mixture of skew-Student t distributions with 3 components.

```
R> arg1 <- c(mu1, sigma2.1, lambda1, nu)
R> arg2 <- c(mu2, sigma2.2, lambda2, nu)
R> arg3 <- c(mu3, sigma2.3, lambda3, nu)
R> y <- rmix(n = 5000, p = pii, family = "Skew.t",
+    arg = list(arg1, arg2, arg3))
```

In the multivariate case, the procedure is similar. First, we define the values of the parameters of interest.

```
R> mu1 <- c(0, 0)
R> Sigma1 <- matrix(c(3, 1, 1, 3), 2, 2)
R> shape1 <-c(4, 4)
R> nu1 <- 4
R> mu2 <- c(5, 5)
R> Sigma2 <- matrix(c(2, 1, 1, 2), 2, 2)
R> shape2 <-c(2, 2)
R> nu2 <- 4
R> pii <- c(0.6, 0.4)
```

Then, we create a list of arguments that will be passed to the function `rmmix()` to generate a sample of size $n = 1000$ from a mixture of bivariate skew-Student t distributions with 2 components.

```
R> arg1 <- list(mu = mu1, Sigma = Sigma1, shape = shape1, nu = nu1)
R> arg2 <- list(mu = mu2, Sigma = Sigma2, shape = shape2, nu = nu2)
R> y <- rmmix(n = 1000, p = pii, family = "Skew.t",
+    arg = list(arg1, arg2))
R> setwd(wd)
```

6. Discussion

In this paper we presented the R package **mixsmsn**, an ensemble of routines useful to analyze data presenting a strong non-normal pattern, including skewness, multimodality and heavy tails, modeling using distributions that are members of an extreme flexible class, which is composed by finite mixtures of distributions that are scale mixtures of the skew-normal distribution. The package allows the user to proceed a full analysis, including point estimation via an EM-type algorithm, estimates of their standard deviations, a proper visualization of the data with immersed estimated densities (in the univariate case) or contours of the estimated densities (in the bivariate case), and generation of artificial data from distributions in the family. The posterior unsupervised classification of the observations is also possible, the output incorporates the grouping criterion proposed by Basso *et al.* (2010) and Cabral *et al.* (2012) as a tool for clustering. We analyzed two real data sets, in order to show the efficacy of the package. We hope that this package can be useful for practitioners in several areas where modeling data with mixtures is applicable, like medicine, image processing, signal processing, genetics, economics, to cite only a few. As in these areas the usage of normal or Student t components is still very popular, although in some cases the nature of the data clearly do not support this, our belief is that the analysis could be substantially improved by modeling using the skew heavy-tailed models provided by the package.

References

Arellano-Valle RB, Azzalini A (2006). "On the Unification of Families of Skew-Normal Distributions." *Scandinavian Journal of Statistics*, **33**, 561–574.

Azzalini A (1985). "A Class of Distributions Which Includes the Normal Ones." *Scandinavian Journal of Statistics*, **12**, 171–178.

Azzalini A (2005). "The Skew-normal Distribution and Related Multivariate Families." *Scandinavian Journal of Statistics*, **32**, 159–188.

Azzalini A (2011). ***sn**: The Skew-Normal and Skew-t Distributions.* R package version 0.4-17, URL http://CRAN.R-project.org/package=sn.

Azzalini A, Bowman AW (1990). "A Look at Some Data on the Old Faithful Geyser." *Journal of the Royal Statistical Society C*, **39**, 357–365.

Azzalini A, Capitanio A (1999). "Statistical Applications of the Multivariate Skew Normal Distribution." *Journal of the Royal Statistical Society B*, **61**, 579–602.

Azzalini A, Valle AD (1996). "The Multivariate Skew-Normal Distribution." *Biometrika*, **83**, 715–726.

Basford KE, Greenway DR, Mclachlan GJ, Peel D (1997). "Standard Errors of Fitted Component Means of Normal Mixtures." *Computational Statistics*, **12**, 1–17.

Basso RM, Lachos VH, Cabral CRB, Ghosh P (2010). "Robust Mixture Modeling Based on Scale Mixtures of Skew-normal Distributions." *Computational Statistics & Data Analysis*, **54**, 2926 – 2941.

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. URL http://www.jstatsoft.org/v32/i06/.

Böhning D, Seidel W (2003). "Editorial: Recent Developments in Mixture Models." *Computational Statistics & Data Analysis*, **41**, 349–357.

Böhning D, Seidel W, Alfó M, Garel B, Patilea V, Walther G (2007). "Editorial: Advances in Mixture Models." *Computational Statistics & Data Analysis*, **51**, 5205–5210.

Branco MD, Dey DK (2001). "A General Class of Multivariate Skew-Elliptical Distributions." *Journal of Multivariate Analysis*, **79**, 99–113.

Cabral CRB, Bolfarine H, Pereira JRG (2008). "Bayesian Density Estimation Using Skew Student-t-Normal Mixtures." *Computational Statistics & Data Analysis*, **52**, 5075–5090.

Cabral CRB, Lachos VH, Prates MO (2012). "Multivariate Mixture Modeling Using Skew-normal Independent Distributions." *Computational Statistics & Data Analysis*, **56**, 126 – 142.

Castillo J, Daoudi J (2009). "The Mixture of Left-Right Truncated Normal Distributions." *Journal of Statistical Planning and Inference*, **139**, 3543–3551.

Chen WC, Maitra R, Melnykov V (2013). ***EMCluster***: *EM Algorithm for Model-Based Clustering of Finite Mixture Gaussian Distribution*. R package version 0.2-3, URL http://CRAN.R-project.org/package=EMCluster.

Denby L, Pregibon D (1987). "An Example of the Use of Graphics in Regression." *The American Statistician*, **41**, 33–38.

Fraley C, Raftery A, Murphy TB, Scrucca L (2012). "**mclust** Version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation." *Technical Report 597*, Department of Statistics, University of Washington.

Frühwirth-Schnatter S (2006). *Finite Mixture and Markov Switching Models*. Springer-Verlag.

Frühwirth-Schnatter S, Pyne S (2010). "Bayesian Inference for Finite Mixtures of Univariate and Multivariate Skew-Normal and Skew-t Distributions." *Biostatistics*, **11**, 317–336.

Garay A, Prates M, Lachos VH (2012). ***nlsmsn***: *Fitting Nonlinear Models with Scale Mixture of Skew-Normal Distributions*. R package version 0.0-3, URL http://CRAN.R-project.org/package=nlsmsn.

Grün B (2011). ***bayesmix***: *Bayesian Mixture Models with* ***JAGS***. R package version 0.7-2, URL http://CRAN.R-project.org/package=bayesmix.

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. URL http://www.jstatsoft.org/v28/i04/.

Ho HJ, Pyne S, Lin TI (2012). "Maximum Likelihood Inference for Mixtures of Skew Student-t-Normal Distributions through Practical EM-Type Algorithms." *Statistics and Computing*, **22**(1), 287–299.

Karlis D, Santourian A (2009). "Model-Based Clustering with Non-Elliptically Contoured Distributions." *Statistics and Computing*, **19**, 73–83.

Komárek A (2009). "A New R Package for Bayesian Estimation of Multivariate Normal Mixtures Allowing for Selection of the Number of Components and Interval-Censored Data." *Computational Statistics & Data Analysis*, **53**, 3932–3947.

Lachos VH, Ghosh P, Arellano-Valle RB (2010). "Likelihood Based Inference for Skew-Normal Independent Linear Mixed Models." *Statistica Sinica*, **20**, 303–322.

Lee S, McLachlan GJ (2013). "Finite Mixtures of Multivariate Skew t-distributions: Some Recent and New Results." *Statistics and Computing*. doi:10.1007/s11222-012-9362-4.

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. URL http://www.jstatsoft.org/v11/i08/.

Lin TI (2009). "Maximum Likelihood Estimation for Multivariate Skew Normal Mixture Models." *Journal of Multivariate Analysis*, **100**, 257–265.

Lin TI (2010). "Robust Mixture Modeling Using Multivariate Skew t Distributions." *Statistics and Computing*, **20**, 343–356.

Lin TI, Lee JC, Hsieh WJ (2007a). "Robust Mixture Modelling Using the Skew t Distribution." *Statistics and Computing*, **17**, 81–92.

Lin TI, Lee JC, Hsieh WJ (2007b). "Robust Mixture Modelling Using the Skew t Distribution." *Statistics and Computing*, **17**, 81–92.

Lin TI, Lee JC, Yen SY (2007c). "Finite Mixture Modelling Using the Skew Normal Distribution." *Statistica Sinica*, **17**, 909–927.

McLachlan G, Peel D (2000). *Finite Mixture Models*. John Wiley & Sons.

Mengersen K, Robert CP, Titterington DM (2011). *Mixtures: Estimation and Applications*. John Wiley & Sons.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Rinehart JS (1969). "Thermal and Seismic Indications of Old Faithful Geyser's Inner Working." *Journal of Geophysical Research*, **74**, 566–573.

Sahu SK, Dey DK, Branco MD (2003). "A New Class of Multivariate Skew Distributions with Applications to Bayesian Regression Models." *The Canadian Journal of Statistics*, **31**, 129–150.

Schlattmann P (2010). *Medical Applications of Finite Mixture Models*. Springer-Verlag.

Silverman BW (1985). "Some Aspects of the Spline Smoothing Approach to Non-Parametric Regression Curve Fitting." *Journal of the Royal Statistical Society B*, **47**, 1–52.

Turner R (2011). ***mixreg**: Functions to Fit Mixtures of Regressions*. R package version 0.0-4, URL http://CRAN.R-project.org/package=mixreg.

Vrbik I, McNicholas PD (2012). "Analytic Calculations for the EM Algorithm for Multivariate Skew-t Mixture Models." *Statistics and Probability Letters*, **82**, 1169–1174.

Zhang F (2011). *Matrix Theory: Basic Results and Techniques*. 2nd edition. Springer-Verlag.

Affiliation:

Marcos Oliveira Prates
Departamento de Estatística
Universidade Federal de Minas Gerais
CEP 31270-901, Belo Horizonte, Minas Gerais, Brazil
E-mail: marcosop@est.ufmg.br

Victor Hugo Lachos
Departamento de Estatística
IMECC, Universidade Estadual de Campinas
Caixa Postal 6065, CEP 13083-859, Campinas, São Paulo, Brazil
E-mail: hlachos@ime.unicamp.br

Celso Rômulo Barbosa Cabral
Departamento de Estatística
Universidade Federal do Amazonas
CEP 69077-000 , Manaus, Amazonas, Brazil
E-mail: celsoromulo@gmail.com

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/

Volume 54, Issue 12 *Submitted:* 2011-12-16
August 2013 *Accepted:* 2013-03-21