A fingerprint of a heterogeneous data set

Advances in Data Analysis and Classification
https://doi.org/10.1007/s11634-021-00452-9

REGULAR ARTICLE

A *fingerprint* of a heterogeneous data set

Matteo Spallanzani[1] · **Gueorgui Mihaylov**[2,3] · **Marco Prato**[4] · **Roberto Fontana**[5]

Received: 6 July 2020 / Revised: 1 June 2021 / Accepted: 8 June 2021

Abstract
In this paper, we describe the *fingerprint* method, a technique to classify bags of mixed-type measurements. The method was designed to solve a real-world industrial problem: classifying industrial plants (individuals at a higher level of organisation) starting from the measurements collected from their production lines (individuals at a lower level of organisation). In this specific application, the categorical information attached to the numerical measurements induced simple mixture-like structures on the global multivariate distributions associated with different classes. The *fingerprint* method is designed to compare the mixture components of a given test bag with the corresponding mixture components associated with the different classes, identifying the most similar generating distribution. When compared to other classification algorithms applied to several synthetic data sets and the original industrial data set, the proposed classifier showed remarkable improvements in performance.

Keywords Bagged data · Mixed-type data · Mixture distributions · Multivariate statistics · Machine learning

Mathematics Subject Classification 62P30 · 62R07 · 62H12 · 62H30

✉ Matteo Spallanzani
spmatteo@ethz.ch

1 Departement Informationstechnologie und Elektrotechnik, ETH Zürich, Gloriastrasse 35, 8092 Zürich, Switzerland

2 GlaxoSmithKline, Brentford, Middlesex TW8 9GS, UK

3 Department of Mathematics, King's College London, Strand, London WC2R 2LS, UK

4 Dipartimento di Scienze Fisiche, Informatiche e Matematiche, Università di Modena e Reggio Emilia, Via Giuseppe Campi 213/B, 41125 Modena, Italy

5 Dipartimento di Scienze Matematiche, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

Published online: 03 July 2021

1 Introduction

Modern data sets describe complex phenomena. It is not uncommon that the records contained in these data sets jointly measure categorical and numerical information. It is also not uncommon that entities in the physical world might be described by multiple records; in this case, it might happen that the samples that describe different entities greatly differ in size.

For instance, mixed-type data can provide a natural representation for the habits of website users, where categorical information can track choices and numerical data can track the time spent on specific pages or the time elapsed in between specific actions. Another use case where mixed-type data can provide richer descriptions is industrial monitoring. In this context, one can think of a piece of equipment that can be configured according to different setups (described by the combinations of the levels of one or more categorical variables) and that produces numerical data about chosen performance metrics.

Sometimes, choosing the single measurement as the relevant point of view of the phenomenon might not be the most informative option. For example, molecules often have multiple isomers; given a specific molecule, observing just one of its possible configurations might not suffice to tell whether the molecule can be used in the development of a target chemical application. Going back to the industrial monitoring example, observing the performance of a single production line in an industrial plant might not be representative of its overall performance. In these cases, the overall distribution of the data might provide a more informative picture.

Given the plethora of potential applications, recent years have witnessed an increase of interest in models and algorithms that can handle mixed-type and *bagged* data (Ahmad and Dey 2007; Hae-Sang and Chi-Hyuck 2009; Abdullin and Nasraoui 2012; Sandhya and PV. 2015).

Standard approaches to process mixed-type data are based on a preprocessing step that maps the original data in a metric space where all their components are of the same type, so that standard statistical or machine learning methods can be applied on top of a homogeneous-type space. For example, the so-called *one-hot encoding* can be used to map purely categorical data (where no natural order relationship can be defined) into numerical spaces; the main disadvantage of this transformation is that it might produce data distributed on extremely-low rank spaces, especially when the converted categorical variables can take on many values. Going in the other direction, *binning* techniques can be used to map numerical data into discrete values; this process destroys the geometry of the data, with the risk of losing potentially important information related to the native metric structure.

Naïve approaches to bag classification can be derived under the assumption of i.i.d. observations. In these cases, a classifier can be trained at the vector-level (also said at the *instance-level*), and its predictions about the elements of a given test bag can then be passed through a *max-win voting* procedure that assigns a bag to the class coinciding with the mode of the predictions.

In both the mixed-type and bagged data scenarios, more advanced classification methods are available. For example, the *mixture composer* (MixtComp) algorithm (Biernacki et al. 2015) can classify mixed-type data by identifying different generative

models for the observations of different classes; in each model, the distribution of each component depends on the class but is independent of the other components. In the original formulation, categorical components are modelled as multinomials, whereas numerical components can be modelled by Gaussian or Weibull distributions. The parameters of the model are then learnt using the classical *expectation-maximisation* algorithm, whereas inference is performed accordingly to the maximum-likelihood principle. The *multiple instance learning* (MIL) framework (Dietterich et al. 1997; Doran and Ray 2014) encompasses a family of techniques built on top of different existing machine learning methods to discriminate bags of samples. In the binary classification setting, a MIL algorithm is tasked with identifying a "critical region" of the domain. The bags sampled from distributions whose supports intersect such region are classified as positive, otherwise they are classified as negative.

These methods are not free of shortcomings. For example, MixtComp might sometimes be too strict in that it imposes the condition of mutual independence on the components of the observations, and therefore can not detect correlation patterns. This assumption is indeed quite strong. For instance, in Sect. 3 we will illustrate and discuss an industrial data set of mixed-type measurements where the values of the categorical components influence the distribution of the numerical components. On the other side of the spectrum, model-free approaches like MIL can sometimes struggle to discriminate patterns that would be much easier to detect by encoding some of the structure of the data into specific features.

In this work, we propose a novel classification algorithm for bagged, mixed-type data called the *fingerprint* method. The algorithm takes advantage of a simple principle: each class is interpreted as a family of probability distributions which can be represented as mixtures, and whose components follow class-dependent distributions. Whenever the algorithm is presented a test sample, it factorises the corresponding distribution according to the attached categorical information, and compares the resulting mixture components to the corresponding mixture components associated with the classes. Then, it assigns the sample to the class for which the corresponding mixture components are most similar. Although the *fingerprint* method follows a model-free approach to classification, it can exploit conditional dependence relationships between the components of mixed-type measurements.

Originally designed to solve a real-world industrial classification problem, the *fingerprint* method performed remarkably better at this task than many standard statistical and machine learning algorithms, and also than algorithms designed specifically to handle mixed-type and bagged data.

The paper is organised as follows:

- in Sect. 2 we introduce the notation and propose the formalisation for the problem of classifying mixed-type, bagged data on which the *fingerprint* method was conceived;
- in Sect. 3 we describe the original data sets that motivated the analysis; this section will clarify the derivation of the formalism and shed light on the specific design choices which characterise our formulation of the *fingerprint* method;

- in Sect. 4 we detail the *fingerprint* method; we also describe a hierarchical version of the method (the *multi-stage fingerprint* method) that is capable of classifying "problematic" bags;
- in Sect. 5 we report the experimental results obtained on a series of toy examples that illustrate some of the strength and weaknesses of the method when compared to a suite of chosen competitor methods; we conclude the paper by comparing the performance of the method to that of the selected competitor methods on the original industrial data set.

2 The problem

2.1 Classifying mixed-type data

Let Q be a finite set that represents the domain of a categorical variable $\mathbf{q}$, and define $N_Q := \#(Q)$. Here, $\#(\cdot)$ is the cardinality operator. Let $d > 0$ be an integer and $X \subseteq \mathbb{R}^d$ be a subset of the d-dimensional Euclidean space, representing the domain of a numerical variable $\mathbf{x}$. We define a **mixed-type measurement** to be a tuple $(\mathbf{q}, \mathbf{x})$, where the numerical information $\mathbf{x}$ is attached categorical information $\mathbf{q}$.

We will now develop a probabilistic framework to discuss the problem of classifying mixed-type data. We use $p(\mathbf{q})$ to denote probability mass functions (PMF) over Q. Also, we work under the assumption that all the "interesting" probability measures $\mu : \mathcal{A}_X \to [0, 1]$ over X (where $\mathcal{A}_X \subseteq \mathcal{P}(X)$ represents an arbitrary σ-algebra over X) are absolutely continuous, so that we can identify them with their probability densities $p(\mathbf{x})$. Complete knowledge of the mixed-type data in $Q \times X$ is available whenever we know the joint distribution

$$p(\mathbf{q}, \mathbf{x}) .$$

Let now Y be a finite set of classes, and define $N_Y := \#(Y)$. Classifying mixed-type measurements $(\mathbf{q}, \mathbf{x})$ into Y can be done applying the maximum likelihood principle:

$$y^* = \arg\max_{y \in Y} \; p(\mathbf{q}, \mathbf{x} \mid y) .$$

This is precisely what is accomplished by the MixtComp algorithm, though under the quite strong assumption of mutual independence between all the components of $\mathbf{q}$ and $\mathbf{x}$.

2.2 Mixed-type data bags

Given an integer $m > 0$, we define a mixed-type data **bag** as a sample

$$\mathcal{B} := \{(\mathbf{q}^{(1)}, \mathbf{x}^{(1)}), \ldots, (\mathbf{q}^{(m)}, \mathbf{x}^{(m)})\}$$

of size m from an unknown distribution $p(\mathbf{q}, \mathbf{x})$. The measurements $(\mathbf{q}^{(i)}, \mathbf{x}^{(i)})$ can appear more than once in each bag. For this reason, bags are also known in the literature as *multi-sets* (Blizard 1991), and are the suitable structure to represent samples from given probability distributions or the databases where such samples are collected.

However, the curly braces notation $\{\cdot\}$ is typically reserved for sets, where the elements must by definition be distinguishable. Hence, we unambiguously represent bags as functions

$$\mathcal{B} : Q \times X \to \mathbb{N}_0 \tag{1}$$

such that $\sum_{(\mathbf{q},\mathbf{x})\in Q\times X} \mathcal{B}(\mathbf{q}, \mathbf{x}) < +\infty$. In this case, the set $Q \times X$ is also called the *base set* of the bag. Intuitively, given a measurement $(\bar{\mathbf{q}}, \bar{\mathbf{x}}) \in Q \times X$, the value $\mathcal{B}(\bar{\mathbf{q}}, \bar{\mathbf{x}})$ represents the multiplicity of $(\bar{\mathbf{q}}, \bar{\mathbf{x}})$ inside the sample represented by bag $\mathcal{B}$.

In general, bags can also be empty (i.e., it might happen that $\mathcal{B}(\mathbf{q}, \mathbf{x}) = 0,\ \forall\, (\mathbf{q}, \mathbf{x}) \in Q \times X$). From a statistical perspective, empty samples are not useful. Hence, although the definitions that we will give in the following also hold for generic bags, the reader can safely consider only non-empty bags when multi-sets are used to represent samples.

In this formalism, a mixed-type data bag can be characterised by its generating distribution $p(\mathbf{q}, \mathbf{x})$ and by the sample size m. Consequently, we can describe a class of bags by defining a probability distribution on a space of probability distributions for mixed-type data:

$$p(p(\mathbf{q}, \mathbf{x}) \mid y)\,. \tag{2}$$

Although the idea of defining a probability distribution on a space of functions, measures or more abstract objects might seem exotic, the axioms of measure theory do not impose any constraint on the nature of the objects in the base set on which the structure of measurable space is defined. For example, thanks to the identification between the possible parametrisations of a discrete categorical distribution on a finite set of size N and the points of the $(N-1)$-dimensional simplex, the Dirichlet distribution describes a probability distribution over the collection of categorical distributions that can be defined on a finite set of size N.

In this framework, classifying the generating distribution $p_{\bar{k}}(\mathbf{q}, \mathbf{x})$ of a given mixed-type data bag (where $\bar{k} \in K$ is an identifier for the bag and K is a discrete, possibly infinite, set of identifiers) can be phrased as a maximum-likelihood problem:

$$y^* = \arg\max_{y\in Y}\ p(p_{\bar{k}}(\mathbf{q}, \mathbf{x}) \mid y)\,. \tag{3}$$

This formalisation is quite general. Taking a model-based approach towards solving this problem might involve several delicate decisions and possibly heavy computational tasks: defining a model for the mixed-type data distributions, selecting a procedure to estimate the best hypothesis from data (or to evolve a distribution over the space of hypothesis), and finally implementing a maximum-likelihood procedure to identify the most likely generating class. Therefore, we propose the following simplification.

By the law of total probability, we can associate to $p_{\bar{k}}(\mathbf{q}, \mathbf{x})$ the corresponding mixture distribution on X:

$$p_{\bar{k}}(\mathbf{x}) := \sum_{\mathbf{q} \in Q} p_{\bar{k}}(\mathbf{x} \mid \mathbf{q}) p_{\bar{k}}(\mathbf{q}) \,. \tag{4}$$

This distribution is completely known if we know the *mixture weights* $p_{\bar{k}}(\mathbf{q})$ and the *mixture components* $p_{\bar{k}}(\mathbf{x} \mid \mathbf{q})$.

In this way, we can simplify problem (3) by assuming that $p_{\bar{k}}(\mathbf{q}, \mathbf{x})$ depends on the class y through the mixture components. More formally, given an arbitrary $p_k(\mathbf{q}, \mathbf{x}) \sim p(p(\mathbf{q}, \mathbf{x}) \mid y)$ such that $p_k(\mathbf{x})$ is the associated mixture (4), then we have

$$p_k(\mathbf{x} \mid \mathbf{q}) = p(\mathbf{x} \mid \mathbf{q}, y) \,,$$

independently of $k \in K$. In other words, we postulate that the bags sampled from different classes are such that the impact of the generating class is detectable from the corresponding mixture components, without the need to analyse the mixture's weights.

This assumption implies that to correctly classify a mixed-type data bag $\mathcal{B}_{\bar{k}}$ whose items are sampled from $p_{\bar{k}}(\mathbf{q}, \mathbf{x})$ it might be sufficient to know the collections

$$\{p_{\mathbf{q},y}(\mathbf{x}) := p(\mathbf{x} \mid \mathbf{q}, y)\}_{\mathbf{q} \in Q} \tag{5}$$

for each class $y \in Y$, and the collection of mixture components

$$\{p_{\bar{k},\mathbf{q}}(\mathbf{x}) := p_{\bar{k}}(\mathbf{x} \mid \mathbf{q})\} \tag{6}$$

associated with the given bag. We call the collections (5) the **class fingerprint distributions**, whereas we call the collection (6) the **bag fingerprint distribution**.

2.3 Bags manipulation

Class *fingerprint* distributions and bag *fingerprint* distributions can not be known exactly, and one needs to estimate them from data. Since bags are the natural structure to represent statistical samples, it is useful to define operations to manipulate mixed-type samples and data sets of bags. To clarify the applicability range of the proposed formalisation, we will accompany the definitions with examples taken from the real-world industrial problem that motivated this research.

In the following, we will denote by k a variable taking values in a discrete set of identifiers K, and we will use $\bar{k}$ to identify a specific bag. In the same way, y will be a variable taking values in the finite set of classes Y, and we will use $\bar{y}$ to identify a specific class. Due to the similarity with the algebra of databases, we will introduce operations on bags using similar terminology.

Given a fixed level $\bar{\mathbf{q}} \in Q$ of the categorical variable, we define a **homogeneous sub-bag** to be any bag of mixed-type measurements

$$\mathcal{S} : Q \times X \to \mathbb{N}_0 \tag{7}$$

such that $\mathcal{S}(\mathbf{q}, \mathbf{x}) = 0$ whenever $\mathbf{q} \neq \bar{\mathbf{q}}$. In other words, homogeneous sub-bags represent samples where the categorical variable $\mathbf{q}$ takes a constant value $\bar{\mathbf{q}} \in Q$. For instance, if Q describes the possible configurations of a sensor, this structure can represent the samples acquired by a sensor whose configuration is fixed to $\bar{\mathbf{q}} \in Q$. Given a configuration $\bar{\mathbf{q}} \in Q$, the corresponding set of homogeneous sub-bags

$$Z_{\bar{\mathbf{q}}} := \left\{ \mathcal{S} \,:\, Q \times X \to \mathbb{N}_0 \,\middle|\, 0 < \sum_{(\mathbf{q},\mathbf{x}) \in Q \times X} \mathcal{S}(\mathbf{q}, \mathbf{x}) < +\infty \wedge \sum_{(\mathbf{q},\mathbf{x}) \in Q \times X \,|\, \mathbf{q} \neq \bar{\mathbf{q}}} \mathcal{S}(\mathbf{q}, \mathbf{x}) = 0 \right\}$$

represents all the possible (non-empty) samples that can be obtained from a sensor whose configuration is $\bar{\mathbf{q}}$.

Bags can be used to represent not only samples but also data sampling setups that commonly emerge in real-world scenarios. Continuing the sensor example, given a collection of $m > 0$ distinct sensors, we define a **bag composition** of size m to be a multi-set

$$\mathcal{C} \,:\, Q \to \mathbb{N}_0 \tag{8}$$

such that $\sum_{\mathbf{q} \in Q} \mathcal{C}(\mathbf{q}) = m$.

Given a bag composition (8), fix $\bar{\mathbf{q}} \in Q$ and consider $m_{\bar{\mathbf{q}}} := \mathcal{C}(\bar{\mathbf{q}})$. In our example, this number counts the sensors in the collection that share the same configuration $\bar{\mathbf{q}}$. We can describe the collection of samples acquired by these sensors with the multi-set

$$\mathcal{T}_{\mathcal{C},\bar{\mathbf{q}}} \,:\, Z_{\bar{\mathbf{q}}} \to \mathbb{N}_0$$

that satisfies the property $\sum_{\mathcal{S} \in Z_{\bar{\mathbf{q}}}} \mathcal{T}_{\mathcal{C},\bar{\mathbf{q}}}(\mathcal{S}) = m_{\bar{\mathbf{q}}}$. Intuitively, $\mathcal{T}_{\mathcal{C},\bar{\mathbf{q}}}$ counts the multiplicity of each sample in the collection of samples acquired by a (finite) number $m_{\bar{\mathbf{q}}}$ of distinct but identically configured sensors.

Releasing the constraint on $\mathbf{q}$, we define a **bag realisation** to be a collection of multi-sets

$$\left\{ \mathcal{T}_{\mathcal{C},\mathbf{q}} \,:\, Z_{\mathbf{q}} \to \mathbb{N}_0 \,\middle|\, \sum_{\mathcal{S} \in Z_{\mathbf{q}}} \mathcal{T}_{\mathcal{C},\mathbf{q}}(\mathcal{S}) = m_{\mathbf{q}} \right\}_{\mathbf{q} \in Q} .$$

Intuitively, this set describes the collection of samples acquired by m distinct sensors whose configurations are described by $\mathcal{C}$.

By combining these concepts, we can represent a generic data set of mixed-type data. Indeed, we can *compose* a data set bag $\mathcal{B}$ by first aggregating homogeneous sub-bags (e.g., the samples gathered from sensors that share the same configuration)

$$\mathcal{S}_{\mathbf{q}} := \sum_{\mathcal{S} \in Z_{\mathbf{q}}} \mathcal{T}_{\mathcal{C},\mathbf{q}}(\mathcal{S}) \mathcal{S} \,,\ \mathbf{q} \in Q \,, \tag{9}$$

and finally aggregating these homogeneous sub-bags into

$$\mathcal{B} = \sum_{\mathbf{q} \in Q} \mathcal{S}_{\mathbf{q}} \,. \tag{10}$$

Despite the fact that the set $Z_{\mathbf{q}}$ appearing in (9) might be infinite, note that the multi-set $\mathcal{T}_{\mathcal{C},\mathbf{q}}$ can assign non-zero multiplicity only to a finite number of samples $\mathcal{S}$ due to the definition of bag composition. Therefore, $\mathcal{S}_{\mathbf{q}}$ is still a finite sample.

The operations in (9) and (10) are instances of the **union** operation on bags. This operation takes in input one or more bags (which must be defined on the same base set), and returns as output a bag (again defined on the same base set).

Although the introduction of the concepts of bag composition and bag realisation might seem unnecessarily complicated, in Sect. 3 we will show how these ideas provide natural tools to represent the collection of samples coming from multiple production lines inside a single industrial plant, and how they enable a simple description of the process that led to the creation of the *fingerprint* method.

As a sort of inverse operation of union, we can define the **partition** operation. We will describe this operation in the specific instance of mixed-type bags, but the generalisation to more general bags should be straightforward. Consider a bag $\mathcal{B} : Q \times X \to \mathbb{N}_0$ defined on the base set $Q \times X$. Define a partition $P \subset \mathcal{P}(Q \times X)$ (i.e., a collection of non-empty subsets of $Q \times X$ such that $S_i \neq S_j \in P$ satisfy $S_i \cap S_j = \emptyset$ and $\cup_{S \in P} S = Q \times X$). The partition operation takes the bag $\mathcal{B}$ and a partition P of its base set, and returns a collection of bags

$$\{\mathcal{B}_S : Q \times X \to \mathbb{N}_0\}_{S \in P}\,, \tag{11}$$

where

$$\mathcal{B}_S(\mathbf{q}, \mathbf{x}) = \begin{cases} 0, & \text{if } (\mathbf{q}, \mathbf{x}) \notin S\,, \\ \mathcal{B}(\mathbf{q}, \mathbf{x}), & \text{if } (\mathbf{q}, \mathbf{x}) \in S\,. \end{cases}$$

An important application of the partition operation is the construction of homogeneous sub-bags starting from a given mixed-type data bag. In this case, the base set is a mixed-type set $Q \times X$, and one can partition the domain starting from the singletons of Q: $P := \{\{\mathbf{q}\} \times X \,|\, \mathbf{q} \in Q\}$.

The careful reader should have noticed that the partition can not always work as the inverse of the union: whereas uniting sub-bags which are homogeneous for different levels of the categorical variable $\mathbf{q}$ as in (10) can be reversed, uniting sub-bags which are homogeneous in the same level $\bar{\mathbf{q}}$ as in (9) destroys the information about the composing sub-bags. For this reason, it is important to define an operation that can label a bag with additional information.

Given a bag $\mathcal{B} : Q \times X \to \mathbb{N}_0$, a set K, and a value $\bar{k} \in K$ (which can represent an identifier for the entity from which the measurements in $\mathcal{B}$ are sampled), we can **join** the information about K to the bag by defining

$$\begin{aligned} \mathcal{B}_{\bar{k}} &: K \times Q \times X \to \mathbb{N}_0 \\ &(k, \mathbf{q}, \mathbf{x}) \mapsto \begin{cases} 0, & \text{if } k \neq \bar{k}\,, \\ \mathcal{B}(\mathbf{q}, \mathbf{x}), & \text{if } k = \bar{k}\,. \end{cases} \end{aligned} \tag{12}$$

In this example, we used the join operation to annotate a bag with identification information. Another example of the utility of the join operation is the classification of bags, where class information $\bar{y}$ is added to the bag.

Once we have access to a bag, we might want to analyse just part of the information it contains. For instance, considering again a mixed-type data bag $\mathcal{B} : Q \times X \to \mathbb{N}_0$, we might be interested just in the numerical part of the measurements it contains. Given a bag $\mathcal{B}$ and one of the factors of its base set, for example X, the **projection** of $\mathcal{B}$ on X is the bag

$$\begin{aligned} \mathcal{R} : X &\to \mathbb{N}_0 \\ \mathbf{x} &\mapsto \sum_{\mathbf{q} \in Q} \mathcal{B}(\mathbf{q}, \mathbf{x}) \,. \end{aligned} \tag{13}$$

In some sense, the projection acts as an inverse of the join operation (although it destroys the information about the discarded parts of the measurements). In the specific case of numerical measurements $\mathbf{x} \in X$, we will also refer to bag projections (13) as **point clouds**, due to the visualisation property enjoyed by such samples in low-dimensional spaces.

We conclude this section by introducing some additional terminology that will be useful in the following parts of the paper. We consider the analysis of a data set composed of bags (12) labelled with bag identity information, which we will call simply bags or *test bags*:

$$\mathcal{B}_{\bar{k}} \,. \tag{14}$$

If we know the class $\bar{y}$ of such a bag, the join operation will produce bags $\mathcal{B}_{\bar{k},\bar{y}}$. By uniting these bags, we obtain the training set $\mathcal{B}$ (a bag defined on the base set $K \times Q \times X \times Y$). We can then partition the complete data set according to the levels of the class variable y, obtaining **class bags**

$$\mathcal{B}_{\bar{y}} \,. \tag{15}$$

We can partition both test and class bags according to the levels $\bar{\mathbf{q}}$ of the categorical variable $\mathbf{q}$. We will refer to the resulting bags $\mathcal{B}_{\bar{k},\bar{\mathbf{q}}}$, $\mathcal{B}_{\bar{\mathbf{q}},\bar{y}}$ as *homogeneous sub-bags of the test bag* and *homogeneous class sub-bags*, respectively.

Finally, we will be interested in the numerical samples associated with test and class bags, and their homogeneous sub-bags. To this end, we will consider the training point cloud $\mathcal{R}$ obtained by projecting the entire training set $\mathcal{B}$ on X. If we consider a test bag $\mathcal{B}_{\bar{k}}$, we can project it on X obtaining the associated point cloud $\mathcal{R}_{\bar{k}}$, or first decompose it into homogeneous sub-bags $\mathcal{B}_{\bar{k},\bar{\mathbf{q}}}$ and then project them on X obtaining a corresponding collection of point clouds

$$\{\mathcal{R}_{\bar{k},\mathbf{q}}\}_{\mathbf{q} \in Q}. \tag{16}$$

These point clouds are called the **homogeneous groups** associated with the test bag. Analogously, given a class $\bar{y} \in Y$, we can define the class point cloud $\mathcal{R}_{\bar{y}}$ and the corresponding homogeneous groups

$$\{\mathcal{R}_{\mathbf{q},\bar{y}}\}_{\mathbf{q} \in Q} \,. \tag{17}$$

In Sect. 4 we will describe an original approach to use (16) and (17) to approximate the comparison between (5) and (6).

3 A heterogeneous and unbalanced data set

3.1 Classifying food processing plants

Tetra Pak is a market leader in the sector of food processing equipment and materials. Tetra Pak's customers are food processing companies that own multiple food processing plants, where raw food (e.g., milk, fruit) is transformed into packaged food. Packaged food is then distributed to retailers, where consumers ultimately buy it. The preferences, habits, and marketing profiles of consumers change over time, and food retailers want to quickly adapt to these changes. Since retailers are supplied by food processing companies, those who manage to implement lean production strategies in their plants can gain long-term economic advantages. Being able to anticipate and facilitate the required infrastructure transitions of customers has therefore become an essential capability for industrial players that, like TetraPak, operate in dynamic *business-to-business* (B2B) markets.

Packaging material and packaging equipment supplies, as well as equipment maintenance, are amongst the most profitable strategic services that Tetra Pak can offer to its customers. Since these services are managed by the food processing companies at the plant level, Tetra Pak considers the food processing plant as the relevant level of organisation. A typical food processing plant can run one or more production lines for food processing and packaging. A food processing and packaging line consists of different pieces of equipment, going from pasteurizers and blenders (which prepare and sterilise the raw food), through filling machines (which wrap processed food into cartons), to conveyor belts and palletizers (which group cartons in units ready to be delivered to retailers). For this project, Tetra Pak provided us with two data sets collected at different time scales (yearly and monthly, respectively) and describing two different levels of organisation: food processing plants and their production lines.

As we said, it is common that food processing plants change their production strategy to adapt to the needs of retailers. From Tetra Pak's perspective, this mostly reflects into changes in the quantity and quality of the packaging material bought by its customers. Such changes are captured by the **packaging material sales data** set (PMSD): each record in the PMSD set includes aggregated indicators about the yearly purchases of packaging material from a specific food processing plant. Tetra Pak can use the information provided by the PMSD to understand and meet its customers' needs.

One of Tetra Pak's business divisions defined a four-class segmentation of food processing plants according to two quantities computed every year for each plant using its PMSD record:

- the production volume, i.e., the total amount of food packages which can be obtained by using the purchased packaging material;

- the production complexity, i.e., the variety of food products that can be obtained by using the purchased packaging material.

Roughly speaking, Segment 1 is composed of plants with high production volumes and low production complexity, Segment 2 is composed of plants with small volumes of highly diversified production, Segment 3 - low volumes and low complexity, Segment 4 - high volumes and high complexity. Due to a non-disclosure agreement, we can disclose neither the specific formulas adopted to quantify the production volume and the production complexity of a customer plant, nor the thresholds used to compute its segment.

If we represent the segments as classes in the finite set

$$Y = \{y_1, y_2, y_3, y_4\},$$

and denote by K the set of plants, the described process amounts to defining a classification $\{(k, y)\}_{k \in K}$.

In some cases, when a customer plant implements a transition in its production strategy, the process might imply a change in packaging material supplies, possibly increasing the purchases of packaging material from one of Tetra Pak's competitors, reducing the purchases from Tetra Pak, or a combination of the two. Such a situation impacts the reliability of the plant's PMSD record, hampering Tetra Pak's visibility on its customer's needs, and ultimately increasing the churn risk. Therefore, for Tetra Pak, it is important to be able to track the strategies of its customers even when the PMSD records of the corresponding plants are not available.

3.2 The PLMS data set

The **packaging line monitoring system** (PLMS) is Tetra Pak's standard data management system to monitor the configuration (e.g., package shape, package volume), the processing parameters (e.g., temperatures, pressures), and the mechanical efficiency (e.g., packaging material waste, equipment stops) of the packaging lines installed at its customers' plants. These quantities are monitored at high frequency, but for storage reasons, only averages over longer time scales are usually retained for long periods.

Tetra Pak's engineers deem the filling machine the critical component of the packaging line: since this is where processed food is wrapped inside carton packages, problems on the filling machine are likely to cause problems also detectable along the packaging line. Thus, Tetra Pak's engineers suggested us to limit our investigation to five categorical indicators and the monthly averages of ten numerical *key performance indicators* (KPIs) associated with filling machines.

We represented the categorical information as an aggregated vector variable

$$\mathbf{q} = (q_1, q_2, q_3, q_4, q_5), \tag{18}$$

where $\mathbf{q} \in Q := \times_{i=1}^{5} Q_i$ and the component q_i takes values in a corresponding finite set Q_i called a **categorical factor**. The elements of the i-th factor Q_i will be referred

to as its **levels**. In particular, the interpretation of the five categorical factors is the following:

1. Q_1 – geographic cluster: where the food processing plant is located (5 different regions); this information is constant for all the production lines inside the same plant;
2. Q_2 – filling machine system: which filling machine model is installed on the packaging line (39 different systems); this information is constant for all the measurements collected from a specific production line;
3. Q_3 – package type: which types of packaging technology and packaging material are used on the line (9 different options);
4. Q_4 – package shape (10 different options);
5. Q_5 – package volume (21 different levels).

We represented the KPIs as ten-dimensional numerical vector variables

$$\mathbf{x} = (x_1, x_2, \ldots, x_{10})' \tag{19}$$

(where the apex denotes transposition), which are supposed to live in some subset $X \subset \mathbb{R}^{10}$ of the ten-dimensional Euclidean space. We call X the **KPI space**.

In this context, it is natural to represent each PLMS measurement as a realisation of a mixed-type vector variable $(\mathbf{q}, \mathbf{x})$ taking values in $Q \times X$, and each plant as a mixed-type data bag (12). In the specific case of the PLMS data, due to the nature of the numerical variables, we will refer to point clouds (13) also as **KPI point clouds**.

3.3 Analysis of the PLMS data

We pointed out that the PMSD of a plant can often be incomplete and sometimes even completely unavailable. Being able to achieve reliable insights into the production strategy of a plant even without access to its PMSD record is a relevant business intelligence problem: *can we determine the commercial segment of a plant by observing the technical performance of its lines*? According to the notation and formalism introduced in Sect. 2, this problem can be phrased as a classification on mixed-type data bags.

The subsets of the PMSD and PLMS data sets we used for this study were collected over three years, and describe 174 food processing plants around the world; some plants were not observed over all the three years. Since the commercial segment is a characteristic determined yearly at plant level (and it can change from one year to another), we define a bag to be the collection of measurements acquired in one year inside a given food processing plant. After the training-test set split, we obtained 429 labelled training bags (32316 vectors) and 83 labelled test bags (6589 vectors).

Food processing plants can run from a few to tens of production lines each. For this reason, the size of their yearly PLMS records can vary from a few dozens to hundreds of measurements.

On average, plants of Segment 1 and Segment 4 run more machines than plants of the other segments. Therefore, their bags usually contain more measurements than the bags of plants in Segment 2 and Segment 3. Moreover, large food processing plants are

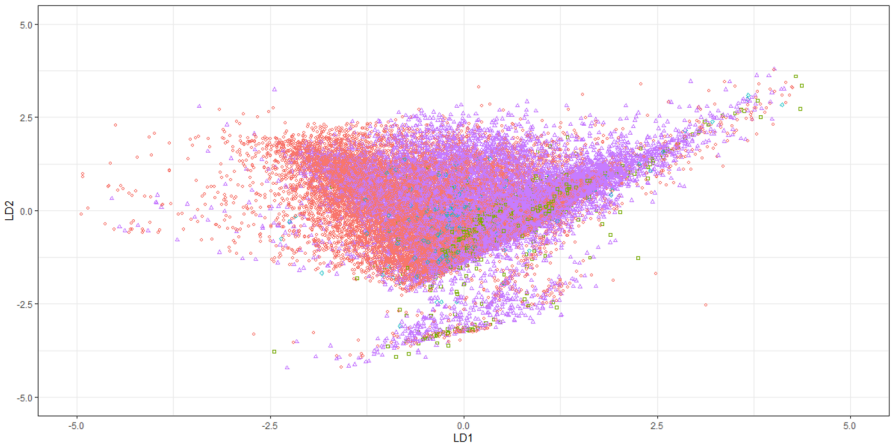


Fig. 1 LDA applied to the labelled KPI measurements. Due to the high overlapping, the majority Segment 1 and Segment 4 (red and violet) hide the minority Segment 2 and Segment 3 (green and cyan)

more numerous than small ones, and large plants are more numerous in Segment 1 and Segment 4, implying that the PLMS data set counts more bags from these segments.

These two facts add up to increase the gap between the number of observations available about Segment 1 and Segment 4 on one side, and Segment 2 and Segment 3 on the other side. The reader can thus understand that the PLMS data set is highly **unbalanced**. In particular, since 35 over the 83 test bags represent plants of Segment 4, the baseline performance for classification is approximately 40%.

To avoid complicating our analysis unnecessarily, we focussed our first investigations on KPI measurements (19), temporarily ignoring the categorical information included in the PLMS measurements.

The first question we asked ourselves is how different the KPI measurements collected from plants belonging to different segments are. To this end, we analysed the class KPI point clouds $\mathcal{R}_y$, $y \in Y$ using standard multivariate analysis of variance (MANOVA) tests (Rencher 2003). These tests showed that the means of their distributions appear to be significantly different (Pillai, Hotelling-Lawley, Roy, and Wilks tests return p-values which are close to zero).

However, even though the class KPI point clouds have different means, their supports are highly overlapped. This property is clearly shown by Fig. 1, which displays the values of the first two linear discriminant functions associated with an LDA model trained on the KPI measurements of the training set.

Even the record of a single plant is usually spread on a large region of the KPI space and is highly overlapped to the records of other plants. A visual example is provided by Fig. 2, where the PCA scores of a plant KPI point cloud are compared to those of the KPI point cloud of the entire training set.

In such circumstances, also considering that the PLMS data set is unbalanced, inferring the commercial segment of a plant from the KPI measurements alone seemed unlikely to attain satisfying performance. This intuition was confirmed by some exper-

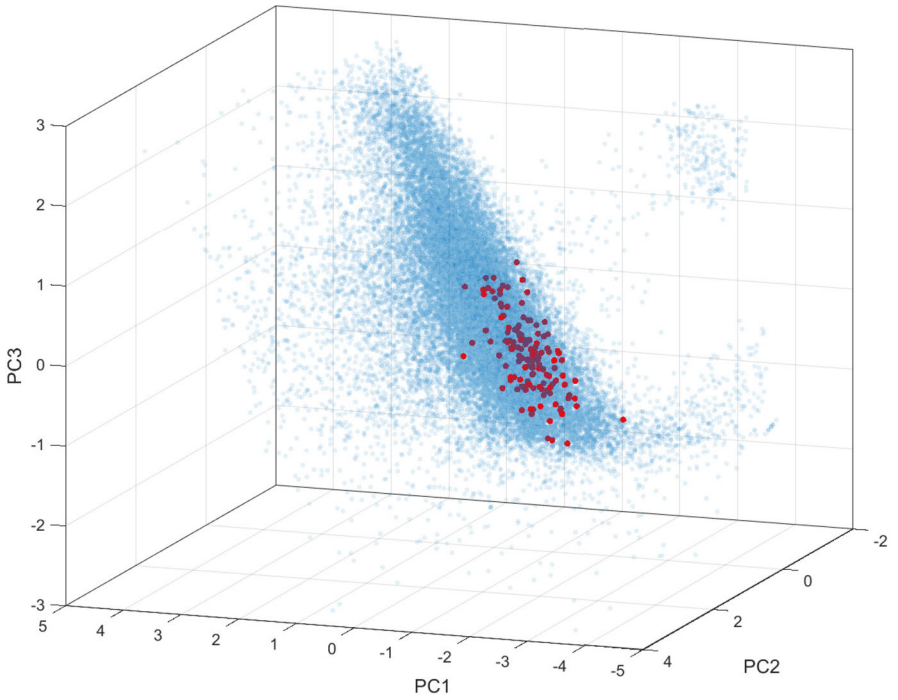


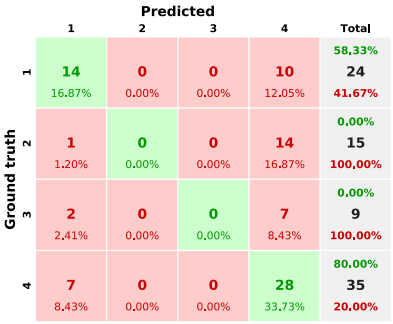
Fig. 2 PCA scores of the KPI point cloud of a food processing plant compared to the PCA scores of the KPI point cloud of the whole training set. The first three principal components are shown

iments with state-of-the-art classification methods which will be described in more detail in Sect. 5.1, whose confusion matrices are reported in Fig. 3.

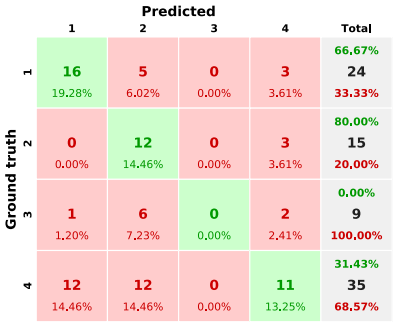
As expected, the minority Segment 2 and Segment 3 were impossible to detect, with the noticeable exception of Segment 2 through the lens of quadratic discriminant analysis (QDA) (that anyway came at the cost of reduced accuracy for plants of Segment 4).

Due to its nature, factor Q_1 is unspecific to the commercial segmentation criteria. Factor Q_2 can be used to discriminate high-throughput machines from low-throughput machines: intuitively, the corresponding levels should provide no information about the production complexity of a plant, though they could correlate with the production volume of a plant. Nevertheless, plant managers can in principle choose to install multiple low-throughput machines to achieve the production volumes which could be obtained by fewer high-throughput machines, or install machines produced by Tetra Pak's competitors. The levels of the remaining factors Q_3, Q_4 and Q_5 can provide some information about the production complexity of a plant, but they are unspecific to the production volumes.

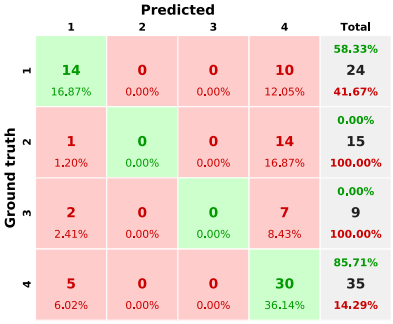
Therefore, we also trained some models using only the categorical measurements to check whether this information could yield satisfying classification performance. The results of our experiments, reported in Fig. 4, led us to rule out this possibility.



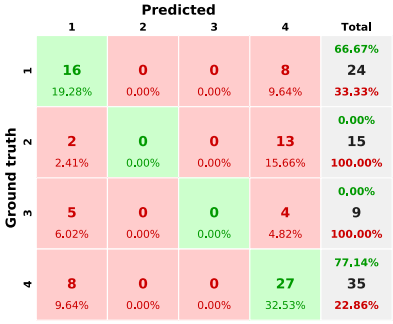
(a) Linear discriminant analysis.



(b) Quadratic discriminant analysis.

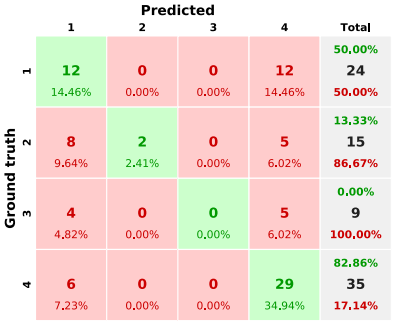


(c) Support vector machine.

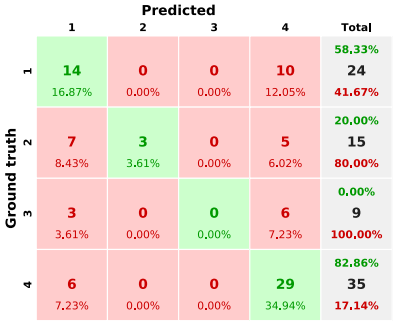


(d) Multi-layer perceptron.

Fig. 3 The classification performance of several classification algorithms applied to the KPI measurements



(a) Support vector machine.



(b) Multi-layer perceptron.

Fig. 4 The classification performance of multiple classification algorithms applied to the categorical measurements

In isolation, the KPI measurements and the categorical information were not sufficient to attain good classification performance. Therefore, we proceeded to investigate the interactions between the categorical factors and the KPI measurements, i.e., the distributions $p(\mathbf{q}, \mathbf{x})$. Since Q is a finite set, we decided to describe these distributions in an "enumerative" way by decomposing them as $p(\mathbf{x} \mid \mathbf{q})p(\mathbf{q})$. This decomposi-

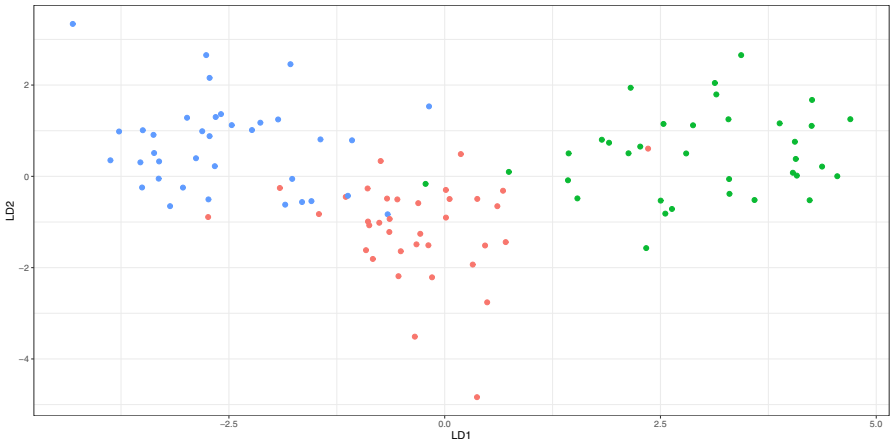


Fig. 5 LDA on PLMS data – impact of the geographic cluster factor. Different point clouds represent packaging lines equipped with the same filling machine system and producing the same kind of package but running in different regions of the world

tion has the additional advantage of enabling the application of standard multivariate statistics techniques to the analysis of the mixture components $p(\mathbf{x} \mid \mathbf{q})$.

Consider the complete training bag $\mathcal{B}$. We analysed the collection $\{\mathcal{R}_\mathbf{q}\}_{\mathbf{q}\in Q}$ of its homogeneous groups with the following procedure. First, we fixed a factor $Q_i, i \in \{1, 2, 3, 4, 5\}$ and fixed the levels $\bar{q}_j$, $j \neq i$ of the remaining four factors. Then, we defined $Q_{\bar{\mathbf{q}}_j, j\neq i} := \{\mathbf{q} \in Q \mid q_j = \bar{q}_j, j \neq i\}$. Finally, we processed the collection $\{\mathcal{R}_\mathbf{q}\}_{\mathbf{q}\in Q_{\bar{\mathbf{q}}_j, j\neq i}}$ of homogeneous groups using linear discriminant analysis (LDA).

The individual impact of the five factors was interestingly revealed by multiple applications of this simple algorithm. Figure 5 shows the LDA scores of three homogeneous groups, differentiated by the geographic cluster factor Q_1.

Figure 6 shows the LDA scores of four homogeneous groups, differentiated by the filling machine system factor Q_2.

Figure 7 shows the LDA scores of three homogeneous groups, differentiated by the package volume factor Q_5.

We also applied LDA to the collections $\{\mathcal{R}_{\bar{k},\mathbf{q}}\}_{\mathbf{q}\in Q}$ of homogeneous groups associated to different plant bags $\mathcal{B}_{\bar{k}}$. As opposed to the apparent lack of structure exhibited by the plant point cloud $\mathcal{R}_{\bar{k}}$ in the KPI space (see Fig. 2), this collection fragmented in interesting mixture-like structures when considering categorical information, as shown in Fig. 8.

Multivariate normality tests (such as Mardia's, Henze-Zirkler's, Royston's, Doornik-Hansen's) showed that the homogeneous groups in the considered data set are typically not normally distributed. In Fig. 9, four different class homogeneous groups $\mathcal{R}_{\mathbf{q},y}$ are analysed.

The two-dimensional contour and the three-dimensional surface graphs represent the sample densities of the first and the second principal component scores. The quantile-quantile plots compare the squared Mahalanobis distances of the data points from the point clouds' mean vectors to the chi-square statistics (a well-known

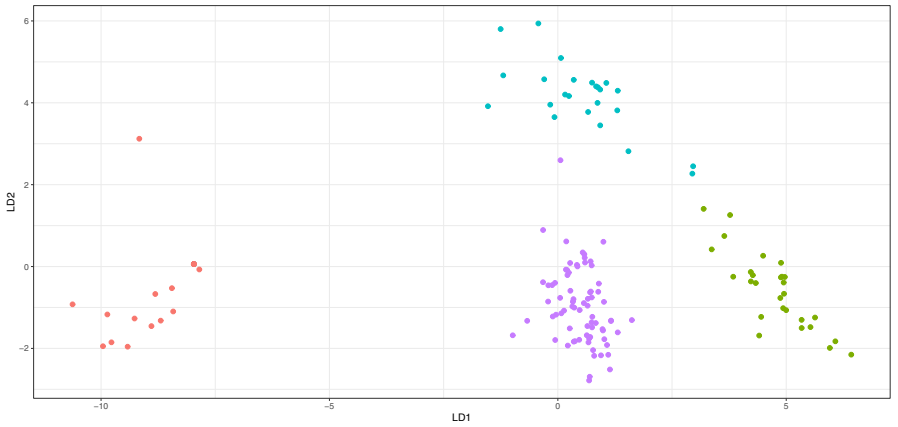


Fig. 6 LDA on PLMS data – impact of the machine system factor. Different point clouds represent packaging lines from a specific region of the world producing the same kind of package but using different filling machine systems

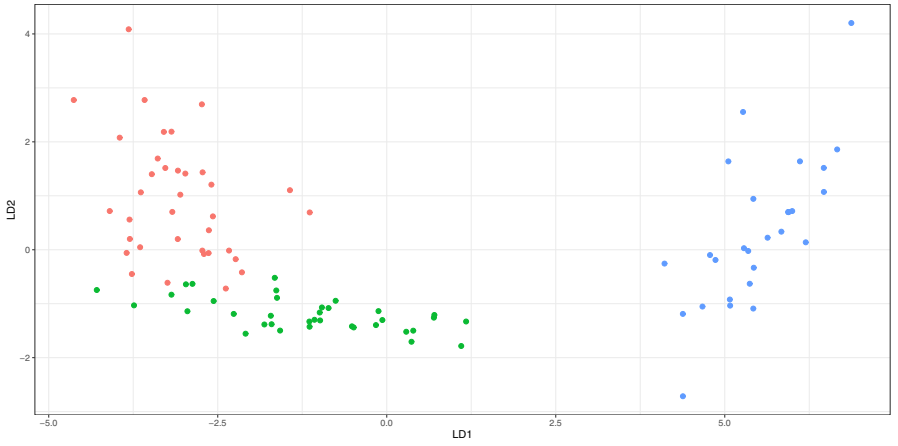


Fig. 7 LDA on PLMS data – impact of the package volume factor. Different point clouds represent packaging lines from a specific region of the world equipped with the same filling machine system and using the same packaging material technology but producing packages of different sizes

multivariate normality criterion). Although the multivariate distributions of the homogeneous groups are not multivariate Gaussians and even not perfectly uni-modal, they usually exhibit one strongly predominant mode.

We interpreted these results in the following way. The productive contexts in which the analysed food processing plants operate are extremely diverse, depending on geographic factors (energy supplies), political factors (work regulations), and specific market contingencies (product seasonality). Moreover, the diversity of the operational conditions of the packaging equipment and the complex interactions between different levels of organisation (packaging line, food processing plant, owner company) also contribute to this diversity. For example, the choice of the package type, shape,

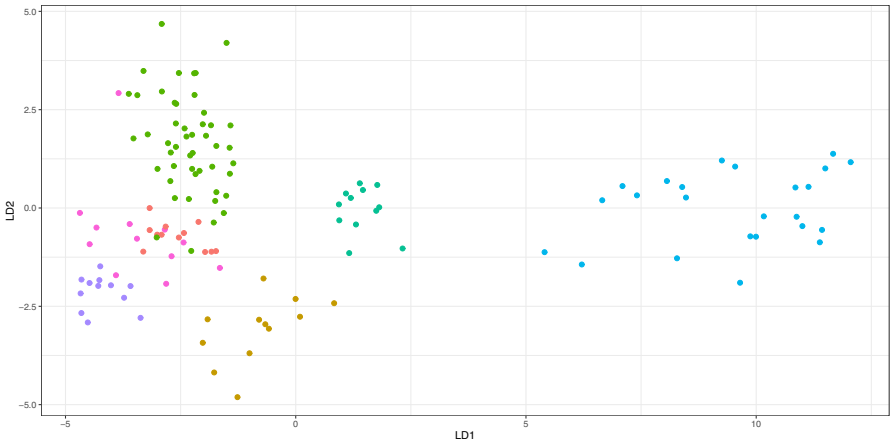


Fig. 8 The representation of a plant point cloud as obtained by applying LDA using the levels of the categorical variable as classes. Different colours identify the different homogeneous groups that partition the point cloud. This is the same plant depicted in red in Fig. 2

and volume can be defined for the single packaging line; the efficiency of the energy supply and the failure recovery procedures are characteristics of the plant; maintenance policies can be defined at an even higher level of organisation. Partitioning the records into groups that are homogeneous with respect to the levels of $\mathbf{q}$ eliminates the deterministic impact of the categorical information, enhancing the uni-modality of sample distributions; vice versa, ignoring qualitative factors gradually reintroduces multi-modality.

4 The *fingerprint* method

4.1 Acceptable groups and the *fingerprint* of a bag

Let $X^* := \{\mathcal{R} : X \to \mathbb{N}_0 \,|\, 0 < \sum_{\mathbf{x}\in X} \mathcal{R}(\mathbf{x}) < +\infty\}$ denote the set of all (finite) samples from the space X. We define a *statistic* to be any function

$$t : X^* \to T$$

taking values in a given set T. For example, if $X := \mathbb{R}^d$ is the d-dimensional Euclidean space, we could set $T = X$ and define t to be the *sample mean*.

It might happen that a given sample does not contain a sufficient number of (or sufficiently good) measurements to compute a given statistic t. For example, if we want to compute the *sample mean* of a point cloud, then it should contain at least one vector. If we want to compute the first *principal component* of a point cloud, then it should contain at least two distinct vectors that define a one-dimensional affine subspace; more generally, if we want to compute the first m principal components, the point cloud should contain at least $m + 1$ distinct vectors that define an m-dimensional

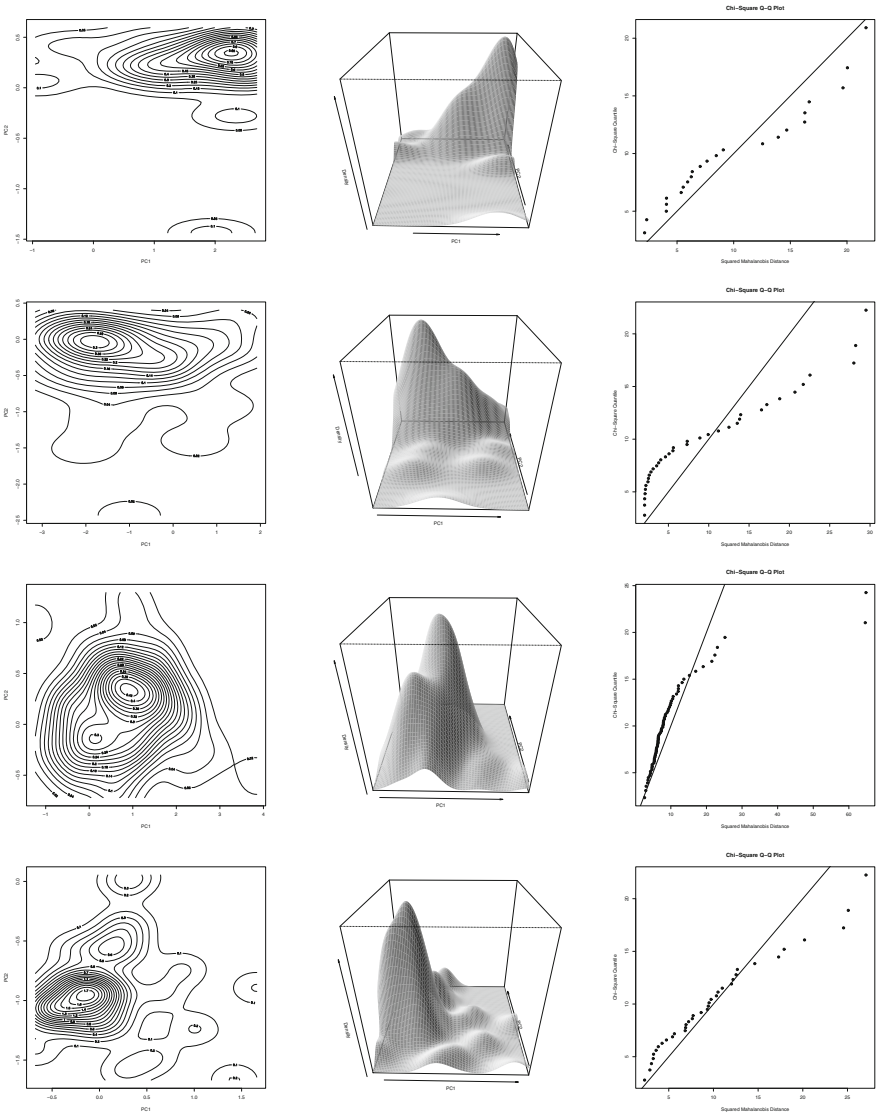


Fig. 9 The first and second columns show the contour graphs and the surface graphs of the sample densities of the first and second principal components of four class homogeneous groups. The third column shows QQ plots for the corresponding 10-dimensional KPI samples, illustrating deviation from normality

affine subspace. Again, if we want to compute the *Mahalanobis distance* (Mahalanobis 1936) of a data point from the mean of its point cloud, then the *sample covariance matrix S* should be invertible.

These requirements become even more stringent when performing computations on a digital computer. To see this, consider the last example. From a theoretical standpoint, a sufficient condition for invertibility of S is that the point cloud should contain at least

d linearly independent vectors (d is the dimensionality of the sample space). But even in this case, the matrix could be computationally singular, meaning that the inverse that can practically be computed (the *computational inverse*) might greatly differ from the true inverse due to numerical instability issues.

Given a statistic t, we need to characterise whether or not it is possible to compute it on a given sample $\mathcal{R} \in X^*$. To this end, we define the *acceptance function*:

$$a_t : X^* \to \{0, 1\}$$
$$\mathcal{R} \mapsto \begin{cases} 0\,, & \text{if } t(\mathcal{R}) \text{ cannot be computed,} \\ 1\,, & \text{otherwise.} \end{cases}$$

We can extend this definition to the case of multiple statistics. Let $t_1, \ldots, t_L$ denote statistics on X^*, and $a_{t_1}, \ldots, a_{t_L}$ the corresponding acceptance functions. If we denote by $\mathbf{t} := (t_1, \ldots, t_n)$ the collection of these statistics, the corresponding acceptance function is

$$a_{\mathbf{t}} : X^* \to \{0, 1\}$$
$$\mathcal{R} \mapsto \begin{cases} 0\,, & \text{if } \exists\, \bar{\ell} \in \{1, \ldots, L\} \text{ s.t. } a_{t_{\bar{\ell}}}(\mathcal{R}) = 0\,, \\ 1\,, & \text{if } a_{t_\ell}(\mathcal{R}) = 1\,,\ \forall\, \ell = 1, \ldots, L\,. \end{cases} \tag{20}$$

Definition 1 Let $\mathbf{t}$ be a collection of statistics defined on X^*, and denote by $a_{\mathbf{t}}$ the corresponding acceptance function (20). We say that the homogeneous group (13) is **acceptable** for the statistics $\mathbf{t}$ whenever

$$a_{\mathbf{t}}(\mathcal{R}) = 1\,.$$

Let $\mathcal{B}_{\bar{k}}$ denote a bag, and $\{\mathcal{R}_{\bar{k},\mathbf{q}}\}_{\mathbf{q}\in Q}$ the collection of its homogeneous groups. Analogously, let $\mathcal{B}_{\bar{y}}$ denote a class bag, and $\{\mathcal{R}_{\mathbf{q},\bar{y}}\}_{\mathbf{q}\in Q}$ the collection of its homogeneous groups.

Definition 2 Let $\mathbf{t}$ denote a given collection of statistics and $a_{\mathbf{t}}$ the corresponding acceptance function. The collection of acceptable homogeneous groups

$$\mathcal{F}_{\bar{k}} := \{\mathcal{R}_{\bar{k},\mathbf{q}} \,|\, a_{\mathbf{t}}(\mathcal{R}_{\bar{k},\mathbf{q}}) = 1\}\,, \tag{21}$$

(respectively, $\mathcal{F}_{\bar{y}} := \{\mathcal{R}_{\mathbf{q},\bar{y}} \,|\, a_{\mathbf{t}}(\mathcal{R}_{\mathbf{q},\bar{y}}) = 1\}$) is called the **bag*fingerprint*** (respectively, the **class*fingeprint***).

Algorithm 1 in Appendix 1details how to build the *fingerprint* of a given bag.

For a given bag *fingerprint* $\mathcal{F}_{\bar{k}}$, we define the symbol

$$\mathcal{Q}_{\bar{k}} := \{\mathbf{q} \in Q \,|\, \mathcal{R}_{\bar{k},\mathbf{q}} \in \mathcal{F}_{\bar{k}}\}$$

to identify the set of the levels of the categorical variable which define its homogeneous groups. Analogously, for a given class *fingerprint* $\mathcal{F}_{\bar{y}}$ we define the set

$$\mathcal{Q}_{\bar{y}} := \{\mathbf{q} \in Q \,|\, \mathcal{R}_{\mathbf{q},\bar{y}} \in \mathcal{F}_{\bar{y}}\}\,.$$

Definition 3 Let $\mathcal{B}_{\bar{k}}$ and $\mathcal{B}_{\bar{y}}$ denote a test bag and a class bag, respectively. Let $\mathcal{F}_{\bar{k}}$ and $\mathcal{F}_{\bar{y}}$ denote the corresponding bag and class *fingerprints*. We say that $\mathcal{F}_{\bar{k}}$ and $\mathcal{F}_{\bar{y}}$ *have a match* whenever

$$\mathcal{Q}_{\bar{k},\bar{y}} := \mathcal{Q}_{\bar{k}} \cap \mathcal{Q}_{\bar{y}} \neq \emptyset\,;$$

$\mathcal{Q}_{\bar{k},\bar{y}}$ is called the **match** of the *fingerprints* of $\bar{k}$ and $\bar{y}$.

4.2 The method

In Sect. 2 we framed the problem of classification of a mixed-type data bag identified by $\bar{k}$ as the result of a comparison between the bag *fingerprint* distributions $\{p_{\bar{k},\mathbf{q}}(\mathbf{x})\}_{\mathbf{q}\in Q}$ and the class *fingerprint* distributions $\{p_{\mathbf{q},y}(\mathbf{x})\}_{\mathbf{q}\in Q}$, where $y \in Y$. Qualitatively speaking, the *fingerprint* method instantiates this operation by comparing specific sample statistics derived from the available bag and class fingerprints.

In the following, we detail its training and inference phases.

Training Given a class $\bar{y} \in Y$, the following statistics are computed for each homogeneous group $\mathcal{R}_{\bar{\mathbf{q}},\bar{y}} \in \mathcal{F}_{\bar{y}}$:

- the sample **mean vector** $\mathbf{m}_{\bar{\mathbf{q}},\bar{y}} := \sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})\mathbf{x} / \sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})$;
- the sample **covariance matrix** $S_{\bar{\mathbf{q}},\bar{y}} := \sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})'$ $/(\sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x}) - 1)$;
- the **average Mahalanobis distance from the mean vector**

$$m_{\bar{\mathbf{q}},\bar{y}} := \frac{\sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})\sqrt{(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})' S^{-1}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})}}{\sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})}\,;$$

- the **standard deviation of these Mahalanobis distances** $s_{\bar{\mathbf{q}},\bar{y}}$;
- the first n_{pcs} **principal directions** $\boldsymbol{v}_{\bar{\mathbf{q}},\bar{y},i}$ (i.e., the eigenversors of $S_{\bar{\mathbf{q}},\bar{y}}$ associated with its n_{pcs} largest eigenvalues).

The process is repeated for each class in Y. See Algorithm 2 in Appendix 1 for additional details.

Inference Given a test bag $\mathcal{B}_{\bar{k}}$ with *fingerprint* $\mathcal{F}_{\bar{k}}$, two steps are needed.

1. **[Comparing homogeneous groups]** Given a class $\bar{y} \in Y$, determine the **matched fingerprint** $\mathcal{F}_{\bar{k},\bar{y}} := \{\mathcal{R}_{\bar{k},\mathbf{q}} \in \mathcal{F}_{\bar{k}} \mid \mathbf{q} \in \mathcal{Q}_{\bar{k},\bar{y}}\}$. Then, for each homogeneous group $\mathcal{R}_{\bar{k},\bar{\mathbf{q}}} \in \mathcal{F}_{\bar{k},\bar{y}}$, compute the following:
   - the average Mahalanobis distance from the mean of the matching homogeneous group in the class *fingerprint*

$$m_{\bar{k},\bar{\mathbf{q}},\bar{y}} := \frac{\sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{k},\bar{\mathbf{q}}}(\mathbf{x})\sqrt{(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})' S^{-1}_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x} - \mathbf{m}_{\bar{\mathbf{q}},\bar{y}})}}{\sum_{\mathbf{x}\in X} \mathcal{R}_{\bar{k},\bar{\mathbf{q}}}(\mathbf{x})};$$

   - the **first moment fit**

$$\alpha_{\bar{k},\bar{\mathbf{q}},\bar{y}} := (m_{\bar{k},\bar{\mathbf{q}},\bar{y}} - m_{\bar{\mathbf{q}},\bar{y}})/s_{\bar{\mathbf{q}},\bar{y}}\,;$$

this statistic aims at capturing the compatibility of the "scatterings" of $p_{\bar{k},\bar{\mathbf{q}}}(\mathbf{x})$ and $p_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})$;
- the sample covariance matrix $S_{\bar{k},\bar{\mathbf{q}}}$ of the homogeneous group $\mathcal{R}_{\bar{k},\bar{\mathbf{q}}}$;
- the first n_{pcs} principal directions $\boldsymbol{\nu}_{\bar{k},\bar{\mathbf{q}},i}$ of $S_{\bar{k},\bar{\mathbf{q}}}$;
- the **second moment fit**

$$\beta_{\bar{k},\bar{\mathbf{q}},\bar{y}} := \frac{\sum_{i=1}^{n_{pcs}} |\cos(\theta_{\bar{k},\bar{\mathbf{q}},\bar{y},i})|}{n_{pcs}},$$

where $|\cos(\theta_{\bar{k},\bar{\mathbf{q}},\bar{y},i})| := |\langle \boldsymbol{\nu}_{\bar{k},\bar{\mathbf{q}},i}, \boldsymbol{\nu}_{\bar{\mathbf{q}},\bar{y},i} \rangle|$ quantifies the alignment between the i-th principal directions of the two point clouds; this statistic aims at capturing the compatibility of the "orientations" of $p_{\bar{k},\bar{\mathbf{q}}}(\mathbf{x})$ and $p_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})$.

This step must be iterated for each class.

2. **[Classification]** Assign the bag to a class $\bar{y} \in Y$ by applying the following three (subordinate) criteria:

   0. $\mathcal{Q}_{\bar{k},\bar{y}} \neq \emptyset$;
   1. the **average of the first moments fits**

$$\sum_{\mathbf{q} \in \mathcal{Q}_{\bar{k},\bar{y}}} \alpha_{\bar{k},\mathbf{q},\bar{y}} / \#(\mathcal{Q}_{\bar{k},\bar{y}})$$

   is minimized;
   2. the **average of the second moments fits**

$$\sum_{\mathbf{q} \in \mathcal{Q}_{\bar{k},\bar{y}}} \beta_{\bar{k},\mathbf{q},\bar{y}} / \#(\mathcal{Q}_{\bar{k},\bar{y}})$$

   is maximised.

   Failure to meet criterion 0 automatically excludes the possibility to assign $\bar{k}$ to $\bar{y}$, since the knowledge about the required distributions $\{p_{\mathbf{q},\bar{y}}(\mathbf{x})\}_{\mathbf{q} \in \mathcal{Q}_{\bar{k}}}$ is insufficient. Criterion 2 is applied only when criterion 1 provides similar answers for two classes $y_i, y_j \in Y$, $y_i \neq y_j$; by similar, we mean that the ratio between the average first moments fits of the two classes lies in a given interval $[1 - \tau, 1]$ (τ is a tunable hyper-parameter of the method). In this way, we make the second-order properties of the point clouds more relevant when the first-order properties are not sufficiently informative.

See Algorithm 3 in Appendix 1for additional details.

Intuitively, the *fingerprint* method assigns an individual to the segment for which the point clouds of the matching homogeneous groups have the most similar sample distributions. Although the chosen statistics are not sufficient to compare complex multivariate distributions, they proved to be sufficently good choices under the assumption of "quasi-uni-modality" of the homogeneous groups discussed in Sect. 3.

Given two different classes $y_i, y_j \in Y$, it might happen that the matches $\mathcal{Q}_{\bar{k},y_i}$ and $\mathcal{Q}_{\bar{k},y_j}$ differ not only in size but also in composition. In principle, the *fingerprint* method can thus compare the collection $\{p_{\bar{k},\mathbf{q}}(\mathbf{x})\}_{\mathbf{q}\in\mathcal{Q}_{\bar{k}}}$ with different sets of distributions $\{p_{\mathbf{q},y_i}(\mathbf{x})\}_{\mathbf{q}\in\mathcal{Q}_{\bar{k},y_i}}$, $\{p_{\mathbf{q},y_j}(\mathbf{x})\}_{\mathbf{q}\in\mathcal{Q}_{\bar{k},y_j}}$, depending on the specific test bag $\bar{k}$.

Although the PLMS data set described in Sect. 3 contained thousands of measurements, the number N_Q of possible levels was still too high to prevent the information about $\mathbf{q}$ from being "sparse". I.e., certain levels of the categorical variable appeared only in certain classes and not in others. At the same time, recall that the classification methods based on purely categorical data did not attain satisfying performance.

Under such circumstances, the set $\bigcap_{y\in Y}\mathcal{Q}_{\bar{k},y}$ is likely to be empty for many test bags. Taking the means of the first and second moments fit over all the levels in $\mathcal{Q}_{\bar{k},\bar{y}}$, possibly depending on the class $\bar{y}$, is meant to combine all the available information about the specific bag-class comparison.

4.3 The *multi-stage fingerprint* method

The *fingerprint* method can classify a plant only when at least one class $\bar{y} \in Y$ is such that $\mathcal{Q}_{\bar{k},\bar{y}} \neq \emptyset$. What happens if no match can be found between the bag *fingerprint* and any of the class *fingerprints*?

When no shared levels of $\mathbf{q}$ can be identified between the bag and the class *fingerprints*, one of the categorical factors composing Q can be dropped so that the bags and homogeneous groups are redefined with respect to the combined levels of the remaining factors. The procedure can be iterated by eliminating one factor at a time until all the categorical factors have been removed. At the last iteration, the *fingerprint* method reduces to a quadratic classifier based on the Mahalanobis distance. We call this version of the algorithm the ***multi-stage fingerprint*** method.

In which order should the categorical factors be dropped? Recall that in Sect. 3 we observed that using all the categorical factors produced distributions $p_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})$ which are approximately uni-modal. The effect of dropping a factor is to merge some previously separated homogeneous groups, and this can turn uni-modal distributions into multi-modal ones. The specific criteria we chose to compare $p_{\bar{k},\bar{\mathbf{q}}}(\mathbf{x})$ to $p_{\bar{\mathbf{q}},\bar{y}}(\mathbf{x})$ (the distribution of the sample Mahalanobis distances and the alignment between the sample principal directions) are sufficiently representative in case of uni-modal distribution, but they can become less informative when transitioning to multi-modal distributions. Indeed, in Sect. 5 we will show that the use of fewer factors during the iterations makes the classifier less precise.

For this reason, we prescribe a heuristic rule to determine the order in which the categorical factors should be dropped: the merging should maximise the number of test bag *fingerprints* $\mathcal{F}_{\bar{k}}$ that have a non-empty match $\mathcal{Q}_{\bar{k},\bar{y}}$ with at least one class *fingerprint* $\mathcal{F}_{\bar{y}}$. In this way, the geometric criteria become applicable to the largest number of individuals while dropping a minimum number of categorical factors, thus reducing the degradation of the uni-modality of the distributions. We remark that computing the drop sequence does not require computing the statistics for all the homogeneous groups in the test bag *fingerprints* and class *fingerprints*. Indeed, it only requires the computation of the matches $\mathcal{Q}_{k,y}, k \in K_{test}, y \in Y$.

We formalise the above considerations as follows. We define

$$Q^{[i]} := Q_1 \times Q_2 \times \cdots \times Q_{i-1} \times Q_{i+1} \times \cdots \times Q_{N_F}$$

as the set obtained from Q by dropping the i-th factor. The homogeneous groups $R^{[i]}_{\bar{k},\bar{\mathbf{q}}}$ and $R^{[i]}_{\bar{\mathbf{q}},\bar{y}}$ are straightforwardly redefined by replacing $\bar{\mathbf{q}} \in Q$ with $\bar{\mathbf{q}} \in Q^{[i]}$. We then define

$$K^{[i]}_{test} := \{k \in K_{test} \mid \exists\, \bar{y} \in Y,\, \mathcal{Q}_{k,\bar{y}} \neq \emptyset\}$$

as the set of plants for which the criteria are applicable for at least one segment. Finally, we select $i \in \{1, \ldots, N_F\}$ to maximise $\#(K^{[i]}_{test})$.

Note that classifying different test sets might define different sequences of eliminations and, consequently, require the computation of a different collection of *fingerprints*. In turn, this difference yields different *multi-stage fingerprint* classifiers. In a sense, the training process for the *multi-stage fingerprint* method depends on the properties of the test set.

We observed an analogy between the *multi-stage fingerprint* method and the formalism of **hierarchical learning**. Hierarchical learning is a conceptual meta-model which can be applied to general statistical and machine learning tasks (Zhang and Zhang 2006). Hierarchical learning suggests that models with stronger inference capabilities can be developed by training multiple models at different scales (local, regional, global). For example, an image can be analysed pixel-by-pixel, by considering sets of neighbouring pixels and analysing the picture as a whole.

The mathematical foundations for hierarchical learning are set by the quotient space theory of problem-solving (Zhang and Zhang 2004). Given a domain X, a *problem view* is a triplet

$$([X], [F], [f])\,, \tag{22}$$

where $[X]$ is quotient space defined by some equivalence relationship on X (which defines the scale or *grain size* of the view), $[F]$ is an algebraic structure on $[X]$ and $[f]$ on $[X]$ is a map called the *feature map* of the problem view. The learning problem is then represented by a semi-lattice whose elements are quotient spaces (22), each of which provides a view of X at a specific grain size.

The *multi-stage fingerprint* method applies the same geometric criteria at different grain sizes, by hierarchically aggregating homogeneous groups. This hierarchical aggregation reflects the definition of multiple equivalence relationships on the space of measurements. Given a training set that includes information about multiple qualitative factors, one can define a combinatorial number of drop sequences for the factors. These sequences generate a tree structure of *fingerprints* which describe the distribution at different grain sizes. Depending on the chosen merging criterion, the *multi-stage fingerprint* method selects one of the possible tree-traversing paths, hence instantiating a specific hierarchical view of the learning problem.

5 Experimental results

To assess the performance of the *fingerprint* method and its multi-stage version, we compared them to other classification algorithms on a series of synthetic data sets plus the original industrial PLMS data set. In all our experiments we set the hyper-parameters of the method to $n_{pcs} = 2$ and $\tau = 0.1$. In the toy examples, it was sufficient to apply a single stage of the *multi-stage fingerprint* method, resulting in the application of the "basic" *fingerprint* method. In the industrial use case, instead, it was necessary to apply all the stages to classify all the test bags and compare the results with those of the competitor methods (which always yield a prediction by construction).

After briefly presenting the selected competitor methods, we will describe the toy data sets and summarise the main insights of the corresponding experiments. Then, we will report the performance of the *multi-stage fingerprint* method on the PLMS data set.

5.1 Competitor methods

We compared the *multi-stage fingerprint* method to a rich variety of standard statistical and machine learning methods, as well as to methods explicitly designed to classify mixed-type and bagged data.

LDA & QDA Linear discriminant analysis and *quadratic discriminant analysis* are standard techniques in multivariate statistics (McLachlan 1992). We used the implementations readily available in the `MASS` package for the R programming language.

MixtComp The *mixture composer* algorithm is a recently proposed method designed to classify mixed-type measurements (Biernacki et al. 2015). We used the implementation provided by the `RMixtComp` package for the R programming language.

SVM Support vector machines are consolidated machine learning systems, with a solid theory and efficient implementations (Boser et al. 1992; Cristianini and Shawe-Taylor 2000). In our experiments, we adopted a Gaussian kernel $K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) := e^{-\gamma \|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\|^2}$, widely used to classify data sets characterised by non-linear decision surfaces. We tuned the kernel sensitivity hyper-parameter γ and the regularisation hyper-parameter C (associated with the numerical optimisation problem) through cross-validation on top of training data. We wrote a MATLAB wrapper around the *parallel gradient projection-based decomposition technique* (parallel GPDT) software for efficient optimisation (Zanni et al. 2006).

MLP Multi-layer perceptrons are simple artificial neural networks consisting of multiple densely connected layers (Hinton 2007). We used MLPs composed of two hidden layers containing 20 and 10 neurons, respectively; the artificial neurons used the hyperbolic tangent as the activation function, whereas the four neurons in the output layer use the (standard) identity activation function. We minimised the *cross-entropy* (CE) loss using the standard combination of the backpropagation algorithm (Rumelhart et al.

1986) and a variant of stochastic gradient descent (in our case, the Adam optimisation algorithm (Kingma and Ba 2014)) with a learning rate $\eta = 0.001$ and a mini-batch size of $n_{batch} = 200$. We used the implementation provided by the `scikit-learn` library for the Python programming language.

RF Random forests (Breiman 2001) are popular ensemble machine learning systems (Schapire 1990) which leverage *decision trees* (Quinlan 1986) as their weak learners. Random forests are natively capable of handling mixed-type data, and have the noticeable property of avoiding overfitting (provided, of course, that the training data distribution is representative of the real-world distribution). We used RFs consisting of $n_{trees} = 100$ DTs. The trees were grown using the maximum information gain branching criterion and applying the maximum depth termination criterion. We used the implementation provided by the `scikit-learn` library for the Python programming language.

MIL The framework of *multiple instance learning* encompasses a series of techniques designed to classify bags of data (Dietterich et al. 1997). To the best of our knowledge, this is the only family of methods explicitly designed to handle bagged data. In our experiments, we used the *MI-SVM* variant (Andrews et al. 2003), which is based on a constrained SVM learning problem designed to take into account the *multiple instance hypothesis*. As in the experiments with standard SVMs, we adopted the widely used Gaussian kernel and tuned the hyper-parameters γ and C using cross-validation on top of training data. We used a publicly available Python implementation associated with a comprehensive literature review on the topic (Doran and Ray 2014).

The methods we considered vastly differ for statistical and computational properties. Most of the methods (LDA, QDA, MixtComp, SVM, MLP, RF) are designed to perform classification at the instance-level, not at the bag level. In particular, some of them are natively capable of handling multi-class classification problems (LDA, QDA, MixtComp, MLP, RF), whereas we needed to adapt the others (SVM, MIL) to implement multiple class-vs-class classifiers, whose results were then fed to a max-win voting (MWV) procedure.

Amongst the considered methods, only MixtComp and RF can natively handle mixed-type data; hence, for the other methods, we needed to implement a type-conversion preprocessing before we could feed the data points to the algorithms. As we have seen in Sect. 3, geometry provides critical information about the PLMS data set and, therefore, preferred to convert categorical measurements into numerical measurements by one-hot encoding their values, instead of binning numerical measurements into categorical or ordinal values.

Finally, some of the chosen algorithms have linear cost in the training data set size (LDA, QDA, MixtComp, MLP), some superlinear (RF), and others quadratic (SVM, MIL). Notice that the higher computational cost for SVM and MIL adds to the need of building multiple class-vs-class classifiers, making these methods less appealing for *big data* regimes.

We conclude this subsection with a couple of technical remarks.

- In most of the experiments, including those on the PLMS data set, one-hot encoding the categorical variables led to computationally singular covariance matrices of

class data, resulting in QDA failing. To simplify the comparisons, we decided not to report the results of this method on the experiments, even for those few where it provided predictions.
- Although, in theory, RF are natively capable of handling categorical data, the chosen `scikit-learn` implementation does not. Hence, also the results for the RFs are computed on top of mixed-type data where the categorical components have been one-hot encoded.

5.2 Toy data sets

To better understand the performance of the *fingerprint* method and the conditions under which it should be preferred to other methods, we designed a parametric algorithm to generate a collection of toy data sets which are representative of the use cases for which the *fingerprint* method is intended.

First, we need to define the set of classes Y and the domain of categorical variables Q. Then, for each pair $(\mathbf{q}, y) \in Q \times Y$, we need to parametrise the distribution $p_{\mathbf{q},y}(\mathbf{x}) = p(\mathbf{x} \mid \mathbf{q}, y)$. In Sect. 3 we showed that the homogeneous group represented in the PLMS data set appeared to be approximately uni-modal. Hence, in order to keep things simple, we chose to model the homogeneous groups $p_{\mathbf{q},y}(\mathbf{x})$ as multivariate Gaussians with mean $\mu_{\mathbf{q},y}$ and covariance matrix $\Sigma_{\mathbf{q},y}$.

These parameters are generated (with a possible degree of stochasticity) according to a simple set of conditioning rules. We parametrised the mean generation to simulate two cases. In the first case, the means of homogeneous groups belonging to the same class (i.e., $\mu_{\mathbf{q}_i,y_i}, \mu_{\mathbf{q}_j,y_j} \mid \mathbf{q}_i \neq \mathbf{q}_j \wedge y_i = y_j$) can be forced to be mutually close or even identical and, at the same time, they are enforced to be far from the means of matching homogeneous groups belonging to different classes. This setup is not interesting, since it can lead to (approximately) linearly separable clusters in the Euclidean space X. What is interesting is imposing that the means of corresponding homogeneous groups belonging to different classes (i.e., $\mu_{\mathbf{q}_i,y_i}, \mu_{\mathbf{q}_j,y_j} \mid \mathbf{q}_i = \mathbf{q}_j \wedge y_i \neq y_j$) are mutually close, and at the same time that they are far from the means of the other homogeneous groups belonging to their same class since this setup can create considerable overlaps between the supports of the distributions of different classes. Hence, we used this second setup in all the reported experiments.

The covariance matrices are generated by combining three factors: the spectrum, the volume (i.e., the magnitude of their largest eigenvalue), and the orientation. In particular, the spectrum and the volume concur to determine the "shape" of the Gaussian clouds. A spectrum with eigenvalues that decay quickly to zero will have a more elongated shape, whereas a spectrum with all equal eigenvalues will return spherical clouds. The volume factor can instead determine how "concentrated" a cloud is since it is the magnitude of the maximum eigenvalue. Finally, the orientation implicitly defines the principal components of the distribution.

Once the class *fingerprints* have been created, we need to generate the actual data set. This process is accomplished by repeatedly generating bags. We split the generation

procedure for a single bag into three steps: deciding the class, generating the bag composition, and finally sampling the numerical measurements.

First, a class y is sampled. Then, the bag composition is generated by first sampling the number m of sub-bags, and then sampling a value $\mathbf{q}^{(j)}$ of the categorical variable $\mathbf{q}$ for each $j = 1, \ldots, m$. Finally, the sample of numerical vectors is generated by sampling a fixed number of measurements from $p_{\mathbf{q}^{(j)},y}(\mathbf{x})$.

The parametrisation also allows to set the class probabilities, the bag size distributions, and even the conditional probabilities $p(\mathbf{q} \mid y)$ used to generate bag compositions. In this way we were able to emulate another characteristic of the original Tetra Pak data set, i.e., the fact that it is unbalanced.

In all our experiments, including the one with PLMS data, we considered 4-class classification problems. The interested reader can find the details of the generation procedure and the exact values for the parameters in the GitHub repository associated with the present work.

5.3 Experiments on toy data sets

In the first experiment, we defined $N_Q = 4$ levels of the categorical variable, yielding $N_Y \cdot N_Q = 16$ homogeneous groups. We generated matching homogeneous groups with close or identical means; the shapes of the groups were slightly ellipsoidal and mutually similar, with the main difference coming from the orientation. In this case, the resulting point clouds resulted very overlapped.

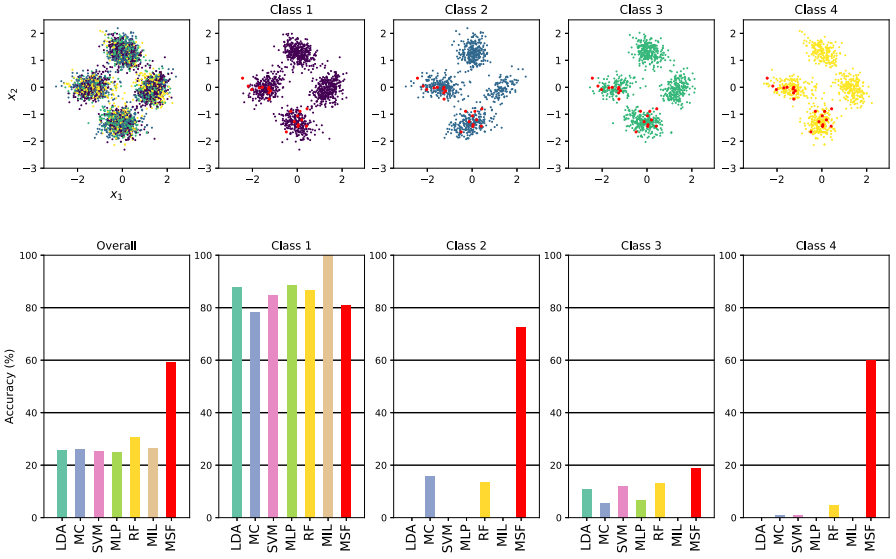


Fig. 10 The setup and experimental results for the first toy data set. Top row: the complete data set (top left image) is partitioned into the class point clouds; the point cloud of a test bag (red points) is overlaid onto the point clouds of the individual classes, which exhibit a mixture-like structure. Bottom row: the performance of the different classification methods is reported, also for the individual classes

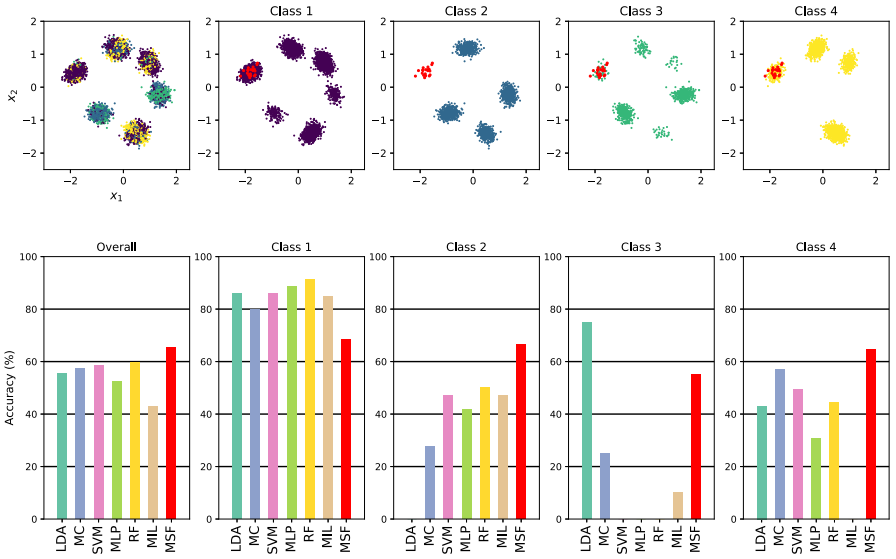


Fig. 11 The setup and experimental results for the second toy data set. Top row: the probability of sampling a bag from Class 1 is 40%, 20% for Class 2, 10% for Class 3, and 30% for Class 4; moreover, some homogeneous groups are extremely unlikely to be sampled when composing bags of Class 2 and Class 4; a test bag (red points) is overlaid to the individual class point clouds. Bottom row: the performance of the different classification methods is reported, also for the individual classes

As can be seen from Fig. 10, the competitor models are prone to overfit the distribution of Class 1 (with the only exception of the RF, which is capable of detecting a weak correlation), whereas the *fingerprint* method is capable of attaining an impressive $\approx 60\%$ accuracy.

In the second experiment, we increased the number of levels of the categorical variable to $N_Q = 6$, for a total number of homogeneous groups of 24. We also unbalanced the data set by setting a slightly higher probability of sampling from classes 1 and 4, and by increasing the average number of sub-bags sampled for each of their bags. These choices added up to globally generate more data points for these two classes. We also unbalanced the conditional probabilities $p(\mathbf{q} \mid y)$ of sampling specific categorical values used during the bag composition stage.

In this case, as can be seen from Fig. 11, all the competitor methods can perform better than random chance (the worst method, MIL, has an accuracy higher than 40%, which is the probability of sampling a bag from Class 1). We ascribe this success to the fact that, since the data set is unbalanced, the overlaps between corresponding homogeneous groups are less severe. Consider a specific cluster of matching homogeneous groups (i.e., one of the eye-detectable clusters in the image at the top left corner of Fig. 11), and the decision surface of a model like an MLP: in this case, the part of the model's decision surface which intersects the support of the cluster distributions is required to discriminate between just two or three classes instead of four, resulting in reduced uncertainty for some of the groups. Note that the SVM, MLP, and RF indeed struggle to determine proper decision surfaces for the underrepresented Class 3 (which

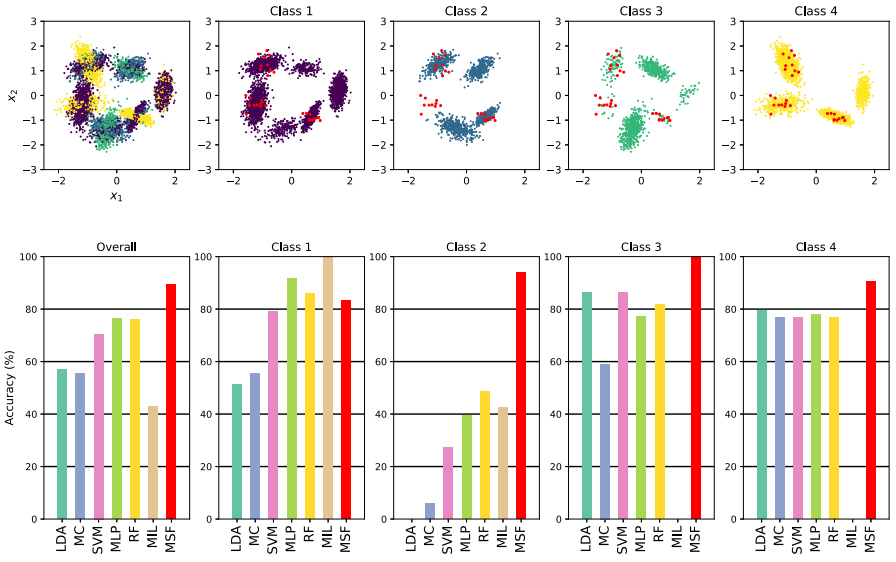


Fig. 12 The setup and experimental results for the third toy data set. Top row: note that the orientation of matching homogeneous groups can greatly differ, although their means remain closer; a test bag (red points) is overlaid to the individual class point clouds. Bottom row: the performance of the different classification methods is reported, also for the individual classes

has a probability of only 10% of being sampled from). Note that the fact that the qualitative factors are unbalanced seems to have favoured the model-based MixtComp approach as well.

In the third and fourth experiments, we modified the second toy data set to investigate how the shape and volume of the homogeneous groups distributions impact performance.

The third data set was generated by enforcing more "elongated" shapes of the Gaussian point clouds (i.e., ellipsoids where the ratios between the longest semi-axis and the shorter semi-axes are larger); nevertheless, note that the volume factors (i.e., the magnitude of the largest eigenvalues) were kept similar.

In this case, probably because the overlaps between matching homogeneous groups are further reduced, the regions where the competitor methods still struggle to discriminate data points are smaller. At the same time, the points which are sampled from a specific class can easily end up having enormous Mahalanobis distances from the means of the homogeneous groups that are not their generating ones. This implies that bags are easier to classify also for the *fingerprint* method. Hence, as depicted in Fig. 12, we observe more correct predictions and overall higher performance for most methods.

In the fourth and last experiment with toy data sets, we restored the shape factors as they were in the second experiment, but explored the effect of changing the volume factors.

Interestingly, while the performance of the competitor methods remained quite satisfying, that of the *fingerprint* method dropped. In particular, as can be seen from

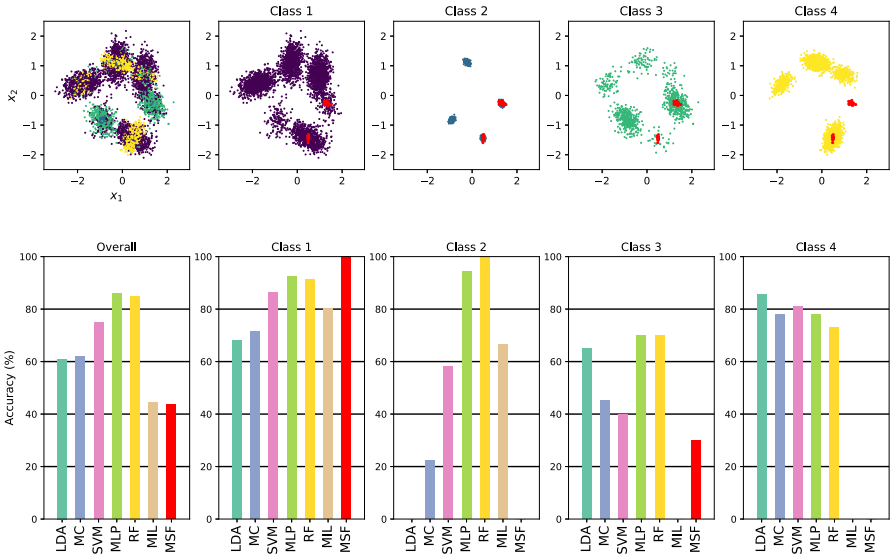


Fig. 13 The setup and experimental results for the fourth toy data set. Top row: note that the homogeneous groups of Class 2 are much more "concentrated" than the matching homogeneous groups of other classes, but their means are also close to those of matching homogeneous groups; a test bag (red points) is overlaid to the individual class point clouds: intuitively, it is easy to assign this bag to Class 2, but the *fingerprint* method consistently fails at classifying such bags. Bottom row: the performance of the different classification methods is reported, also for the individual classes; note that the *fingerprint* method is not capable of classifying correctly bags that belong to Class 2 and Class 4

Fig. 14 Confusion matrix of the *multi-stage fingerprint* method applied to the fourth toy data set

Ground truth \ Predicted	1	2	3	4	Total
1	81 40.50%	0 0.00%	0 0.00%	0 0.00%	100.00% 81 0.00%
2	17 8.50%	0 0.00%	19 9.50%	0 0.00%	0.00% 36 100.00%
3	14 7.00%	0 0.00%	6 3.00%	0 0.00%	30.00% 20 70.00%
4	32 16.00%	0 0.00%	31 15.50%	0 0.00%	0.00% 63 100.00%

Fig. 13, we observe that the classification performance for Class 2 and Class 4 (the ones whose homogeneous groups are more "concentrated") suffers the most. Why does this happen?

Looking at the confusion matrix reported in Fig. 14, we see that most of the bags in Class 2 are misclassified as belonging to Class 1 or Class 3. Noticing that the means of its homogeneous groups are very close to the means of the (larger) matching homoge-

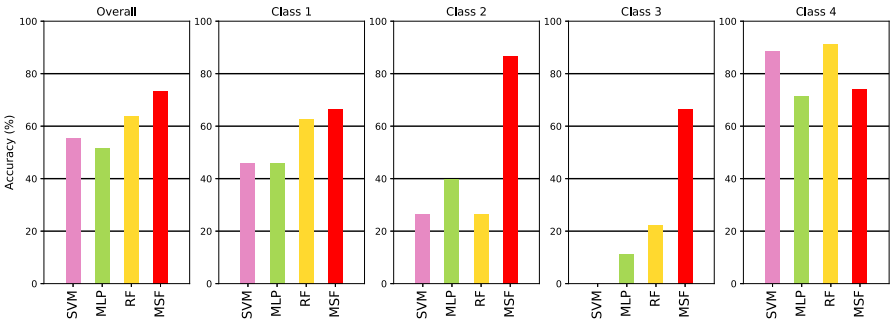


Fig. 15 Performance of the competitor methods (SVM, MLP, RF) and the *fingerprint* method on the mixed-type PLMS data. Some of the methods that we applied to the toy examples could not be trained on the PLMS data set: the fact that the training set is unbalanced led to computational issues for MIL (namely, the software reported computationally singular matrices during optimisation), whereas MixtComp failed at computing the likelihood scores for certain test points since the levels of their categorical variables had never been observed in the training set (and the corresponding multinomial probabilities thus estimated zero probability for those levels)

neous groups of Class 1 and Class 3, we formulated the following hypothesis. Since the first criterion on the *fingerprint* methods is based on the Mahalanobis distance, the points sampled from a homogeneous group in Class 2 or Class 4 are much farther (under the metric defined by the associated Mahalanobis distance) from its mean than from the means of the matching homogeneous groups of Class 1 or Class 3 (under the metrics defined by the corresponding Mahalanobis distances). Since this fact is likely to hold for most of the homogeneous groups in Class 2 and Class 4, a sample of points from one of these classes is likely never classfied as belonging to it. At the same time, standard machine learning methods like SVM, MLP, and RF show good performance at classifying bags from Class 2. We interpret this as the fact that their learning algorithms are capable of drawing "efficient" decision surfaces around the homogeneous groups of Class 2. By this use of the term, we mean that these decision surfaces are likely very tight around the clusters containing most of the points of the homogeneous groups of Class 2: they "accept" to misclassify a few points which are sampled from matching homogeneous groups of other classes in these localised regions, but they are then able to compensate for these errors by classifying most of the remaining points in the test bags correctly (which, if sampled from other classes, are unlikely to fall in these tiny regions).

5.4 Experimenting with the PLMS data set

We conclude this section by reporting and commenting on the results of the experiment on the industrial PLMS data set. Recall that the test set for this problem is highly unbalanced in favour of Segment 1 and Segment 4: over 83 test bags, 24 are labelled as Segment 1 and 35 are labelled as Segment 4, 15 as Segment 2, and only 9 as Segment 3. Thus, the baseline performance would be $\approx 40\%$ classification accuracy (by always predicting Segment 4).

In this experiment, the *multi-stage fingerprint* method far outmatched the competitor methods: the gap with respect to the best competitor model (a random forest achieving an accuracy of 63.86%) amounts to almost ten accuracy points (the *multi-stage fingerprint* method achieves an accuracy of 73.49%). The accuracies shown in Fig. 15 reveal that this gap is mostly due to the fact that the *fingerprint* method classifies correctly most plants of the under-represented Segment 2 and Segment 3 (classes 2 and 3), which are usually missed by the competitor methods. As shown be the first and second toy examples in the previous sub-section, this might be because the matching homogeneous groups of the corresponding segments are highly overlapped, and only the evaluation of higher-order statistics of the homogeneous groups (considered as wholes) can allow to discriminate them.

By design, the *multi-stage fingerprint* method had to be able to classify all the bags in a given test set: we did not allow the scenario of leaving unclassified bags. The classification of all the test bags can be achieved at the last stage of the method when all the qualitative factors have been dropped. As anticipated in Sect. 4, Fig. 16 shows that the use of less qualitative criteria at later stages makes the classifier less precise on the remaining test bags. In our experiments, the initial definition of the homogeneous groups involved all the five available factors. In such circumstances, the test set contained several bags whose *fingerprints* did not have any match with the class *fingerprints* (10.8% of the total test individuals). After the first stage, 68.7% of the test bags were correctly classified (remarkably, 77.0% of the bags whose *fingerprint* had a non-empty match with at least one class *fingerprint* were correctly classified). At the second stage, the geographic cluster factor Q_5 was dropped, and 72.3% of the total test bags were correctly classified (77.9% of the test bags which could be classified), while the class of 7.2% of the bags (six plants) remained unknown. Note that all the newly classified plants were correctly classified. At the third stage, only one of the three newly classified bags was correctly classified. None of the mergings that were possible at the fourth and fifth stages allowed to classify other plants. At the last stage, when all the factors had been dropped, none of the three remaining bags was correctly classified: as we expected by merging multiple homogeneous groups, the precision is greatly reduced with respect to the previous stages.

We recall from Sect. 3 that being unbalanced and heterogeneous are key characteristics of the PLMS data set. Classification methods for unbalanced data sets can be classified in two families: re-sampling approaches and model-based approaches. Re-sampling approaches aim at balancing the available data by under-sampling the majority classes to match the size of the minority classes, over-sampling the minority classes to match the size of the majority classes, or combinations of these two techniques. We refer the interested reader to Liu et al. (2006), Khoshgoftaar et al. (2007) for additional details. On the other hand, model-based approaches aim at developing algorithms that can directly cope with the fact that the available data is unbalanced. Some examples include weighing more the classification errors for points belonging to the minority classes during training, *rare events modelling*, and anomaly detection. The *fingerprint* method takes a model-based approach to the unbalance problem. Note that the definition of acceptance function (20) has the intrinsic advantage of reducing the impact of the data set unbalance: if a homogeneous group $\mathcal{R}_{\tilde{q},\bar{y}}$ is acceptable,

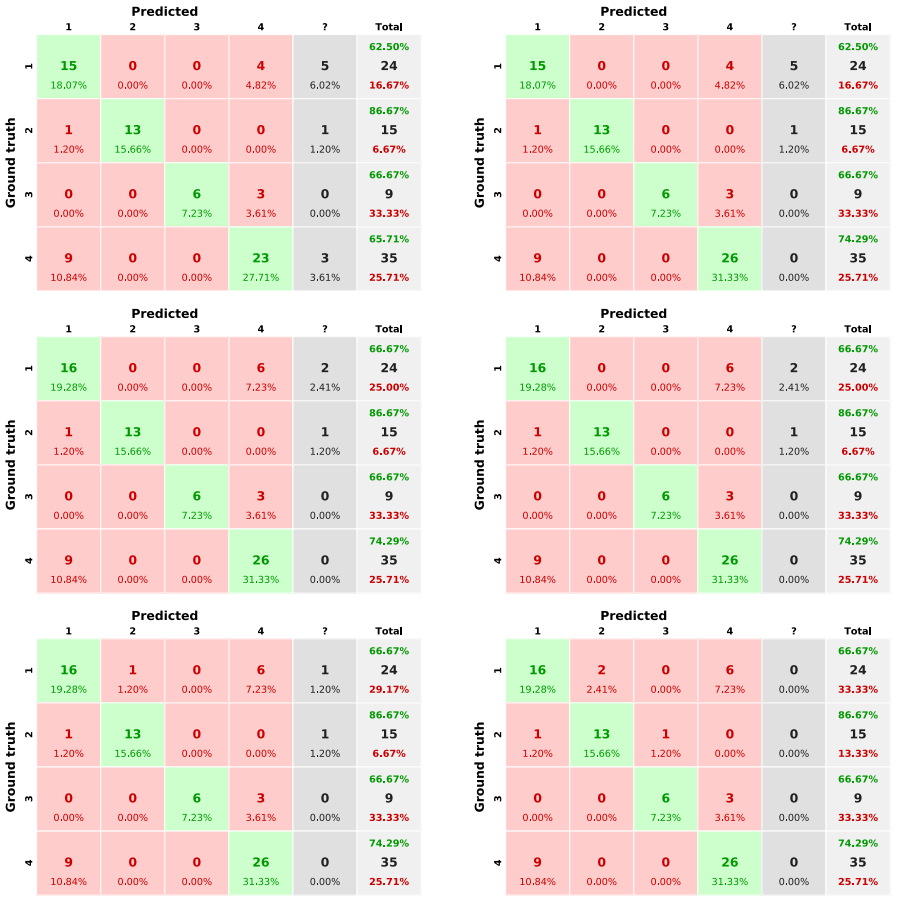


Fig. 16 Confusion matrices of different stages of the *multi-stage fingerprint* method applied to the PLMS data set. The six figures represent, in lexicographic order, the results obtained by dropping one categorical factor at the time, starting from all factors (top left) and concluding with no factors (bottom right)

the corresponding sample statistics are independent of the number of points that it contains (although more points might make the statistics more precise).

A reader might anyway wonder why we did not explore re-sampling approaches in the present study. We hypothesised that straightforward over-sampling the minority classes by using interpolation or nearest-neighbours techniques might alter the mixture-like structure of the PLMS data set; some support for this hypothesis comes from the degradation in performance that we witness when merging different homogeneous groups in the multi-stage version of the method.

6 Conclusions

We have presented the *fingerprint* method, a new classifier that can process variable-length records (*bags*) of mixed-type measurements. Inspired by and applied to a real-world industrial problem, it performed remarkably better than standard statistical and machine learning methods (linear discriminant analysis, support vector machines, multi-layer perceptrons, random forests), and also than methods specifically designed to handle mixed-type and bagged data (mixture composer, multiple instance learning).

The specific criteria chosen to approximate the comparison between the bag *fingerprint* distributions $\{p_{\bar{k},\mathbf{q}}(\mathbf{x})\}_{\mathbf{q}\in Q}$ and the class *fingerprint* distributions $\{p_{\mathbf{q},\bar{y}}(\mathbf{x})\}_{\mathbf{q}\in Q}$ (the Mahalanobis distances and the principal components) are simple heuristics conceived according to specific properties of the original industrial data set but proved effective in both synthetic and real-world experiments.

The *fingerprint* method can achieve remarkable accuracy levels even when matching generating distributions (i.e., distributions $p(\mathbf{x} \mid \mathbf{q}_i, y_j)$, $p(\mathbf{x} \mid \mathbf{q}_i, y_j) \mid \mathbf{q}_1 = \mathbf{q}_2 \wedge y_1 \neq y_2$) have highly-overlapping supports, cases where the competitor methods completely fail (i.e., their accuracy is the same of a random guess) or can detect only weak correlations (see the first toy example). On the other hand, the choice to use the Mahalanobis distance as a classification criterion might sometimes result in the method to fail. In particular, this weakness emerged when matching homogeneous groups had similar means but whose covariance matrices had maximum eigenvalues which differed by orders of magnitude (see the last toy example).

There are some directions in which the *fingerprint* method can be further developed. The chosen heuristics to compare the bag distributions $\{p_{\bar{k},\mathbf{q}}(\mathbf{x})\}_{q\in Q}$ to the class *fingerprint* distributions $\{p_{\mathbf{q},\bar{y}}(\mathbf{x})\}_{\mathbf{q}\in Q}$ can be improved. A possible investigation approach can be briefly outlined here. Once the match $\mathcal{Q}_{\bar{k},\bar{y}}$ has determined the comparisons bag-class that have to take place, one could apply a series of multivariate two-sample Kolmogorov-Smirnov tests: the sample distribution associated with a bag homogeneous group can be compared to the sample distributions of matching class homogeneous groups, and the class whose sample distribution is most likely to be obtained from the same ideal population can be detected. This is expected to provide a more robust theoretical substitute for the proposed heuristics. More general questions regarding the handling of different matches $\mathcal{Q}_{\bar{k},y_i} \neq \mathcal{Q}_{\bar{k},y_j}$, rules for aggregating the (possibly probabilistic) outcomes of the comparisons at the homogeneous group level into a bag level response, and the related computational challenges seem non-trivial.

At the moment, the determination of the drop sequence for qualitative factors in the *multi-stage fingerprint* method is based on a greedy heuristic prescription (classifying the largest number of test bags by removing the least number of factors). The rationale for this choice is that the chosen geometric comparison criteria are more reliable on uni-modal mixture components. This approach has performed well so far, but there are opportunities for implementing a more advanced feature selection procedure, possibly related to hierarchical learning.

Going back to the original industrial problem, there is one characteristic of the PLMS records which has not been exploited in this research. The measurements collected from a single production line are indexed over time. In the scope of the broader

study of which this work represents a part, we detected and analysed some interesting temporal patterns (e.g., seasonal variations, different inventory management strategies) in the PMSD set. These patterns are possibly reflected in the PLMS data, but this could also introduce additional complications in the formalisation and experimental activities. For instance, the fact that in the southern and the northern hemisphere some seasonality-related patterns are shifted might force to design dedicated "alignment" procedures for time series. Hence, our approach to classification was developed around bags of measurements instead of time series. Although interesting, such an extension goes beyond the scope of the present paper.

The formalisation of the *fingerprint* method we provided in Sect. 2 is general enough to suggest that the method could be applied to other mixed-type bagged data sets. As a first example, anomaly detection can be considered. The performance parameters of a given system in different operational regimes can differ quite substantially. Imagine a jet turbofan, with its operational regimes of takeoff, cruise, and landing (interpreted as the levels of a categorical variable): the *fingerprint* of the machine can be exploited to distinguish normal and anomalous states of the system (interpreted as classes). From this viewpoint, the *fingerprint* of the system can be interpreted as the result of a physical diffusion process which starts from a set of centroids (the initial conditions) that describe different ideal states.

Wearable devices for health monitoring might provide another example. In this case, a collection of sensors placed on different body parts (head, wrists, ankles) and measuring different biological signals (EEG, ECG, O2 sats) can be represented as a bag composition. The *fingerprint* of such a system can be used to monitor the health state of an individual.

Acknowledgements The results presented in this work were achieved during a joint research project between Tetra Pak Packaging Solutions S.p.A., the Polytechnic University of Turin and the University of Modena and Reggio Emilia. The authors wish to thank Gabriele Molari (Tetra Pak) for having made its realisation possible and for having obtained Tetra Pak's permission to access this data set and publish the related results. Marco Prato has been partially funded by the ECSEL JU programme under the PRYSTINE Project Grant Agreement No. 783190, and the UNIMORE FAR 2019 "Risk assessment in the EU: new indices based on machine learning methods". The Italian INdAM GNSAGA and GNCS research groups are also kindly acknowledged. Finally, the authors want to thank professor Gian Paolo Leonardi, Menelaos Kanakis, and the anonymous reviewers for their constructive and insightful comments, which greatly contributed to the improvement of the manuscript.

Funding Open Access funding provided by ETH Zurich.

Declarations

Software and data The R, Python, and MATLAB scripts used to generate the toy data sets and perform the experiments are available on GitHub (https://github.com/spallanzanimatteo/Fingerprint). The industrial data used for the experiments is property of Tetra Pak Packaging Solutions S.p.A., and access must be authorised by the company.

Conflict of interest The authors declare that they have no conflict of interest.

A: Algorithms

Algorithm 1 Given an acceptance function $a_{\mathbf{t}}$, split a bag $\mathcal{B}$ (list of structures) into its *fingerprint* $\mathcal{F}$ (dictionary of lists).

Function `compute_fingerprint`
Input $a_{\mathbf{t}}, \mathcal{B}$
Output $\mathcal{F}$

$\mathcal{Q} \leftarrow list()$
$\mathcal{F} \leftarrow dict()$
for $b \in \mathcal{B}$ **do**
  $\mathbf{q} \leftarrow b.\mathbf{q}$
  if $\mathbf{q} \notin \mathcal{Q}$ **then**
    $\mathcal{Q}.append(\mathbf{q})$
    $\mathcal{F}[\mathbf{q}] \leftarrow list()$
  end if
  $\mathcal{F}[\mathbf{q}].append(b.\mathbf{x})$
end for
for $\mathbf{q} \in \mathcal{Q}$ **do**
  if $a_{\mathbf{t}}(\mathcal{F}[\mathbf{q}]) = 0$ **then**
    `delete`$(\mathcal{F}, \mathbf{q})$
  end if
end for
return $\mathcal{F}$

Algorithm 2 Given the class bag $\mathcal{B}_{\bar{y}}$ (list of structures), compute the statistics $\mathcal{T}_{\bar{y}}$ of its *fingerprint* (dictionary of structures).

Input $a_{\mathbf{t}}, \mathcal{B}_{\bar{y}}, n_{pcs}$
Output $\mathcal{T}_{\bar{y}}$

$\mathcal{F}_{\bar{y}} \leftarrow$ `compute_fingerprint`$(a_{\mathbf{t}}, \mathcal{B}_{\bar{y}})$
$\mathcal{T}_{\bar{y}} \leftarrow dict()$
for $\mathbf{q} \in \mathcal{F}_{\bar{y}}.keys()$ **do**
  $\mathcal{R}_{\mathbf{q},\bar{y}} \leftarrow \mathcal{F}_{\bar{y}}[\mathbf{q}]$
  $\mathbf{m}_{\mathbf{q},\bar{y}} \leftarrow mean(\mathcal{R}_{\mathbf{q},\bar{y}})$
  $S_{\mathbf{q},\bar{y}} \leftarrow cov(\mathcal{R}_{\mathbf{q},\bar{y}})$
  $m_{\mathbf{q},\bar{y}} \leftarrow mean(\{\sqrt{(\mathbf{x}-\mathbf{m}_{\mathbf{q},\bar{y}})' S^{-1}_{\mathbf{q},\bar{y}} (\mathbf{x}-\mathbf{m}_{\mathbf{q},\bar{y}})} \mid \mathbf{x} \in \mathcal{R}_{\mathbf{q},\bar{y}}\})$
  $s_{\mathbf{q},\bar{y}} \leftarrow std(\{\sqrt{(\mathbf{x}-\mathbf{m}_{\mathbf{q},\bar{y}})' S^{-1}_{\mathbf{q},\bar{y}} (\mathbf{x}-\mathbf{m}_{\mathbf{q},\bar{y}})} \mid \mathbf{x} \in \mathcal{R}_{\mathbf{q},\bar{y}}\})$
  $N_{\mathbf{q},\bar{y}} \leftarrow \{\boldsymbol{v}_{\mathbf{q},\bar{y},i}\}_{i=1,\ldots,n_{pcs}} = PCA(S_{\mathbf{q},\bar{y}}, n_{pcs})$
  $stats \leftarrow (\mathbf{m}_{\mathbf{q},\bar{y}}, S_{\mathbf{q},\bar{y}}, m_{\mathbf{q},\bar{y}}, s_{\mathbf{q},\bar{y}}, N_{\mathbf{q},\bar{y}})$
  $\mathcal{T}_{\bar{y}}[\mathbf{q}] \leftarrow stats$
end for
return $\mathcal{T}_{\bar{y}}$

Algorithm 3 Assign a test bag $\mathcal{B}_{\bar{k}}$ (list of structures) to a class y^* (enumerated; -1 represents "unknown"), given the trained statistics $\mathcal{T}$ of the class *fingerprints* (dictionary of dictionary of structures).

Input: $a_{\mathbf{t}}, \mathcal{B}_{\bar{k}}, \mathcal{T}, n_{pcs}, \tau$
Output: y^*

$\mathcal{F}_{\bar{k}} \leftarrow$ `compute_fingerprint`$(a_{\mathbf{t}}, \mathcal{B}_{\bar{k}})$
$\mathcal{Q}_{\bar{k}} \leftarrow set(\mathcal{F}_{\bar{k}}.keys())$
$\mathcal{M} \leftarrow dict()$
for $y \in Y$ **do**
  $\mathcal{Q}_y \leftarrow set(\mathcal{T}[y].keys())$
  $\mathcal{Q}_{\bar{k},y} \leftarrow \mathcal{Q}_{\bar{k}} \cap \mathcal{Q}_y$
  if $\mathcal{Q}_{\bar{k},y} = \emptyset$ **then**
    $\mathcal{M}[y] \leftarrow$ `NULL`
  else
    $\alpha \leftarrow list()$
    $\beta \leftarrow list()$
    for $\mathbf{q} \in \mathcal{Q}_{\bar{k},y}$ **do**
      $\mathcal{R}_{\bar{k},\mathbf{q}} \leftarrow \mathcal{F}_{\bar{k}}[\mathbf{q}]$
      $(\mathbf{m}_{\mathbf{q},y}, S_{\mathbf{q},y}, m_{\mathbf{q},y}, s_{\mathbf{q},y}, N_{\mathbf{q},y}) \leftarrow \mathcal{T}[y][\mathbf{q}]$
      $m_{\bar{k},\mathbf{q},y} \leftarrow mean(\{\sqrt{(\mathbf{x}-\mathbf{m}_{\mathbf{q},y})' S^{-1}_{\mathbf{q},y} (\mathbf{x}-\mathbf{m}_{\mathbf{q},y})} \,|\, \mathbf{x} \in \mathcal{R}_{\bar{k},\mathbf{q}}\})$
      $\alpha.append((m_{\bar{k},\mathbf{q},y} - m_{\mathbf{q},y})/s_{\mathbf{q},y})$
      $S_{\bar{k},\mathbf{q}} \leftarrow cov(\mathcal{R}_{\bar{k},\mathbf{q}})$
      $N_{\bar{k},\mathbf{q}} \leftarrow \{\boldsymbol{\nu}_{\bar{k},\mathbf{q},i}\}_{i=1,\dots,n_{pcs}} = PCA(S_{\bar{k},\mathbf{q}}, n_{pcs})$
      $\beta.append(mean(\{|\langle \boldsymbol{\nu}_{\bar{k},\mathbf{q},i}, \boldsymbol{\nu}_{\mathbf{q},y,i} \rangle|\}_{i=1,\dots,n_{pcs}}))$
    end for
    $\mathcal{M}[y].\alpha \leftarrow mean(\alpha)$
    $\mathcal{M}[y].\beta \leftarrow mean(\beta)$
  end if
end for
$Y_{\bar{k}} \leftarrow \{y \in \mathcal{M}.keys() \,|\, \mathcal{M}[y] \neq$ `NULL`$\}$
if $Y_{\bar{k}} = \emptyset$ **then**
  $y^* \leftarrow -1$
else
  if $\#(Y_{\bar{k}}) = 1$ **then**
    $y^* \leftarrow y \,|\, Y_{\bar{k}} = \{y\}$
  else
    $y_A \leftarrow \underset{y \in Y_{\bar{k}}}{arg\,min} \mathcal{M}[y].\alpha$
    $y_B \leftarrow \underset{y \in Y_{\bar{k}} \setminus \{y_A\}}{arg\,min} \mathcal{M}[y].\alpha$
    if $\mathcal{M}[y_A].\alpha / \mathcal{M}[y_B].\alpha \geq 1 - \tau$ **then**
      if $\mathcal{M}[y_B].\beta > \mathcal{M}[y_A].\beta$ **then**
        $y^* \leftarrow y_B$
      else
        $y^* \leftarrow y_A$
      end if
    else
      $y^* \leftarrow y_A$
    end if
  end if
end if
return y^*

B: Experimental results on the mixed-type PLMS data set

(a) SVM vector-level performance.

(b) SVM bag-level performance.

(c) MLP vector-level performance.

(d) MLP bag-level performance.

(e) RF vector-level performance.

(f) RF bag-level performance.

Fig. 17 Confusion matrices of SVM (top row), MLP (central row) and RF (bottom row) applied to the mixed-type PLMS data

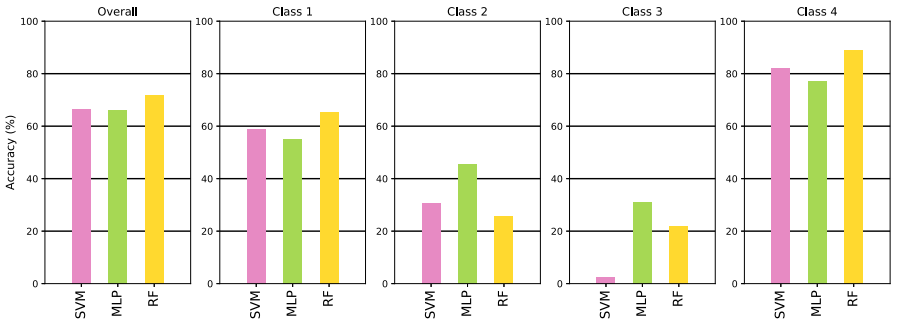


Fig. 18 Vector-level performance of the competitor methods (SVM, MLP, RF) on the mixed-type PLMS data

References

Abdullin A, Nasraoui O (2012) Clustering heterogeneous data sets. In: Proceedings of the 2012 Eighth Latin American Web Congress, IEEE

Ahmad A, Dey L (2007) A *k*-mean clustering algorithm for mixed numeric and categorical data. Data Knowl Eng 63:503–527

Andrews S, Tsochantaridis I, Hofmann T (2003) Support vector machines for multiple-instance learning. In: Advances in Neural Information Processing Systems 15, Neural Information Processing Systems (NIPS)

Biernacki C, Deregnaucourt T, Kubicki V (2015) Model-based clustering with mixed/missing data using the new software MixtComp. In: 8th International Conference of the ERCIM WG on Computational and Methodological Statistics (CMStatistics), ERCIM

Blizard WD (1991) The development of multiset theory. Modern Logic 1:319–352

Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: 5th Annual Workshop on Computational Learning Theory, ACM

Breiman L (2001) Random forests. Mach Learn 45:5–32

Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines and other Kernel-based learning methods. Cambridge University Press, Cambridge

Dietterich TG, Lathrop LH, Lozano-Pérez T (1997) Solving the multiple instance problem with axis-parallel rectangles. Artif Intell 89:31–71

Doran G, Ray S (2014) A theoretical and empirical analysis of support vector machine methods for multiple-instance classification. Mach Learn 97:79–102

Hae-Sang P, Chi-Hyuck J (2009) A simple and fast algorithm for *k*-medoids clustering. Expert Syst Appl 36:3336–3341

Hinton GE (2007) Learning multiple layers of representation. Trends Cognit Sci 11:428–434

Khoshgoftaar TM, Golawala M, van Hulse J (2007) An empirical study of learning from imbalanced data using random forest. In: 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), IEEE

Kingma DP, Ba JL (2014) Adam: a method for stochastic optimization. CoRR arXiv:1412.6980v9

Liu Y, An A, Huang X (2006) Boosting prediction accuracy on imbalanced datasets with SVM ensembles. In: Advances in Knowledge Discovery and Data Mining, Springer

Mahalanobis PC (1936) On the generalized distance in statistics. Proc National Instit Sci India 2:49–55

McLachlan GJ (1992) Discriminant analysis and statistical pattern recognition. Wiley, New Jersey

Quinlan JR (1986) Induction of decision trees. Mach Learn 1:81–106

Rencher AC (2003) Methods of multivariate analysis. Wiley, New Jersey

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536

Sandhya H, Sandhya PV (2015) *k*-medoid clustering for heterogeneous datasets. Proc Computer Sci 70:226–237

Schapire RE (1990) The strength of weak learnability. Mach Learn 5:197–227

Zanni L, Serafini T, Zanghirati G (2006) Parallel software for training large scale support vector machines on multiprocessor systems. J Mach Learn Res 7:1467–1492
Zhang L, Zhang B (2004) The quotient space theory of problem solving. Fundam Inf 59:287–298
Zhang L, Zhang B (2006) Hierarchical machine learning – a learning methodology inspired by human intelligence. In: Rough Sets and Knowledge Technology, Springer