PUBLISHED VERSION

Sharon X. Lee, Geoffrey J. McLachlan
EMMIXuskew: an R package for fitting mixtures of multivariate skew t distributions via the EM algorithm
Journal of Statistical Software, 2013; 55(12):1-22

Originally published at: http://doi.org/10.18637/jss.v055.i12

Journal of Statistical Software

November 2013, Volume 55, Issue 12. *http://www.jstatsoft.org/*

EMMIXuskew: An R Package for Fitting Mixtures of Multivariate Skew t Distributions via the EM Algorithm

Sharon X. Lee
University of Queensland

Geoffrey J. McLachlan
University of Queensland

Abstract

This paper describes an algorithm for fitting finite mixtures of unrestricted Multivariate Skew t (FM-uMST) distributions. The package **EMMIXuskew** implements a closed-form expectation-maximization (EM) algorithm for computing the maximum likelihood (ML) estimates of the parameters for the (unrestricted) FM-MST model in R. **EMMIXuskew** also supports visualization of fitted contours in two and three dimensions, and random sample generation from a specified FM-uMST distribution.

Finite mixtures of skew t distributions have proven to be useful in modelling heterogeneous data with asymmetric and heavy tail behaviour, for example, datasets from flow cytometry. In recent years, various versions of mixtures with multivariate skew t (MST) distributions have been proposed. However, these models adopted some restricted characterizations of the component MST distributions so that the E-step of the EM algorithm can be evaluated in closed form. This paper focuses on mixtures with unrestricted MST components, and describes an iterative algorithm for the computation of the ML estimates of its model parameters. Its implementation in R is presented with the package **EMMIXuskew**.

The usefulness of the proposed algorithm is demonstrated in three applications to real datasets. The first example illustrates the use of the main function `fmmst` in the package by fitting a MST distribution to a bivariate unimodal flow cytometric sample. The second example fits a mixture of MST distributions to the Australian Institute of Sport (AIS) data, and demonstrates that **EMMIXuskew** can provide better clustering results than mixtures with restricted MST components. In the third example, **EMMIXuskew** is applied to classify cells in a trivariate flow cytometric dataset. Comparisons with some other available methods suggest that **EMMIXuskew** achieves a lower misclassification rate with respect to the labels given by benchmark gating analysis.

Keywords: mixture models, skew distributions, multivariate t distribution, EM algorithm, flow cytometry, R.

1. Introduction

In many practical problems, data are often skewed, heterogeneous, and/or contain outliers. Finite mixture of skewed distributions have become increasingly popular in modelling and analyzing such data. This use of finite mixture distributions to model heterogeneous data has undergone intensive development in the past decades, as witnessed by the numerous applications in various scientific fields such as bioinformatics, cluster analysis, genetics, information processing, medicine, and pattern recognition. For a comprehensive survey on mixture models and their applications see, for example, the monographs by Everitt and Hand (1981), Titterington, Smith, Markov, and E. (1985), McLachlan and Basford (1988), Lindsay (1995), Böhning (2000), McLachlan and Peel (2000), and Frühwirth-Schnatter (2006), the edited volume of Mengersen, Robert, Titterington, and M. (2011), and also the papers by Banfield and Raftery (1993) and Fraley and Raftery (1998).

In recent years, finite mixtures of skew t distributions have been exploited as an effective tool in modelling high-dimensional multimodal and asymmetric datasets; see, for example, Pyne *et al.* (2009a) and Frühwirth-Schnatter and Pyne (2010). Following the introduction of the skew normal (SN) distribution by Azzalini (1985), several authors have studied skewed extensions of the t distribution. Finite mixture models with multivariate skew t (MST) components was first proposed by Pyne *et al.* (2009a) in a study of an automated approach to the analysis of flow cytometry data. Wang, McLachlan, Ng, and Peel (2012) has given a package **EMMIX-skew** for the implementation in R (R Core Team 2013) of their algorithm. More recently, Basso, Lachos, Cabral, and Ghosh (2010) studied a class of mixture models where the components densities are scale mixtures of univariate skew normal distributions, known as the skew normal/independent (SNI) family of distributions, which include the (univariate) skew normal and skew t distributions as special cases. This work was later extended to the multivariate case in Cabral, Lachos, and Prates (2012), and was implemented in an R package **mixsmsn**. However, in these characterizations, restrictions were imposed on the component skew t distributions in order to obtain manageable analytical expressions for the conditional expectations involved in the E-step of the EM algorithm. These versions of the skew t distributions are known as the 'restricted' form of the MST distribution; see Lee and McLachlan (2013a) for further discussion on this.

In this paper, we present an algorithm for the fitting of the unrestricted skew t mixture model. We show that an EM algorithm can be implemented exactly without restricting the characterizations of the component MST distributions. Closed form expressions can be obtained for the E-step conditional expectations by recognizing that they can be formulated as moments of a multivariate non-central truncated t variate, which can be further expressed in terms of central t distributions. The algorithm is implemented in R in the package **EMMIXuskew**, available from the Comprehensive R Archive Network at `http://CRAN.R-project.org/package=EMMIXuskew`.

The package **EMMIXuskew** consists of three main functions: `fmmst`, `rfmmst`, and `contour.fmmst`. The main function `fmmst` fits a mixture of unrestricted MST (uMST) distributions using an EM algorithm described in Section 3. The function `rfmmst` generates random samples from mixtures of uMST distributions. For a user friendly visualisation of the fitted models, `fmmst.contour` provides 2D contour maps of the fitted bivariate densities and 3D displays with interactive viewpoint navigation facility for trivariate densities.

The remainder of this paper is organized as follows. Section 2 provides a brief description

of the uMST distribution and defines the FM-uMST model. Section 3 presents an EM algorithm for fitting the FM-uMST model. In the next section, an explanation of how to fit, visualize, and interpret the FM-uMST models using **EMMIXuskew** is presented. The usage of **EMMIXuskew** is illustrated with three applications and comparisons are made with some restricted FM-MST models and other clustering methods. Finally, we conclude with a brief summary of our results.

2. Finite mixtures of multivariate skew t distributions

We begin by defining the (unrestricted) multivariate skew t density. Let $\boldsymbol{Y}$ be a p-dimensional random vector. Then $\boldsymbol{Y}$ is said to follow a p-dimensional unrestricted skew t distribution (Sahu, Dey, and Branco 2003) with $p \times 1$ location vector $\boldsymbol{\mu}$, $p \times p$ scale matrix $\boldsymbol{\Sigma}$, $p \times 1$ skewness vector $\boldsymbol{\delta}$, and (scalar) degrees of freedom ν, if its probability density function (pdf) is given by

$$f_p(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\delta}, \nu) = 2^p t_{p,\nu}\left(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Omega}\right) T_{p,\nu+p}\left(\boldsymbol{y}^*; \boldsymbol{0}, \boldsymbol{\Lambda}\right), \tag{1}$$

where

$$\begin{aligned}
\boldsymbol{\Delta} &= \text{diag}(\boldsymbol{\delta}),\\
\boldsymbol{\Omega} &= \boldsymbol{\Sigma} + \boldsymbol{\Delta}^2,\\
\boldsymbol{y}^* &= \boldsymbol{q}\sqrt{\frac{\nu + p}{\nu + d\left(\boldsymbol{y}\right)}},\\
\boldsymbol{q} &= \boldsymbol{\Delta}\boldsymbol{\Omega}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}),\\
d\left(\boldsymbol{y}\right) &= (\boldsymbol{y} - \boldsymbol{\mu})^\top \boldsymbol{\Omega}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}),\\
\boldsymbol{\Lambda} &= \boldsymbol{I}_p - \boldsymbol{\Delta}\boldsymbol{\Omega}^{-1}\boldsymbol{\Delta}.
\end{aligned}$$

Here the operator diag($\boldsymbol{\delta}$) denotes a diagonal matrix with diagonal elements specified by the vector $\boldsymbol{\delta}$. Also, we let $t_{p,\nu}(.; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ be the p-dimensional t distribution with location vector $\boldsymbol{\mu}$, scale matrix $\boldsymbol{\Sigma}$, and degrees of freedom ν, and $T_{p,\nu}(.; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ the corresponding (cumulative) distribution function. The notation $\boldsymbol{Y} \sim \text{uMST}_{p,\nu}(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\delta})$ will be used. Note that when $\boldsymbol{\delta} = \boldsymbol{0}$, (1) reduces to the symmetric t density $t_{p,\nu}(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Also, when $\nu \to \infty$, we obtain the (unrestricted) skew normal distribution.

Various versions of the multivariate skew t density have been proposed in recent years. It is worth noting that the versions considered by Azzalini and Capitanio (2003), Gupta (2003), and Lachos, Ghosh, and Arellano-Valle (2010), among others, are different from (1). These versions are simpler in that the skew t density is defined in terms involving only the univariate t distribution function instead of the multivariate form of the latter as used in (1). These simplified characterizations have the advantage of having closed form expressions for the conditional expectations that have to be calculated on the E-step. The reader is referred to Lee and McLachlan (2013a,b) for a discussion on different forms of skew t distributions. We shall adopt the unrestricted form (1) of the MST distribution here as proposed by Sahu *et al.* (2003), and describe a computationally efficient EM algorithm for fitting this model.

A g-component finite mixture of uMST distributions has density given by

$$f\left(\boldsymbol{y}; \boldsymbol{\Psi}\right) = \sum_{h=1}^{g} \pi_h f_p\left(\boldsymbol{y}; \boldsymbol{\mu}_h, \boldsymbol{\Sigma}_h, \boldsymbol{\delta}_h, \nu_h\right), \tag{2}$$

where $f_p\left(\boldsymbol{y};\boldsymbol{\mu}_h,\boldsymbol{\Sigma}_h,\boldsymbol{\delta}_h,\nu_h\right)$ denotes the h-th uMST component of the mixture model as defined by (1), with location parameter $\boldsymbol{\mu}_h$, scale matrix $\boldsymbol{\Sigma}_h$, skew parameter $\boldsymbol{\delta}_h$, and degrees of freedom ν_h. The mixing proportions π_h satisfy $\pi_h \geq 0$ $(h = 1,\ldots,g)$ and $\sum_{h=1}^{g}\pi_h = 1$. We shall denote the model defined by (2) by the FM-uMST (finite mixture of uMST) distributions. Let $\boldsymbol{\Psi}$ contain all the unknown parameters of the FM-uMST model; that is, $\boldsymbol{\Psi} = \left(\pi_1,\ldots,\pi_{g-1},\boldsymbol{\theta}_1^\top,\ldots,\boldsymbol{\theta}_g^\top\right)^\top$ where now $\boldsymbol{\theta}_h$ consists of the unknown parameters of the h-th component density function. The density values for a uMST and FM-uMST distribution can be evaluated using the functions `dmst` and `dfmmst` in **EMMIXuskew**.

Random samples of uMST variates can be generated by adopting a stochastic representation of (1) (Lin 2010). If $\boldsymbol{Y} \sim uMST_{p,\nu}(\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\delta})$, then

$$\boldsymbol{Y} = \boldsymbol{\mu} + \frac{1}{\sqrt{w}}\boldsymbol{\Delta}\left|\boldsymbol{U}_1\right| + \frac{1}{\sqrt{w}}\boldsymbol{U}_0, \tag{3}$$

where the random variables

$$\boldsymbol{U}_0 \sim N_p(\boldsymbol{0},\boldsymbol{\Sigma}), \tag{4}$$
$$\boldsymbol{U}_1 \sim N_p(\boldsymbol{0},\boldsymbol{I}_p), \tag{5}$$
$$w \sim \text{gamma}\left(\frac{\nu}{2},\frac{\nu}{2}\right), \tag{6}$$

are independent, and gamma(α,β) denotes the gamma distribution with shape and scale parameters given by α and β respectively. Sampling of uMST and FM-uMST variates are implemented in **EMMIXuskew** in the `rmst` and `rfmmst` functions, respectively.

3. The EMMIXuskew algorithm

From (3) to (6), the uMST distribution admits a convenient hierarchical characterization that facilitates the computation of the maximum likelihood estimator (MLE) of the unknown model parameters using the EM algorithm, namely,

$$\begin{aligned}
\boldsymbol{Y} \mid \boldsymbol{u}, w &\sim N_p\left(\boldsymbol{\mu}+\boldsymbol{\Delta}\boldsymbol{u},\frac{1}{w}\boldsymbol{\Sigma}\right),\\
\boldsymbol{U} \mid w &\sim HN_p\left(\boldsymbol{0},\frac{1}{w}\boldsymbol{I}_p\right),\\
W &\sim \text{gamma}\left(\frac{\nu}{2},\frac{\nu}{2}\right),
\end{aligned}$$

where $HN_p(\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes the p-dimensional half-normal distribution with location parameter $\boldsymbol{\mu}$ and scale matrix $\boldsymbol{\Sigma}$.

3.1. Fitting of FM-uMST model via the EM algorithm

Let $\boldsymbol{Y}_1,\cdots,\boldsymbol{Y}_n$ be n independent observations of $\boldsymbol{Y}$. To formulate the estimation of the unknown parameters as an incomplete-data problem in the EM framework, we introduce a set of latent component labels $\boldsymbol{z}_j = (z_{1j},\ldots,z_{gj})$ $(j = 1,\ldots,n)$ in addition to the unobservable variables $\boldsymbol{u}_j$ and w_j, where each element z_{hj} is a zero-one indicator variable with $z_{hj} = 1$ if $\boldsymbol{y}_j$ belongs to the h-th component, and zero otherwise. Thus, $\sum_{h=1}^{g} z_{hj} = 1$ $(j = 1,\ldots,n)$.

It follows that the random vector $\boldsymbol{Z}_j$ corresponding to $\boldsymbol{z}_j$ follows a multinomial distribution with one trial and cell probabilities $\pi_1, \ldots, \pi_g$; that is, $\boldsymbol{Z}_j \sim \text{Mult}_g(1; \pi_1, \ldots, \pi_g)$.

The complete-data log likelihood function can be factored into the marginal densities of the $\boldsymbol{z}_j$, the conditional densities of the w_j given $\boldsymbol{z}_j$, and the conditional densities of the $\boldsymbol{y}_j$ given $\boldsymbol{u}_j$, w_j, and $\boldsymbol{z}_j$. Accordingly, the complete-data log likelihood is given by

$$\log L_c(\boldsymbol{\Psi}) = \log L_{1c}(\boldsymbol{\Psi}) + \log L_{2c}(\boldsymbol{\Psi}) + \log L_{3c}(\boldsymbol{\Psi}), \tag{7}$$

where

$$\begin{aligned}
L_{1c}(\boldsymbol{\Psi}) &= \sum_{h=1}^{g}\sum_{j=1}^{n} z_{hj} \log(\pi_h), \\
L_{2c}(\boldsymbol{\Psi}) &= \sum_{h=1}^{g}\sum_{j=1}^{n} z_{hj} \left[\left(\frac{\nu_h}{2}\right) \log\left(\frac{\nu_h}{2}\right) + \left(\frac{\nu_h}{2} + p - 1\right) \log(w_j) \right. \\
&\quad \left. - \log\Gamma\left(\frac{\nu_h}{2}\right) - \left(\frac{w_j}{2}\right)\nu_h \right], \\
L_{3c}(\boldsymbol{\Psi}) &= \sum_{h=1}^{g}\sum_{j=1}^{n} z_{hj} \left\{ -p\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_h| \right. \\
&\quad \left. - \frac{w_j}{2}\left[d_h(\boldsymbol{y}_j) + (\boldsymbol{u}_j - \boldsymbol{q}_{hj})^\top \boldsymbol{\Lambda}_h^{-1} (\boldsymbol{u}_j - \boldsymbol{q}_{hj}) \right] \right\},
\end{aligned} \tag{8}$$

and where

$$\begin{aligned}
d_h(\boldsymbol{y}_j) &= (\boldsymbol{y}_j - \boldsymbol{\mu}_h)^\top \boldsymbol{\Omega}_h^{-1} (\boldsymbol{y}_j - \boldsymbol{\mu}_h), \\
\boldsymbol{q}_{hj} &= \boldsymbol{\Delta}_h \boldsymbol{\Omega}_h^{-1} (\boldsymbol{y}_j - \boldsymbol{\mu}_h), \\
\boldsymbol{\Lambda}_h &= \boldsymbol{I}_p - \boldsymbol{\Delta}_h \boldsymbol{\Omega}_h^{-1} \boldsymbol{\Delta}_h, \\
\boldsymbol{\Omega}_h &= \boldsymbol{\Sigma}_h + \boldsymbol{\Delta}_h^2.
\end{aligned}$$

Here $\boldsymbol{\Psi}$ contains all the unknown parameters of the FM-uMST model.

The implementation of the EM algorithm requires alternating repeatedly the E- and M-steps until convergence in the case where the changes in the log likelihood values are less than some specified small value. The E-step calculates the expectation of the complete-data log likelihood given the observed data $\boldsymbol{y}$ using the current estimate of the parameters, known as the Q-function, given by

$$Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) \quad = \quad E_{\boldsymbol{\Psi}^{(k)}} \{ \log L_c(\boldsymbol{\Psi}) \mid \boldsymbol{y} \}.$$

The M-step then maximizes the Q-function with respect to the parameters $\boldsymbol{\Psi}$.

On the $(k+1)$-th iteration, the E-step requires the calculation of the conditional expectations

$$e_{1,j}^{(k)} = E_{\boldsymbol{\theta}^{(k)}}(W_j \mid \boldsymbol{y}_j), \tag{9}$$

$$\boldsymbol{e}_{2,j}^{(k)} = E_{\boldsymbol{\theta}^{(k)}}(W_j \boldsymbol{U}_j \mid \boldsymbol{y}_j), \tag{10}$$

$$\boldsymbol{e}_{3,j}^{(k)} = E_{\boldsymbol{\theta}^{(k)}}\left(W_j \boldsymbol{U}_j \boldsymbol{U}_j^\top \mid \boldsymbol{y}_j\right). \tag{11}$$

The conditional expectation of Z_{hj} given the observed data, is given, using Bayes' Theorem, by

$$\tau_{hj}^{(k)} = \frac{\pi_h^{(k)} f_p\left(\boldsymbol{y}_j; \boldsymbol{\mu}_h^{(k)}, \boldsymbol{\Sigma}_h^{(k)}, \boldsymbol{\delta}_h^{(k)}, \nu_h^{(k)}\right)}{\sum_{i=1}^{g} \pi_i^{(k)} f_p\left(\boldsymbol{y}_j; \boldsymbol{\mu}_i^{(k)}, \boldsymbol{\Sigma}_i^{(k)}, \boldsymbol{\delta}_i^{(k)}, \nu_i^{(k)}\right)}. \tag{12}$$

which can be interpreted as the posterior probability of membership of the h-th component by $\boldsymbol{y}_j$, using the current estimate $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$.

It can be shown that the conditional expectations $e_{1,j}^{(k)}$, $\boldsymbol{e}_{2,j}^{(k)}$, and $\boldsymbol{e}_{3,j}^{(k)}$ are given by

$$e_{1,hj}^{(k)} = \left(\frac{\nu_h^{(k)} + p}{\nu_h^{(k)} + d_h^{(k)}\left(\boldsymbol{y}_j\right)}\right) \frac{T_{p,\nu_h^{(k)}+p+2}\left(\boldsymbol{q}_{hj}^{(k)} \sqrt{\frac{\nu_h^{(k)}+p+2}{\nu_h^{(k)}+d_h^{(k)}(\boldsymbol{y}_j)}}; \boldsymbol{0}, \boldsymbol{\Lambda}_h^{(k)}\right)}{T_{p,\nu_h^{(k)}+p}\left(y_{hj}^{*(k)}; \boldsymbol{0}, \boldsymbol{\Lambda}_h^{(k)}\right)}, \tag{13}$$

$$\boldsymbol{e}_{2,hj}^{(k)} = e_{1,hj}^{(k)} E(\boldsymbol{X}), \tag{14}$$

and

$$\boldsymbol{e}_{3,hj}^{(k)} = e_{1,hj}^{(k)} E(\boldsymbol{X}\boldsymbol{X}^{\top}), \tag{15}$$

where $\boldsymbol{X}$ is a p-dimensional t-variate truncated to the positive hyperplane $\mathbb{R}^{+}$, which is distributed as

$$\boldsymbol{X} \sim tt_{p,\nu_h^{(k)}+p+2}\left(\boldsymbol{q}_{hj}^{(k)}, \left(\frac{\nu_h^{(k)} + d_h^{(k)}(\boldsymbol{y}_j)}{\nu_h^{(k)} + p + 2}\right) \boldsymbol{\Lambda}_h^{(k)}; \mathbb{R}^{+}\right), \tag{16}$$

where $tt_{p,\nu}(\boldsymbol{\mu}, \boldsymbol{\Sigma}; \mathbb{R}^{+})$ denotes the positively truncated t distribution with location vector $\boldsymbol{\mu}$, scale matrix $\boldsymbol{\Sigma}$, and ν degrees of freedom. The truncated moments $E(\boldsymbol{X})$ and $E(\boldsymbol{X}\boldsymbol{X}^{\top})$ can be swiftly evaluated by noting that they can be expressed in terms of the distribution function of a (non-truncated) multivariate central t random vector; Lee and McLachlan (2011, 2013a). Recently, Ho, Lin, Chen, and Wang (2012) have considered the moments of of the doubly truncated multivariate t distribution, but their result corresponding to (16) is incorrect; see Lee and McLachlan (2013a) for further details.

The $(k+1)$-th M-step consists of maximization of the Q-function with respect to $\boldsymbol{\Psi}$. It follows that an updated estimate of the unknown parameters of the FM-uMST model is given by

$$\boldsymbol{\mu}_h^{(k)} = \frac{\sum_{j=1}^{n} \tau_{hj}^{(k)} \left[e_{1,hj}^{(k)} \boldsymbol{y}_j - \boldsymbol{\Delta}_h^{(k)} \boldsymbol{e}_{2,hj}^{(k)}\right]}{\sum_{j=1}^{n} \tau_{hj}^{(k)} e_{1,hj}^{(k)}}, \tag{17}$$

$$\boldsymbol{\delta}_h^{(k+1)} = \left(\boldsymbol{\Sigma}_h^{(k)^{-1}} \odot \sum_{j=1}^{n} \tau_{hj}^{(k)} \boldsymbol{e}_{3,hj}^{(k)}\right)^{-1} \operatorname{diag}\left(\boldsymbol{\Sigma}_h^{(k)^{-1}} \sum_{j=1}^{n} \tau_{hj}^{(k)} \left(\boldsymbol{y}_j - \boldsymbol{\mu}_h^{(k+1)}\right) \boldsymbol{e}_{2,hj}^{(k)^{\top}}\right), \tag{18}$$

and

$$\begin{aligned}\boldsymbol{\Sigma}_h^{(k+1)} = \frac{1}{\sum_{j=1}^{n} \tau_{hj}^{(k)}} \sum_{j=1}^{n} \tau_{hj}^{(k)} \Big[&\boldsymbol{\Delta}_h^{(k+1)} \boldsymbol{e}_{3,hj}^{(k)^{\top}} \boldsymbol{\Delta}_h^{(k+1)^{\top}} - \left(\boldsymbol{y}_j - \boldsymbol{\mu}_h^{(k+1)}\right) \boldsymbol{e}_{2,hj}^{(k)^{\top}} \boldsymbol{\Delta}_h^{(k+1)} \\ &- \boldsymbol{\Delta}_h^{(k+1)} \boldsymbol{e}_{2,hj}^{(k)} \left(\boldsymbol{y}_j - \boldsymbol{\mu}_h^{(k+1)}\right)^{\top} + \left(\boldsymbol{y}_j - \boldsymbol{\mu}_h^{(k+1)}\right) \left(\boldsymbol{y}_j - \boldsymbol{\mu}_h^{(k+1)}\right)^{\top} e_{1,hj}^{(k)}\Big],\end{aligned} \tag{19}$$

where $\odot$ denotes element-wise matrix product. Note that (18) and also (16) are given incorrectly in Lee and McLachlan (2011).

An update $\nu_h^{(k+1)}$ of the degrees of freedom is obtained by solving iteratively the equation

$$\log\left(\frac{\nu_h^{(k+1)}}{2}\right) - \psi\left(\frac{\nu_h^{(k+1)}}{2}\right) = \frac{\sum_{j=1}^n \tau_{hj}^{(k)} \left[\log\left(\frac{\nu_h^{(k)}+d_h^{(k)}(\boldsymbol{y}_j)}{2}\right) - \psi\left(\frac{\nu_h^{(k)}+p}{2}\right) + \frac{\nu_h^{(k)}+p}{\nu_h^{(k)}+d_h^{(k)}(\boldsymbol{y}_j)}\right]}{\sum_{j=1}^n \tau_{hj}^{(k)}}, \tag{20}$$

where $\psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}$ is the Digamma function. This last equation has been simplified by making use of a one-step-late approximation (Green 1990) in updating the estimate of ν_h. As a consequence, it can affect the monotonicity of the likelihood function. Our experience suggests that this rarely happens. The monotonicity of the likelihood can be preserved by working with the exact expression as given by Equation (73) in Lee and McLachlan (2013a). The algorithm described in this section is implemented as the `fmmst` function in **EMMIXuskew**.

3.2. Choosing initial values

It is important to obtain suitable initial values in order for `fmmst` to converge quickly. In **EMMIXuskew**, starting values for the model parameters are based on an initial clustering given by k-means. Twenty attempts of k-means are performed, and the starting component labels $\boldsymbol{z}_j^{(0)}$ $(j = 1, \ldots, n)$ are initialized according to the clustering result with the highest relative log likelihood (see Lee and McLachlan (2013a)). The other parameters are initialized as follows:

$$\begin{aligned}
\boldsymbol{\Sigma}^{(0)} &= \boldsymbol{S}_h - (a-1)\,\mathrm{diag}\,(\boldsymbol{s}_h)\,, \\
\boldsymbol{\delta}^{(0)} &= \mathrm{sign}(\boldsymbol{\gamma}_h)\sqrt{\frac{(1-a)\pi}{\pi-2}}\;\boldsymbol{s}_h^*, \\
\boldsymbol{\mu}^{(0)} &= \bar{\boldsymbol{y}} - \sqrt{\frac{2}{\pi}}\boldsymbol{\delta}^{(0)}, \\
\nu^{(0)} &= 40,
\end{aligned} \tag{21}$$

where $\boldsymbol{S}_h$ is the sample covariance of the h-th component, and where $\boldsymbol{\gamma}_h$ is the sample skewness of the h-th component, whose i-th element is given by

$$\gamma_i = \frac{n^{-1}\sum_{j=1}^n (y_{ij}-\mu_i)^3}{\left(n^{-1}\sum_{j=1}^n (y_{ij}-\mu_i)^2\right)^{\frac{3}{2}}} \quad (i = 1, \ldots, p),$$

and where y_{ij} denotes the i-th element of the j-th observation, and μ_i is the i-th element of $\boldsymbol{\mu}$. Here, $\boldsymbol{s}_h$ denotes the vector created by extracting the main diagonal of $\boldsymbol{S}_h$, and the vector $\boldsymbol{s}_h^*$ is created by taking the square root of each element in $\boldsymbol{s}_h$. The scalar a is varied systematically across the interval $(0, 1)$ to search for a (relatively) optimal set of starting values for the model parameters.

Parameter	R arguments	Dimensions	Description
$\boldsymbol{\mu}$	`mu`	$p \times 1 \times g$	Location parameter
$\boldsymbol{\Sigma}$	`sigma`	$p \times p \times g$	Scale matrix
$\boldsymbol{\delta}$	`delta`	$p \times 1 \times g$	Skewness parameter
ν	`dof`	$g \times 1$	Degrees of freedom
π	`pro`	$g \times 1$	Mixing proportions

Table 1: Structure of the model parameters in **EMMIXuskew**.

3.3. Stopping rule

EMMIXuskew adopts a traditional stopping criterion which is based on the absolute change in the size of the log likelihood. An Aitken acceleration-based strategy is described in Lin (2010). The algorithm is terminated when the absolute difference between the log likelihood value and the asymptotic log likelihood value is less than a sepcified tolerance, ϵ, that is

$$\left| L_{\infty}^{(k+1)} - L^{(k+1)} \right| < \epsilon, \tag{22}$$

where $L_{\infty}^{(k+1)}$ is the asymptotic estimate of the log likelihood at the $(k+1)$-th iteration, given by $L_{\infty}^{(k+1)} = L^{(k)} + \frac{L^{(k+1)} - L^{(k)}}{1-\alpha^{(k)}}$, and $\alpha^{(k)} = \frac{L^{(k+1)} - L^{(k)}}{L^{(k)} - L^{(k-1)}}$ is the Aitken's acceleration at the k-th iteration. The default tolerance is $\epsilon = 10^{-3}$, but the user can specify a different value.

4. Using the EMMIXuskew package

The parameters of the FM-uMST model in **EMMIXuskew** are specified as a list structure containing the elements described in Table 1. The parameters $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{\delta}$ are each implemented as a list of `g` matrices, where `g` is the number of components in the fitted model. For example, `mu[[2]]` is a $p \times 1$ matrix representing $\boldsymbol{\mu}_2$. Each `sigma[[h]]` $(h = 1, \ldots, g)$ is a $p \times p$ matrix representing the symmetric positive definite scale matrix of the h-th component. The parameters `dof` and `pro` are g by 1 arrays, representing the vector of degrees of freedom and the vector of mixing proportions for each component, respectively.

The probability density function of a multivariate skew t distribution is calculated by the `dmst` function. The parameter `dat` is an $n \times p$ matrix, containing the coordinates of the n point(s) at which the density is to be evaluated. The following command will return a vector of n density values.

```
dmst(dat, mu, sigma, delta, dof)
```

For a FM-uMST density, the function `dfmmst` can be used.

```
dfmmst(dat, mu, sigma, delta, dof, pro)
```

4.1. Generating samples from a FM-uMST distribution

Consider generating a random sample of n p-dimensional uMST observations, with location parameter $\boldsymbol{\mu}$, scale matrix $\boldsymbol{\Sigma}$, skewness parameter $\boldsymbol{\delta}$, and degrees of freedom ν. The function `rfmmst` supports two types of inputs – the parameters can be passed as separate arguments, or as a single list argument `known` with elements as specified in Table 1:

```
rfmmst(g, n, mu, sigma, delta, dof, pro, known = NULL, ...)
```

As an example, suppose that $\boldsymbol{\mu} = (1, 2)^\top$, $\boldsymbol{\Sigma}$ is the identity matrix, $\boldsymbol{\delta} = (-1, 1)^\top$, and $\nu = 4$. Then the following command will generate a random sample of 500 observations from the $uMST_{2,4}(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\delta})$ distribution,

```
R> rfmmst(1, 500, c(1, 2), diag(2), c(-1, 1), 4, 1)
```

To generate a mixture of uMST random samples, the above command can be issued. Alternatively, the parameters can be specified in a list structure (Table 1) `obj` as follows:

```
R> obj <- list()
R> obj$mu <- list(c(17, 19), c(5, 22), c(6, 10))
R> obj$sigma <- list(diag(2), matrix(c(2, 0, 0, 1), 2),
+    matrix(c(3, 7, 7, 24), 2))
R> obj$delta <- list(c(3, 1.5), c(5, 10), c(2, 0))
R> obj$dof <- c(1, 2, 3)
R> obj$pro <- c(0.25, 0.25, 0.5)
R> rfmmst(3, 500, known = obj)
```

An output of the `rfmmst` function consists of $p + 1$ columns. The first p columns are the coordinates of the generated sample. The last column indicates from which component each data point is generated. Executing the above command will generate an output similar to the following:

```
           [,1]       [,2] [,3]
  [1,] 17.310999 18.6616688    1
  [2,] 17.723334 18.4303338    1
  [3,] 19.831565 20.3413001    1
  [4,] 20.017125 19.0167033    1
  [5,] 17.567501 19.6295725    1
  [6,] 19.793005 19.6365686    1
  [7,] 17.427954 21.0046329    1
  [8,] 21.482355 19.0482537    1
  [9,] 15.778796 18.9365259    1
 [10,] 17.894952 20.8721242    1
     ... rest omitted ...
```

4.2. Fitting a single multivariate skew t distribution

To fit a specified FM-uMST model, the core function in **EMMIXuskew**, `fmmst`, is used. This implements the algorithm described in Section 3. A typical function call of `fmmst` is:

```
fmmst(g, dat, initial = NULL, known = NULL, itmax = 100, eps = 1e-3,
  nkmeans = 20, tmethod = 1, print = TRUE)
```

The main arguments used within this function are:

- `g`: a scalar that specifies the number of uMST components to be fitted.
- `dat`: an $n \times p$ matrix containing the data.
- `initial`: a list that specifies the initial values used to start the algorithm.
- `known`: a list that specifies any model parameters that are known and so not required to be estimated.
- `itmax`: a scalar that specifies the maximum number of iterations to be used.
- `eps`: a scalar that specifies the termination criterion of the EM algorithm loop.
- `nkmeans`: an integer that specify the number of k-means trials to be used to select the best set of initial values. (20) is to be used for the update of the degrees of freedom.

Note that if the initial values of the model parameters are provided by the user, the argument `initial` is expected to be structured as described in Table 1. Similarly, `known` is expected to have the same structure. When `initial = NULL`, `fmmst` will generate a set of initial values using the procedure described in Section 3.2. Any parameters specified in `known` are taken as known parameters and hence are not estimated by `fmmst`. There is no need to specify the values of all the parameters in `initial` and `known` when only some of the parameters are known. Parameters that are not specified in the function call are estimated by `fmmst`. By default, `fmmst` performs 20 k-means attempts when searching for the best initial value. The user can specify a different value using `nkmeans`. The termination criterion for the **EMMIXuskew** algorithm is controlled by the parameters `itmax` and `eps`. The EM loop terminates when either one of the two criterion is satisfied, whichever occurs first: (a) the EM loop reaches `itmax` iterations (default is 100 iterations), or (b) the absolute difference between the current log likelihood value and that the asymptotic log likelihood value is smaller than `eps` (default is `1e-3`). The last argument of `fmmst` is `print`. When the option `print` is set to `TRUE` (default), `fmmst` prints the log likelihood value at each iteration and displays a summary of the parameters of the fitted model after termination. To turn off the print mode, simply set `print = FALSE`. For further details of the arguments of `fmmst`, including `tmethod` which selects the method for computing values of the multivariate t distribution function, the reader is referred to the documentation of `fmmst`. This can be accessed by typing `?fmmst` at the R command prompt.

We consider now the T-cell phosphorylation dataset (Maier, Anderson, Jager, Wicker, and Hafler 2007) as an example of asymmetrically distributed data, available from Pyne, Hu, Wang, Rossin, Lin, Maier, Baecher-Allan, McLachlan, Tamayo, Hafler, Jager, and Mesirow (2009b). The data contain measurements of blood samples stained with four antibodies, CD4, CD45RA, SLP76, and ZAP70. For illustration, we randomly select 500 observations and focus on two of the variables, CD4 and ZAP70. To fit a MST model to this bivariate Lymphoma dataset, under the default settings, the following command is issued:

```
R> RNGversion("3.0.2"); set.seed(12345)
R> data("Lympho")
R> LymphoSample <- Lympho[sample(1:nrow(Lympho), 500), ]
R> Fit <- fmmst(1, LymphoSample)
```

A summary of the output of the fitted model can be obtained using the `summary` function. This prints the values of the fitted model parameters for each component. For a fitted uMST model, the weighting proportion (which is 1) is not printed. The following output shows a typical summary of a fitted single component uMST model.

```
R> summary(Fit)

Finite Mixture of Multivariate Skew t-distribution
with 1 component
Mean:
         [,1]
[1,] 4.808390
[2,] 5.500602

Scale matrix:
[[1]]
           [,1]       [,2]
[1,] 0.06792546 0.03733140
[2,] 0.03733140 0.04827617

Skewness parameter:
           [,1]
[1,] -0.7085353
[2,] -0.7994653

Degrees of freedom:
5.867415
```

To view a more detailed output of the `fmmst` function, the `print` function is called. This outputs a list containing 11 elements. The first five elements give the estimates of the parameters of the fitted FM-uMST model, as described in Table 1.

The posterior probability of component membership is given by the output argument `tau`, a $g \times n$ matrix where the rows corresponds to the component number. The final partition of each data point, based on `tau`, is stored as `clusters`. The value of the log likelihood function, evaluated with the current parameter estimates, is given by `loglik`. The last two arguments `aic` and `bic` are the values of the Akaike information criterion (AIC) and the Bayes information criterion (BIC), respectively. The following output shows an excerpt from the second part of the `print` output of the fitted model.

```
R> print(Fit)

Finite Mixture of Multivariate Skew t-distributions
with 1 component
... first five components omitted ...

$tau
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
```

```
[1,]    1    1    1    1    1    1    1    1    1    1    1    1    1
... rest omitted ...

$clusters
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
... rest omitted ...

$loglik
[1] -880.7115
$lk
 [1] -925.9661 -918.5664 -914.9227 -912.8726 -911.5528 -910.5935 -909.8236
... rest omitted ...

$iter
[1] 98

$eps
[1] 0.000967241

$aic
[1] 1773.423

$bic
[1] 1798.711

attr(,"class")
[1] "fmmst"
```

As mentioned previously, initial values for the EM algorithm can be specified by the user. Suppose an initial guess of $\boldsymbol{\mu}$ for the above example is $(5,6)^\top$, then one can specify $\boldsymbol{\mu}^{(0)}$ to be $(5,6)^\top$ by issuing the command:

```
R> obj2 <- list()
R> obj2$mu <- list(c(5, 6))
R> fmmst(1, LymphoSample, initial = obj2)
```

This will start the EM algorithm with the specified value for $\boldsymbol{\mu}^{(0)}$, and the other parameters using (21). The user can further demand more k-means trials to be performed by increasing `nkmeans`, for example, to 50 trials. This can be achieved by issuing the following command.

```
R> fmmst(1, LymphoSample, nkmeans = 50)
```

4.3. Fitting mixtures of multivariate skew t distributions

This section presents an illustration of fitting a mixture of unrestricted skew t distributions to some bivariate bimodal asymmetric data. We consider the Australian Institute of Sport (AIS) data from Cook and Weisberg (1994), where thirteen body measurements on 102 male

and 100 female athletes were recorded. In this example, we consider the clustering of the data with a two component skew t mixture model based on the two variables Height and Body fat. By setting `print = TRUE`, we can examine the value of the log likelihood function at each iteration.

```
R> Fit2 <- fmmst(2, ais[, c(2, 12)], print = TRUE)

Finite Mixture of Multivariate Skew t-distributions
with 2 components
  ------------------------------------------------
  Iteration  1 : loglik =  -1372.719
  Iteration  2 : loglik =  -1370.503
  Iteration  3 : loglik =  -1368.776
  Iteration  4 : loglik =  -1367.395
  Iteration  5 : loglik =  -1366.256
       ... rest omitted ...
  ------------------------------------------------
  Iteration 100: loglik = -1343.534

Component means:
           [,1]      [,2]
[1,] 181.451777 181.91337
[2,]   5.814342  13.68289

Component scale matrices:
[[1]]
          [,1]      [,2]
[1,] 61.629119 2.3166120
[2,]  2.316612 0.1513647

[[2]]
         [,1]     [,2]
[1,] 26.36822 18.76079
[2,] 18.76079 16.12127

Component skewness parameters:
         [,1]      [,2]
[1,] 3.585828 -9.577637
[2,] 5.728121  5.974213

Component degrees of freedom:
28.99095 60.00839

Component mixing proportions:
0.5898522 0.4101478
```

We compare the results with two other model-based clustering methods provided by the package **mixsmsn** (Prates, Lachos, and Cabral 2012) and **EMMIX-skew** (Wang *et al.* 2012).

As mentioned previously, this two models are based on mixture of restricted versions of the multivariate skew t distributions. The first model adopts the skew normal/independent skew t distribution (Cabral *et al.* 2012) as its component densities, which is equivalent to the restricted skew t distribution (Pyne *et al.* 2009a) used in the second model. However, it should be noted that, in the ECME algorithm implemented in the package **mixsmsn**, the component degrees of freedom are constrained to be the same. A comparison of the table of cluster labels (permuted where necessary to minimize the number of misallocations) with the true class labels (given by `ais$Sex` in this example) reveals that the FM-uMST model has a higher number of correct allocations (183 compared to 162 and 157 given by **mixsmsn** and **EMMIX-skew**, respectively). Thus, the unrestricted FM-uMST model in **EMMIXuskew** gives a more accurate clustering in this case.

```
R> library("mixsmsn")
R> Fit3 <- smsn.mmix(ais[c(2, 12)], g = 2, family = "Skew.t", group = TRUE)
R> Fit4 <- EmSkew(ais[c(2,12)], 2, "mst", debug = FALSE)
R> table(ais$Sex, Fit3$group)

     1  2
  0 91 11
  1 29 71

R> table(ais$Sex, Fit4$clust)

     1  2
  0 89 13
  1 32 68

R> table(ais$Sex, Fit2$clusters)

     1  2
  0 97  5
  1 14 86
```

4.4. Testing for the significance of the skewness parameter

When we set $\boldsymbol{\delta} = \mathbf{0}$ in (1), we obtain the multivariate t density. The function

```
fmmt(g, dat, initial = NULL, known = NULL, itmax = 100, eps = 1e-3,
  nkmeans = 20, print = TRUE)
```

implements the EM algorithm for fitting finite mixtures of multivariate t (FM-MT) distributions (McLachlan and Peel 2000).

To test whether the skewness parameter in the FM-uMST model is significant, one can construct a likelihood ratio test for the null hypothesis $H_0 : \boldsymbol{\delta}_1 = \ldots = \boldsymbol{\delta}_g = \mathbf{0}$ versus the alternative hypothesis where at least one of $\boldsymbol{\delta}_h$ $(h = 1, \ldots, g)$ is different from $\mathbf{0}$. This leads to the test statistic

$$LR = -2\left(L_t - L_{st}\right), \tag{23}$$

where L_t and L_{st} denote the log likelihood value associated with the FM-MT model and the FM-uMST model, respectively. It follows that the test statistics is asymptotically distributed as χ^2_r, where r is the difference between the number of parameters under the alternative and null hypotheses. This test is implemented in the function `delta.test(stmodel = NULL, tmodel = NULL, stloglik, tloglik, r)`, where the first two arguments are the output from `fmmst` and `fmmt` respectively. Alternatively, the user can provide the log likelihood values of the two models and the value of r directly by specifying the last three arguments of `delta.test()`. The output of the function is the p value of the test.

Consider again the AIS example in Section 4.6. If we examine the cluster labels given by the FM-MT model, we can see that it yields a noticeably higher number of misallocations than the skew t mixture model. A test for $\boldsymbol{\delta} = \mathbf{0}$ can be performed by issuing the following commands. In this case, the small p value suggests there is strong evidence that the skewness parameter in the FM-uMST fit is significantly different from zero.

```
R> Fit5 <- fmmt(2, ais[, c(2, 12)])
R> table(ais$Sex, Fit5$clusters)

     1  2
  0 88 14
  1 23 77

R> delta.test(Fit2, Fit5)

[1] 0.0003360263
```

4.5. Discriminant analysis

Discriminant analysis based on a specified FM-uMST model can be performed using the `fmmstDA` function.

```
fmmstDA(g, dat, model)
```

The data in `dat` are assigned to the cluster corresponding to the component of the FM-uMST model with the highest posterior probability. Specifications of the model parameters must be provided in `model`, which is typically an output from `fmmst`. Optionally, `model` can be specified by the user as a list of at least six elements: the five model parameters, and a vector of cluster labels `clusters`. The following commands shows an example using `fmmstDA`. A random sample of FM-uMST variables is generated from `rfmmst`, the first part of which is used as training set, and the second is a testing set. The FM-uMST model fitted to the training set is then used for classifying the data in the testing set.

```
R> RNGversion("3.0.2"); set.seed(732)
R> X <- rfmmst(3, 200, known = obj)
R> Ind <- sample(1:nrow(X), 175)
R> train <- X[Ind, ]
R> test <- X[-Ind, ]
R> trainmodel <- fmmst(3, train[, 1:2])
R> results <- fmmstDA(3, test[, 1:2], trainmodel)
R> table(test[, 3], results)
```

```
   results
     1  2  3
1    0  6  0
2    0  0  5
3   14  0  0
```

4.6. Visualization of fitted contours

The **EMMIXuskew** package supports visualization of the contours of a FM-uMST model in 2D and 3D. The plots are generated by the functions `fmmst.contour.2d` and `fmmst.contour.3d`,

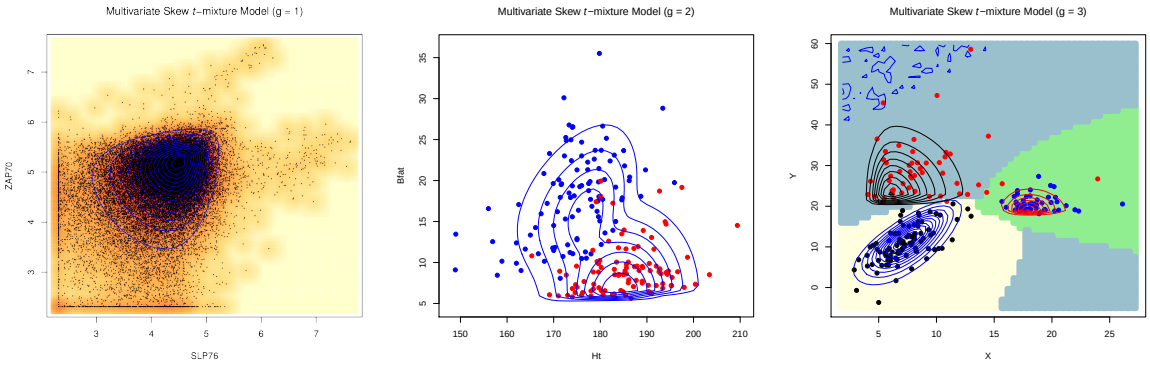
```
fmmst.contour.2d(dat, model, grid = 50, drawpoints = TRUE, clusters = NULL,
  levels = 10, component = NULL, map = c("scatter", "heat", "cluster"), ...)
fmmst.contour.3d(dat, model, grid = 20, drawpoints = TRUE, clusters = NULL,
  levels = 0.9, component = NULL, ...)
```

In `fmmst.contour.2d` (`fmmst.contour.3d`), the first argument `dat` is a matrix of coordinates with two (three) columns. The second argument `model`, similar to that in `fmmstDA()`, is either an output from `fmmst()`, or a list containing the five model parameters and the cluster labels. The grid size is determined by `grid`. By default, the data points are included in the plot. If only the contour are required, the option `drawpoints = FALSE` should be set. When including the points in a plot, `clusters` specifies the component labels of each point according to which the data points will be coloured. The argument `levels` is either an integer specifying the number of contour lines to be plotted, or a vector of quantile values. For `fmmst.contour.3d`, only the 90-th percentile contour is plotted by default. If more contours are required, the argument `levels` should be a vector of the required quantiles. For example, if a plot of the 25-th, 50-th, and 75-th percentiles are required, then `levels = c(0.25, 0.5, 0.75)`. Bivariate data have the option of being plotted as an intensity map (via **KernSmooth**, Wand and Ripley 2013) instead of scatter plot. This can be obtained by setting `map = "heat"`. There is also an option for plotting a cluster map of a fitted model using the option `map = "cluster"`. Plots for specific components of a mixture model can be requested with the argument `component`. When `component = NULL` (which is default), the mixture contour is plotted. When `component` is a vector with length between 1 and g, the specified components are plotted and the mixing proportion is not taken into account. The last argument of the `fmmst.contour` functions "`...`" allows the user to pass additional arguments to the `plot` function, such as the colour and size of the points.

Figure 1a shows the contour of the fitted MST model to the Lymphoma data. Here a heatmap of the original data is used. This plot can be generated via the command,

```
R> fmmst.contour.2d(Lympho, model = Fit, map = "heat", xlab = "SLP76",
+    ylab = "ZAP70")
```

The default `fmmst.contour.2d` function will return a scatter plot of the data in 2D superimposed with the contours of the fitted mixture model. For example, the following command generates a contour plot of the fitted FM-uMST model to the `ais` data in Section 4.2 (Figure 1b). Note that `fmmst.contour.2d` coloured the sample points according to the clustering given by the argument `clusters`:



(a) (b) (c)

Figure 1: 2D contour plots generated by the `fmmst.contour.2d` function. (a) The fitted contour of the single component uMST model plotted over the hue intensity diagram of the Lymphoma dataset; (b) the default mixture contour plot of the fitted two-component FM-uMST model of the AIS dataset; (c) the contour of the individual components of the three-component model fitted to a bivariate synthetic sample plotted over the cluster map of the sample.

```
R> label <- abs(unclass(ais$Sex) - 2)
R> fmmst.contour.2d(ais[, c(2, 12)], model = Fit2, clusters = label,
+    xlab = "Ht", ylab = "Bfat")
```

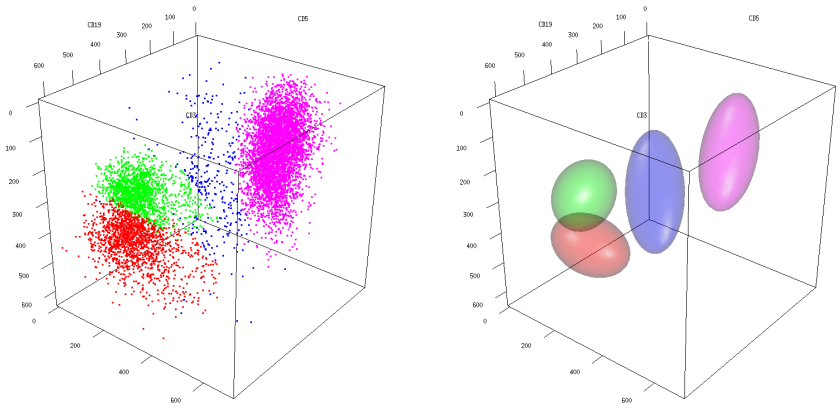
Suppose we are interested in visualizing a clustering map of the fitted model to the simulated data in Section 4.5. This plot can be generated by issuing the following command.

```
R> fmmst.contour.2d(X, model = trainmodel, clusters = X[, 3],
+    map = "cluster", component = 1:3)
```

The output is given in Figure 1c.

To demonstrate the use of `fmmst.contour.3d`, we consider the clustering of a trivariate Diffuse Large B-cell Lymphoma (DLBCL) dataset provided by the British Columbia Cancer Agency (Aghaeepour *et al.* 2013; Spidlen, Breuer, Rosenberg, Kotecha, and Brinkman 2012). The data contain fluorescent intensities of multiple conjugated antibodies (known as markers) stained on a sample of over 8000 cells derived from the lymph nodes of patients diagnosed with DLBCL. In flow cytometric analysis, these parallel measurements of fluorescent intensities can be used to study the differential expression of different surface and intracellular proteins of a given blood sample. The analysis typically involves the identification of cell populations from the multidimensional dataset, currently performed manually by visually separating regions (gates) of interests on a series of sequential bivariate projections of the data, a process known as *gating*. Due to the subjective and time-consuming nature of this approach, and the difficulty in detecting higher-dimensional inter-marker relationships, many efforts have been made to develop computational methods to automate the gating process.

The DLBCL samples here were stained with three markers CD3, CD5, and CD19. The task is to automatically gate the cells by clustering the data into four groups. Hence we fit a four-component FM-uMST model to the data. The maximum number of iterations was increased to 300.



(a) (b)

Figure 2: 3D contours plot of the DLBCL dataset generated by the `fmmst.contour.3d` function. (a) A scatterplot of the data coloured according to the the true clustering labels of the DLBCL dataset; (b) fitted contour of the three component FM-uMST model for the DLBCL dataset.

A scatterplot of the data is shown in Figure 2, where the dots are coloured according to the clustering provided by human experts, which are considered as the 'true' class labels. Figure 2b shows the 98-th percentile density contours of the four components of the fitted model which are displayed with matching colours. The 3D plot uses the **rgl** (Adler and Murdoch 2013) visualization device system, and hence supports user friendly interactive navigation. The plots can be rotated in real-time to select a suitable viewpoint. The following code can be used to generate the 3D plots in Figure 2.

```
R> RNGversion("3.0.2"); set.seed(53)
R> Fit6 <- fmmst(4, DLBCL, nkmeans = 1)
R> fmmst.contour.3d(DLBCL, model = Fit6, level = 0.985, drawpoints = FALSE,
+    xlab = "CD3", ylab = "CD5", zlab = "CD19", component = 1:4, grid = 50)
```

The effectiveness of a clustering can be obtained by comparing its error rate with the cluster labels from manual expert gating taken to be the true class labels. This error rate is calculated for each permutation of the cluster labels of the clustering result under consideration and the rate reported is the minimum value over all such permutations. Note that dead cells were removed before evaluating the error rate against the benchmark results. For comparison, we calculated the error rate associated with the clustering results given by three other methods – FLAME (Pyne *et al.* 2009a), flowClust (Lo, Brinkman, and Gottardo 2008) and flowMeans (Aghaeepour, Nikoloc, Hoos, and Brinkman 2011). From Table 2, the FM-uMST model clearly shows superior performances in this dataset.

Method	FLAME	flowClust	flowMeans	FM-uMST
Error rate	0.302	0.203	0.304	0.056

Table 2: Error rate of misclassification of four methods for the DLBCL dataset.

5. Concluding remarks

We have presented the R package **EMMIXuskew** for fitting finite mixtures of unrestricted multivariate skew t distributions to heterogeneous asymmetric data. The package implements a closed-form EM algorithm for fitting FM-uMST models and provides user-friendly visualization of the fitted contours in 2D and 3D. The major features of the software have been demonstrated on three real examples on the T-cell phosphorylation data, the Australian Institute of Sports (AIS) data, and the DLBCL dataset. The clustering results were compared to those obtained via mixtures of restricted multivariate skew t distributions and other methods. In both the AIS and DLBCL illustrations, the unrestricted model gave better clustering results with respect to the true class labels.

It should be noted that the fitting of the unrestricted skew t mixture model can be quite slow in higher dimensional applications, due to the computationally intensive procedure involved in the calculation of multivariate t distribution function values. The algorithm would benefit from further research on applicable acceleration techniques, for example, the implementation of the SQUAREM strategy (Varadhan and Roland 2008).

Acknowledgments

This work is supported by a grant from the Australian Research Council. Also, we would like to thank Professor Seung-Gu Kim for comments and corrections, and Drs. Kui (Sam) Wang, Saumyadipta Pyne, and Felix Lamp for their helpful discussions on this topic.

References

Adler D, Murdoch D (2013). ***rgl**: 3D Visualization Device System (OpenGL)*. R package version 0.93.986, URL http://CRAN.R-project.org/package=rgl.

Aghaeepour N, Finak G, The FLOWCAP Consortium, The DREAM Consortium, Hoos H, Mosmann T, Gottardo R, Brinkman RR, Scheuermann RH (2013). "Critical Assessment of Automated Flow Cytometry Analysis Techniques." *Nature Methods*, **10**, 228–238.

Aghaeepour N, Nikoloc R, Hoos HH, Brinkman RR (2011). "Rapid Cell Population Identification in Flow Cytometry Data." *Cytometry A*, **79**, 6–13.

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

Azzalini A, Capitanio A (2003). "Distributions Generated by Perturbation of Symmetry with Emphasis on a Multivariate Skew t Distribution." *Journal of the Royal Statistical Society B*, **65**, 367–389.

Banfield JD, Raftery AE (1993). "Model-Based Gaussian and non-Gaussian Clustering." *Biometrics*, **49**, 803–821.

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.

Böhning D (2000). *Computer Assisted Analysis of Mixtures and Applications: Meta-Analysis, Disease Mapping, and Others*. Chapman and Hall/CRC, London.

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

Cook RD, Weisberg S (1994). *An Introduction to Regression Graphics*. John Wiley & Sons, New York.

Everitt BS, Hand DJ (1981). *Finite Mixture Distributions*. Chapman & Hall, London.

Fraley C, Raftery AE (1998). "How Many Clusters? Which Clustering Methods? Answers via Model-Based Cluster Analysis." *Computer Journal*, **41**, 578–588.

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

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.

Green PJ (1990). "On Use of the EM Algorithm for Penalized Likelihood Estimation." *Journal of the Royal Statistical Society B*, **52**, 443–452.

Gupta AK (2003). "Multivariate Skew-t Distribution." *Statistics*, **37**, 359–363.

Ho HJ, Lin TI, Chen HY, Wang WL (2012). "Some Results on the Truncated Multivariate t Distribution." *Journal of Statistical Planning and Inference*, **142**, 25–40.

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 (2011). "On the Fitting of Mixtures of Multivariate Skew t-Distributions via the EM Algorithm." *Technical report*, arXiv e-prints. arXiv: 1109.4706 [stat.ME], URL http://arXiv.org/abs/1109.4706.

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

Lee SX, McLachlan GJ (2013b). "On Mixtures of Skew-Normal and Skew t-Distributions." *Advances in Data Analysis and Classification*, **7**, 241–266.

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

Lindsay BG (1995). *Mixture Models: Theory, Geometry, and Applications*. NSF-CBMS Regional Conference Series in probability and Statistics, Vol. 5 (Institute of Mathematical Statistics and the American Statistical Association), Alexandria, VA.

Lo K, Brinkman RR, Gottardo R (2008). "Automated Gating of Flow Cytometry Data via Robust Model-Based Clustering." *Cytometry A*, **73**, 321–332.

Maier LM, Anderson DE, Jager PLD, Wicker LS, Hafler DA (2007). "Allelic Variant in CTLA4 Alters T Cell Phosphorylation Patterns." *Proceedings of the National Academy of Sciences of the USA*, **104**, 18607–18612.

McLachlan GJ, Basford KE (1988). *Mixture Models: Inference and Applications*. Marcel Dekker, New York.

McLachlan GJ, Peel D (2000). *Finite Mixture Models*. 2nd edition. John Wiley & Sons, New York.

Mengersen KL, Robert CP, Titterington, M D (2011). *Mixtures: Estimation and Applications*. John Wiley & Sons, New York.

Prates M, Lachos V, Cabral C (2012). ***mixsmsn**: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions*. R package version 1.0-2, URL http://CRAN.R-project.org/package=mixsmsn.

Pyne S, Hu X, Wang K, Rossin E, Lin TI, Maier LM, Baecher-Allan C, McLachlan GJ, Tamayo P, Hafler DA, Jager PLD, Mesirow JP (2009a). "Automated High-Dimensional Flow Cytometric Data Analysis." *Proceedings of the National Academy of Sciences USA*, **106**, 8519–8524.

Pyne S, Hu X, Wang K, Rossin E, Lin TI, Maier LM, Baecher-Allan C, McLachlan GJ, Tamayo P, Hafler DA, Jager PLD, Mesirow JP (2009b). *FLAME: Flow Analysis with Automated Multivariate Estimation*. URL http://www.broadinstitute.org/cancer/software/genepattern/modules/FLAME/published_data.

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/.

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.

Spidlen J, Breuer K, Rosenberg C, Kotecha N, Brinkman RR (2012). "FlowRepository – A Resource of Annotated Flow Cytometry Datasets Associated with Peer-Reviewed Publications." *Cytometry A*, **81**, 727–731.

Titterington DM, Smith AFM, Markov, E U (1985). *Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons, New York.

Varadhan R, Roland C (2008). "Simple and Globally Convergent Methods for Accelerating the Convergence of Any EM Algorithm." *Scandinavian Journal of Statistics*, **35**, 335–353.

Wand M, Ripley BD (2013). ***KernSmooth**: Functions for Kernel Smoothing for Wand & Jones (1995)*. R package version 2.23-10, URL http://CRAN.R-project.org/package=KernSmooth.

Wang K, McLachlan GJ, Ng SK, Peel D (2012). ***EMMIX-skew**: EM Algorithm for Mixture of Multivariate Skew Normal/t Distributions*. R code version 1.0.16, URL http://www.maths.uq.edu.au/~gjm/mix_soft/EMMIX-skew.

Affiliation:

Geoffrey J. McLachlan
Department of Mathematics
University of Queensland
Brisbane, Australia
E-mail: g.mclachlan@uq.edu.au

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 55, Issue 12 *Submitted:* 2012-02-03
November 2013 *Accepted:* 2013-11-15