Minimally Informed Linear Discriminant Analysis: training an LDA model with unlabelled data

Nicolas Heintz, Tom Francart, and Alexander Bertrand, *Senior Member, IEEE*

***Abstract*—Linear Discriminant Analysis (LDA) is one of the oldest and most popular linear methods for supervised classification problems. In this paper, we demonstrate that it is possible to compute the exact projection vector from LDA models based on unlabelled data, if some minimal prior information is available. More precisely, we show that only one of the following three pieces of information is actually sufficient to compute the LDA projection vector if only unlabelled data are available: (1) the class average of one of the two classes, (2) the difference between both class averages (up to a scaling), or (3) the class covariance matrices (up to a scaling). These theoretical results are validated in numerical experiments, demonstrating that this minimally informed Linear Discriminant Analysis (MILDA) model closely matches the performance of a supervised LDA model. Furthermore, we show that the MILDA projection vector can be computed in a closed form with a computational cost comparable to LDA and is able to quickly adapt to non-stationary data, making it well-suited to use as an adaptive classifier.**

***Index Terms*—unsupervised learning, classification, linear discriminant analysis, adaptive learning**

I. Introduction

Fisher's Linear Discriminant Analysis (LDA) model is a standard algorithm in data science for linear classification [1]. Its goal is to find the optimal linear projection of the data such that two classes are maximally separated. Despite the emergence of more advanced techniques, LDA remains a very popular data analysis tool in all scientific fields [2]–[8] because it is easy to understand, cheap to compute, and has an elegant and interpretable formulation. Its low computational complexity also makes it a popular tool for signal processing, in particular in real-time or low-resource environments [4], [5].

In his original LDA paper, Fisher proposed a transformation that maximises the distance between class averages and minimises the variance within each class [9]. Computing Fisher's LDA model requires four (or two) parameters: the two class averages (or their difference) and the class covariance matrices of both classes (or their sum). These parameters can either be known via prior knowledge or estimated using labelled data. Herein lies one of the main drawbacks of LDA (and any other supervised model): as soon as one of these parameters is unknown, a labelled dataset is required to train the LDA model. For many applications, this requires manual labelling by experts, which is time-consuming and costly. Furthermore, when a problem is non-stationary, i.e. when the statistics of the data change over time, it is often important to regularly re-train the model, which only exacerbates the aforementioned issues.

Alternatively, one can resort to unsupervised classification algorithms that attempt to classify given data without requiring labels. Such unsupervised classification problems are generally tackled with clustering algorithms such as, e.g., Gaussian Mixture Models (GMMs) [10], K-means [11], and its extension K-means++ [12]. However, the modelling power and robustness of LDA, in particular in high-dimensional spaces, have motivated the incorporation of LDA in such unsupervised frameworks. A popular approach is to iteratively estimate a set of pseudo-labels using some unsupervised clustering method and then train the LDA model with these labels [13]–[15].

Another solution to the labelling problem is to only label a limited amount of samples and then train the model on a small number of labelled samples and a large number of unlabelled samples, which is called semi-supervised learning. For semi-supervised LDA, the small number of labelled samples can then give a rough indication of where the two classes are supposed to be located. This information can then be merged with a much better description of the general geometry of the data delivered by the vast amount of unlabelled data through some objective function to provide an estimation of the LDA projection, as done in [16], [17].

All these proposed methods require (pseudo-)labels in order to compute Fisher's LDA projections. However, there exist many scenarios where some of the relevant ground-truth statistics of the data are known (or can be estimated without labels), but not all of them:

- Only the **class average** of a **single class** is known: This could occur, e.g., when there is a (partial) model for one of the classes, but not for the other. One example could be a detection problem where the noise is known to be zero-mean, but where its covariance structure is unknown. In the absence of a detection target (class 0), the mean vector is then known to be 0. Another example is hypothesis testing, where the mean of the null hypothesis distribution is assumed to be known.
- Only the **relative difference** between the **class averages** is known (up to a scaling): This is relevant when it is known in what direction the features are expected to move when shifting from one class to the other, without

N. Heintz and A. Bertrand are with KU Leuven, Department of Electrical Engineering (ESAT), STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics, Belgium.

N. Heintz and T. Francart are with KU Leuven, Department of Neurosciences, ExpORL, Belgium.

This research is funded by Aspirant Grant 1S31522N (for N. Heintz) from the Research Foundation - Flanders (FWO), the Research Foundation - Flanders (FWO) project No G0A4918N and G081722N, the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 802895 and grant agreement No 637424), and the Flemish Government (AI Research Program). The scientific responsibility is assumed by its authors.

N. Heintz, T. Francart and A. Bertrand are also affiliated with Leuven.AI - KU Leuven institute for AI, Belgium.

knowing how large the change is expected to be, what the exact class averages are or what variance is expected. This is the case, for example, when a binary amplitude-modulated signal must be detected with a sensor array. The steering vector of this signal is often known up to an unknown scalar, while there can simultaneously be ambiguity in the amplitude of the source signal.

- Only the **covariances** for each of the **two classes** are known (up to a scaling): Such a scenario is common when, e.g., the noise covariance statistics are known (possibly up to an unknown calibration factor). This also includes the common case where the noise is independent and identically distributed across the feature space.

In general, the availability of one such ground-truth statistic is not enough to train a classifier from unlabelled data, in which case the remaining statistics still need to be estimated from labelled data. One exception is the last case, with the additional assumption that the classes have a Gaussian distribution and that they are sufficiently separated. In this case, it is enough to know the class covariance matrices to compute LDA without labels [18], [19].

In this paper, we present a mathematical formulation that actually makes it possible to compute the LDA projection with only *one* of the aforementioned ground-truth statistics, which we refer to as 'minimally informed' LDA (MILDA). In contrast to [18], [19], it does not assume Gaussian class distributions, does not require a good class separability and is applicable if any of the aforementioned ground-truth statistics are known.

The proposed MILDA framework still results in closed-form expressions, which are elegant and cheap to compute. This is in line with the main advantages of the standard LDA. MILDA optimises a slightly more general objective function than LDA, and has the original LDA objective as a special case when either the classes are balanced or if the homoscedasticity assumption is satisfied. However, even if neither of these assumptions hold, MILDA empirically still achieves similar accuracies as LDA. Furthermore, MILDA becomes more robust to estimation errors in the ground-truth statistic when the classes are less separated or imbalanced. This is in stark contrast with most unsupervised clustering methods. Up to our knowledge, such a result has not been described.

The aforementioned properties make MILDA especially suited for use in adaptive modelling. Typical examples are, e.g., classifying a non-stationary signal with stationary noise or vice versa. The stationary statistics can then be estimated and used as ground-truth statistics, while MILDA automatically adapts to the non-stationary statistics.

The outline of the paper is as follows. In Section II, we describe the main mathematical framework behind the MILDA model. In Section III we provide a theoretical analysis of the sensitivity of MILDA to errors in the used ground-truth statistic. In Section IV, we evaluate MILDA in numerical experiments and demonstrate its performance on a target signal detection task and on non-stationary data. The computational cost of MILDA is briefly discussed in Section V. Finally, in Section VI, we summarise the main conclusions from this paper and discuss possibilities for future work.

As a notational convention, we use regular letters (a) to denote scalars, bold letters ($\mathbf{a}$) for vectors, capital letters (A) for constants and matrices (depending on the context) and Greek letters (α) for parameters and latent scalars to shorten equations.

II. Minimally informed LDA (MILDA)

In this section, we will first briefly review the main mathematical framework of LDA as proposed by Fisher [9]. In Section II-B, we show how the same projection can be found with the temporary (and overly strict) assumption that the two class averages are proportionate to each other. This generates an intermediate result that will be needed in the derivation of MILDA. Since this assumption of proportionality is unrealistic in real-life applications, we show in Section II-C how this assumption can be dropped when only a single ground-truth statistic is known. No other assumptions on the data are necessary to prove that the MILDA and LDA projections are equivalent.

A. Linear Discriminant Analysis

Let $\{\mathbf{x}_i\}_i$ with $\mathbf{x}_i \in \mathbb{R}^D$ be a set of D-variate samples, where each sample belongs to one of two classes. The goal of Linear Discriminant Analysis (LDA) is to find a projection vector $\mathbf{w} \in \mathbb{R}^D$ that maximally separates two classes after projecting each data point onto $\mathbf{w}$. An intuitive way to do this is by searching for the projection that maximises the distance between the class averages and minimises the variance within each class. By writing these two joint objectives as a ratio, Fisher proposed the following objective:

$$\arg\max_{\mathbf{w}} \frac{\mathbf{w}^\top (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)}{\mathbf{w}^\top (\Sigma_+ + \Sigma_-)\,\mathbf{w}}, \tag{1}$$

where $\boldsymbol{\mu}_+$, $\boldsymbol{\mu}_-$ are the class averages and Σ_+, Σ_- are the covariance matrices for each class [9]. If a labelled set of samples $\mathbf{x}_i$ is available, then the class averages and class covariances can be estimated as[1]:

$$\boldsymbol{\mu}_\pm \triangleq \frac{1}{N_\pm} \sum_{i \in \mathcal{C}_\pm} \mathbf{x}_i \tag{2}$$

$$\Sigma_\pm \triangleq \frac{1}{N_\pm} \sum_{i \in \mathcal{C}_\pm} (\mathbf{x}_i - \boldsymbol{\mu}_\pm)(\mathbf{x}_i - \boldsymbol{\mu}_\pm)^\top, \tag{3}$$

where $\mathcal{C}_+$ and $\mathcal{C}_-$ contain the sample indices of the positive and negative class, respectively, with cardinalities $N_+ = |\,\mathcal{C}_+\,|$ and $N_- = |\,\mathcal{C}_-\,|$. The solution of (1) is then given by:

$$\mathbf{w}^* \propto (\Sigma_+ + \Sigma_-)^{-1} (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-), \tag{4}$$

assuming $\Sigma_+ + \Sigma_-$ is full rank[2] [9]. Here we use the notation $\propto$ to denote that the left- and right-hand sides are equal up to an arbitrary non-zero scaling.

[1] We use the biased covariance estimator throughout this paper for notational convenience.

[2] If the summed covariance matrix is not full rank, a dimensionality reduction should be performed first, e.g., via principal component analysis.

It was later shown that this solution minimises the classification error if both classes have a Gaussian distribution and if they are homoscedastic ($\Sigma_+ = \Sigma_-$), i.e., no other classifier can result in better classification performance [1]. In this case, $\mathbf{w} \propto \Sigma^{-1}(\boldsymbol{\mu}_+ \text{-} \boldsymbol{\mu}_-)$, with $\Sigma = \Sigma_+ = \Sigma_-$. Confusingly, this solution is often also referred to as *Linear Discriminant Analysis*, but it is an edge case of the more general LDA formulation provided by Fisher. We note that Gaussianity and homoscedasticity should not be viewed as limiting assumptions in order to use LDA, i.e., without these assumptions, the LDA objective (1) remains relevant and often results in a good classifier [1], [2].

In this paper, we will show that MILDA provides a projection vector that optimises a slightly more general objective (compared to (1)):

$$\arg\max_{\mathbf{w}} \frac{\mathbf{w}^\top(\boldsymbol{\mu}_+ \text{-} \boldsymbol{\mu}_-)}{\mathbf{w}^\top(q\Sigma_+ + (1-q)\Sigma_-)\,\mathbf{w}}, \tag{5}$$

with $q = N_+/N$ the fraction of samples in a dataset that belong to the positive class and $N = N_+ + N_-$ the total number of samples in a class. The solution of (5) is given by:

$$\mathbf{w}^* \propto \hat{\Sigma}^{-1}(\boldsymbol{\mu}_+ \text{-} \boldsymbol{\mu}_-), \tag{6}$$

$$\hat{\Sigma} \triangleq q\Sigma_+ + (1-q)\Sigma_-. \tag{7}$$

Note that (6) is identical to the original LDA solution (4) if either $\Sigma_+ \propto \Sigma_-$ (e.g. when the classes are homoscedastic) or if $q = 0.5$, meaning that the classes are balanced. When this is not the case, MILDA will put more weight on minimising the class variance of the class containing more samples compared to the class containing fewer samples than LDA. This could theoretically be beneficial when little data are available, as in that case, the covariance matrices are sensitive to estimation errors. However, in practice we found no noteworthy difference in performance between (6) and (4), as demonstrated in Section IV.

B. Preliminaries to MILDA

The projection from (6) requires knowledge of Σ_+, Σ_-, the class difference $\boldsymbol{\mu}_+ \text{-} \boldsymbol{\mu}_-$, and q. In practice, these parameters can usually only be estimated on labelled data in a supervised setting. However, in this section we will show that the solution (6) can be computed without labels if the two class averages are proportional[3], i.e., $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$. While the latter assumption may seem artificial and overly strict at first sight, we will show in Section II-C that it can be relaxed to more realistic conditions based on knowledge of some ground-truth statistics.

[3]We use $\mathbf{x} \sim \mathbf{y}$ to denote that either $\mathbf{x} = \alpha\mathbf{y}$ or $\mathbf{y} = \beta\mathbf{x}$ with $\alpha, \beta \in \mathbb{R}$, whereas we use $\mathbf{x} \propto \mathbf{y}$ to denote that $\mathbf{x} = \alpha\mathbf{y}$ with $\alpha \in \mathbb{R}\setminus\{0\}$. In the former case, it is allowed that either $\mathbf{x}$ or $\mathbf{y}$ is the zero vector, whereas in the latter both vectors should be non-zero.

Lemma II.1. *If $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$, then the solution of* (5) *can be computed as*

$$\mathbf{w}^* \propto \bar{\Sigma}^{-1}\,\bar{\boldsymbol{\mu}} \tag{8}$$

$$\textit{with } \bar{\boldsymbol{\mu}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \tag{9}$$

$$\bar{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \bar{\boldsymbol{\mu}})(\mathbf{x}_i - \bar{\boldsymbol{\mu}})^\top \tag{10}$$

Proof. With some algebraic manipulations, we can rewrite the covariance matrix $\bar{\Sigma}$ in (10) and class covariance matrices Σ_+ and Σ_- in (3) as:

$$\bar{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^\top - \bar{\boldsymbol{\mu}}\,\bar{\boldsymbol{\mu}}^\top \tag{11}$$

$$\Sigma_+ = \frac{1}{qN}\sum_{i\in\mathcal{C}_+}\mathbf{x}_i\mathbf{x}_i^\top - \boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top \tag{12}$$

$$\Sigma_- = \frac{1}{(1-q)N}\sum_{i\in\mathcal{C}_-}\mathbf{x}_i\mathbf{x}_i^\top - \boldsymbol{\mu}_-\,\boldsymbol{\mu}_-{}^\top. \tag{13}$$

Based on (7), (11)-(13), it can be verified that:

$$\begin{aligned}\bar{\Sigma} &= q(\Sigma_+ + \boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top) + (1-q)(\Sigma_- + \boldsymbol{\mu}_-\,\boldsymbol{\mu}_-{}^\top) - \bar{\boldsymbol{\mu}}\,\bar{\boldsymbol{\mu}}^\top \\ &= \hat{\Sigma} + q\,\boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top + (1-q)\,\boldsymbol{\mu}_-\,\boldsymbol{\mu}_-{}^\top - \bar{\boldsymbol{\mu}}\,\bar{\boldsymbol{\mu}}^\top.\end{aligned} \tag{14}$$

Furthermore, from (2) and (9), we can also write

$$\bar{\boldsymbol{\mu}} = q\,\boldsymbol{\mu}_+ + (1-q)\,\boldsymbol{\mu}_-. \tag{15}$$

Since $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$, we define[4] $\boldsymbol{\mu}_- = \alpha\,\boldsymbol{\mu}_+$ with $\alpha \in \mathbb{R}\setminus\{1\}$. Note that $\alpha = 1$ is excluded, since this would imply that the mean of both classes coincides, in which case (5) is ill-defined. By substituting $\boldsymbol{\mu}_- = \alpha\,\boldsymbol{\mu}_+$ in (14) and (15), we can simplify $\bar{\boldsymbol{\mu}}$ and $\bar{\Sigma}$ as:

$$\begin{aligned}\bar{\boldsymbol{\mu}} &= (q + (1-q)\alpha)\,\boldsymbol{\mu}_+ &&\triangleq \beta\,\boldsymbol{\mu}_+ \\ \bar{\Sigma} &= \hat{\Sigma} + (q + (1-q)\alpha^2 - \beta^2)\,\boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top \triangleq \hat{\Sigma} + \gamma\,\boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top,\end{aligned}$$

with $\gamma = q(1-q)(\alpha-1)^2 > 0$.

When plugging this in the right-hand side of (8), we get.

$$\begin{aligned}\mathbf{w}^* &\propto \bar{\Sigma}^{-1}\,\bar{\boldsymbol{\mu}} \\ &\propto (\hat{\Sigma} + \gamma\,\boldsymbol{\mu}_+\,\boldsymbol{\mu}_+{}^\top)^{-1}\,\boldsymbol{\mu}_+,\end{aligned} \tag{16}$$

where β is absorbed in the $\propto$ sign.

We will now show that (16) is indeed a solution of (5). First, the matrix inverse in this expression can be rewritten using the Sherman-Morrison identity, which is a special case

[4]If $\boldsymbol{\mu}_+ = \mathbf{0}$, the roles of $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_-$ should be switched in the proof.

of the Woodbury identity for rank-1 updates [20]. This allows us to write (16) as:

$$\begin{aligned}\mathbf{w}^* &\propto \left(\hat{\Sigma}^{-1} - \frac{\gamma\,\hat{\Sigma}^{-1}\boldsymbol{\mu}_+\boldsymbol{\mu}_+{}^\top\hat{\Sigma}^{-1}}{1+\gamma\,\boldsymbol{\mu}_+{}^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_+}\right)\boldsymbol{\mu}_+ \\ &\propto \hat{\Sigma}^{-1}\boldsymbol{\mu}_+ - \frac{\gamma\,\boldsymbol{\mu}_+{}^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_+}{1+\gamma\,\boldsymbol{\mu}_+{}^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_+}\hat{\Sigma}^{-1}\boldsymbol{\mu}_+ \\ &\propto \hat{\Sigma}^{-1}\boldsymbol{\mu}_+, \end{aligned} \tag{17}$$

where the irrelevant scaling factor between the left- and right-hand side were absorbed in the $\propto$ sign.

Finally, since $\boldsymbol{\mu}_+ - \boldsymbol{\mu}_- = (1-\alpha)\,\boldsymbol{\mu}_+ \propto \boldsymbol{\mu}_+$, we can rewrite (17) as:

$$\mathbf{w}^* \propto \hat{\Sigma}^{-1}(\boldsymbol{\mu}_+ \text{-}\, \boldsymbol{\mu}_-).$$

This results in (6), which is a solution of (5), proving the lemma. □

As stated before, the vector $\mathbf{w}^*$ in (6) or (8) is proportional to Fisher's LDA defined in (4) if the classes are balanced ($q = 0.5$) or if $\Sigma_+ \propto \Sigma_-$. Otherwise, this vector optimises the objective formulated in (5).

C. The MILDA framework

Section II-B showed that the LDA projection vector can be computed using the label-free expression in (8) if the two class averages are proportional to each other or if either class is zero-mean, i.e. if $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$. However, in practice this rather artificial assumption is rarely satisfied. Fortunately, it is possible to satisfy this assumption by transforming the data using one out of the three possible ground-truth statistics: the class average of a single class, the relative difference between the class averages (up to a scaling) or the covariances for each of the two classes (up to a scaling). The latter case also assumes either balanced classes (or knowledge of the class imbalance), or a relaxed homoscedasticity condition (i.e. $\Sigma_+ \propto \Sigma_-$). We will go over these cases one by one and derive the required transformations. We will denote the transformed data and its corresponding statistics using the notation $(.)'$, i.e., the mean of the positive class after transformation is denoted as $\boldsymbol{\mu}'_+$. All transformations are summarised in Table I for convenience of the reader. The MILDA algorithm, which is summarised in Algorithm 1, then simply requires applying the corresponding transformation on the data and then computing (8).

1) The class average of a single class is known: Assume without loss of generality that the class average $\boldsymbol{\mu}_+$ is known. In this case, you can guarantee that $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$ after transforming each feature vector $\mathbf{x}_i$ to:

$$\mathbf{x}'_i \longleftarrow \mathbf{x}_i - \boldsymbol{\mu}_+\,. \tag{18}$$

Indeed, in this case $\boldsymbol{\mu}'_+ = \mathbf{0}$. Therefore $\boldsymbol{\mu}'_+ = 0\boldsymbol{\mu}'_-$, making $\boldsymbol{\mu}'_+ \sim \boldsymbol{\mu}'_-$ trivially true. The new global average $\bar{\boldsymbol{\mu}}' = \bar{\boldsymbol{\mu}} - \boldsymbol{\mu}_+ \neq \mathbf{0}$ as long as $\boldsymbol{\mu}_- \neq \boldsymbol{\mu}_+$, which ensures that the MILDA transformation from (8) has a solution. If $\boldsymbol{\mu}_- = \boldsymbol{\mu}_+$, the two classes are not separable with either LDA or MILDA and another classifier should be used.

2) The relative difference between the class averages is known: In this case, $\boldsymbol{\mu}_+ - \boldsymbol{\mu}_- = \delta\mathbf{d}$, with $\delta \in \mathbb{R}$ an unknown scalar and $\mathbf{d}$ a known vector. It is thus unknown by how much the classes are separated. We can guarantee that $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$ by transforming each data sample $\mathbf{x}_i$ to:

$$\mathbf{x}'_i \longleftarrow \mathbf{x}_i - (\bar{\boldsymbol{\mu}} - \alpha\mathbf{d}), \alpha \neq 0. \tag{19}$$

Though parameter $\alpha \neq 0$ can be freely chosen, it is best to choose α such that $\|\alpha\mathbf{d}\| \approx \|\mathbf{x}_i\|$ to avoid ill-conditioning.

After the transformation and using (15), we can write the new class averages as:

$$\begin{aligned}\boldsymbol{\mu}'_+ &= \boldsymbol{\mu}_+ - \bar{\boldsymbol{\mu}} + \alpha\mathbf{d} = (1-q)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-) + \alpha\mathbf{d} \\ &= ((1-q)\delta + \alpha)\mathbf{d} \\ \boldsymbol{\mu}'_- &= \boldsymbol{\mu}_- - \bar{\boldsymbol{\mu}} + \alpha\mathbf{d} = -q\,\boldsymbol{\mu}_+ + q\,\boldsymbol{\mu}_- + \alpha\mathbf{d} \\ &= (-q\delta + \alpha)\mathbf{d}.\end{aligned}$$

This shows indeed that $\boldsymbol{\mu}'_+ \sim \boldsymbol{\mu}'_-$. Furthermore, using (15) for the transformed data, we see that $\bar{\boldsymbol{\mu}}' \neq \mathbf{0}$ as long as $\alpha \neq 0$.

3) The covariances for each of the two classes are known: In this section, we assume that the covariances Σ_+ and Σ_- are known (up to a scaling), i.e., we assume the availability of the matrices $S_+ = \beta\Sigma_+$ and $S_- = \beta\Sigma_-$ with β an unknown scalar. In addition, we assume that the classes are balanced or that the class proportions (i.e. q) are known in case they are unbalanced. We will later see that this latter assumption can be removed if a relaxed homoscedasticity condition (i.e. $\Sigma_+ \propto \Sigma_-$) is satisfied.

We propose the following double transformation, which will later be shown (see Lemma II.2) to result in $\boldsymbol{\mu}''_+ \sim \boldsymbol{\mu}''_-$:

$$\mathbf{x}'_i \longleftarrow \hat{S}^{-1/2}\mathbf{x}_i \tag{20}$$

$$\mathbf{x}''_i \longleftarrow \mathbf{x}'_i - (\bar{\boldsymbol{\mu}}' - \alpha\mathbf{d}), \alpha \neq 0, \tag{21}$$

$$\text{with} \quad \hat{S} = qS_+ + (1-q)S_- \propto \hat{\Sigma} \tag{22}$$

$$\bar{\boldsymbol{\mu}}' = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}'_i$$

$$\bar{\Sigma}' = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}'_i\mathbf{x}'^{\top}_i - \bar{\boldsymbol{\mu}}'\,\bar{\boldsymbol{\mu}}'^{\top}$$

$$\bar{\Sigma}'\,\mathbf{d} = \lambda_{max}\mathbf{d},$$

where λ_{max} is the largest eigenvalue of $\bar{\Sigma}'$. In other words, we first transform the data via (20). Then, we compute the eigenvector $\mathbf{d}$ that corresponds to the largest eigenvalue λ_{max} from the new covariance matrix $\bar{\Sigma}'$. In Lemma II.2 below, we show that this vector $\mathbf{d}$ has the same direction as $\boldsymbol{\mu}'_+ - \boldsymbol{\mu}'_-$. Therefore, once this direction is known, we can apply the same transformation as (19) to the data $\mathbf{x}_i$, resulting in (21).

Lemma II.2. *Let $\bar{\Sigma}$ be defined as in* (11)*, and let $\hat{S}$ be defined as in* (22)*. If $\mathbf{d}$ is the eigenvector of $\bar{\Sigma}' = \hat{S}^{-1/2}\,\bar{\Sigma}\,\hat{S}^{-1/2}$ with the largest eigenvalue, then:*

$$\mathbf{d} \propto \hat{S}^{-1/2}(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-) = \boldsymbol{\mu}'_+ - \boldsymbol{\mu}'_-.$$

Proof. First, we will compute the effect of the first transformation on the covariance matrix. Substituting (15) in (14), we obtain

$$\begin{aligned}\bar{\Sigma} &= \hat{\Sigma} + q(1-q)\left(\boldsymbol{\mu}_+ \boldsymbol{\mu}_+^\top + \boldsymbol{\mu}_- \boldsymbol{\mu}_-^\top - 2\,\boldsymbol{\mu}_+ \boldsymbol{\mu}_-^\top\right) \\ &= \hat{\Sigma} + q(1-q)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)^\top. \end{aligned} \tag{23}$$

After a transformation $\mathbf{x}_i' \longleftarrow A\mathbf{x}_i$, the new average and covariance are respectively $\bar{\boldsymbol{\mu}}' = A\,\bar{\boldsymbol{\mu}}$ and $\bar{\Sigma}' = A\,\bar{\Sigma}\,A^\top$. In this case, (23) becomes:

$$\begin{aligned}\bar{\Sigma}' &= A\,\hat{\Sigma}\,A^\top + q(1-q)(A\,\boldsymbol{\mu}_+ - A\,\boldsymbol{\mu}_-)(A\,\boldsymbol{\mu}_+ - A\,\boldsymbol{\mu}_-)^\top. \\ &\triangleq A\,\hat{\Sigma}\,A^\top + q(1-q)(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')^\top. \end{aligned} \tag{24}$$

Since $\hat{S}$ is the sum of two symmetric positive definite matrices, it is also symmetric positive definite. Therefore, a symmetric matrix $\hat{S}^{-1/2}$ exists. From (22), we know that $\hat{S}^{-1/2} \propto \hat{\Sigma}^{-1/2}$. By defining $A = \hat{S}^{-1/2}$, (24) becomes:

$$\bar{\Sigma}' = \beta^{-1} I + q(1-q)(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')^\top, \tag{25}$$

where β is an unknown scalar due to the proportionality $\hat{S} = \beta\,\hat{\Sigma}$. This shows that the transformation $\mathbf{x}_i' \longleftarrow \hat{S}^{-1/2}\mathbf{x}_i$ results in a covariance matrix that is equal to a rank-1 update on a scaled identity matrix. Note that if $\Sigma_+ \propto \Sigma_- \propto S$, the first transformation is proportional to $\mathbf{x}_i' \longleftarrow S^{-1/2}\mathbf{x}_i$, independent of q. The class imbalance q must thus only be known if $\Sigma_+ \not\propto \Sigma_-$.

Finally, we need to show that the vector $\mathbf{d} \propto \boldsymbol{\mu}_+' - \boldsymbol{\mu}_-'$ is indeed the eigenvector of $\bar{\Sigma}'$ with the largest eigenvalue. By construction, the rank-1 matrix $B = q(1-q)(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')^\top$ has only one eigenvector $\mathbf{d} \propto \boldsymbol{\mu}_+' - \boldsymbol{\mu}_-'$ with a non-zero eigenvalue $\lambda_{B,1} = q(1-q)(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-')^\top(\boldsymbol{\mu}_+' - \boldsymbol{\mu}_-') > 0$. Note that $\bar{\Sigma}'$ is the matrix obtained by adding a scaled identity matrix $\beta^{-1}I$ to B. The impact of adding $\beta^{-1}I$ is that all eigenvalues are incremented with β^{-1}, while the eigenvectors remain unchanged. Therefore, the eigenvector $\mathbf{d}$ will still be the eigenvector of $\bar{\Sigma}'$ corresponding to the largest eigenvalue. □

Algorithm 1: A summary of the MILDA algorithm

Data: N samples $\mathbf{x}_i \in \mathbb{R}^{D\times 1}$
Result: Projection vector $\mathbf{w} \in \mathbb{R}^{D\times 1}$
$\mathbf{x}_i' \longleftarrow f(\mathbf{x}_i)$ as defined in Table I
$\bar{\boldsymbol{\mu}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i$,
$\bar{\Sigma} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i\mathbf{x}_i^\top - \bar{\boldsymbol{\mu}}\,\bar{\boldsymbol{\mu}}^\top$
$\mathbf{w} \propto \bar{\Sigma}^{-1}\,\bar{\boldsymbol{\mu}}$

Known	**Transformation** $f(\mathbf{x}_i)$
1) Class average $\boldsymbol{\mu}_+$	$\mathbf{x}_i' \longleftarrow \mathbf{x}_i - \boldsymbol{\mu}_+$
2) $\mathbf{d} \propto \boldsymbol{\mu}_+ - \boldsymbol{\mu}_-$	$\mathbf{x}_i' \longleftarrow \mathbf{x}_i - (\bar{\boldsymbol{\mu}} - \alpha\mathbf{d}), \alpha \neq 0$
3) $S_+ = \beta\Sigma_+,\ S_- = \beta\Sigma_-$ and q	$\mathbf{x}_i' \longleftarrow (qS_+ + (1-q)S_-)^{-1/2}\mathbf{x}_i$ $\mathbf{x}_i'' \longleftarrow \mathbf{x}_i' - (\bar{\boldsymbol{\mu}}' - \alpha\mathbf{d}), \alpha \neq 0,$
4) $S \propto \Sigma_+ \propto \Sigma_-$	$\mathbf{x}_i' \longleftarrow S^{-1/2}\mathbf{x}_i$ $\mathbf{x}_i'' \longleftarrow \mathbf{x}_i' - (\bar{\boldsymbol{\mu}}' - \alpha\mathbf{d}), \alpha \neq 0,$ with $\mathbf{d}$ defined in (21).

TABLE I: A summary of the transformations required given the known piece of information.

III. Sensitivity to errors in the prior knowledge

Until now, we have always assumed that MILDA has access to the exact ground-truth statistic for each of the 4 cases in Table I. In practice, often only an estimate of this statistic is available. In this section, we will therefore study how these approximation errors affect the projection vector from MILDA compared to the exact LDA projection vector (5). More precisely, we will study these effects both in a mathematical and experimental context in Sections III-A and III-B respectively. Amongst others, we will show that such approximation errors matter less when classes are more difficult to separate or more imbalanced, which are precisely the cases that are the hardest for unsupervised models to tackle.

A. Mathematical sensitivity analysis

Equation (8) states that MILDA computes the projection vector:

$$\mathbf{w}_{MILDA} \propto \bar{\Sigma}^{-1}\,\bar{\boldsymbol{\mu}}, \tag{26}$$

with $\bar{\boldsymbol{\mu}}$, $\bar{\Sigma}$ defined as in (9),(10). Furthermore, from (23), we know that

$$\begin{aligned}\bar{\Sigma} &= \hat{\Sigma} + q(1-q)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)^\top \\ &\triangleq \hat{\Sigma} + q(1-q)\,\boldsymbol{\mu}_\Delta\,\boldsymbol{\mu}_\Delta^\top, \end{aligned} \tag{27}$$

where we introduced a new notation $\boldsymbol{\mu}_\Delta \triangleq \boldsymbol{\mu}_+ - \boldsymbol{\mu}_-$ for notational convenience.

Using (27) in (26), applying the Sherman-Morrison identity [20] and rearranging the order of the terms gives:

$$\begin{aligned}\mathbf{w}_{MILDA} &\propto \left(\hat{\Sigma}^{-1} - \frac{q(1-q)\,\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}}{1 + q(1-q)\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta}\right)\bar{\boldsymbol{\mu}} \\ &= \hat{\Sigma}^{-1}\,\bar{\boldsymbol{\mu}} - \frac{q(1-q)\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}\bar{\boldsymbol{\mu}}}{1 + q(1-q)\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta}\,\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta \\ &= \hat{\Sigma}^{-1}(\bar{\boldsymbol{\mu}} + \phi\,\boldsymbol{\mu}_\Delta), \\ &\text{with } \phi \triangleq -\frac{q(1-q)\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}\bar{\boldsymbol{\mu}}}{1 + q(1-q)\,\boldsymbol{\mu}_\Delta^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta}. \end{aligned} \tag{28}$$

When comparing (28) with (6), we find that MILDA optimises (5) if and only if $\bar{\boldsymbol{\mu}}$ is proportional to $\boldsymbol{\mu}_\Delta$. It can be verified that if $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$, then $\bar{\boldsymbol{\mu}} \sim \boldsymbol{\mu}_\Delta$. This indeed confirms Lemma II.1, and this is what the different transformations in Section II-C aim to achieve.

Let us now consider the case where the assumption $\boldsymbol{\mu}_+ \sim \boldsymbol{\mu}_-$ (and thus $\bar{\boldsymbol{\mu}} \sim \boldsymbol{\mu}_\Delta$) is not (perfectly) satisfied. This happens for example if the transformations in Section II-C are computed based on approximations or estimates of the ground-truth statistics in the left column. In this case, we decompose $\bar{\boldsymbol{\mu}}$ into a component proportional to $\boldsymbol{\mu}_\Delta$ and a component orthogonal to $\boldsymbol{\mu}_\Delta$, i.e., $\bar{\boldsymbol{\mu}} = \delta\,\boldsymbol{\mu}_\Delta + \mathbf{e}$, with $\delta \in \mathbb{R}$ a scaling term and $\mathbf{e}$ an error vector orthogonal to $\boldsymbol{\mu}_\Delta$. $\mathbf{w}_{MILDA}$ can then be written as:

$$\mathbf{w}_{MILDA} \propto \hat{\Sigma}^{-1}\left((\phi + \delta)\,\boldsymbol{\mu}_\Delta + \mathbf{e}\right). \tag{29}$$

Depending on the statistics of the data, the same error does not always affect the projection vector $\mathbf{w}$ equally. The error

will have a small effect on $\mathbf{w}_{MILDA}$ when the angle between $\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta$ and $\hat{\Sigma}^{-1}\mathbf{e}$ is small and/or when $\frac{1}{|\phi+\delta|}\frac{\|\mathbf{e}\|_2}{\|\boldsymbol{\mu}_\Delta\|_2}$ is small. By using the definition of ϕ in (28), we can further expand the latter expression as:

$$\begin{aligned}&\frac{1}{|\phi+\delta|}\frac{\|\mathbf{e}\|_2}{\|\boldsymbol{\mu}_\Delta\|_2}\\&=\frac{1}{\left|1-\frac{q(1-q)\,\boldsymbol{\mu}_\Delta{}^\top\hat{\Sigma}^{-1}(\boldsymbol{\mu}_\Delta+\frac{\mathbf{e}}{\delta})}{1+q(1-q)\,\boldsymbol{\mu}_\Delta{}^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta}\right|}\frac{\|\mathbf{e}\|_2}{\delta\|\boldsymbol{\mu}_\Delta\|_2}\\&=\left|\frac{1+q(1-q)\,\boldsymbol{\mu}_\Delta{}^\top\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta}{1-q(1-q)\,\boldsymbol{\mu}_\Delta{}^\top\hat{\Sigma}^{-1}\frac{\mathbf{e}}{\delta}}\right|\frac{\|\mathbf{e}\|_2}{\|\bar{\boldsymbol{\mu}}-\mathbf{e}\|_2}.\end{aligned}\tag{30}$$

Using (29) and (30), we can conclude that the following properties reduce the sensitivity of MILDA to errors in the ground-truth statistics:

- Property 1: $\hat{\Sigma}^{-1}\mathbf{e}$ and $\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta$ form a small angle. Since $\mathbf{e}$ and $\boldsymbol{\mu}_\Delta$ are orthogonal (by construction), more skewed class covariances can reduce the impact of the error (although this is not guaranteed). Note that the third and fourth transformations of Table I first transform the data such that $\hat{\Sigma}=I$. For these transformations, $\hat{\Sigma}^{-1}\boldsymbol{\mu}_\Delta$ will thus always be orthogonal to $\hat{\Sigma}^{-1}\mathbf{e}$.
- Property 2: $\|\mathbf{e}\|_2$ is small relative to $\|\bar{\boldsymbol{\mu}}\|_2$. This is trivial: the error matters less when it is smaller. The worst case is when $\boldsymbol{\mu}_\Delta\perp\bar{\boldsymbol{\mu}}$, since in this case $\bar{\boldsymbol{\mu}}=\mathbf{e}$, such that $\mathbf{w}_{MILDA}\perp\mathbf{w}_{LDA}$ if $\hat{\Sigma}=I$. From (15), we see that $\boldsymbol{\mu}_\Delta{}^\top\bar{\boldsymbol{\mu}}=q\|\boldsymbol{\mu}_+\|_2^2+(1-2q)\,\boldsymbol{\mu}_+{}^\top\boldsymbol{\mu}_-(1-q)\|\boldsymbol{\mu}_-\|_2^2$. Since $\boldsymbol{\mu}_\Delta{}^\top\bar{\boldsymbol{\mu}}=0$ if $\boldsymbol{\mu}_\Delta\perp\bar{\boldsymbol{\mu}}$, this shows that this worst-case scenario does not occur when the class average of the class with the most samples is much larger than the other class average. If the classes are approximately balanced, it suffices that the norm of any one of the two class averages is much larger than the other.
- Property 3: There is a large class imbalance. Indeed, if $q\approx 0$ or $q\approx 1$, the first term in (30) becomes minimal (i.e. close to 1), therefore reducing the sensitivity to errors.
- Property 4: $\boldsymbol{\mu}_\Delta{}^\top\Sigma^{-1}\boldsymbol{\mu}_\Delta$ is small. This also minimises the first term in (30). Interestingly, this implies that the error will have a reduced effect on $\mathbf{w}_{MILDA}$ when there is a lot of overlap between the two classes. MILDA has thus the nice property that approximation errors matter less when the classes are harder to separate, which are exactly the cases where fully unsupervised methods do not perform well.

Beware that, in some cases, the beneficial effect of the latter two properties can be counteracted by other detrimental effects. This happens in the following cases.

- A large class imbalance will increase the error after transformation 1 from Table I (1 class average known) if the unknown class average is also the class with the least samples. Indeed, after the transformation the class with the most samples will be close to zero-mean resulting in a small $\bar{\boldsymbol{\mu}}'$, which directly counteracts the second aforementioned property.
- Both a large class imbalance and a worse separability make it harder to correctly estimate $\mathbf{d}$ when using transformations 3 or 4 from Table I (the covariance methods), since both will reduce the impact of the rank-1 update on the overall covariance matrix in (25).

B. Experimental sensitivity analysis

To visualise the findings from Section III-A, the angle between the LDA and MILDA projection vectors is estimated in a two-dimensional space. To achieve this, we keep $\boldsymbol{\mu}_+$ fixed at $[0,1]^\top$. Meanwhile, $\boldsymbol{\mu}_-=[x,y]^\top$ is varied in the ranges $x,y\in[-2,2]$. For each $\boldsymbol{\mu}_-$, we compute the projection vectors $\mathbf{w}_{MILDA}\propto\bar{\Sigma}^{-1}\bar{\boldsymbol{\mu}}$ and $\mathbf{w}_{LDA}\propto(\Sigma_++\Sigma_-)^{-1}(\boldsymbol{\mu}_+-\boldsymbol{\mu}_-)$, assuming balanced classes. We then measure the similarity between the two projections using the angle η between $\mathbf{w}_{MILDA}$ and $\mathbf{w}_{LDA}$ using the formula $\cos(\eta)=\frac{\mathbf{w}_{LDA}^\top\mathbf{w}_{MILDA}}{\|\mathbf{w}_{LDA}\|_2\|\mathbf{w}_{MILDA}\|_2}$. The closer η is to 0, the more the directions of $\mathbf{w}_{LDA}$ and $\mathbf{w}_{MILDA}$ are similar, and thus the more their respective projections are equivalent.

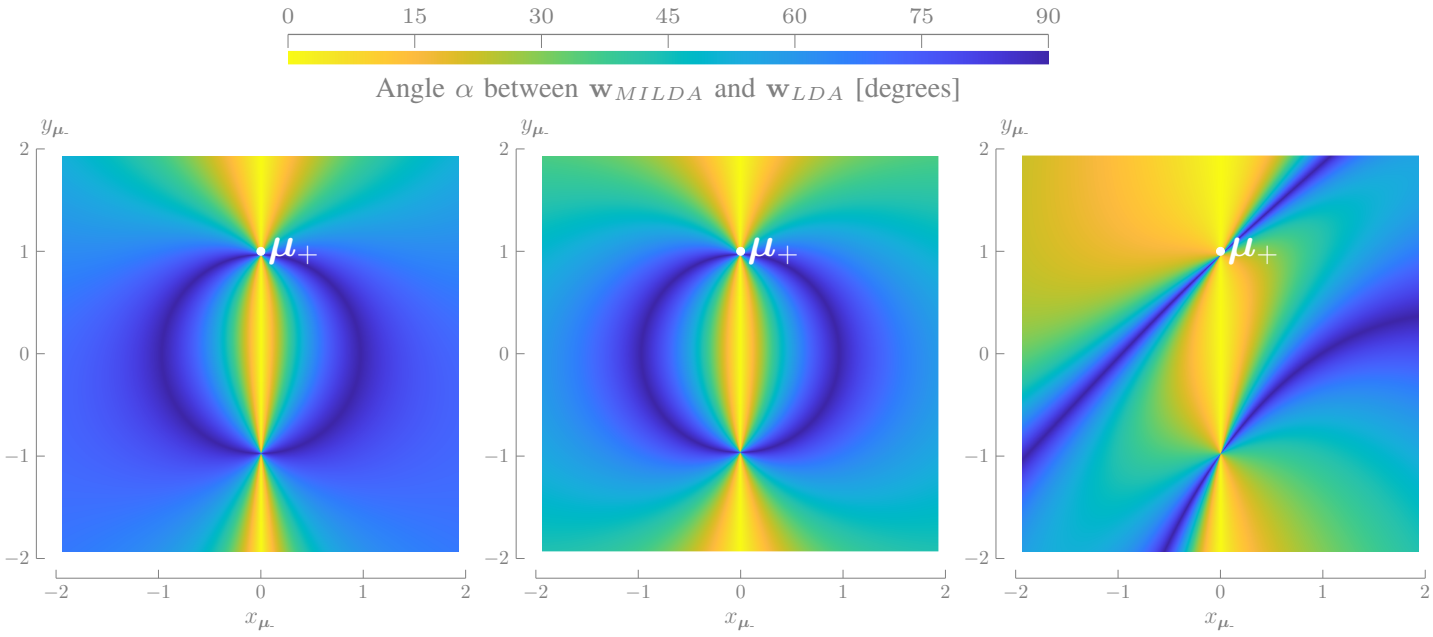
This set-up allows us to simulate a wide array of possible violation of the assumption $\boldsymbol{\mu}_+\sim\boldsymbol{\mu}_-$ up to an irrelevant scaling and rotation[5]. To study how the class covariance influences the sensitivity of MILDA to estimation errors, we will repeat this process for three different sets of class covariances. In the first experiment, $\hat{\Sigma}_1=I$. In the second experiment, $\hat{\Sigma}_2=5I$. Here, the variance is larger, making the classes less separable. In the third experiment, $\hat{\Sigma}_3=0.3I+0.7J$, with I the identity matrix and J a matrix with all ones. In this case, the class covariances are very skewed, allowing us to verify that this indeed decreases the influence of the error in many cases. For each experiment, $\Sigma_+=\Sigma_-=\hat{\Sigma}$ and $q=0.5$. To avoid any noise blurring the visualisation, no data was generated in this process. Instead, both the LDA and the MILDA projections were directly computed from $\Sigma_\pm$ and $\boldsymbol{\mu}_\pm$.

Figure 1 confirms the insights obtained from the mathematical analysis in Section III-A. When $\hat{\Sigma}\propto I$ (Fig. 1a), the blue circle at the positions where $\|\boldsymbol{\mu}_+\|\approx\|\boldsymbol{\mu}_-\|$ clearly shows the worst-case scenario where $\mathbf{w}_{MILDA}\perp\mathbf{w}_{LDA}$ in the area where $\bar{\boldsymbol{\mu}}=\mathbf{e}$ (see property 2). The region where $\boldsymbol{\mu}_+\approx-\boldsymbol{\mu}_-$ is particularly interesting, as here the alignment assumption $\boldsymbol{\mu}_+\sim\boldsymbol{\mu}_-$ is quite well satisfied, yet the sensitivity of MILDA to small deviations from such perfect alignment is very high because $\bar{\boldsymbol{\mu}}\approx\mathbf{0}$.

The ideal location for $\boldsymbol{\mu}_-$ to be, is either around $[0,0]^\top$ (if $\boldsymbol{\mu}_+$ is non-zero) or $\pm\kappa\,\boldsymbol{\mu}_+$ with $\kappa\gg 1$ In these regions, the misalignment error between $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_-$ can be larger without having much effect on the solution.

Finally, we also clearly observe that MILDA becomes more robust against errors as the class covariances increase (Figure 1b) or become less white (Figure 1c), as predicted in Property 1 and Property 4 in the mathematical error analysis. This is interesting, as these are the typical scenario's where unsupervised methods suffer the most.

[5]Scaling and rotation are irrelevant because LDA and MILDA transformations are affine invariant (proof omitted).



(a) $\hat{\Sigma} = I$ **(b)** $\hat{\Sigma} = 5I$ **(c)** $\hat{\Sigma} = 0.3I + 0.7J$

Fig. 1: Heatmap representing the angle between $\mathbf{w}_{LDA}$ and $\mathbf{w}_{MILDA}$ as a function of the location of $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_-$. The angle between the LDA and MILDA projection vector decreases when $\boldsymbol{\mu}_- << \boldsymbol{\mu}_+$ or $\boldsymbol{\mu}_- >> \boldsymbol{\mu}_+$, when the covariance increases, and when it is less white.

IV. Experiments

In this section, we demonstrate that MILDA closely matches the performance of a fully supervised LDA. In the first set of experiments, the two models are compared to each other on a vast set of simulated data with varying parameters. This allows us to accurately investigate the influence of each parameter on the model's performance. In a second experiment, we apply MILDA in the context of target detection using a sensor array. This is a common problem for e.g. radar detection, wireless communication and brain-computer interfaces [21]–[23]. Finally, MILDA and LDA are compared in a third experiment on non-stationary data, highlighting how the label-free MILDA model can easily adapt to changing statistics.

Note that the LDA and MILDA models would not be used interchangeably. LDA requires all statistics to be known, whereas MILDA is specifically designed for cases where much less information is available. Therefore, LDA should be seen as a benchmark that MILDA should ideally approach as closely as possible (except in the third experiment).

A. Ablation study

1) Experiment design: In this experiment, the models are validated on a simulated dataset where the two classes are sampled from two different Gaussian distributions. By default, $N = 1000$ samples are sampled from the distributions using the following hyperparameters: $\boldsymbol{\mu}_+ = [1, \mathbf{0}]^\top \in \mathbb{R}^{D\times 1}$, $\boldsymbol{\mu}_- = [-1, \mathbf{0}]^\top \in \mathbb{R}^{D\times 1}$, $\Sigma_+ = (1-\rho)I + \rho J$, with $I \in \mathbb{R}^{D\times D}$ the identity matrix and $J \in \mathbb{R}^{D\times D}$ a matrix of all ones. $\rho \in [0, 1]$ controls how correlated the different features are ($\rho = 0.3$ by default). $D = 10$ is the data dimensionality. Σ_- is computed by using the same eigenvalues and eigenvectors as Σ_+, but pairing each eigenvector with a different eigenvalue than in Σ_+. This generates two classes with the same covariance structure but rotated such that the maximal covariance directions are orthogonal. By default, both classes are balanced ($q = 0.5$).

Each time, one specific parameter in the model is varied to study its influence on the behaviour of MILDA. The following parameters are studied:

- The effect of the class separation: $\boldsymbol{\mu}_- = [i, \mathbf{0}]^\top, i \in [-3, 1]$ (results shown in Fig. 2).
- The effect of the correlation between the features $\rho \in [0, 1]$ (results shown in Fig. 3).
- The degree to which Σ_+ and Σ_- are similar (results shown in Fig. 4). This is controlled by choosing $\Sigma_- = (1-\sigma)\Sigma_+ + \sigma\Sigma_{-,default}, \sigma \in [0, 1]$.
- The data dimensionality $D \in [2, 100]$ (results shown in Fig. 5).
- The influence of the class imbalance, modelled by the fraction of samples from the positive class $\mathcal{C}_+$ $q \in [0.01, 0.99]$ (results shown in Fig. 6).

Given that MILDA is a label-free method, we also compare it to two other popular unsupervised classification methods: K-means and Gaussian mixture models (GMMs). Where possible, these models were modified[6] to incorporate the same prior knowledge as MILDA to make the comparison as fair as

[6]For all experiments in this paper, we have verified that these modifications do not deteriorate the results compared to the original K-means and GMM algorithm. Using all default parameters, the accuracy of the original and adapted K-means algorithms are respectively 64% and 84%. The accuracy of the original and adapted GMM algorithms are respectively 66% and 69%.

Known	K-means update	GMM update
1) Class average $\boldsymbol{\mu}_+$	$\boldsymbol{\mu}_1^{(i)} \longleftarrow (1-\lambda)\boldsymbol{\mu}_1^{(i)} + \lambda\,\boldsymbol{\mu}_+$	$\boldsymbol{\mu}_1^{(i)} \longleftarrow (1-\lambda)\boldsymbol{\mu}_1^{(i)} + \lambda\,\boldsymbol{\mu}_+$
2) $\mathbf{d} \propto \boldsymbol{\mu}_+ - \boldsymbol{\mu}_-$	$\boldsymbol{\mu}_{1/2}^{(i)} \longleftarrow \left((1-\lambda)I + \lambda\mathbf{d}\mathbf{d}^\top\right)(\boldsymbol{\mu}_{1/2}^{(i)} - \bar{\boldsymbol{\mu}})$	$\boldsymbol{\mu}_{1/2}^{(i)} \longleftarrow \left((1-\lambda)I + \lambda\mathbf{d}\mathbf{d}^\top\right)(\boldsymbol{\mu}_{1/2}^{(i)} - \bar{\boldsymbol{\mu}})$
3) Σ_+, Σ_- and q or 4) $\Sigma_+ \propto \Sigma_-$	$\boldsymbol{\mu}_{1/2}^{(i)} \longleftarrow \boldsymbol{\mu}_{1/2}^{(i)}$	$\Sigma_{1/2}^{(i)} = (1-\lambda)\Sigma_{1/2}^{(i)} + \lambda\Sigma_{+/-}$

TABLE II: The updating schemes for K-means and GMM to incorporate known pieces of information as a prior, with $\lambda \in [0, 1]$ and $\|\mathbf{d}\|_2 = 1$. By default, $\lambda = 0.7$ is chosen, as it empirically produced the best results. In contrast to MILDA, $\Sigma_\pm$ were used for the GMM instead of $S_\pm \propto \Sigma_\pm$ in transformations 3 and 4, since the algorithm is not invariant to scaling. Classes 1/2 are paired with classes +/- after the K-means++ initialisation such that $\|\boldsymbol{\mu}_+ - \boldsymbol{\mu}_1^{(1)}\|_2$ is minimal when $\boldsymbol{\mu}_+$ is known and such that $\|\Sigma_+ - \Sigma_1^{(1)}\|_F + \|\Sigma_- - \Sigma_2^{(1)}\|_F$ is minimal when Σ_+, Σ_- are known. No explicit pairing is necessary when $\mathbf{d}$ is known.

possible. This is achieved by updating specific parameters at each iteration, following the updating procedure from Table II. All other design choices for K-means and GMM are identical to the default Matlab R2021a implementation. While LDA is the only true benchmark here (as this is MILDA's target performance), these additional comparisons with K-means and GMMs are added to provide some insights into the strengths and limitations of MILDA as a label-free classifier compared to traditional unsupervised methods.

The LDA model is always trained on the ground-truth statistics that are used to generate the samples, making it the *ideal* LDA model. Similarly, MILDA always uses the known ground-truth statistics to compute the transformations from Table I. This reflects real-life limitations, as sample statistics of test data are generally unknown.

Each experiment is repeated 50 times for the three ground-truth statistics (one class average known, the relative difference between class averages known and the class covariances known). The results are then averaged and shown in Figures 2 to 6. The shaded areas represent the standard deviation for each model.

2) Class separation: Unsurprisingly, Figure 2 shows that a better class separation improves the performance of all models. MILDA performs significantly better than K-means and GMM when the class separation is large enough (in this example when $\|\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-\|_2 > 0.1$). MILDA is slightly worse than LDA for most accuracies. The difference in their accuracy is at most 1%, except for the case where only the class covariances are known (cases 3/4) if the class separation is small. In that case, the MILDA model is very sensitive to disparities between the ground-truth covariances and the estimated sample covariances. However, K-means and GMM models never outperform MILDA, as they suffer even more from such data with poorly separated classes.

3) Correlation between features: From Figure 3, we again see that MILDA and LDA achieve very similar performances, no matter how correlated the features are. K-means is not significantly worse than MILDA when both class covariances are perfectly white, but quickly drops in performance when this is no longer the case. It is significantly worse than MILDA when the correlation between features is high (in this example when $\rho > 0.1$). The GMM improves as features become more correlated, but never comes near the performance of LDA or MILDA.

4) Similarity between covariance matrices: The degree to which Σ_- and Σ_+ are different has no influence on either MILDA or LDA, as shown in Figure 4. MILDA thus reaches similar performances as LDA, no matter how different the class covariances are. K-means does improve as the covariances become more orthogonal.

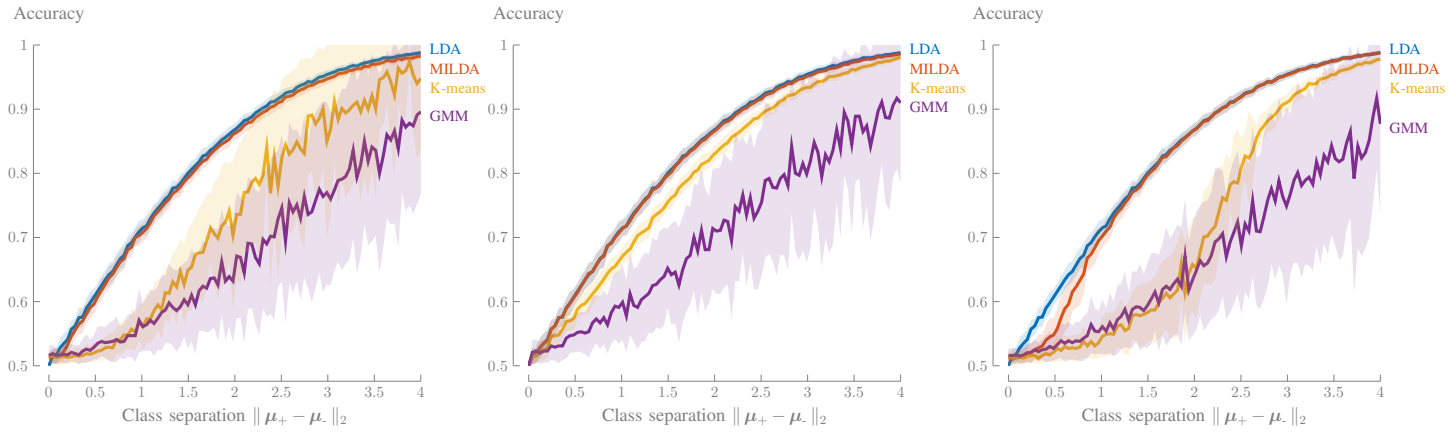
5) Data dimensionality: While the performance of K-means and GMM quickly drops with an increasing data dimensionality in Figure 5 due to the curse of dimensionality [2], MILDA only slowly loses performance. The slight drop in performance for MILDA compared to LDA is caused by the cumulative sum of small disparities between the ground-truth and sample statistics over an increasing data dimensionality. Note that LDA has access to all ground-truth statistics and thus doesn't suffer from these estimation errors. However, in reality, its statistics are often estimated on labelled data. In this case, LDA will also suffer from the curse of dimensionality.

6) Class imbalance: Although LDA and MILDA no longer optimise the same objective function when $q \neq 0.5$ and $\Sigma_+ \not\propto \Sigma_-$, as explained in Section II, they still produce very similar results. This is shown in Figure 6. MILDA is the most different from LDA in the cases where only 1 class average is known (case 1) and when there are very few samples from the other class. This is expected, as here, MILDA knows hardly anything about the second class, making it very difficult to properly estimate the projection vector, as explained in Section III. A similar effect can be seen in K-means. All models reach near 100% accuracy when the classes are extremely imbalanced, which is simply because the chance level also reaches 100%.

B. Example use case: target signal detection

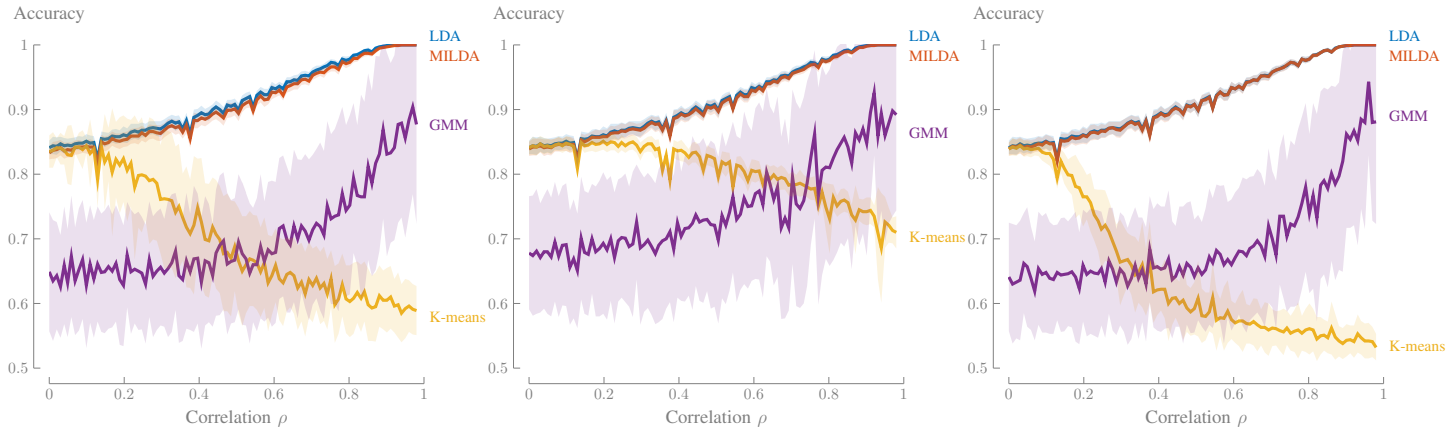
1) Experiment design: To illustrate the use of MILDA in a concrete application use case, we consider a target signal detection problem in a D-channel sensor array. Given a target signal $s(t)$, the observed signal $\mathbf{x}(t) \in \mathbb{R}^{D\times 1}$ can be modeled as $\mathbf{x}(t) = s(t)\mathbf{a} + \mathbf{n}(t)$, with $\mathbf{a} \in \mathbb{R}^{D\times 1}$ the steering vector and $\mathbf{n}(t) \in \mathbb{R}^{D\times 1}$ the noise. All signals are assumed to be stationary. To illustrate the different ways that MILDA can be used, we propose three sub-problems:

- In the first problem, we attempt to detect whether or not the target signal is present, i.e. $s(t) \in \{0, 1\}$. The steering vector $\mathbf{a}$ is unknown, but assumed fixed within the observation window that contains the N samples that have to be classified. Even though the general noise statistics are unknown, the sensors are calibrated such that the noise is zero-mean. One class average ($\boldsymbol{\mu}_{noise} = \mathbf{0}$) is thus known, allowing us to use MILDA with transformation 1 from Table I.



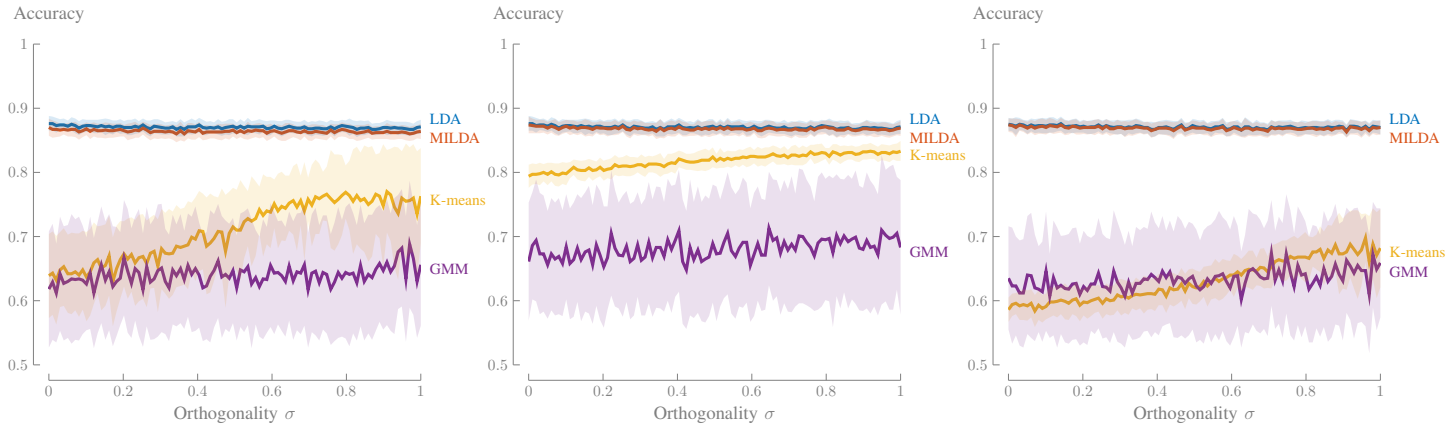
(a) μ_+ known **(b)** $\mathbf{d}$ known **(c)** $\Sigma_+,\ \Sigma_-,\ q$ known

Fig. 2: Accuracy as a function of class separation. All models improve as the class separation grows. LDA and MILDA perform almost equally well, except when only the covariance is known and the class separation is small (Figure 2c). K-means and GMM perform significantly worse for class separations above 0.1.



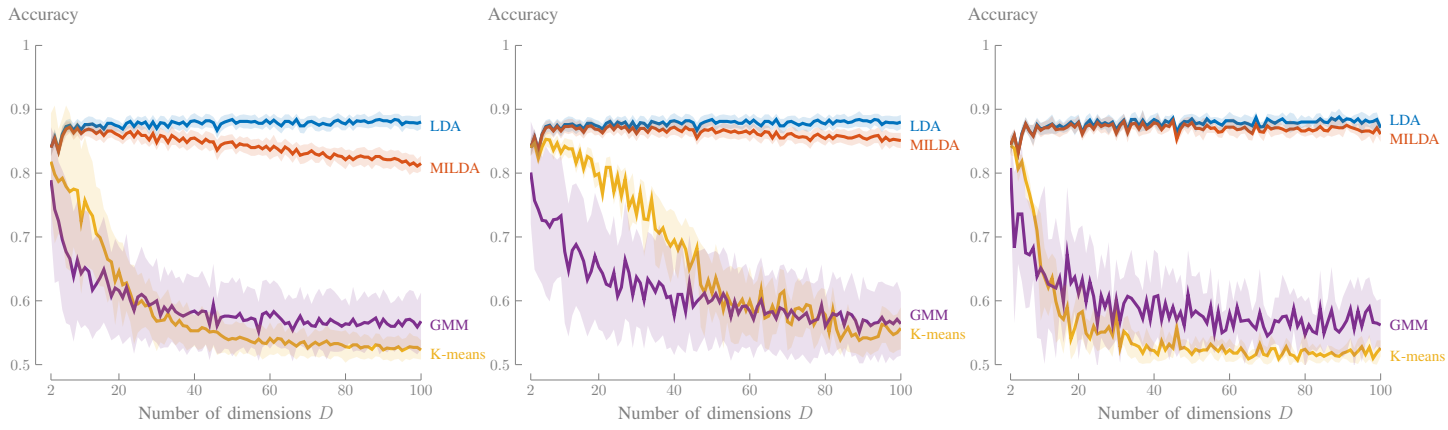
(a) μ_+ known **(b)** $\mathbf{d}$ known **(c)** $\Sigma_+,\ \Sigma_-,\ q$ known

Fig. 3: Accuracy as a function of the correlation between features. If features are more correlated, the classes tend to become more separable. LDA, MILDA and GMM, therefore, improve in performance. K-means drops in performance because it implicitly assumes white class covariance matrices. LDA and MILDA perform nearly equally well for all settings, outperforming K-means and GMM.



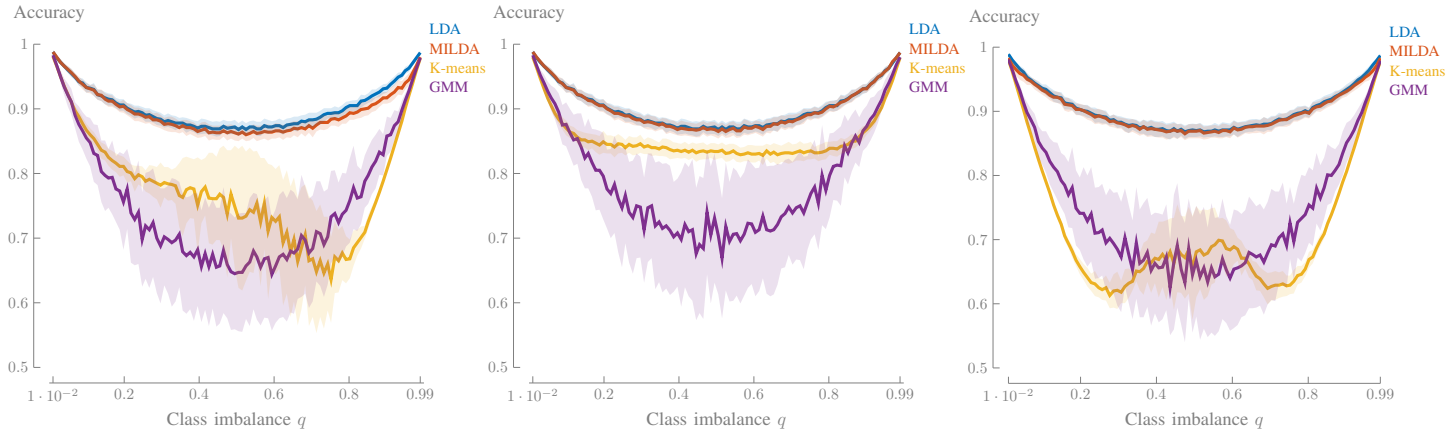
(a) μ_+ known **(b)** $\mathbf{d}$ known **(c)** $\Sigma_+,\ \Sigma_-,\ q$ known

Fig. 4: Accuracy as a function of how similar the class covariances are. Whether or not the two classes have similar class covariances or not has little influence on the performance of any model. Only K-means improves noticeably as the two class covariances become more different.



(a) $\boldsymbol{\mu}_+$ known **(b)** $\mathbf{d}$ known **(c)** Σ_+, Σ_-, q known

Fig. 5: Accuracy as a function of the data dimensionality. GMM and K-means models suffer more from the curse of dimensionality than MILDA. LDA, having access to all ground-truth statistics, is unaffected.



(a) $\boldsymbol{\mu}_+$ known **(b)** $\mathbf{d}$ known **(c)** Σ_+, Σ_-, q known

Fig. 6: Accuracy as a function of the class imbalance. Although MILDA and LDA optimise a slightly different objective when the classes are not balanced, their difference in performance is nearly nonexistent, no matter the class imbalance.

- In the second problem, a binary signal $s(t) \in \{-1/2, 1/2\}$ is sent from a single location. In this case, the signal $s(t)$ is always active (i.e., it is never zero). The steering vector $\mathbf{a}$ is known (up to an arbitrary scaling) [21]. The statistics of $s(t)$ and $\mathbf{n}(t)$ are assumed to be unknown. In this case, we know that the relative difference between both classes averages is proportional to the steering vector ($\boldsymbol{\mu}_+ - \boldsymbol{\mu}_- \propto \mathbf{a}$). We can therefore use MILDA with transformation 2 from Table I.
- In the third problem, two different locations interchangeably transmit the signal $s(t) = 1$ while the other location is silent. For each observation $\mathbf{x}(t)$, we aim to detect which location transmitted the signal. The locations (and corresponding steering vectors) are unknown, similar to case 1. The noise covariance matrix $S = \mathbb{E}[\mathbf{n}(t)\mathbf{n}^\top(t)]$ is assumed to be known. It is therefore possible to use MILDA with transformation 4 from Table I.

We set $D = 10$ and perform multiple Monte-Carlo runs. In each run, $N = 1000$ samples are created with equal probability between the two detection classes of the corresponding experiment, i.e., $N/2$ of the samples belong to class 0, the other $N/2$ belong to class 1. The noise is modelled as $\mathbf{n}(t) = \mathbf{g}(t) + \mathbf{f}(t)$, with Gaussian noise $\mathbf{g}(t) \sim \mathcal{N}(0.3\,\mathrm{diag}(\boldsymbol{\sigma}) + 0.7J)$ and a high-frequency interference signal $\mathbf{f}(t) = \boldsymbol{\alpha} \odot \sin(\pi/2t + \boldsymbol{\phi})$. The variances $\boldsymbol{\sigma} \in \mathbb{R}^{D\times 1}$ and amplitudes $\boldsymbol{\alpha} \in \mathbb{R}^{D\times 1}$ are randomly sampled from the uniform distribution $\mathcal{U}_D(\mathbf{0}, \mathbf{1})$. The phase for each dimension $\boldsymbol{\phi} \in \mathbb{R}^{D\times 1}$ is sampled from $\mathcal{U}_D(\mathbf{0}, \mathbf{2\pi})$. The steering vectors are sampled from $\mathcal{U}_D(-\mathbf{1}, \mathbf{1})$. Each experiment is repeated 100 times. All models are trained in the same way as in Section IV-A. Since LDA has access to information that wouldn't be available in the described scenarios, it merely serves as a benchmark.

This experiment is conducted with the primary objective of demonstrating the practical applicability of MILDA within a specific use-case context. The intent is not to assert superiority over state-of-the-art target detection algorithms. The experiments are purposefully simplified for clarity.

2) Results: The detection accuracy for each model and each experiment is displayed in Table III. Similar to the previous experiments, LDA and MILDA achieve comparable

performances and consistently outperform the GMM and K-means models. The inclusion of non-Gaussian noise also demonstrates that MILDA does indeed not require the assumption of Gaussianity to be comparable to LDA.

Problem	LDA	MILDA	K-means	GMM
Zero-mean noise	93.4 (4.4)	92.8 (4.4)	63.4 (5.6)	87.5 (14.3)
Binary signal	83.2 (4.4)	82.9 (4.4)	76.6 (5.9)	64.4 (9.1)
Noise covariance	90.3 (4.6)	90.2 (4.6)	64.6 (11.1)	63.9 (10.2)

TABLE III: The average accuracy (%) and standard deviation between brackets for each model. MILDA performs only slightly worse than LDA, even though it requires significantly less information. K-means and GMM always perform significantly worse, even with access to the same ground-truth information as MILDA.

C. Comparison on non-stationary data

1) Experiment design: Until now, all experiments were performed on stationary data. However, there is a vast amount of classification problems where the stationarity assumption does not hold. Instead, the class distributions can shift and morph over time. Ideally, the classification models should adapt to these changes. While this is generally possible for MILDA (assuming the ground-truth statistic is stationary), this is rarely the case for LDA. It rarely has access to labels to re-estimate the relevant statistics at inference time to recompute the LDA projection vector.

To build further on the first problem from Section IV-B, we illustrate the effects of non-stationary data on binary target signal detection in zero-mean noise by introducing three changes in the data and tracking the accuracy over time:

- The first change is a sudden change in the noise covariance from Σ_1 to Σ_2.
- The second change is a *gradual* drift in both the steering vector from $\mathbf{a}_1$ to $\mathbf{a}_2$ and the noise covariance from Σ_2 to Σ_3.
- The third change is a sudden change in the steering vector from $\mathbf{a}_2$ to $\mathbf{a}_3$.

Similar to the first problem, $s(t) \in \{0, 1\}$ and $\mathbf{a} \sim \mathcal{U}_D(-\mathbf{1}, \mathbf{1})$. The noise is modelled in the same way as in Section IV-B. The signal and noise statistics are changed by respectively resampling $\mathbf{a}$ and $\boldsymbol{\sigma}$, $\boldsymbol{\alpha}$, $\boldsymbol{\phi}$. During the gradual change, each parameter is changed by linearly interpolating between the value at the start and at the end of the gradual change. The norm of the steering vector is also changed by linearly interpolating between $\|\mathbf{a}_1\|$ and $\|\mathbf{a}_2\|$. This guarantees that the expected class separation is constant during the gradual shift.

We performed 1000 Monte-Carlo runs. 10000 samples are classified during each run, with a sudden change in noise covariance at sample 2500, a gradual change between samples 5000 and 7500 and a sudden change in the steering vector at sample 7500. MILDA, K-means and GMM are adaptively retrained using a moving window covering the last 500 samples. LDA is trained using the original statistics Σ_1 and $\mathbf{a}_1$ and thus does not adapt to the changes in statistics (as there are no labels available to update the model). Since the noise is always zero-mean, MILDA, K-means and GMM use $\boldsymbol{\mu}_{noise} = \mathbf{0}$ as the stationary ground-truth statistic.

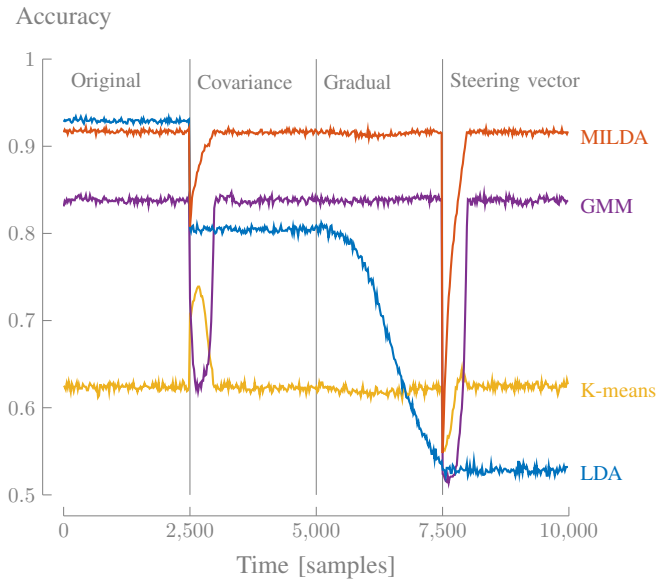


Fig. 7: Comparison of LDA, MILDA, K-means and GMM on non-stationary data. Since MILDA, K-means and GMM do not require labels to train, it can adapt to both sudden and gradual changes in the statistics.

2) Results: The average accuracy of the four models is shown in Figure 7. As expected, LDA slightly outperforms MILDA on samples that are drawn from the same distribution as it was trained on. LDA and MILDA equally drop in performance when sudden changes are introduced in the statistics. However, MILDA completely recovers after 500 samples, which is the width of the moving window MILDA was trained on. Remarkably, MILDA barely drops in performance during the gradual change in statistics (on average only 0.25%), showing that the model is well able to track gradual changes in statistics. This is also true for GMM and K-means, although they always perform significantly worse than MILDA ($p < 1e-5$). As expected, LDA continuously drops in performance as more statistics are changed. Since it is unable to adapt due to the lack of labels, it is also unable to recover from these changes.

V. Computational cost

Simple models such as K-means and LDA are often chosen over more complex models such as neural networks because of their low computational cost, which is particularly relevant in adaptive realizations where the classifier is continuously updated when new data becomes available. It is therefore relevant to study whether MILDA has a similar computational cost. Table IV shows the average time (in ms) each model required to train using all default parameters from Section IV. MILDA is only slightly slower than LDA, but this is mainly because LDA has access to the ground-truth covariance matrices. If LDA also needs to estimate Σ_+ and Σ_- from data, there is no longer a difference in computation time. The K-means and GMM models take over 10x longer to compute in this specific experiment.

To examine the computational cost to train each model at much larger scales, it can be useful to examine their asymptotic complexity. It is well known that LDA has a time complexity of $\mathcal{O}(ND^2) + \mathcal{O}(D^3)$, with N the number of samples and D the number of features (i.e. the dimension of $\mathbf{x}_i$). Similarly, K-means has a time complexity of $\mathcal{O}(ND)$ [12] and GMM's have a time complexity of $\mathcal{O}(ND^2I) + \mathcal{O}(D^3I)$, with I the number of iterations that GMM's train (assuming naive computations of Σ^{-1} for LDA and GMM's). MILDA has the same time complexity as LDA, namely $\mathcal{O}(ND^2) + \mathcal{O}(D^3)$. Since GMM's need to make similar computations as MILDA, but repeated at every iteration, a GMM model will always be significantly slower than a MILDA model. MILDA is also expected to be faster than K-means if the data dimensionality D is relatively small. In the aforementioned experiments, K-means was faster than MILDA for a data dimensionality $D > 80$ at $N = 1000$ samples and $D > 120$ for $N = 10000$ samples (although K-means no longer performed better than chance in these cases, while MILDA barely dropped in performance, as was shown in Figure 5).

When MILDA is used in an adaptive context, its computational cost can be further reduced to $\mathcal{O}(D^2)$ per iteration by using similar techniques as used in, e.g., recursive least-squares (RLS) [1]. Furthermore, only the global covariance matrix and global average need to be stored in memory, making the model much less memory-intensive than e.g. an adaptive K-means model, where all the data needs to be stored.

Even though the transformations required in MILDA do require some additional computations compared to a simple LDA model, these transformations are much cheaper to compute than the estimation of the covariance matrices (assuming $N \gg D$). MILDA is, therefore, not significantly more expensive than LDA when LDA also needs to estimate the covariance matrix from the data.

Prior knowledge	LDA	MILDA	K-means	GMM
$\boldsymbol{\mu}_+$	0.24	0.31	3.0	4.7
$\mathbf{d}$	0.24	0.44	5.6	4.7
$\Sigma_{+/-}$	0.24	0.46	5.7	4.8

TABLE IV: MILDA and LDA require about 10x less time to classify the data than K-means and GMM (time in [ms]).

VI. Conclusion & Future Work

When discriminating between two classes, (Fisher's) Linear Discriminant Analysis (LDA) requires knowledge of the class averages and class covariance matrices to work optimally, which usually have to be estimated from the data based on available class labels. If no labels are available, one would need to know all ground-truth statistics up to a high precision to use LDA, which rarely is the case in practice. However, at least one ground-truth statistic is often known, e.g. a single class average, the relative difference between two class averages or the class covariance matrices. In this paper, we have proposed the minimally informed LDA (MILDA) model that can calculate the same projection vector as LDA without requiring labelled data, assuming prior knowledge of only one of these ground-truth statistics.

We have mathematically proven that the MILDA projection vector is equivalent to the LDA projection vector under the mild assumption that the classes are balanced or that the class covariances have a similar shape. When both assumptions are violated, MILDA optimises a slightly different objective function, yet we could not find a noteworthy difference in performance between both. Furthermore, we have shown both mathematically and empirically that, in general, MILDA has the nice property of being quite robust to approximation errors in cases where the classes are harder to separate, the class covariance matrices are less white or when the classes are more unbalanced, which are all cases where unsupervised methods typically have difficulties with.

The performance of MILDA is comparable to the performance of an ideal LDA projection vector in nearly all studied cases. The relative difference in performance is nearly unaffected by the class separation, the correlation between different features, how similar the covariance matrices of both classes are or the class imbalance. MILDA is somewhat sensitive to outliers, except for the case where the relative difference between the class averages is known.

The MILDA projection vector can be computed in closed form with a computational cost comparable to the LDA projection vector. Because MILDA has a low computational cost to train and does not require any labels, it is well-suited for use in adaptive classifiers that update themselves over time using the most recent data.

Although MILDA is promising on its own and can be used in a wide array of applications, the necessity to know at least one ground-truth statistic reduces the utility of MILDA in some research fields. To go from a minimally informed model to a truly unsupervised model, MILDA should replace the necessity to know one ground-truth statistic with an assumption on the data. Some assumptions are trivial to incorporate in MILDA, such as the assumption that the class covariances are white or that one class average is zero (respectively use case 4 and case 1 from Table I). Other assumptions, such as the assumption that the classes are Gaussian or that the features are uncorrelated, but not white, are not as trivial to incorporate. Therefore, further research on how these assumptions can be leveraged in MILDA can be vital to further expand the applicability of MILDA.

References

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY: Springer New York, 2009.

[2] C. Bouveyron, G. Celeux, T. B. Murpy, and A. E. Raftery, *Model-Based Clustering and Classification for Data Science*. Cambridge University Press, 2019.

[3] N. Auguin, D. Morales-Jimenez, and M. McKay, "Large-Dimensional Characterization of Robust Linear Discriminant Analysis," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2625–2638, 2021.

[4] J. Liu, X. Wang, R. Wang, C. Xu, R. Zhao, H. Li, S. Zhang, and X. Yao, "Near-infrared auto-fluorescence spectroscopy combining with Fisher's linear discriminant analysis improves intraoperative real-time identification of normal parathyroid in thyroidectomy," *BMC Surgery*, vol. 20, no. 1, pp. 1–7, 1 2020.

[5] C. W. Antuvan and L. Masia, "An LDA-Based Approach for Real-Time Simultaneous Classification of Movements Using Surface Electromyography," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 27, no. 3, pp. 552–561, 2019.

[6] C. Ricciardi, A. S. Valente, K. Edmund, V. Cantoni, R. Green, A. Fiorillo, I. Picone, S. Santini, and M. Cesarelli, "Linear discriminant analysis and principal component analysis to predict coronary artery disease," *Health Informatics Journal*, vol. 26, no. 3, pp. 2181–2192, 2020.

[7] F. Zhu, J. Gao, J. Yang, and N. Ye, "Neighborhood linear discriminant analysis," *Pattern Recognition*, vol. 123, p. 108422, 2022.

[8] K. Elkhalil, A. Kammoun, R. Couillet, T. Y. Al-Naffouri, and M. S. Alouini, "A Large Dimensional Study of Regularized Discriminant Analysis," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2464–2479, 2020.

[9] R. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 9 1936.

[10] R. O. Duda and P. E. Hart, *Pattern classification and scene analysis*. JSTOR, 1974, no. 4.

[11] S. P. Lloyd, "Least Squares Quantization in PCM," in *IEEE Transactions on Information Theory*, vol. 28, no. 2, 1982, pp. 129–137.

[12] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[13] F. De La Torre and T. Kanade, "Discriminative cluster analysis," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 241–248.

[14] P. Deng, H. Wang, T. Li, S. J. Horng, and X. Zhu, "Linear discriminant analysis guided by unsupervised ensemble learning," *Information Sciences*, vol. 480, pp. 211–221, 4 2019.

[15] F. Wang, Q. Wang, F. Nie, Z. Li, W. Yu, and R. Wang, "Unsupervised Linear Discriminant Analysis for Jointly Clustering and Subspace Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 3, pp. 1276–1290, 2021.

[16] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," *Proceedings of the IEEE International Conference on Computer Vision*, 2007.

[17] S. Wang, J. Lu, X. Gu, H. Du, and J. Yang, "Semi-supervised linear discriminant analysis for dimension reduction and classification," *Pattern Recognition*, vol. 57, pp. 179–189, 2016.

[18] D. Peña and F. J. Prieto, "The kurtosis coefficient and the linear discriminant function," *Statistics & Probability Letters*, vol. 49, no. 3, pp. 257–261, 9 2000.

[19] R. Martín-Clemente and V. Zarzoso, "LDA via L1-PCA of Whitened Data," *IEEE Transactions on Signal Processing*, vol. 68, p. 225, 2020.

[20] M. A. Woodbury, *Inverting modified matrices*. Princeton, NJ: Department of Statistics, Princeton University, 1950.

[21] S. Markovich, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 6, pp. 1071–1086, 8 2009.

[22] Y. Gu and A. Leshem, "Robust adaptive beamforming based on interference covariance matrix reconstruction and steering vector estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3881–3885, 7 2012.

[23] A. Khabbazibasmenj, S. A. Vorobyov, and A. Hassanien, "Robust adaptive beamforming based on steering vector estimation with as little as possible prior information," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2974–2987, 6 2012.