# A robust approach to model-based classification based on trimming and constraints

**3 authors**, including:

Andrea Cappozzo
Politecnico di Milano
**17** PUBLICATIONS **29** CITATIONS

SEE PROFILE

Francesca Greselin
Università degli Studi di Milano-Bicocca
**94** PUBLICATIONS **511** CITATIONS

SEE PROFILE

**Some of the authors of this publication are also working on these related projects:**

Income Inequality in Canada View project

Mixtures of skew normal distributions and an efficient EM algorithm to robustly estimate them View project

# A robust approach to model-based classification based on trimming and constraints

## Semi-supervised learning in presence of outliers and label noise

Andrea Cappozzo[*]    Francesca Greselin[*]    Thomas Brendan Murphy[†]

August 6, 2019

### Abstract

In a standard classification framework a set of trustworthy learning data are employed to build a decision rule, with the final aim of classifying unlabelled units belonging to the test set. Therefore, unreliable labelled observations, namely outliers and data with incorrect labels, can strongly undermine the classifier performance, especially if the training size is small. The present work introduces a robust modification to the Model-Based Classification framework, employing impartial trimming and constraints on the ratio between the maximum and the minimum eigenvalue of the group scatter matrices. The proposed method effectively handles noise presence in both response and exploratory variables, providing reliable classification even when dealing with contaminated datasets. A robust information criterion is proposed for model selection. Experiments on real and simulated data, artificially adulterated, are provided to underline the benefits of the proposed method.

## 1   Introduction

In statistical learning, we define classification as the task of assigning group memberships to a set of unlabelled observations. Whenever a labelled sample (i.e., the training set) is available, the information contained in such dataset is exploited to classify the remaining unlabelled observations (i.e., the test set), either in a supervised or in a semi-supervised manner, depending whether the information contained in the test set are included in building the classifier (e.g. McNicholas, 2016). Either way, the presence of unreliable data points can be detrimental for the classification process, especially if the training size is small (Zhu and Wu, 2004).

Broadly speaking, noise is anything that obscures the relationship between the attributes and the class membership (Hickey, 1996). In a classification context, Wu (1995) distinguishes between two types of noise: attribute noise and class noise. The former is related to contamination in the exploratory variables, that is when observations present unusual values on their predictors; whereas the latter refers to samples whose associated labels are wrong. Zhu and Wu (2004) and the recent work of Prati et al. (2018) offer an extensive review on the topic and the methods that have been proposed in the literature to deal with attribute noise and class noise, respectively. Generally, three main approaches can be employed when building a classifier from a noisy dataset: cleaning the data, modeling the noise and using robust estimators of model parameters (Bouveyron and Girard, 2009).

The approach presented in this paper is based on a robust estimation of a Gaussian mixture model with parsimonious structure, to account for both attribute and label noise. Our conjecture is that the contaminated observations would be the least plausible units under the robustly estimated model: the corrupted subsample will be revealed by detecting those

---

[*]Department of Statistics and Quantitative Methods, University of Milano-Bicocca, `a.cappozzo@campus.unimib.it, francesca.greselin@unimib.it`

[†]School of Mathematics & Statistics and Insight Research Centre, University College Dublin, `brendan.murphy@ucd.ie`

observations with the lowest contributions to the associated likelihood. Impartial trimming (Gordaliza, 1991a,b; Cuesta-Albertos et al., 1997) is employed for robustifying the parameter estimates, being a well established technique to treat mild and gross outliers in the clustering literature (García-Escudero et al., 2010) and here used, for the first time, to additionally account for label noise in a classification framework. A semi-supervised approach is developed, where information contained in both labelled and unlabelled samples is combined for improving the classifier performance and for defining a data-driven method to identify outlying observations possibly present in the test set.

The rest of the manuscript is organized as follows. A brief review on model-based discriminant analysis and classification is given in Section 2, Section 3 introduces the robust updating classification rules, covering the model formulation, inference aspects and model selection. Simulation studies to compare the method introduced in Section 3 with other popular classification methods are reported in Section 4. Finally, in Section 5 our proposal is employed in performing classification and adulteration detection in a food authenticity context, dealing with contaminated samples of Irish honey. Concluding notes and further research directions are outlined in Section 6. The proof of Proposition 1, details on the parameters values for simulation study II and efficient algorithms for enforcing the eigen-ratio constraint for different patterned models are deferred respectively to appendices A, B and C.

# 2 Model-Based Discriminant Analysis and Classification

In this Section we review the main concepts of supervised classification based on mixture models, with particular focus on Eigenvalue Decomposition Discriminant Analysis and its semi-supervised formulation, as introduced in Dean et al. (2006). This approach is the basis of the novel robust semi-supervised classifier introduced in Section 3.

## 2.1 Eigenvalue Decomposition Discriminant Analysis

Model-based discriminant analysis (McLachlan, 1992; Fraley and Raftery, 2002) is a probabilistic approach for supervised classification, in which a classifier is built from a complete set of learning observations $\{(\mathbf{x}_1, \mathbf{l}_1), \ldots, (\mathbf{x}_N, \mathbf{l}_N)\}$; where $\mathbf{x}_n$ and $\mathbf{l}_n$, $n = 1, \ldots, N$, are independent realizations of random vectors $\mathcal{X} \in \mathbb{R}^p$ and $\mathcal{G} \in \{1, \ldots, G\}$, respectively. That is, $\mathbf{x}_n$ denotes a $p$-variate observation and $\mathbf{l}_n$ its associated class label, such that $l_{ng} = 1$ if observation $n$ belongs to group $g$ and 0 otherwise, $g = 1, \ldots, G$. Considering a Gaussian framework, the probabilistic mechanism that is assumed to have generated the data is as follows:

$$\begin{aligned} \mathcal{G} &\sim Mult_G(1; \tau_1, \ldots, \tau_G) \\ \mathcal{X}|\mathcal{G} = g &\sim \mathcal{N}_p(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) \end{aligned} \tag{1}$$

where $\mathcal{G}$ is multinomially distributed with $\tau_g$ probability of observing class $g$ and the conditional density of $\mathcal{X}$ given $\mathcal{G}$ is multivariate normal with mean vector $\boldsymbol{\mu}_g$ and variance covariance matrix $\boldsymbol{\Sigma}_g$. Therefore, the joint density of $(\mathbf{x}_n, \mathbf{l}_n)$ is given by:

$$f(\mathbf{x}_n, \mathbf{l}_n; \boldsymbol{\Theta}) = \prod_{g=1}^{G} [\tau_g \phi(\mathbf{x}_n; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)]^{l_{ng}} \tag{2}$$

where $\phi(\cdot; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ denotes the multivariate normal density and $\boldsymbol{\Theta}$ represents the collection of parameters to be estimated, $\boldsymbol{\Theta} = \{\tau_1, \ldots, \tau_G, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G\}$. Discriminant analysis makes use of data with known labels to estimate model parameters for creating a classification rule. The trained classifier is subsequently employed for assigning a set of unlabelled observations $\mathbf{y}_m$, $m = 1, \ldots, M$ to the class $g$ with the associated highest posterior probability:

$$z_{mg} = \mathbb{P}(\mathcal{G} = g | \mathcal{X} = \mathbf{y}_m) = \frac{\tau_g \phi(\mathbf{y}_m; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)}{\sum_{j=1}^{G} \tau_j \phi(\mathbf{y}_m; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \tag{3}$$

Table 1: Nomenclature, covariance structure and number of free parameters in $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G$: $\gamma$ denotes the number of parameters related to the orthogonal rotation and $\delta$ the number of parameters related to the eigenvalues. The last column indicates whether the eigenvalue-ratio (ER) constraint is required.

| Model | $\boldsymbol{\Sigma}_g$ | $\gamma$ | $\delta$ | ER |
|-------|------|------|------|------|
| EII | $\lambda\boldsymbol{I}$ | - | $1$ | Not required |
| VII | $\lambda_g\boldsymbol{I}$ | - | $G$ | Required |
| EEI | $\lambda\boldsymbol{A}$ | - | $p$ | Not required |
| VEI | $\lambda_g\boldsymbol{A}$ | - | $G+p-1$ | Required |
| EVI | $\lambda\boldsymbol{A}_g$ | - | $Gp-(G-1)$ | Required |
| VVI | $\lambda_g\boldsymbol{A}_g$ | - | $Gp$ | Required |
| EEE | $\lambda\boldsymbol{D}\boldsymbol{A}\boldsymbol{D}'$ | $p(p-1)/2$ | $p$ | Not required |
| VEE | $\lambda_g\boldsymbol{D}\boldsymbol{A}\boldsymbol{D}'$ | $p(p-1)/2$ | $G+p-1$ | Required |
| EVE | $\lambda\boldsymbol{D}\boldsymbol{A}_g\boldsymbol{D}'$ | $p(p-1)/2$ | $Gp-(G-1)$ | Required |
| EEV | $\lambda\boldsymbol{D}_g\boldsymbol{A}\boldsymbol{D}'_g$ | $Gp(p-1)/2$ | $p$ | Not required |
| VVE | $\lambda_g\boldsymbol{D}\boldsymbol{A}_g\boldsymbol{D}'$ | $p(p-1)/2$ | $Gp$ | Required |
| VEV | $\lambda_g\boldsymbol{D}_g\boldsymbol{A}\boldsymbol{D}'_g$ | $Gp(p-1)/2$ | $G+p-1$ | Required |
| EVV | $\lambda\boldsymbol{D}_g\boldsymbol{A}_g\boldsymbol{D}'_g$ | $Gp(p-1)/2$ | $Gp-(G-1)$ | Required |
| VVV | $\lambda_g\boldsymbol{D}_g\boldsymbol{A}_g\boldsymbol{D}'_g$ | $Gp(p-1)/2$ | $Gp$ | Required |

using the maximum a posteriori (MAP) rule. The afore-described framework is widely employed in classification tasks, thanks to its probabilistic formulation and well-established efficacy.

The number of parameters in the component variance covariance matrices grows quadratically with the dimension $p$. Thus, Bensmail and Celeux (1996) introduced a parsimonious parametrization proposing to enforce additional assumptions on the matrices structure, based on the eigen-decomposition of Banfield and Raftery (1993) and Celeux and Govaert (1995):

$$\boldsymbol{\Sigma}_g = \lambda_g\boldsymbol{D}_g\boldsymbol{A}_g\boldsymbol{D}'_g \tag{4}$$

where $\boldsymbol{D}_g$ is an orthogonal matrix of eigenvectors, $\boldsymbol{A}_g$ is a diagonal matrix such that $|\boldsymbol{A}_g| = 1$ and $\lambda_g = |\boldsymbol{\Sigma}_g|^{1/p}$. This elements correspond respectively to the orientation, shape and volume (alternatively called scale) of the different Gaussian components. Allowing each parameter in (4) to be equal or different across groups, Bensmail and Celeux (1996) define a family of 14 patterned models, listed in Table 1. Such class of models is particularly flexible, as it includes very popular classification methods like Linear Discriminant Analysis and Quadratic Discriminant Analysis as special cases for the EEE and VVV models, respectively (Hastie and Tibshirani, 1996). Eigenvalue Decomposition Discriminant Analysis (EDDA) is implemented in the `mclust R` package (Fop et al., 2016).

## 2.2 Updating Classification Rules

Exploiting the assumption that the data generating process outlined in (1) is the same for both labelled and unlabelled observations, Dean et al. (2006) propose to include also the data whose memberships are unknown in the parameter estimation. That is, information about group structure that may be contained in both labelled and unlabelled samples is combined in order to improve the classifier performance, in a semi-supervised manner.

Under the framework defined in Section 2.1, and given the set of available information

$\{(\mathbf{x}_n, \mathbf{l}_n)|n = 1, \ldots, N\} \cup \{\mathbf{y}_m|m = 1, \ldots, M\}$, the *observed log-likelihood* is

$$\ell(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}|\mathbf{X}, \mathbf{Y}, \mathbf{l}) = \sum_{n=1}^{N} \sum_{g=1}^{G} l_{ng} \log\left[\tau_g \phi(\mathbf{x}_n; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right]+$$
$$+ \sum_{m=1}^{M} \log\left[\sum_{g=1}^{G} \tau_g \phi(\mathbf{y}_m; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right] \tag{5}$$

in which both labelled and unlabelled samples are accounted for in the likelihood definition. Treating the (unknown) labels $z_{mg}$, $m = 1, \ldots, M$, $g = 1, \ldots, G$ as missing data and including them in the likelihood specification defines the so called *complete-data log-likelihood:*

$$\ell_C(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}|\mathbf{X}, \mathbf{Y}, \mathbf{l}, \mathbf{z}) = \sum_{n=1}^{N} \sum_{g=1}^{G} l_{ng} \log\left[\tau_g \phi(\mathbf{x}_n; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right]+$$
$$+ \sum_{m=1}^{M} \sum_{g=1}^{G} z_{mg} \log\left[\tau_g \phi(\mathbf{y}_m; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right] \tag{6}$$

Maximum likelihood estimates for (5) are obtained through the EM algorithm (Dempster et al., 1977), iteratively computing the expected value for the unknown labels given the current set of parameter estimates (E-Step), and employing (6) to find maximum likelihood estimates for the unknown parameters (M-Step). The unlabelled data are then classified according to $\hat{z}_{mg}$, using the MAP. The updating classification rules was demonstrated to give improved classification performance over the classical model-based discriminant analysis in some food authenticity applications, particularly when the training size is small. An implementation of this can be found in the `upclass R` package (Russell et al., 2014).

# 3 Robust Updating Classification Rules

We introduce here a Robust modification to the Updating Classification Rule described in Section 2.2, with the final aim of developing a classifier whose performance is not affected by contaminated data, either in the form of label noise and outlying observations.

## 3.1 Model Formulation

The main idea of the proposed approach is to employ techniques originated in the branch of robust statistics to obtain a model-based classifier in which parameters are robustly estimated and outlying observations identified. We are interested in providing a method that jointly accounts for noise on response and exploratory variables, where the former might be present in the labelled set and the latter in both the labelled and unlabelled sets. We propose to modify the log-likelihood in (5) with a *trimmed mixture log-likelihood* (Neykov et al., 2007) and to employ impartial trimming and constraints on the covariance matrices for achieving both robust parameter estimation and identification of the unreliable sub-sample. Impartial trimming is enforced by considering the distinct structure of the likelihoods associated to the labelled and unlabelled sets, accounting for the possible label noise that might be present in the labelled sample (see Section 3.2 for details). Following the same notation introduced in Section 2.1, we aim at maximizing the *trimmed observed data log-likelihood*:

$$\ell_{trim}(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}|\mathbf{X}, \mathbf{Y}, \mathbf{l}) = \sum_{n=1}^{N} \zeta(\mathbf{x}_n) \sum_{g=1}^{G} l_{ng} \log\left[\tau_g \phi(\mathbf{x}_n; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right]+$$
$$+ \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \log\left[\sum_{g=1}^{G} \tau_g \phi(\mathbf{y}_m; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\right] \tag{7}$$

where $\zeta(\cdot)$, $\varphi(\cdot)$ are 0-1 trimming indicator functions, that express whether observation $\mathbf{x}_n$ and $\mathbf{y}_m$ are trimmed off or not. A fixed fraction $\alpha_l$ and $\alpha_u$ of observations, belonging to the labelled and unlabelled set respectively, is unassigned by setting $\sum_{n=1}^{N} \zeta(\mathbf{x}_n) = \lceil N(1 - \alpha_l) \rceil$ and $\sum_{m=1}^{M} \varphi(\mathbf{y}_m) = \lceil M(1 - \alpha_u) \rceil$. In this way, the less plausible samples under the currently

4

estimated model are tentatively trimmed out at each step of the iterations that leads to the final estimate. The *labelled trimming level* $\alpha_l$ and the *unlabelled trimming level* $\alpha_u$ account for possible adulteration in both sets. At the end of the iterations, a value of $\zeta(\mathbf{x}_n) = 0$ or $\varphi(\mathbf{y}_m) = 0$ corresponds to identify $\mathbf{x}_n$ or $\mathbf{y}_m$, respectively, as unreliable observations. Notice that impartial trimming automatically deals with both class noise and attribute noise, as observations that suffer from either noise structure will give low contribution to the associated likelihood.

Maximization of (7) is carried out via the EM algorithm, in which an appropriate Concentration Step (Rousseeuw and Driessen, 1999) is performed in both labelled and unlabelled sets at each iteration to enforce the impartial trimming. In addition, we protect the parameter estimation from spurious solutions, that may arise whenever one component of the mixture fits a random pattern in the data. We consider the eigenvalue-ratio restriction:

$$\mathbf{M}_n/\mathbf{m}_n \leq c \tag{8}$$

where $\mathbf{M}_n = \max_{g=1\ldots G} \max_{l=1\ldots p} d_{lg}$ and $\mathbf{m}_n = \min_{g=1\ldots G} \min_{l=1\ldots p} d_{lg}$, with $d_{lg}$, $l = 1, \ldots, p$ being the eigenvalues of the matrix $\mathbf{\Sigma}_g$ and $c \geq 1$ being a fixed constant (Ingrassia, 2004). Constraint (8) simultaneously controls differences between groups and departures from sphericity, by forcing the relative length of the axes of the equidensity ellipsoids, based on the multivariate normal distribution, to be smaller than $\sqrt{c}$ (García-Escudero et al., 2014). Notice that the constraint in (8) is still needed whenever either the shape or the volume is free to vary across components (García-Escudero et al., 2017), that is for all models in Table 1 that present "Required" entry in the ER column. The considered approach is the (semi)-supervised version of the methodology proposed in Dotto and Farcomeni (2019), which is framed in a completely unsupervised scenario. Feasible and computationally efficient algorithms for enforcing the eigen-ratio constraint for different patterned models are reported in the Appendix C.

## 3.2 Estimation Procedure

The EM algorithm for obtaining Maximum Trimmed Likelihood Estimates of the robust updating classification rule involves the following steps:

- *Robust Initialization:* set $k = 0$. Employing only the labelled data, we obtain robust starting values for the mean vector $\boldsymbol{\mu}_g$ and covariance matrix $\mathbf{\Sigma}_g$ of the multivariate normal density for each group $g$, $g = 1, \ldots, G$, employing the following procedure:

  1. For each class $g$, draw a random $(p+1)$-subset $J_g$ and compute its empirical mean $\hat{\boldsymbol{\mu}}_g^{(0)}$ and variance covariance matrix $\hat{\mathbf{\Sigma}}_g^{(0)}$ according to the considered parsimonious structure. This procedure yields better initial subsets than drawing random $\lceil N(1-\alpha_l) \rceil$-subsets directly, because the probability of drawing an outlier-free $(p+1)$-subset is much higher than that of drawing an outlier-free $\lceil N(1-\alpha_l) \rceil$-subset (Hubert et al., 2018).

  2. Set
     $$\hat{\boldsymbol{\theta}} = \{\hat{\tau}_1, \ldots, \hat{\tau}_G, \hat{\boldsymbol{\mu}}_1, \ldots, \hat{\boldsymbol{\mu}}_G, \hat{\mathbf{\Sigma}}_1, \ldots, \hat{\mathbf{\Sigma}}_G\} =$$
     $$= \{\hat{\tau}_1^{(0)}, \ldots, \hat{\tau}_G^{(0)}, \hat{\boldsymbol{\mu}}_1^{(0)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(0)}, \hat{\mathbf{\Sigma}}_1^{(0)}, \ldots, \hat{\mathbf{\Sigma}}_G^{(0)}\}$$
     where $\hat{\tau}_1^{(0)} = \ldots = \hat{\tau}_G^{(0)} = 1/G$.

  3. For each $\mathbf{x}_n$, $n = 1, \ldots, N$, compute the conditional density
     $$f(\mathbf{x}_n|l_{ng} = 1; \hat{\boldsymbol{\theta}}) = \phi\left(\mathbf{x}_n; \hat{\boldsymbol{\mu}}_g, \hat{\mathbf{\Sigma}}_g\right) \quad g = 1, \ldots, G. \tag{9}$$

     $\lfloor N\alpha_l \rfloor \%$ of the samples with lowest value of (9) are temporarily discarded as possible outliers, namely label noise and/or attribute noise. That is, $\zeta(\mathbf{x}_n) = 0$ for such observations.

  4. The parameter estimates are updated, based on the non-discarded observations:
     $$\hat{\tau}_g = \frac{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng}}{\lceil N(1 - \alpha_l) \rceil} \quad g = 1, \ldots, G \tag{10}$$

     $$\hat{\boldsymbol{\mu}}_g = \frac{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} \mathbf{x}_n}{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng}} \quad g = 1, \ldots, G. \tag{11}$$

Estimation of $\mathbf{\Sigma}_g$ depends on the considered patterned model, details are given in Bensmail and Celeux (1996).

5. Iterate $3-4$ until the $\lfloor N\alpha_l \rfloor$ discarded observations are exactly the same on two consecutive iterations, then stop (usually, $\leq 3$ iterations are required).

The procedure described in steps $1-5$ is performed $\texttt{nsamp}$ times, and the parameter estimates $\hat{\boldsymbol{\theta}}^R$ that lead to the highest value of the objective function $\ell_{trim}(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}|\mathbf{X}, l) = \sum_{n=1}^{N} \zeta(\mathbf{x}_n) \sum_{g=1}^{G} l_{ng} \log \left[ \hat{\tau}_g \phi(\mathbf{x}_n; \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Sigma}}_g) \right]$, out of $\texttt{nsamp}$ repetitions, are retained. The afore-described procedure stems from the ideas of the FastMCD algorithm of Rousseeuw and Driessen (1999), here adapted for dealing with parsimonious structures in the covariance matrices. Retaining $\hat{\boldsymbol{\theta}}^R$ as final estimates leads to a fully supervised robust model-based method, called REDDA hereafter (see Section 4.1.1). Then, if the selected patterned model allows for heteroscedastic $\mathbf{\Sigma}_g$ and (8) is not satisfied, constrained maximization is enforced, see Appendix C for details.

- *EM Iterations:* denote by $\hat{\boldsymbol{\theta}}^{(k)} = \{\hat{\tau}_1^{(k)}, \ldots, \hat{\tau}_G^{(k)}, \hat{\boldsymbol{\mu}}_1^{(k)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(k)}, \hat{\boldsymbol{\Sigma}}_1^{(k)}, \ldots, \hat{\boldsymbol{\Sigma}}_G^{(k)}\}$ the parameter estimates at the $k$-th iteration of the algorithm.

  - *Step 1 - Concentration:* the trimming procedure is implemented by discarding the $\lfloor N\alpha_l \rfloor$ observations $\mathbf{x}_n$ with smaller values of

$$D\left(\mathbf{x}_n; \hat{\boldsymbol{\theta}}^{(k)}\right) = \prod_{g=1}^{G} \left[ \phi\left(\mathbf{x}_n; \hat{\boldsymbol{\mu}}_g^{(k)}, \hat{\boldsymbol{\Sigma}}_g^{(k)}\right) \right]^{l_{ng}} \quad n = 1, \ldots, N \tag{12}$$

  and discarding the $\lfloor M\alpha_u \rfloor$ observations $\mathbf{y}_m$ with smaller values of

$$D\left(\mathbf{y}_m; \hat{\boldsymbol{\theta}}^{(k)}\right) = \sum_{g=1}^{G} \hat{\tau}_g^{(k)} \phi\left(\mathbf{y}_m; \hat{\boldsymbol{\mu}}_g^{(k)}, \hat{\boldsymbol{\Sigma}}_g^{(k)}\right) \quad m = 1, \ldots, M. \tag{13}$$

  - *Step 2 - Expectation:* for each non-trimmed observation $\mathbf{y}_m$ compute the posterior probabilities

$$\hat{z}_{mg}^{(k+1)} = \frac{\hat{\tau}_g^{(k)} \phi\left(\mathbf{y}_m; \hat{\boldsymbol{\mu}}_g^{(k)}, \hat{\boldsymbol{\Sigma}}_g^{(k)}\right)}{D\left(\mathbf{y}_m; \hat{\boldsymbol{\theta}}^{(k)}\right)} \quad g = 1, \ldots, G; \quad m = 1, \ldots, M. \tag{14}$$

  - *Step 3 - Constrained Maximization:* the parameter estimates are updated, based on the non-discarded observations and the current estimates for the unknown labels:

$$\hat{\tau}_g^{(k+1)} = \frac{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg}^{(k+1)}}{\lceil N(1-\alpha_l) \rceil + \lceil M(1-\alpha_u) \rceil} \quad g = 1, \ldots, G \tag{15}$$

$$\hat{\boldsymbol{\mu}}_g^{(k+1)} = \frac{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} \mathbf{x}_n + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg}^{(k+1)} \mathbf{y}_m}{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg}^{(k+1)}} \quad g = 1, \ldots, G. \tag{16}$$

  Estimation of $\mathbf{\Sigma}_g$ depends on the considered patterned model and on the eigenvalues-ratio constraint. Details are given in Bensmail and Celeux (1996) and, if (8) is not satisfied, in Appendix C.

  - *Step 4 - Convergence of the EM algorithm:* check for algorithm convergence (see Section 3.3). If convergence has not been reached, set $k = k + 1$ and repeat steps 1-4.

Notice how the trimming step differs between the labelled and unlabelled observations. We implicitly assume that a label in the training set conveys a sound meaning about the presence of a class of objects. Therefore, in the labelled set, we opted for trimming the samples with lowest conditional density $f(\mathbf{x}_n|l_{ng} = 1; \hat{\boldsymbol{\theta}}^{(k)}) = \phi\left(\mathbf{x}_n; \hat{\boldsymbol{\mu}}_g^{(k)}, \hat{\boldsymbol{\Sigma}}_g^{(k)}\right)$. The alternative choice of considering the joint density $f(\mathbf{x}_n, l_{ng}; \hat{\boldsymbol{\theta}}^{(k)}) = \prod_{g=1}^{G} [\tau_g \phi(\mathbf{x}_n; \boldsymbol{\mu}_g, \mathbf{\Sigma}_g)]^{l_{ng}}$ is instead prone to trim off completely groups with small prior probability $\tau_g$ for large enough value of $\alpha_l$, and should be discarded. Note that with (12) we are both discriminating label noise (i.e., observations that are likely to belong to the mixture model but whose associated label is wrong) and outliers. In the unlabelled set, on the other hand, trimming is based on the

marginal density $f(\mathbf{y}_m; \hat{\boldsymbol{\theta}}^{(k)}) = \sum_{g=1}^{G} \hat{\tau}_g^{(k)} \phi\left(\mathbf{y}_m; \hat{\boldsymbol{\mu}}_g^{(k)}, \hat{\boldsymbol{\Sigma}}_g^{(k)}\right)$, having no prior information on the group membership of the samples.

Once convergence is reached, the estimated values $\hat{z}_{mg}$ provide a classification for the unlabelled observations $\mathbf{y}_m$, assigning observation $m$ into group $g$ if $\hat{z}_{mg} > \hat{z}_{mg'}$ for all $g' \neq g$. Final values of $\zeta(\mathbf{x}_n) = 0$, and $\varphi(\mathbf{y}_m) = 0$, classify $\mathbf{x}_n$ and $\mathbf{y}_m$ respectively, as outlying observations.

The routines for estimating the robust updating classification rules have been written in R language (R Core Team, 2018): the source code is available at https://github.com/AndreaCappozzo/rupclass. The estimation procedure detailed in this Section implies the monotonicity of the algorithm, according to:

**Proposition 1:** If the values $\zeta(\mathbf{x}_n)$, $\varphi(\mathbf{y}_m)$, $n = 1, \ldots, N$, $m = 1, \ldots, M$ are kept fixed, the EM algorithm described in Section 3.2 implies $\ell_{trim}(\hat{\boldsymbol{\theta}}^{(k+1)}|\mathbf{X}, \mathbf{Y}, \mathbf{l}) \geq \ell_{trim}(\hat{\boldsymbol{\theta}}^{(k)}|\mathbf{X}, \mathbf{Y}, \mathbf{l})$ at any $k$.

The proof is reported in Appendix A. Furthermore, our estimation procedure reduces possible incorrect modes of the optimization function (spurious maximizers) and offers a constructive way to obtain a maximizer $\hat{\theta}_n$ for the sample problem, that converges to the global maximizer for the population, see García-Escudero et al. (2008) and García-Escudero et al. (2015).

## 3.3 Convergence Criterion

We assess whether the EM algorithm has reached convergence evaluating at each iteration how close the trimmed log-likelihood is to its estimated asymptotic value, using the Aitken acceleration (Aitken, 1926):

$$a^{(k)} = \frac{\ell_{trim}^{(k+1)} - \ell_{trim}^{(k)}}{\ell_{trim}^{(k)} - \ell_{trim}^{(k-1)}} \tag{17}$$

where $\ell_{trim}^{(k)}$ is the trimmed observed data log-likelihood from iteration $k$. The asymptotic estimate of the trimmed log-likelihood at iteration $k$ is given by (Bohning et al., 1994):

$$\ell_{\infty_{trim}}^{(k)} = \ell_{trim}^{(k)} + \frac{1}{1 - a^{(k)}} \left( \ell_{trim}^{(k+1)} - \ell_{trim}^{(k)} \right). \tag{18}$$

The EM algorithm is considered to have converged when $|\ell_{\infty_{trim}}^{(k)} - \ell_{trim}^{(k)}| < \varepsilon$; a value of $\varepsilon = 10^{-5}$ has been chosen for the experiments reported in the next Sessions.

## 3.4 Model Selection

A robust likelihood-based criterion is employed for choosing the best model among the 14 patterned covariance structures listed in Table 1 and a reasonable value for the constraint $c$ in (8):

$$RBIC = 2\ell_{trim}(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}) - v_{XXX}^c \log\left(\lceil N(1 - \alpha_l) \rceil + \lceil M(1 - \alpha_u) \rceil\right) \tag{19}$$

where $\ell_{trim}(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ denotes the maximized trimmed observed data log-likelihood and $v_{XXX}^c$ a penalty term whose definition is:

$$v_{XXX}^c = Gp + G - 1 + \gamma + (\delta - 1)\left(1 - \frac{1}{c}\right) + 1. \tag{20}$$

That is, $v_{XXX}^c$ depends on the total number of parameters to be estimated: $\gamma$ and $\delta$ for every $XXX$ patterned model are given in Table 1. It also accounts for the trimming levels and for the eigen-ratio constraint $c$, according to Cerioli et al. (2018). Note that, when $c \to +\infty$ and $\alpha_l = \alpha_u = 0$, (19) is the Bayesian Information Criterion (Schwarz, 1978).
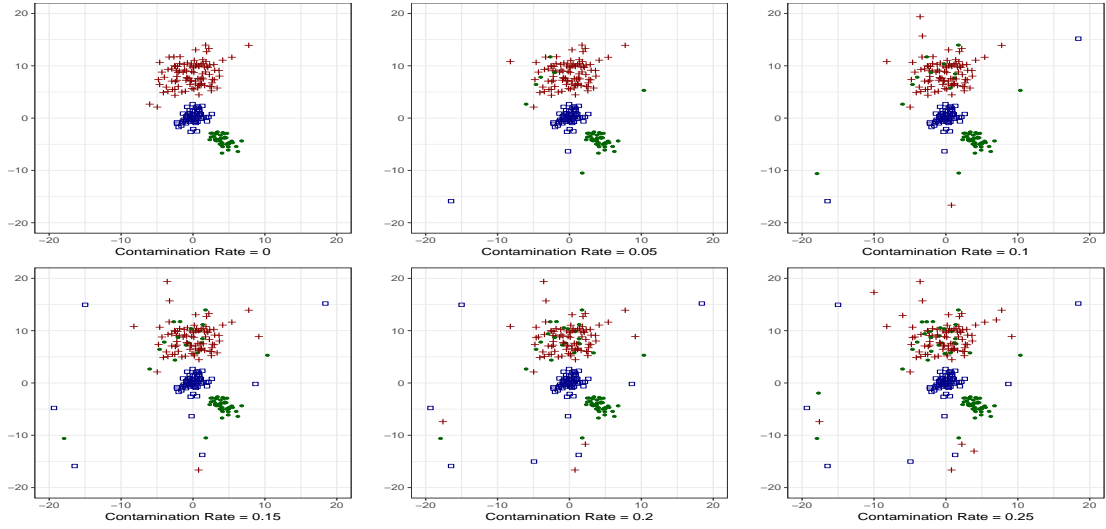
Figure 1: Simulated data considering the Simulation Setup described in Section 4.1.1, varying Contamination Rate $\eta$
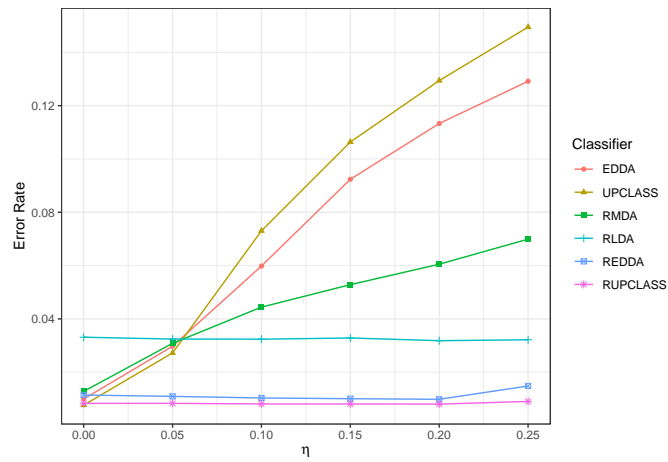


Figure 2: Average misclassification errors on $B = 1000$ runs for different classification methods, increasing contamination rate.

Table 2: Average misclassification errors on $B = 1000$ runs, varying method and contamination rate $\eta$. Standard errors are reported in parenthesis.

| $\eta$ | 0 | 0.05 | 0.10 | 0.15 | 0.20 | 0.25 |
|---|---|---|---|---|---|---|
| EDDA | 0.009 | 0.031 | 0.053 | 0.079 | 0.099 | 0.112 |
| | (0.005) | (0.026) | (0.043) | (0.051) | (0.054) | (0.05) |
| UPCLASS | 0.008 | 0.041 | 0.091 | 0.142 | 0.166 | 0.186 |
| | (0.004) | (0.056) | (0.088) | (0.088) | (0.08) | (0.067) |
| RMDA | 0.009 | 0.045 | 0.049 | 0.052 | 0.07 | 0.08 |
| | (0.005) | (0.072) | (0.063) | (0.057) | (0.068) | (0.073) |
| RLDA | 0.027 | 0.027 | 0.026 | 0.026 | 0.026 | 0.067 |
| | (0.009) | (0.009) | (0.009) | (0.008) | (0.009) | (0.037) |
| REDDA | 0.01 | 0.01 | 0.01 | 0.009 | 0.024 | 0.042 |
| | (0.005) | (0.005) | (0.005) | (0.005) | (0.014) | (0.014) |
| RUPCLASS | 0.01 | 0.009 | 0.009 | 0.008 | 0.019 | 0.044 |
| | (0.005) | (0.005) | (0.005) | (0.005) | (0.013) | (0.014) |

## 4 Simulation studies

In this Section, we present two simulated data experiments: Simulation Study I compares the performances of several model-based classification methods in a low dimensional setting when dealing with noisy data at different contamination rates; Simulation Study II considers a higher dimensional scenario in which the accuracy performance of some popular classification methods is assessed, at a fixed contamination rate. In both scenarios we consider a joint noise structure on response and exploratory variables.

### 4.1 Simulation Study I

#### 4.1.1 Experimental Setup

We consider a data generating process given by a mixture of $G = 3$ components of bivariate normal distributions, according to the following parameters:

$$\boldsymbol{\tau} = (0.3, 0.2, 0.5)', \quad \boldsymbol{\mu}_1 = (0, 0)', \quad \boldsymbol{\mu}_2 = (4, -4)', \quad \boldsymbol{\mu}_3 = (0, 8)'$$

$$\boldsymbol{\Sigma}_1 = \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix} \quad \boldsymbol{\Sigma}_2 = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix} \quad \boldsymbol{\Sigma}_3 = \begin{bmatrix} 6.71 & 2.09 \\ 2.09 & 6.71 \end{bmatrix}.$$

600 observations were generated from the model, randomly assigning $N = 200$ to the labelled set and $M = 400$ to the unlabelled set. The labelled set was subsequently adulterated with contamination rate $\eta$ (ranging from 0 to 0.25), wrongly assigning $\lceil \eta/2N \rceil$ of the third group units to the first class and adding $\lceil \eta/2N \rceil$ randomly labelled points generated from a Uniform distribution on the square with vertices $[(-20, -20), (-20, 20), (20, -20), (20, 20)]$. The contamination is therefore twofold, involving jointly label switching and outliers for a total of $\eta N$ adulterated labelled units. Examples of labelled datasets with different contamination rates are reported in Figure 1. Performances of 6 model-based classification methods are considered:

- EDDA: Eigenvalue Decomposition Discriminant Analysis (Bensmail and Celeux, 1996)

- UPCLASS: Updating Classification Rules (Dean et al., 2006)

- RMDA: Robust Mixture Discriminant Analysis (Bouveyron and Girard, 2009)

- RLDA: Robust Linear Discriminant Analysis (Hawkins and McLachlan, 1997)

- REDDA: Robust Eigenvalue Decomposition Discriminant Analysis. This is the supervised version of the model described in Section 3, where only the labelled observations are used for parameter estimation obtained via the robust initialization detailed in Section 3.2.
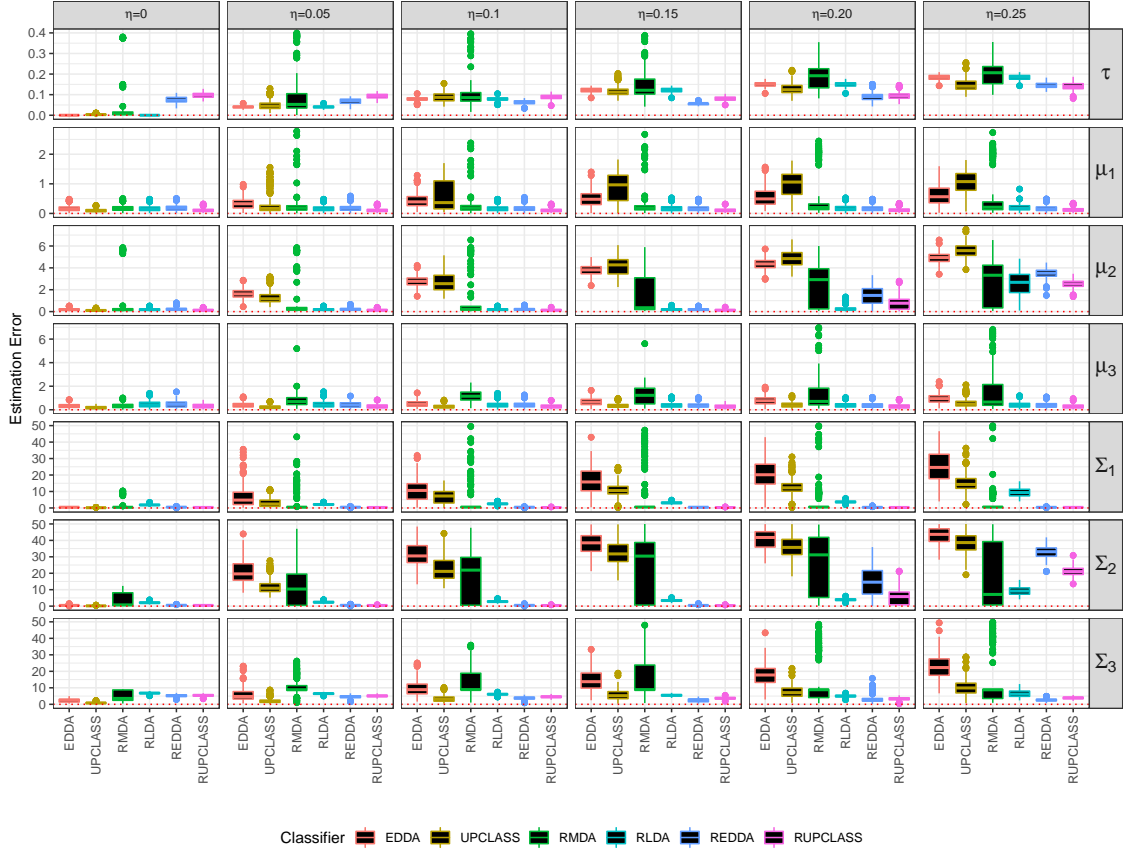
9

Figure 3: Box plots of the simulated estimation errors for the parameters of the mixture, computed via Euclidean norms for the proportion vector $\boldsymbol{\tau}$, the mean vectors $\boldsymbol{\mu}_g$ and covariance matrices $\boldsymbol{\Sigma}_g$, $g = 1, \ldots, 3$ for the different models, varying contamination rate $\eta$ from 0 to 0.25.

- RUPCLASS: Robust Updating Classification Rules. The semi-supervised method described in Section 3.

To make a fair performance comparison, a level of $\alpha_l = 0.15$ (REDDA and RUPCLASS) and $\alpha_u = 0.05$ (RUPCLASS) have been kept fixed throughout the simulation study. Nevertheless, exploratory tools such as Density-Based Silhouette plot (Menardi, 2011) and trimmed likelihood curves (García-Escudero et al., 2011) could be employed to validate and assess the choice of $\alpha_l$ and $\alpha_u$. A more automatic approach, like the one introduced in Dotto et al. (2018), could also be adapted to our framework. This, however, goes beyond the scope of the present manuscript, it will nonetheless be addressed in the future. A value of $c = 20$ was selected for the eigenvalue-ratio restriction in (8). Simulation study results are presented in the following subsections.

### 4.1.2 Classification Performance

Average misclassification errors for the different methods and for varying contamination rates are reported in Table 2 and in Figure 2. The error rate is computed on the unlabelled dataset and averaged over the $B = 1000$ simulations. As expected, the misclassification error is fairly equal to all methods when there is no contamination rate, with the only exception being RLDA: this is due to the implicit model assumption that $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_3$, which is not the case in our simulated scenario. As the contamination rate increases, so does the error rate for the non-robust methods (EDDA and UPCLASS), whereas for RLDA and RMDA it has a lower increment rate. Nevertheless, such methods fail to jointly cope with both sources

of adulteration, namely class and attribute noise. Our proposals REDDA and RUPCLASS, thanks to the trimming step enforced in the estimation process, have always higher correct classification rates, on average, at any adulteration level. Notice that, to compare results of robust and non-robust methods, also the trimmed observations were classified a-posteriori according to the Bayes rule, assigning them to the component $g$ having greater value of $\hat{\tau}_g \phi(\mathbf{y}_m; \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Sigma}}_g)$.

On average, the robust semi-supervised approach performs better than the supervised counterpart, due to the information incorporated from genuine unlabelled data in the estimation process. Interestingly, the same behavior is not reflected in the non-robust counterparts, where the detrimental effect of contaminated labelled units magnifies the bias of the UPCLASS method. Therefore, robust solutions are even more paramount when a semi-supervised approach is considered.

### 4.1.3 Parameter Estimation

Figure 3 reports the box plots of the simulated estimation error over $B = 1000$ Monte Carlo repetitions for the parameters of the mixture model, computing Euclidean norms for the proportion vector $\boldsymbol{\tau}$, the mean vectors $\boldsymbol{\mu}_g$ and covariance matrices $\boldsymbol{\Sigma}_g$, $g = 1, \ldots, 3$. The estimated values for the mixing proportion are mildly affected when increasing contamination is considered; conversely, the estimation of $\boldsymbol{\mu}_2$ is on average heavily influenced by the adulterating process, and also the robust methods fail to estimate it correctly as soon as the contamination rate $\eta$ is larger than the trimming level $\alpha_l = 0.15$. Clearly, the estimation of the variance covariance matrices is as well badly affected in most extreme scenarios, where their entries are inflated in order to accommodate more and more bad points. Our robust proposals are less affected by the harmful effect of adding anomalous observations, also in the most adulterated scenario.

## 4.2 Simulation Study II

### 4.2.1 Experimental Setup

We consider here a simulating model with a larger number of features ($d = 10$), where the data generating process is given by a mixture of $G = 4$ components of a multivariate t-distribution with $\nu = 6$ degrees of freedom. More details on the parameter values are contained in Appendix B. 1000 observations were generated from the model, randomly assigning $N = 250$ to the labelled set and $M = 750$ to the unlabelled set. The training set was subsequently adulterated wrongly labelling 10 units and adding 15 randomly labelled outlying points, uniformly generated in the $d$-dimensional hypercube over $[10, 15]^{10}$. We therefore consider a scenario in which 10% of the learning units are contaminated, via both label and attribute noise.

Together with the model-based methods previously described in Section 4.1.1, we included in the performance evaluation widely used classification techniques that, even though not engineered to achieve robustness, are noise tolerant. Particularly, the ensemble learner AdaBoost (Freund and Schapire, 1997) and the kernel method Support Vector Machine (Cortes and Vapnik, 1995) were added to the comparison. Furthermore, the robust adaptation of the SIMCA method for high-dimensional classification (Vanden Branden and Hubert, 2005) was also considered. The classification performance of the afore-described techniques are tested against the proposed methodologies, under different combinations of $(c, \alpha_l, \alpha_u)$: accuracy results are reported in the next Section.

### 4.2.2 Classification Performance

Boxplots of the misclassification errors for the considered methods are reported in Figure 4. The error rate is computed on the $M = 750$ units of the test set, under $B = 1000$ repetitions of the generating process and subsequent adulteration scheme described in Section 4.2.1. As it was already apparent from the previous simulation study, accuracy for non-robust methods is badly affected by the contamination present in the learning set. Even though not specifically designed for dealing with adulterated datasets, SVM and AdaBoost perform better than the non-robust model-based approaches, thanks to their non-parametric nature and flexibility. As expected, the best classification accuracy are obtained by the robust
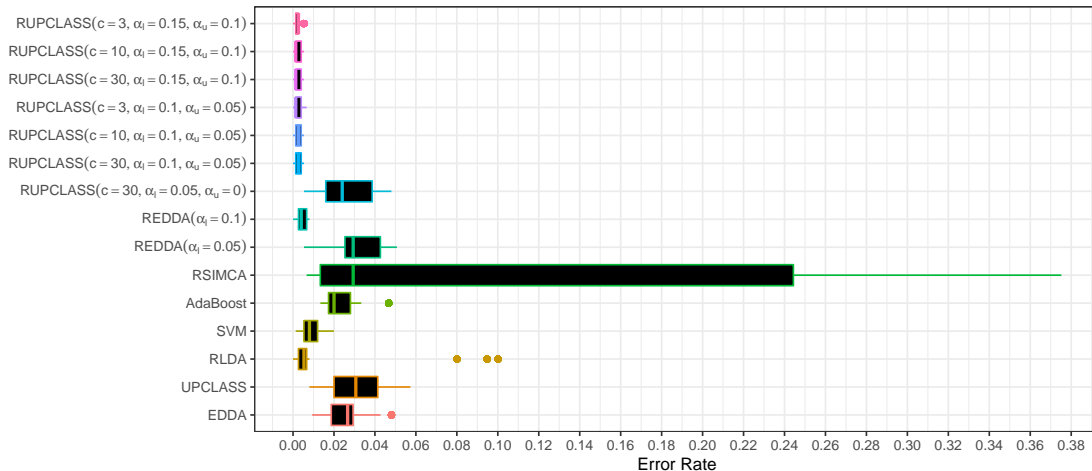
Figure 4: Box plots of the misclassification errors under $B = 1000$ repetitions of the simulating experiment II. Error rate is computed on the $M = 750$ data points of the test set for different classification methods.

methodologies, namely RLDA and our proposals REDDA and RUPCLASS. We also check the sensitivity of our techniques comparing different combinations of $(c, \alpha_l, \alpha_u)$. As it is easily visible in the boxplots, setting a smaller than needed labelled trimming level $\alpha_l$ leads to a loss in prediction accuracy, as a portion of adulterated units still affects the learning phase. Once the corrupted observations are correctly trimmed (i.e., $\alpha_l$ is set $\geq 0.1$), accuracy seems to remain stable with little influence induced by the choice of $c$ and $\alpha_u$, with only a slight preference for the semi-supervised RUPCLASS over its supervised version REDDA. This shows that setting a higher value of $\alpha_l$ is less detrimental than underestimating it, and that the impartial trimming almost exactly identifies the corrupted units when $\alpha_l = 0.1$, that is the true adulteration proportion. The bad performance of RSIMCA is only due to the simulating process: given the fact that data truly lie on a 10-dimensional space, performing (robust) dimensional reduction prior to classification evidently leads to a concealment in the grouping structure.

The proposed methodologies were shown to be capable of dealing with data whose distribution is not exactly Gaussian, but where an effective robust decision rule can be built employing Gaussian mixture models.

# 5 Application to Midinfrared Spectroscopy of Irish Honey

The semi-supervised method introduced in Section 3 is employed in performing adulteration detection and classification in a food authenticity context: we consider the task of discriminating between pure and adulterated Irish Honey, where the training set itself contains unreliable samples.

## 5.1 Honey Samples

Honey is defined as "the natural sweet substance, produced by honeybees from the nectar of plants or from secretions of living parts of plants, or excretions of plant-sucking insects on the living parts of plants, which the bees collect, transform by combining with specific substances of their own, deposit, dehydrate, store and leave in honeycombs to ripen and mature" (Alimentarius, 2001). Being a relatively expensive commodity to produce and extremely variable in nature, honey is prone to adulteration for economic gain: in 2015 the European Commission organized an EU coordinated control plan to assess the prevalence on the market of honey adulterated with sugars and honeys mislabelled with regard to their botanical source
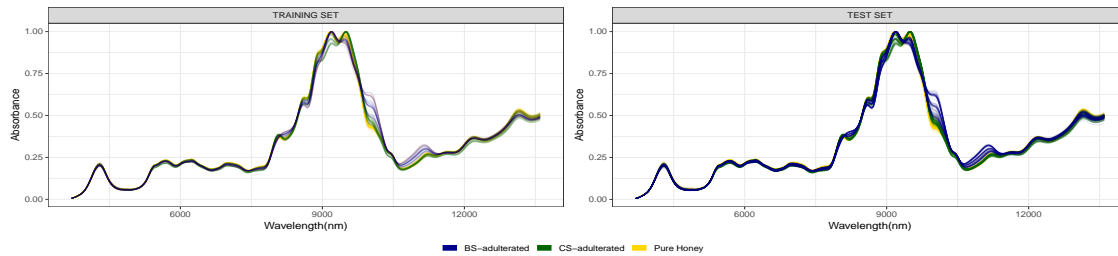
Figure 5: Midinfrared spectra for pure and contaminated honey, Irish Honey data.
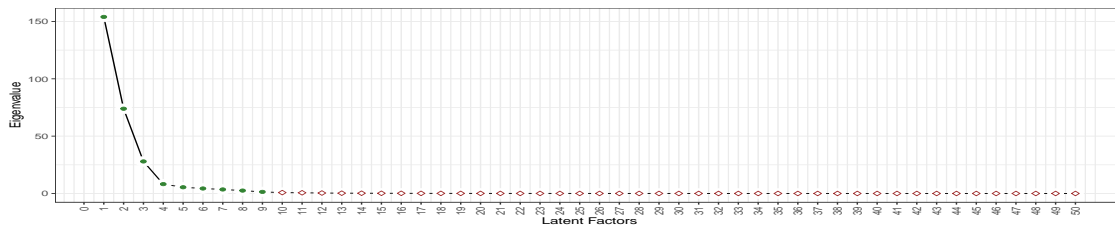


Figure 6: Cattell's scree plot (Cattell, 1966) for the first 50 eigenvalues of the robustly estimated correlation matrix, Irish Honey data. Green solid dots denote eigenvalues bigger than 1.

or geographical origin. It is therefore of prime interest to employ robust analytical methods to protect food quality and uncover its illegal adulteration.

We consider here a dataset of midinfrared spectroscopic measurements of 530 Irish honey samples. Midinfrared spectroscopy is a fast, non-invasive method for examining substances that does not require any sample preparation, it is therefore an effective procedure for collecting data to be subsequently used in food authenticity studies (Downey, 1996). The spectra measurements lie in the wavelength range of 3700 nm and 13600 nm, recorded at intervals of 35 nm, with a total of 285 absorbance values. The dataset contains 290 Pure Honey observations, while the rest of the samples are honey diluted with adulterant solutions: 120 with Dextrose Syrup and 120 with Beet Sucrose, respectively. Kelly et al. (2006) gives a thorough explanation of the adulteration process. The aim of the study is to discriminate pure honey from the adulterated samples, when varying sample size of the labelled set whilst including a percentage of wrongly labelled units. Such a scenario is plausible to be encountered in real situations, since in a context in which the final purpose is to detect potential adulterated samples it may happen that the learning data is itself not fully reliable. An example of the data structure is reported in Figure 5.

## 5.2 Robust Dimensional Reduction

Prior to perform classification and adulteration detection, a preprocessing step is needed due to the high-dimensional nature of the considered dataset ($p = 285$ variables). To do so, we robustly estimate a factor analysis model, retaining a set $d$ of factors, $d \ll p$, to be subsequently employed with the Robust Updating Classification Rules. Formally, for each Honey sample $\mathbf{x}_i$, we postulate a factor model of the form:

$$\mathbf{x}_i = \boldsymbol{\mu} + \boldsymbol{\Lambda} \boldsymbol{u}_i + \boldsymbol{e}_i \tag{21}$$

where $\boldsymbol{\mu}$ is a $p \times 1$ mean vector, $\boldsymbol{\Lambda}$ is a $p \times d$ matrix of factor loadings, $\boldsymbol{u}_i$ are the unobserved factors, assumed to be realizations of a $d$-variate standard normal and the errors $\boldsymbol{e}_i$ are independent realizations of $\mathcal{N}(\mathbf{0}, \boldsymbol{\Psi})$, with $\boldsymbol{\Psi}$ a $p \times p$ diagonal matrix. In such a way, the observed variables are assumed independent given the factors. For a general review on factor analysis, see for example Chapter 9 in (Mardia et al., 1979). Parameters in (21) are estimated employing a robust procedure based on trimming and constraints (García-Escudero et al., 2016), yielding dimensionality reduction at the same time. Given the robustly estimated

13

parameters, the latent traits are computed using the regression method (Thomson, 1939):

$$\hat{\boldsymbol{u}}_i = \hat{\boldsymbol{\Lambda}}' \left( \hat{\boldsymbol{\Lambda}} \hat{\boldsymbol{\Lambda}}' + \hat{\boldsymbol{\Psi}} \right)^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}) \tag{22}$$

The estimated factors scores $\hat{\boldsymbol{u}}_i$ will be used for the classification task reported in the upcoming Section. For the considered dataset, after a graphical exploration of Cattell's scree plot for the correlation matrix robustly estimated via MCD (Rousseeuw and Driessen, 1999), reported in Figure 6, we deem sufficient to set the number $d$ of latent factors equal to 10. Parameters were estimated setting a trimming level $\alpha = 0.1$ and $c_{noise} = 1000$.

## 5.3 Classification Performance

SVM, AdaBoost and RSIMCA are designed to optimally perform in a high dimensional setting. Therefore, to respect the specificity of each family of methodologies, we directly applied SVM, AdaBoost and RSIMCA on the whole spectra. For EDDA, UPCLASS, RMDA, RLDA, REDDA and RUPCLASS we preprocessed the data with the dimension reduction method described in Section 5.2. To discriminate between pure and adulterated honey samples, we divided the available data into a training (labelled) sample and a validation (unlabelled) sample. We investigated the effect of having different sample sizes in the labelled set, both in terms of classification accuracy and adulteration detection. Particularly, 3 proportions have been considered: 50% - 50% , 25% - 75% and 10% - 90% for splitting data into training and validation set, respectively, within each group. For each split, 10% of the Beet Sucrose adulterated samples were incorrectly labelled as Pure Honey in the training set, adding class noise in the discrimination task. The trimming levels $\alpha_l$ and $\alpha_u$ were set equal to 0.12 and 0.05, respectively. Table 3 and 4 summarize the accuracy results employing different classification approaches under the described scenarios. Careful investigation has been dedicated to measuring the ability of the robust methodologies in correctly determining (i.e., trimming) the 10% of incorrectly labelled samples, that is, units adulterated with Beet Sucrose and erroneously labelled as Pure Honey: such information, only relevant for RSIMCA, REDDA and RUPCLASS models, is reported in Table 4. *% Correctly Trimmed* indicates the class noise percentage correctly detected by the impartial trimming. For the recognized class noise, *% Correctly Assigned* indicates the percentage of units properly a-posteriori assigned to the Beet Sucrose group. RSIMCA performs remarkably well in identifying the adulterated units, even though the classification accuracy is lower than the one obtained employing RUPCLASS model. As expected, the semi-supervised approach performs much better in terms of classification rate when the labelled sample size is small. Comparing the error rate of the robust techniques with the other methods in Table 3 we notice how powerful classifiers like SVM and AdaBoost work well also in dealing with adulterated datasets: SVM error rate in the 50% Tr - 50% Te is on average lower than the one obtained with RUPCLASS. However, when the labelled sample size decreases a semi-supervised approach is preferable: RUPCLASS reports the lowest error rate for both 25% Tr - 75% Te and 10% Tr - 90% Te scenarios. VEV and VVV models have been almost always chosen: model selection was performed through the Robust criteria defined in Section 3.4.

Results in Table 4 show that the proposed methodology is effective not only for accurately robustifying the parameter estimates, but also for efficiently detecting observations affected by class noise, firstly by trimming and subsequently by correctly assigning them: a critical information that cannot be obtained with standard classification methods like SVM and AdaBoost.

# 6    Concluding Remarks

In this paper we have proposed a robust modification to a family of semi-supervised patterned models, for performing classification in presence of both class and attribute noise.

We have shown that our methodology effectively addresses the issues generated by these two noise types, by identifying wrongly labelled units (noise in the response variable) and corrupted attributes in units (noise in the explanatory variables). Robust parameter estimates can therefore be obtained by excluding the noisy observations from the estimation procedure, both in the training set, and in the test set. Our proposal has been based on incorporating

Table 3: Misclassification rates in the unlabelled set for different classification methods. Average values for 50 random splits in training and validation (three proportions are considered), standard deviations reported in parentheses.

| Error Rate | EDDA | UPCLASS | RMDA | RLDA | SVM | AdaBoost |
|---|---|---|---|---|---|---|
| 50% Tr - 50% Te | 0.033 | 0.065 | 0.291 | 0.1 | 0.025 | 0.036 |
| | (0.012) | (0.049) | (0.091) | (0.02) | (0.008) | (0.011) |
| 25% Tr - 75% Te | 0.078 | 0.112 | 0.303 | 0.12 | 0.048 | 0.042 |
| | (0.025) | (0.028) | (0.08) | (0.04) | (0.021) | (0.012) |
| 10% Tr - 90% Te | 0.24 | 0.126 | 0.375 | 0.157 | 0.109 | 0.058 |
| | (0.031) | (0.023) | (0.065) | (0.08) | (0.036) | (0.021) |

Table 4: Misclassification rates in the unlabelled set, % of wrongly labelled samples correctly trimmed in the labelled set and % of those correctly trimmed observations properly a-posteriori assigned to the Beet Sucrose group. Average values for 50 random splits in training and validation (three proportions are considered), standard deviations reported in parentheses.

| | | RSIMCA | REDDA | RUPCLASS |
|---|---|---|---|---|
| 50% Tr - 50% Te | Error Rate | 0.069 | 0.05 | 0.029 |
| | | (0.029) | (0.013) | (0.01) |
| | % Correctly Trimmed | 1 | 0.977 | 1 |
| | | (0) | (0.075) | (0) |
| | % Correctly Assigned | 1 | 1 | 1 |
| | | (0) | (0) | (0) |
| | | | | |
| 25% Tr - 75% Te | Error Rate | 0.075 | 0.053 | 0.032 |
| | | (0.038) | (0.034) | (0.009) |
| | % Correctly Trimmed | 1 | 0.88 | 0.96 |
| | | (0) | (0.25) | (0.145) |
| | % Correctly Assigned | 1 | 0.963 | 1 |
| | | (0) | (0.162) | (0) |
| | | | | |
| 10% Tr - 90% Te | Error Rate | 0.111 | 0.121 | 0.053 |
| | | (0.051) | (0.039) | (0.038) |
| | % Correctly Trimmed | 0.99 | 0.47 | 0.73 |
| | | (0.071) | (0.238) | (0.381) |
| | % Correctly Assigned | 0.99 | 0.72 | 0.84 |
| | | (0.019) | (0.071) | (0.37) |

impartial trimming and eigenvalue-ratio constraints in previous semi-supervised methods. We have adapted the trimming procedure to the two different frameworks, i.e., for the labelled units and the unlabelled ones. After completing the robust estimation process, trimmed observations can be classified as well, by the usual Bayes rule. This final step allows the researcher to detect whether one observation is indeed extreme in terms of its attributes or it has been wrongly assigned to a different class. Such feature seems particularly desirable in food authenticity applications, where, due to imprecise readings and fraudulent units, it is likely to have label noise also within the labelled set. Some simulations, and a study on real data from pure and adulterated Honey samples, have shown the effectiveness of our proposal.

As an open point for further research, an automatic procedure for selecting reasonable values for the labelled and unlabelled trimming levels, along the lines of Dotto et al. (2018), is under study. Additionally, a robust wrapper variable selection for dealing with high-dimensional problems could be useful for further enhancing the discriminating power of the proposed methodology.

## Acknowledgements

# Appendix A

**Proof of Proposition 1:** Considering the random variable $\mathcal{Z}_{mg}$ corresponding to $z_{mg}$, the E-step on the $(k+1)$th iteration requires the calculation of the conditional expectation of $\mathcal{Z}_{mg}$ given $\mathbf{y}_m$:

$$
\begin{aligned}
E_{\hat{\boldsymbol{\theta}}^{(k)}}(\mathcal{Z}_{mg}|\mathbf{y}_m) = \mathbb{P}\left(\mathcal{Z}_{mg}=1|\mathbf{y}_m;\hat{\theta}^{(k)}\right) = \\
= \frac{\mathbb{P}\left(\mathbf{y}_m|\mathcal{Z}_{mg}=1;\hat{\theta}^{(k)}\right)\mathbb{P}\left(\mathcal{Z}_{mg}=1;\hat{\theta}^{(k)}\right)}{\sum_{j=1}^{G}\mathbb{P}\left(\mathbf{y}_m|\mathcal{Z}_{mj}=1;\hat{\theta}^{(k)}\right)\mathbb{P}\left(\mathcal{Z}_{mj}=1;\hat{\theta}^{(k)}\right)} = \\
= \frac{\hat{\tau}_g^{(k)}\phi\left(\mathbf{y}_m;\hat{\boldsymbol{\mu}}_g^{(k)},\hat{\boldsymbol{\Sigma}}_g^{(k)}\right)}{\sum_{j=1}^{G}\hat{\tau}_j^{(k)}\phi\left(\mathbf{y}_m;\hat{\boldsymbol{\mu}}_j^{(k)},\hat{\boldsymbol{\Sigma}}_j^{(k)}\right)} = \\
= \hat{z}_{mg}^{(k+1)} \qquad g=1,\ldots,G; \quad m=1,\ldots,M.
\end{aligned}
\tag{23}
$$

Therefore, the Q function, to be maximized with respect to $\boldsymbol{\theta}$ in the M-step, is given by

$$
\begin{aligned}
Q(\boldsymbol{\theta};\hat{\boldsymbol{\theta}}^{(k)}) = \sum_{n=1}^{N}\zeta(\mathbf{x}_n)\sum_{g=1}^{G}l_{ng}\log\left[\tau_g\phi(\mathbf{x}_n;\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g)\right]+ \\
+ \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\sum_{g=1}^{G}\hat{z}_{mg}\log\left[\tau_g\phi(\mathbf{y}_m;\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g)\right].
\end{aligned}
\tag{24}
$$

The maximization of (24) according to the mixture proportion $\tau_g$, $\sum_{j=1}^{G}\tau_j=1$ is solved considering the Lagrangian $\mathcal{L}(\boldsymbol{\theta},\kappa)$:

$$
\mathcal{L}(\boldsymbol{\theta},\kappa) = Q(\boldsymbol{\theta};\hat{\boldsymbol{\theta}}^{(k)}) - \kappa\left(\sum_{j=1}^{G}\tau_j-1\right)
\tag{25}
$$

with $\kappa$ the Lagrangian coefficient. The partial derivative of (25) with respect to $\tau_g$ has the form:

$$
\frac{\partial}{\partial\tau_g}\mathcal{L}(\boldsymbol{\theta},\kappa) = \frac{\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng}}{\tau_g} + \frac{\sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg}}{\tau_g} - \kappa
\tag{26}
$$

and setting (26) equal to 0 for all $g=1,\ldots,G$ we obtain:

$$
\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng} + \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg} - \kappa\tau_g = 0.
\tag{27}
$$

Summing (27) over $g$, $g=1,\ldots,G$, provides the value of $\kappa = \lceil N(1-\alpha_l)\rceil + M(1-\alpha_u)\rceil$ and substituting it in the previous expression yields the ML estimate for $\tau_g$:

$$
\hat{\tau}_g^{(k+1)} = \frac{\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng} + \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg}^{(k+1)}}{\lceil N(1-\alpha_l)\rceil + \lceil M(1-\alpha_u)\rceil} \qquad g=1,\ldots,G.
\tag{28}
$$

The partial derivative of (24) with respect to the mean vector $\boldsymbol{\mu}_g$ reads:

$$
\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\mu}_g}Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)}) = -\boldsymbol{\Sigma}_g^{-1}\left[\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng}\left(\mathbf{x}_n-\boldsymbol{\mu}_g\right) + \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg}^{(k+1)}\left(\mathbf{y}_m-\boldsymbol{\mu}_g\right)\right] = \\
= -\boldsymbol{\Sigma}_g^{-1}\left[\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng}\mathbf{x}_n + \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg}^{(k+1)}\mathbf{y}_m+ \right. \\
\left. -\boldsymbol{\mu}_g\left(\sum_{n=1}^{N}\zeta(\mathbf{x}_n)l_{ng} + \sum_{m=1}^{M}\varphi(\mathbf{y}_m)\hat{z}_{mg}^{(k+1)}\right)\right].
\end{aligned}
\tag{29}
$$

Equating (29) to 0 and rearranging terms provides the ML estimate of $\boldsymbol{\mu}_g$:

$$\hat{\boldsymbol{\mu}}_g^{(k+1)} = \frac{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} \mathbf{x}_n + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg}^{(k+1)} \mathbf{y}_m}{\sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg}^{(k+1)}} \quad g = 1, \ldots, G. \tag{30}$$

Discarding quantities that do not depend on $\boldsymbol{\Sigma}_g$, we can rewrite (24) as follows:

$$\sum_{n=1}^{N} \sum_{g=1}^{G} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) \left[ -\log |\boldsymbol{\Sigma}_g|^{1/2} - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_g)' \boldsymbol{\Sigma}_g^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_g) \right] +$$

$$+ \sum_{m=1}^{M} \sum_{g=1}^{G} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \left[ -\log |\boldsymbol{\Sigma}_g|^{1/2} - \frac{1}{2} (\mathbf{y}_m - \boldsymbol{\mu}_g)' \boldsymbol{\Sigma}_g^{-1} (\mathbf{y}_m - \boldsymbol{\mu}_g) \right] =$$

$$= -\frac{1}{2} \left[ \sum_{n=1}^{N} \sum_{g=1}^{G} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) \log |\boldsymbol{\Sigma}_g| + \sum_{n=1}^{N} \sum_{g=1}^{G} \zeta(\mathbf{x}_n) l_{ng} \underbrace{\left[ (\mathbf{x}_n - \boldsymbol{\mu}_g)' \boldsymbol{\Sigma}_g^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_g) \right]}_{\text{a scalar}} + \right.$$

$$\left. + \sum_{m=1}^{M} \sum_{g=1}^{G} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \log |\boldsymbol{\Sigma}_g| + \sum_{m=1}^{M} \sum_{g=1}^{G} \varphi(\mathbf{y}_m) \hat{z}_{mg} \underbrace{\left[ (\mathbf{y}_m - \boldsymbol{\mu}_g)' \boldsymbol{\Sigma}_g^{-1} (\mathbf{y}_m - \boldsymbol{\mu}_g) \right]}_{\text{a scalar}} \right] =$$

$$= -\frac{1}{2} \left[ \sum_{g=1}^{G} \log |\boldsymbol{\Sigma}_g| \left( \sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \right) + \right.$$

$$+ \sum_{n=1}^{N} \sum_{g=1}^{G} \zeta(\mathbf{x}_n) l_{ng} tr \left[ \boldsymbol{\Sigma}_g^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_g) (\mathbf{x}_n - \boldsymbol{\mu}_g)' \right] +$$

$$\left. + \sum_{m=1}^{M} \sum_{g=1}^{G} \varphi(\mathbf{y}_m) \hat{z}_{mg} tr \left[ \boldsymbol{\Sigma}_g^{-1} (\mathbf{y}_m - \boldsymbol{\mu}_g) (\mathbf{y}_m - \boldsymbol{\mu}_g)' \right] \right] =$$

$$= -\frac{1}{2} \left[ \sum_{g=1}^{G} \log |\boldsymbol{\Sigma}_g| \left( \sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \right) + \right.$$

$$\left. + \sum_{g=1}^{G} tr \left[ \boldsymbol{\Sigma}_g^{-1} \boldsymbol{W}_g^X \right] + \sum_{g=1}^{G} tr \left[ \boldsymbol{\Sigma}_g^{-1} \boldsymbol{W}_g^Y \right] \right] =$$

$$= -\frac{1}{2} \left[ \sum_{g=1}^{G} \log |\boldsymbol{\Sigma}_g| \left( \sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \right) + \sum_{g=1}^{G} tr \left[ \boldsymbol{\Sigma}_g^{-1} \left( \boldsymbol{W}_g^X + \boldsymbol{W}_g^Y \right) \right] \right] \tag{31}$$

where $\boldsymbol{W}_g^X = \sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} \left[ (\mathbf{x}_n - \boldsymbol{\mu}_g) (\mathbf{x}_n - \boldsymbol{\mu}_g)' \right]$ and $\boldsymbol{W}_g^Y = \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg} \left[ (\mathbf{y}_m - \boldsymbol{\mu}_g) (\mathbf{y}_m - \boldsymbol{\mu}_g)' \right]$. Finally, considering the eigenvalue decomposition $\boldsymbol{\Sigma}_g = \lambda_g \boldsymbol{D}_g \boldsymbol{A}_g \boldsymbol{D}_g'$, (31) simplifies to:

$$-\frac{1}{2} \left[ \sum_{g=1}^{G} p \log \lambda_g \left( \sum_{n=1}^{N} \zeta(\mathbf{x}_n) l_{ng} (\mathbf{x}_n) + \sum_{m=1}^{M} \varphi(\mathbf{y}_m) \hat{z}_{mg} (\mathbf{y}_m) \right) + \right.$$

$$\left. + \sum_{g=1}^{G} \frac{1}{\lambda_g} tr \left[ \boldsymbol{D}_g \boldsymbol{A}^{-1} \boldsymbol{D}_g' \left( \boldsymbol{W}_g^X + \boldsymbol{W}_g^Y \right) \right] \right] \tag{32}$$

The partial derivative of (32) with respect to $(\lambda_g, \boldsymbol{A}_g, \boldsymbol{D}_g)$ depends on the considered patterned structure: for a thorough derivation the reader is referred to Bensmail and Celeux (1996). If (8) is not satisfied, the constraints are enforced as detailed in Appendix C. Lastly, notice that in performing the concentration step the optimal observations of both training and test sets are retained, i.e. the ones with the highest contribution to the objective function.

The afore-described procedure falls within the structure of a general EM algorithm, for which the likelihood function does not decrease after an EM iteration, as shown in Dempster et al. (1977) and reported in page 78 of McLachlan and Krishnan (2008).
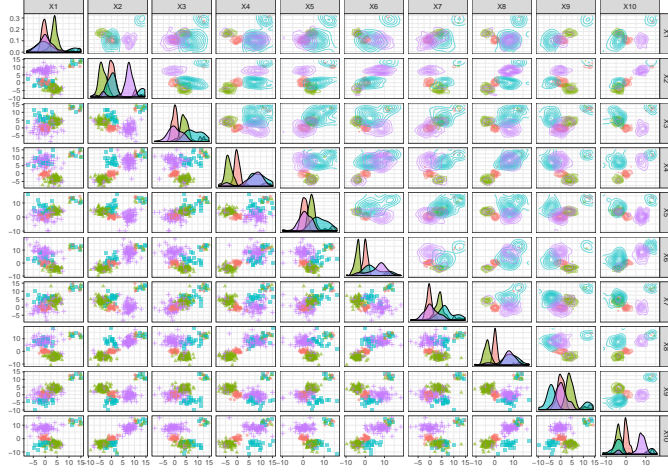
$\square$

Figure 7: Generalized pairs plot of the simulated data under the Simulation Setup described in 4.2.1. Both label noise and outliers are present in the data units.

# Appendix B

This appendix details the structure of the Simulation Study in Section 4.2.1. We consider a data generating process given by a mixture of $G = 4$ components of multivariate t-distributions (McLachlan and Peel, 1998; Peel and McLachlan, 2000), according to the following parameters:

$$\boldsymbol{\tau} = (0.2, 0.4, 0.1, 0.3)', \quad \nu = 6,$$

$$\boldsymbol{\mu}_1 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)',$$

$$\boldsymbol{\mu}_2 = (4, -4, 4, -4, 4, -4, 4, -4, 4, -4)',$$

$$\boldsymbol{\mu}_3 = (0, 0, 7, 7, 7, 3, 6, 8, -4, -4)',$$

$$\boldsymbol{\mu}_4 = (8, 0, 8, 0, 8, 0, 8, 0, 8, 0, 8)',$$

$$\boldsymbol{\Sigma}_1 = diag(1, 1, 1, 1, 1, 1, 1, 1, 1, 1),$$

$$\boldsymbol{\Sigma}_2 = diag(2, 2, 2, 2, 2, 2, 2, 2, 2, 2),$$

$$\boldsymbol{\Sigma}_3 = \boldsymbol{\Sigma}_4 = \begin{bmatrix} 5.05 & 1.26 & -0.35 & -0.00 & -1.04 & -1.35 & 0.29 & 0.07 & 0.69 & 1.17 \\ 1.26 & 2.57 & 0.17 & 0.00 & 0.27 & 0.11 & 0.61 & 0.11 & 0.59 & 0.89 \\ -0.35 & 0.17 & 6.74 & -0.00 & -0.26 & -0.31 & -0.01 & 0.00 & 0.08 & 0.14 \\ -0.00 & 0.00 & -0.00 & 5.47 & -0.00 & -0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ -1.04 & 0.27 & -0.26 & -0.00 & 6.80 & -0.76 & -0.12 & -0.01 & 0.09 & 0.21 \\ -1.35 & 0.11 & -0.31 & -0.00 & -0.76 & 7.75 & -0.26 & -0.04 & -0.03 & 0.03 \\ 0.29 & 0.61 & -0.01 & 0.00 & -0.12 & -0.26 & 4.76 & 0.06 & 0.38 & 0.60 \\ 0.07 & 0.11 & 0.00 & 0.00 & -0.01 & -0.04 & 0.06 & 4.18 & 0.07 & 0.11 \\ 0.69 & 0.59 & 0.08 & 0.00 & 0.09 & -0.03 & 0.38 & 0.07 & 3.23 & 0.60 \\ 1.17 & 0.89 & 0.14 & 0.00 & 0.21 & 0.03 & 0.60 & 0.11 & 0.60 & 3.24 \end{bmatrix}.$$

A generalized pairs plot of contaminated labelled units under the afore-described Simulation Setup is reported in Figure 7.

# Appendix C

This final Section presents feasible and computationally efficient algorithms for enforcing the eigenvalue-ratio constraint according to the different patterned models in Table 1. At the $k-$th iteration of the M step, the goal is to update the estimates for the variance-covariance matrices $\hat{\boldsymbol{\Sigma}}_g^{(k+1)} = \hat{\lambda}_g^{(k+1)} \hat{\boldsymbol{D}}_g^{(k+1)} \hat{\boldsymbol{A}}_g^{(k+1)} \hat{\boldsymbol{D}}_g^{'(k+1)}$, $g = 1, \dots, G$ such that,

$$\frac{\max_{g=1\dots G} \max_{l=1\dots p} \hat{\lambda}_g^{(k+1)} \hat{a}_{lg}^{(k+1)}}{\min_{g=1\dots G} \min_{l=1\dots p} \hat{\lambda}_g^{(k+1)} \hat{a}_{lg}^{(k+1)}} \leq c \tag{33}$$

where $\hat{a}_{lg}^{(k+1)}$ indicates the diagonal entries of matrix $\hat{\boldsymbol{A}}_g^{(k+1)}$. Denote with $\hat{\Sigma}_g^U = \hat{\lambda}_g^U \hat{\boldsymbol{D}}_g^U \hat{\boldsymbol{A}}_g^U \hat{\boldsymbol{D}}_g'^U$ the estimates for the variance covariance matrices obtained following Bensmail and Celeux (1996) without enforcing the eigenvalues-ratio restriction in (33). Lastly, denote with $\hat{\boldsymbol{\Delta}}_g^U = \hat{\lambda}_g^U \hat{\boldsymbol{A}}_g^U$ the matrix of eigenvalues for $\hat{\boldsymbol{\Sigma}}_g^U$, with diagonal entries $\hat{d}_{lg}^U = \hat{\lambda}_g^U \hat{a}_{lg}^U$, $l = 1, \ldots, p$.

## Constrained maximization for VII, VVI and VVV models

1. Compute $\boldsymbol{\Delta}_g$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\left\{ \hat{\boldsymbol{\Delta}}_1^U, \ldots, \hat{\boldsymbol{\Delta}}_G^U \right\}$, under condition (33)

2. Set $\hat{\lambda}_g^{(k+1)} = |\boldsymbol{\Delta}_g|^{1/p}$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \frac{1}{\hat{\lambda}_g^{(k+1)}} \boldsymbol{\Delta}_g$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \hat{\boldsymbol{D}}_g^U$

## Constrained maximization for VVE model

1. Compute $\boldsymbol{\Delta}_g$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\left\{ \hat{\boldsymbol{\Delta}}_1^U, \ldots, \hat{\boldsymbol{\Delta}}_G^U \right\}$, under condition (33)

2. Given $\boldsymbol{\Delta}_g$, compute the common principal components $\boldsymbol{D}$ via, for example, a majorization-minimization (MM) algorithm (Browne and McNicholas, 2014)

3. Set $\hat{\lambda}_g^{(k+1)} = |\boldsymbol{\Delta}_g|^{1/p}$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \frac{1}{\hat{\lambda}_g^{(k+1)}} \boldsymbol{\Delta}_g$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \boldsymbol{D}$

## Constrained maximization for EVI, EVV models

1. Compute $\boldsymbol{\Delta}_g$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\left\{ \hat{\boldsymbol{\Delta}}_1^U, \ldots, \hat{\boldsymbol{\Delta}}_G^U \right\}$, under condition (33)

2. Compute $\boldsymbol{\Delta}_g^\star$ constraining $\boldsymbol{\Delta}_g$ such that $\boldsymbol{\Delta}_g^\star = \lambda^\star \boldsymbol{A}_g^\star$. That is, constraining $|\boldsymbol{\Delta}_g^\star|$ to be equal across groups (Maronna and Jacovkis, 1974; Gallegos, 2002). Details are given in Section 3.2 of Fritz et al. (2012)

3. Iterate $1 - 2$ until (33) is satisfied

4. Set $\hat{\lambda}_g^{(k+1)} = \lambda^\star$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \boldsymbol{A}_g^\star$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \hat{\boldsymbol{D}}_g^U$

## Constrained maximization for EVE model

1. Compute $\boldsymbol{\Delta}_g$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\left\{ \hat{\boldsymbol{\Delta}}_1^U, \ldots, \hat{\boldsymbol{\Delta}}_G^U \right\}$, under condition (33)

2. Compute $\boldsymbol{\Delta}_g^\star$ constraining $\boldsymbol{\Delta}_g$ such that $\boldsymbol{\Delta}_g^\star = \lambda^\star \boldsymbol{A}_g^\star$. Details are given in Section 3.2 of Fritz et al. (2012)

3. Iterate $1 - 2$ until (33) is satisfied

4. Given $\boldsymbol{A}_g^\star$, compute the common principal components $\boldsymbol{D}$ via, for example, a majorization-minimization (MM) algorithm (Browne and McNicholas, 2014)

5. Set $\hat{\lambda}_g^{(k+1)} = \lambda^\star$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \boldsymbol{A}_g^\star$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \boldsymbol{D}$

## Constrained maximization for VEI, VEV models

1. Set $\boldsymbol{\Delta}_g = \hat{\boldsymbol{\Delta}}_g^U$

2. Set $\lambda_g^\star = \hat{\lambda}_g^U$, $g = 1, \ldots, G$

3. Compute $\boldsymbol{\Delta}_g^\star$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\{\boldsymbol{\Delta}_1, \ldots, \boldsymbol{\Delta}_G\}$, under condition (33)

4. Compute $\boldsymbol{A}^\star = \sum_{g=1}^G \frac{1}{\lambda_g^\star} \boldsymbol{\Delta}_g^\star \Big/ \left| \sum_{g=1}^G \frac{1}{\lambda_g^\star} \boldsymbol{\Delta}_g^\star \right|^{1/p}$

5. Compute $\lambda_g^\star = \frac{1}{p} tr \left( \boldsymbol{\Delta}_g^\star \boldsymbol{A}^{\star-1} \right)$

6. Set $\boldsymbol{\Delta}_g = \lambda_g^\star \boldsymbol{A}^\star$

7. Iterate $3 - 6$ until (33) is satisfied

8. Set $\hat{\lambda}_g^{(k+1)} = \lambda_g^\star$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \boldsymbol{A}^\star$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \hat{\boldsymbol{D}}_g^U$

## Constrained maximization for VEE model

1. Set $\boldsymbol{K}_g = \hat{\boldsymbol{\Sigma}}_g^U$

2. Set $\lambda_g^\star = \hat{\lambda}_g^U$, $g = 1, \ldots, G$

3. Compute $\boldsymbol{K}_g^\star$ applying the *optimal truncation operator* defined in Fritz et al. (2013) to $\{\boldsymbol{K}_1, \ldots, \boldsymbol{K}_G\}$, under condition (33)

4. Compute $\boldsymbol{C}^\star = \sum_{g=1}^G \frac{1}{\lambda_g^\star} \boldsymbol{K}_g^\star \Big/ \left| \sum_{g=1}^G \frac{1}{\lambda_g^\star} \boldsymbol{K}_g^\star \right|^{1/p}$

5. Compute $\lambda_g^\star = \frac{1}{p} tr \left( \boldsymbol{K}_g^\star \boldsymbol{C}^{\star -1} \right)$

6. Set $\boldsymbol{K}_g = \lambda_g^\star \boldsymbol{C}^\star$

7. Iterate $3 - 6$ until (33) is satisfied

8. Considering the spectral decomposition for $\boldsymbol{C}^\star = \boldsymbol{D}^\star \boldsymbol{A}^\star \boldsymbol{D}^{\star'}$, set $\hat{\lambda}_g^{(k+1)} = \lambda_g^\star$, $\hat{\boldsymbol{A}}_g^{(k+1)} = \boldsymbol{A}^\star$, $\hat{\boldsymbol{D}}_g^{(k+1)} = \boldsymbol{D}^\star$

# References

Aitken AC (1926) A series formula for the roots of algebraic and transcendental equations. Proceedings of the Royal Society of Edinburgh 45(01):14–22

Alimentarius C (2001) Revised codex standard for honey. Codex stan 12:1982

Banfield JD, Raftery AE (1993) Model-based Gaussian and non-Gaussian clustering. Biometrics 49(3):803

Bensmail H, Celeux G (1996) Regularized Gaussian discriminant analysis through eigenvalue decomposition. Journal of the American Statistical Association 91(436):1743–1748

Bohning D, Dietz E, Schaub R, Schlattmann P, Lindsay BG (1994) The distribution of the likelihood ratio for mixtures of densities from the one-parameter exponential family. Ann Inst Statist Math 46(2):373–388

Bouveyron C, Girard S (2009) Robust supervised classification with mixture models: Learning from data with uncertain labels. Pattern Recognition 42(11):2649–2658

Browne RP, McNicholas PD (2014) Estimating common principal components in high dimensions. Adv Data Anal Classif 8:217–226

Cattell RB (1966) The scree test for the number of factors. Multivariate Behavioral Research 1(2):245–276

Celeux G, Govaert G (1995) Gaussian parsimonious clustering models. Pattern Recognition 28(5):781–793

Cerioli A, García-Escudero LA, Mayo-Iscar A, Riani M (2018) Finding the number of normal groups in model-based clustering via constrained likelihoods. Journal of Computational and Graphical Statistics 27(2):404–416

Cortes C, Vapnik V (1995) Support-vector networks. Machine Learning 20(3):273–297

Cuesta-Albertos JA, Gordaliza A, Matrán C (1997) Trimmed k-means: An attempt to robustify quantizers. Annals of Statistics 25(2):553–576

Dean N, Murphy TB, Downey G (2006) Using unlabelled data to update classification rules with applications in food authenticity studies. Journal of the Royal Statistical Society Series C: Applied Statistics 55(1):1–14

Dempster A, N Laird, Rubin D (1977) Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39(1):1–38

Dotto F, Farcomeni A (2019) Robust inference for parsimonious model-based clustering. Journal of Statistical Computation and Simulation 89(3):414–442

Dotto F, Farcomeni A, García-Escudero LA, Mayo-Iscar A (2018) A reweighting approach to robust clustering. Statistics and Computing 28(2):477–493

Downey G (1996) Authentication of food and food ingredients by near infrared spectroscopy. Journal of Near Infrared Spectroscopy 4(1):47

Fop M, Murphy TB, Raftery AE (2016) mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. The R Journal XX(August):1–29

Fraley C, Raftery AE (2002) Model-based clustering, discriminant analysis, and density estimation. Journal of the American Statistical Association 97(458):611–631

Freund Y, Schapire RE (1997) A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. Journal of Computer and System Sciences 55(1):119–139

Fritz H, García-Escudero LA, Mayo-Iscar A (2012) tclust : An R Package for a Trimming Approach to Cluster Analysis. Journal of Statistical Software 47(12):1–26

Fritz H, García-Escudero LA, Mayo-Iscar A (2013) A fast algorithm for robust constrained clustering. Computational Statistics and Data Analysis 61:124–136

Gallegos MT (2002) Maximum likelihood clustering with outliers. In: Classification, Clustering, and Data Analysis, Springer, pp 247–255

García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2008) A general trimming approach to robust cluster Analysis. The Annals of Statistics 36(3):1324–1345

García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2010) A review of robust clustering methods. Advances in Data Analysis and Classification 4(2-3):89–109

García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2011) Exploring the number of groups in robust model-based clustering. Statistics and Computing 21(4):585–599

García-Escudero LA, Gordaliza A, Mayo-Iscar A (2014) A constrained robust proposal for mixture modeling avoiding spurious solutions. Advances in Data Analysis and Classification 8(1):27–43

García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2015) Avoiding spurious local maximizers in mixture modeling. Statistics and Computing 25(3):619–633

García-Escudero LA, Gordaliza A, Greselin F, Ingrassia S, Mayo-Iscar A (2016) The joint role of trimming and constraints in robust estimation for mixtures of Gaussian factor analyzers. Computational Statistics & Data Analysis 99:131–147

García-Escudero LA, Gordaliza A, Greselin F, Ingrassia S, Mayo-Iscar A (2017) Eigenvalues and constraints in mixture modeling: geometric and computational issues. Advances in Data Analysis and Classification pp 1–31

Gordaliza A (1991a) Best approximations to random variables based on trimming procedures. Journal of Approximation Theory 64(2):162–180

Gordaliza A (1991b) On the breakdown point of multivariate location estimators based on trimming procedures. Statistics & Probability Letters 11(5):387–394

Hastie T, Tibshirani R (1996) Discriminant analysis by Gaussian mixtures. Journal of the Royal Statistical Society Series B (Methodological) 58(1):155–176

Hawkins DM, McLachlan GJ (1997) High-breakdown linear discriminant analysis. Journal of the American Statistical Association 92(437):136

Hickey RJ (1996) Noise modelling and evaluating learning from examples. Artificial Intelligence 82(1-2):157–179

Hubert M, Debruyne M, Rousseeuw PJ (2018) Minimum covariance determinant and extensions. Wiley Interdisciplinary Reviews: Computational Statistics 10(3):1–11

Ingrassia S (2004) A likelihood-based constrained algorithm for multivariate normal mixture models. Statistical Methods and Applications 13(2):151–166

Kelly JD, Petisco C, Downey G (2006) Application of Fourier transform midinfrared spectroscopy to the discrimination between Irish artisanal honey and such honey adulterated with various sugar syrups. Journal of Agricultural and Food Chemistry 54(17):6166–6171

Mardia KV, Kent JT, Bibby JM (1979) Multivariate analysis. Academic Press London; New York

Maronna R, Jacovkis PM (1974) Multivariate clustering procedures with variable metrics. Biometrics 30(3):499

McLachlan GJ (1992) Discriminant analysis and statistical pattern recognition, Wiley Series in Probability and Statistics, vol 544. John Wiley & Sons, Inc., Hoboken, NJ, USA

McLachlan GJ, Krishnan T (2008) The EM Algorithm and Extensions, Wiley Series in Probability and Statistics, vol 54. John Wiley & Sons, Inc., Hoboken, NJ, USA

McLachlan GJ, Peel D (1998) Robust cluster analysis via mixtures of multivariate t-distributions. pp 658–666

McNicholas PD (2016) Mixture Model-Based Classification. Chapman and Hall/CRC

Menardi G (2011) Density-based Silhouette diagnostics for clustering methods. Statistics and Computing 21(3):295–308

Neykov N, Filzmoser P, Dimova R, Neytchev P (2007) Robust fitting of mixtures using the trimmed likelihood estimator. Computational Statistics & Data Analysis 52(1):299–308

Peel D, McLachlan GJ (2000) Robust mixture modelling using the t distribution. Statistics and Computing 10(4):339–348

Prati RC, Luengo J, Herrera F (2018) Emerging topics and challenges of learning from noisy data in nonstandard classification: a survey beyond binary class noise. Knowledge and Information Systems pp 1–35

R Core Team (2018) R: A Language and Environment for Statistical Computing

Rousseeuw PJ, Driessen KV (1999) A fast algorithm for the minimum covariance determinant estimator. Technometrics 41(3):212–223

Russell N, Cribbin L, Murphy TB (2014) upclass: An R Package for updating model-based classification rules. Cran R-Project Org

Schwarz G (1978) Estimating the dimension of a model. The Annals of Statistics 6(2):461–464

Thomson G (1939) The factorial analysis of human ability. British Journal of Educational Psychology 9(2):188–195

Vanden Branden K, Hubert M (2005) Robust classification in high dimensions based on the SIMCA Method. Chemometrics and Intelligent Laboratory Systems 79(1-2):10–21

Wu X (1995) Knowledge acquisition from databases. Intellect books, Westport, CT, USA

Zhu X, Wu X (2004) Class noise vs. attribute noise: A quantitative study. Artificial Intelligence Review 22(3):177–210