# Learning from Low Rank Tensor Data: A Random Tensor Theory Perspective

**Mohamed El Amine Seddik**[1] **Malik Tiomoko**[2] **Alexis Decurninge**[2] **Maxim Panov**[1] **Maxime Guillaud**[3]

[1]Technology Innovation Institute, PO Box: 9639, Masdar City, Abu Dhabi, UAE
[2]Huawei Technologies France, Paris, France
[3]Inria / CITI Laboratory, 6 avenue des Arts, 69621 Villeurbanne, France

## Abstract

Under a simplified data model, this paper provides a theoretical analysis of learning from data that have an underlying low-rank tensor structure in both supervised and unsupervised settings. For the supervised setting, we provide an analysis of a Ridge classifier (with high regularization parameter) with and without knowledge of the low-rank structure of the data. Our results quantify analytically the gain in misclassification errors achieved by exploiting the low-rank structure for denoising purposes, as opposed to treating data as mere vectors. We further provide a similar analysis in the context of clustering, thereby quantifying the exact performance gap between tensor methods and standard approaches which treat data as simple vectors.

## 1 INTRODUCTION

The current era of artificial intelligence tackles learning tasks leveraging millions or even billions of data. These data lie in high-dimensional spaces and often come from multiple *modes*, such as multiple modalities, multiple sensors, multiple sources, multiple types, and multiple (space, time, frequency, etc.) domains. In other words, these data can naturally be seen as tensors, in which vectors and matrices are simply the 1-mode and 2-mode versions.

Tensors are a natural way to store data and their inner geometric structure is richer than the one-dimensional and the two-dimensional algebra [Landsberg, 2012]. In particular, unlike matrices, low-rank tensor factorization is essentially unique under mild assumptions when the number of modes is greater than three. Their ubiquity in numerous applications makes them increasingly important [Sun et al., 2014], leading to a growing interest in tensor data analysis in the statistical learning community.

A large part of previous works on tensor theory applied to machine learning problems assume a low-rank representation of input data [Anandkumar et al., 2014, Kadmon and Ganguli, 2019] and estimate this representation using as main ingredient the CANDECOMP/PARAFAC decomposition (CPD; Hitchcock [1927]). Indeed, the low-rank tensor structure is a natural sparsity hypothesis in the modeling of real data seen through high-dimensional inputs [Kolda and Bader, 2019]. However, faced with tensor-structured data, a simple and commonly used approach consists in neglecting the tensor structure and reshaping them into a set of vectors, to which a classical machine learning algorithm is then applied. In this work, we aim at analyzing simple machine-learning methods and quantifying their exact theoretical performances when neglecting versus considering the low-rank structure, thereby quantifying the theoretical gap between tensor methods and their vectorized counterparts.

In the literature, the low-rank tensor structure has been considered for example in tensor regression in a supervised setting [Zhou et al., 2013] or clustering in an unsupervised setting [Sun and Li, 2019]. The tensor structure has been shown to enhance the performance of learning models as a key ingredient of more complex learning architectures e.g. for multi-modal data or multi-spectral images [Liang et al., 2019, Chen et al., 2020], or in the design of advanced neural network architectures by replacing the flattening operation in fully connected layers of a Convolutional Neural Network by CPD-based operations [Kossaifi et al., 2020].

On top of the performance gain shown by Kossaifi et al. [2020], the reduction of the number of parameters needed to describe the learned model is also significant. Indeed, the gain in the size of the parameter space can be seen when the data samples are $k$-order tensors and have for example a rank-one underlying structure. In this case, if the dimensions of the tensor are $p_1 \times \cdots \times p_k$, the dimension of the parameter space can be significantly reduced from $\prod_{j=1}^k p_j$ to $\sum_{j=1}^k p_j$.

All this literature motivates the analysis of learning algo-

*Accepted for the 39th Conference on Uncertainty in Artificial Intelligence* (UAI 2023).

rithms when processing low-rank tensor structured data. To do so, we consider a simple framework where data are supposed to be low-rank tensors perturbed by some additive noise. The proposed framework directly extends the fundamental settings of binary classification in the vector case Mignacco et al. [2020], Wang and Thrampoulidis [2022]. Then, based on some random tensor theory tools (recalled in Section 2 of the supplementary material), we characterize the theoretical performance of linear methods (in both supervised and unsupervised settings) with and without incorporating the knowledge of the low-rank structure. We show analytically that the incorporation of this knowledge allows us to considerably improve the performance of the studied methods, in particular when the amount of training samples is limited or equivalently when data are high-dimensional. Thus, exploiting the structure of the data allows for obtaining equivalent performance with far fewer samples.

In this work, we limit our attention to a simple, tractable framework where data are generated as rank-one tensors with additive Gaussian noise (see Section 2). The main contributions brought by this paper are two-fold:

1. We first consider a supervised learning setting where we provide a theoretical analysis of a Ridge classifier with and without incorporating the low-rank tensor structure of the data; see Section 3.1. The results extend the known misclassification in a vector case. Importantly, we show that the clever usage of a low-rank structure allows for significant improvement in classification performance, which is further quantified.

2. We also consider an unsupervised setting by analyzing a linear clustering approach and a low-rank tensor counterpart (Section 3.2). Our analysis provides the theoretical conditions for which efficient clustering is possible both theoretically and algorithmically. In passing, we precisely quantify the performance gap between linear versus tensor-based clustering, thereby demonstrating the superiority of the latter in the considered setting.

To the best of our knowledge, few works in the literature were focused on the *exact characterization* of the performance of ML methods when processing tensor data with low-rank structures, even under our considered setting. This paper suggests new directions to fill in this gap leveraging on recent advances in random tensor theory (RTT). We demonstrate how RTT allows for the exact characterization of the performance of the considered methods while confirming practical insights about learning from low-rank tensor data. In particular, our results highlight that *it takes fewer training samples to achieve better performances when the low-rank tensor structure of the data is leveraged.*

**Notations:** $[n]$ denotes the set $\{1, \dots, n\}$. Scalars are denoted by lowercase letters as $a, b, c$. Vectors are denoted by bold lowercase letters as $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$. Matrices are denoted by bold uppercase letters as $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$. Tensors are denoted as $\mathbf{A}, \mathbf{B}, \mathbf{C}$. $T_{i_1,\dots,i_d}$ denotes the entry $(i_1, \dots, i_d)$ of the tensor $\mathbf{T}$. The inner product between two order-$d$ tensors $\mathbf{A}$ and $\mathbf{B}$ is denoted $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{i_1,\dots,i_d} A_{i_1\dots i_d} B_{i_1\dots i_d}$. The $\ell_2$-norm of $\mathbf{A}$ is denoted $\|\mathbf{A}\| = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$. For any vectors $\boldsymbol{u}_1, \dots, \boldsymbol{u}_d$, contractions of a tensor $\mathbf{A}$ are denoted by $\mathbf{A}(\boldsymbol{u}_1, \dots, \boldsymbol{u}_d) = \sum A_{i_1\dots i_d} u_{1i_1} \dots u_{di_d}$. The notation $\bigotimes_{i=1}^k \boldsymbol{v}_i$ stands for the tensor outer product between the vectors $\boldsymbol{v}_1, \dots, \boldsymbol{v}_k$ with $[\bigotimes_{i=1}^k \boldsymbol{v}_i]_{i_1\dots i_k} = \prod_{j=1}^k v_{ji_j}$. $\mathrm{Mat}_i(\mathbf{T})$ denotes the matrix obtained by unfolding the tensor $\mathbf{T}$ w.r.t. its $i$-th mode. $\mathbf{T} \times_i \boldsymbol{u}$ denotes the contraction of the tensor $\mathbf{T}$ on the vector $\boldsymbol{u}$ through mode $i$. $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$ corresponds to the Gaussian tail function. $\xrightarrow{\text{a.s.}}$ stands for the almost sure convergence and $\xrightarrow{\mathcal{D}}$ for the convergence in distribution. $\mathbb{S}^{d-1}$ stands for the unit sphere in dimension $d$. We refer the reader to the supplementary material for definitions of tensor notations.

## 2 STATISTICAL DATA MODEL

Let us start from the classical prototypical model for the binary classification with the covariate $\boldsymbol{x} \in \mathbb{R}^p$ belonging to one of the two classes $\mathcal{C}_1$ or $\mathcal{C}_2$:

$$\boldsymbol{x} = (-1)^a \boldsymbol{\mu} + \boldsymbol{z} \in \mathbb{R}^p \tag{1}$$

with $a = 1$ for class $\mathcal{C}_1$ or $a = 2$ for class $\mathcal{C}_2$ (thus, class centroids are $\pm\boldsymbol{\mu} \in \mathbb{R}^p$), and random noise $\boldsymbol{z} \in \mathbb{R}^p$. The optimal estimation procedures and rates in this model were studied extensively in the literature. Recent works of Mignacco et al. [2020] and Wang and Thrampoulidis [2022] showed that asymptotically the optimal misclassification error behaves as $Q\left(\frac{m}{\sigma}\right)$ with $m = \sqrt{\frac{n}{p}} \|\boldsymbol{\mu}\|^2$ and $\sigma = \sqrt{\frac{n}{p} \|\boldsymbol{\mu}\|^2 + 1}$, where $n$ is a sample size.

In this work, we aim to extend the fundamental result above to more complex tensor-structured data. Let the observed samples be $n$ independent tensors $\mathbf{X}_1, \dots, \mathbf{X}_n$ each of order $k$ and of dimension $p_1 \times \dots \times p_k$. We denote the dimensions $p = \sum_{j=1}^k p_j$ and $P = \prod_{j=1}^k p_j$. We suppose that the $\mathbf{X}_i$'s are distributed in two classes $\mathcal{C}_1$ and $\mathcal{C}_2$ (of cardinality $n_1$ and $n_2$ respectively – that is $n = n_1 + n_2$), such that for $\mathbf{X}_i \in \mathcal{C}_a$ with $a \in \{1, 2\}$,

$$\mathbf{X}_i = (-1)^a \bigotimes_{j=1}^k \boldsymbol{\mu}_j + \mathbf{Z}_i \in \mathbb{R}^{p_1 \times \dots \times p_k}, \tag{2}$$

where $\mathbf{Z}_i$ is a random tensor with i.i.d. standard Gaussian entries, $\boldsymbol{\mu}_j \in \mathbb{R}^{p_j}$ for $j \in [k]$ are independent from the $\mathbf{Z}_i$'s and $\mathbf{M} = \bigotimes_{j=1}^k \boldsymbol{\mu}_j$ stands for the outer product between all the $\boldsymbol{\mu}_j$'s. Here, the rank-1 tensor term represents the informative part of the data, while $\mathbf{Z}_i$ models corruption by

additive noise. In the context of supervised binary classification, we are further given a vector of labels $\boldsymbol{y} \in \mathbb{R}^n$ such that $y_i = -1$ for $\mathbf{X}_i \in \mathcal{C}_1$ and $y_i = 1$ for $\mathbf{X}_i \in \mathcal{C}_2$. Importantly, the model for the vector case (1) is a particular instance of the tensor model (2) with $k = 1$.

Note that in this formulation, the noise variance is assumed constant, and the difficulty of the classification problem is controlled by the between-class distance $\|\mathbf{M}\|$. Specifically, when $\|\mathbf{M}\| = 0$ the classification is impossible whereas when $\|\mathbf{M}\|$ is very large the classification becomes trivial. We also highlight that the classical high dimensional statistical model corresponds to the case $k = 1$, and we consider a more general setting by taking any $k \geq 1$.

We denote the observed data tensor $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_n] \in \mathbb{R}^{p_1 \times \cdots \times p_k \times n}$ by concatenating all the $\mathbf{X}_i$ along the $(k+1)$-th mode of dimension $n$. $\mathbf{X}$ expresses in tensor form as

$$\mathbf{X} = \mathbf{M} \otimes \boldsymbol{y} + \mathbf{Z}, \tag{3}$$

where $\mathbf{Z} = [\mathbf{Z}_1, \ldots, \mathbf{Z}_n] \in \mathbb{R}^{p_1 \times \cdots \times p_k \times n}$. Given the rank-one structure of the tensor mean $\mathbf{M}$, the outer product $\mathbf{M} \otimes \boldsymbol{y}$ results in a rank-one tensor of order $k+1$. As such, the data tensor $\mathbf{X}$ is a *rank-one spiked random tensor model* of order $k+1$, where the signal part is $\mathbf{M} \otimes \boldsymbol{y}$ and $\mathbf{Z}$ corresponds to the noise part.

**Remark 2.1** (On the data model)**.** *Note that the RTT analysis presented below (following [Seddik et al., 2021a]) extends trivially to a more general (rank-$r$) data model of the form $\sum_{i=1}^r \bigotimes_{j=1}^k \boldsymbol{\mu}_j^{(i)} + \mathbf{Z}$ as long as the $\boldsymbol{\mu}_j^{(i)}$'s are orthogonal and $r$ of order $O(1)$. On the other hand, for arbitrary $\boldsymbol{\mu}_j^{(i)}$'s, the analysis is non-trivial – see the end of Section 3.1 and the supplementary material for more details.*

Throughout the following sections, we assume a high-dimensional regime, i.e., the number of training samples $n$ scales linearly with the tensor dimensions $p_j$ while $\|\boldsymbol{\mu}_j\|$ remains constant.

**Assumption 2.2** (Growth rate)**.** *For all $j \in [k]$, $\frac{p_j}{n} = \mathcal{O}_n(1)$ and $\|\boldsymbol{\mu}_j\| = \mathcal{O}_n(1)$*[1]*.*

This is a classical assumption in learning theory and random matrix theory [Pennington and Worah, 2017, Louart et al., 2018, Ali and Couillet, 2017, Mai and Couillet, 2018, Tiomoko et al., 2020, Seddik et al., 2021b], which considers that the feature size scales linearly with the number of samples. Indeed, such scaling coincides with the case $k = 1$ in Assumption 2.2. Moreover, Assumption 2.2 is more realistic from the practical viewpoint in scenarios where a limited amount of samples is available, contrary to classical statistical settings which make the assumption that $p_i$ is fixed while $n \to \infty$.

[1]The notation $a = \mathcal{O}_n(1)$ means that $a$ converges to a constant not depending on $n$ if $n \to \infty$.

# 3 MAIN RESULTS

## 3.1 SUPERVISED SETTING

Given the training data tensor $\mathbf{X}$ in (3) and the corresponding labels vector $\boldsymbol{y}$, a simple learning approach consists in reshaping $\mathbf{X}$ into a data matrix $\boldsymbol{X} \equiv \mathrm{Mat}_{k+1}(\mathbf{X}) \in \mathbb{R}^{n \times P}$ with $P = \prod_{j=1}^k p_j$, and then training a Ridge classifier with some regularization parameter $\gamma \geq 0$, i.e.,

$$\min_{\boldsymbol{w}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2 + \gamma \|\boldsymbol{w}\|^2, \tag{4}$$

the solution of which writes explicitly as $\boldsymbol{w}^* = \left(\boldsymbol{X}^\top \boldsymbol{X} + \gamma \boldsymbol{I}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$. Since the two classes corresponding to the data model in (2) are only separable through their means ($-\mathbf{M}$ and $\mathbf{M}$) and have the same covariance, we consider the study of the Ridge classifier for $\gamma \gg \|\mathbf{X}^\top \mathbf{X}\|$ which we refer to as $\infty$-*Ridge classifier*[2]. Therefore, the $\infty$-Ridge classifier consists in projecting the data matrix $\boldsymbol{X}$ on the labels $\boldsymbol{y}$ as[3]

$$\boldsymbol{w} = \frac{1}{\sqrt{np}} \boldsymbol{X}^\top \boldsymbol{y}, \tag{5}$$

where we recall that $p = \sum_{j=1}^k p_j$, for which the decision function (for a new datum $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$) is given by $f_{\mathrm{R}}(\tilde{\mathbf{X}}_i) = \langle \boldsymbol{w}, \mathrm{vec}(\tilde{\mathbf{X}}_i) \rangle$. This is equivalent in tensor notations to

$$f_{\mathrm{R}}(\tilde{\mathbf{X}}_i) = \langle \mathbf{W}, \tilde{\mathbf{X}}_i \rangle \underset{\mathcal{C}_2}{\overset{\mathcal{C}_1}{\lessgtr}} 0, \quad \mathbf{W} \equiv \frac{1}{\sqrt{np}} \mathbf{X} \times_{k+1} \boldsymbol{y}. \tag{6}$$

As such, the $\infty$-Ridge classifier does not leverage the low-rank tensor structure of the underlying data model and treats the data as mere vectors. Our first result consists in characterizing the theoretical performance of the $\infty$-Ridge classifier for the data model in (3):

**Theorem 3.1** (Performance of the $\infty$-Ridge classifier)**.** *Under Assumption 2.2, for $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$ with $a \in \{1, 2\}$ independent from the training set $\mathbf{X}$,*

$$\frac{1}{\sigma}\left(f_R(\tilde{\mathbf{X}}_i) - m_a\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, 1),$$

*where $m_a = (-1)^a \|\mathbf{M}\|^2 \sqrt{\frac{n}{p}}$ and $\sigma = \sqrt{\frac{n}{p}\|\mathbf{M}\|^2 + \frac{P}{p}}$. Moreover, the misclassification error verifies with probability one $\mathbb{P}\left((-1)^a f_R(\tilde{\mathbf{X}}_i) < 0 \mid \tilde{\mathbf{X}}_i \in \mathcal{C}_a\right) - Q\left(\frac{|m_a|}{\sigma}\right) \to 0$.*

*Proof.* See supplementary material. □

[2]Known as the matched filter classifier in some literature and is proven to be optimal for the model in (2) when $k = 1$, as stated in [Tiomoko et al., 2021].

[3]The normalization by $\sqrt{np}$ is considered for convenience and does not affect the performances of the considered methods. Moreover, under Assumption 2.2 the quantities $n$ and $p$ are of the same order which is equivalent to the standard normalization by $n$.

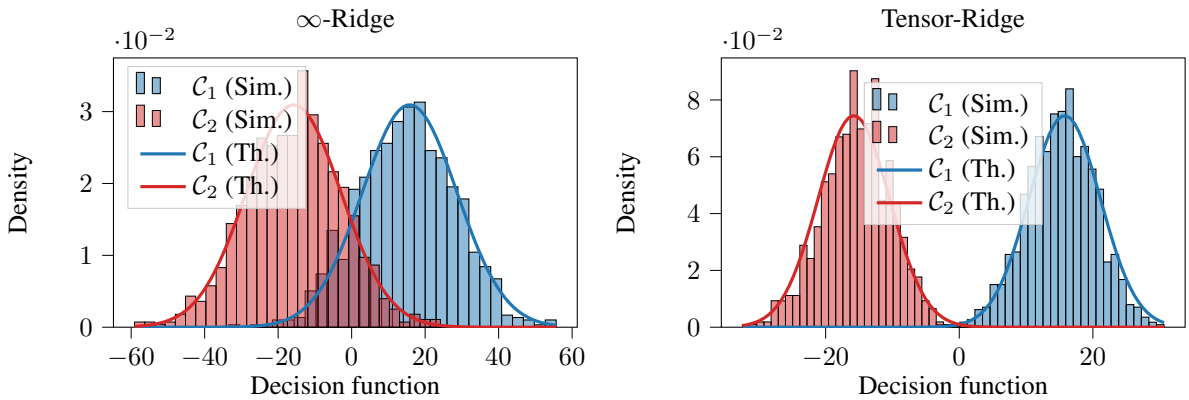


Figure 1: Theoretical versus empirical histogram of the decision function $f_{\mathrm{R}}(\tilde{\mathbf{X}}_i)$ for the $\infty$-Ridge classifier as per Theorem 3.1 (left) and for the Tensor-Ridge classifier as per Theorem 3.3 (right). We considered $n = 200$ training data ($n_1 = n_2 = 100$) that are tensors of shape $(15, 30, 20)$, distributed as the rank-one tensor model in (2) with the $\boldsymbol{\mu}_j$'s being randomly sampled vectors from spheres such that $\|\mathbf{M}\| = 3$.

Theorem 3.1 states that the performance of the $\infty$-Ridge classifier depends solely on $\|\mathbf{M}\|$ and the dimension ratios $\frac{n}{p}$ and $\frac{P}{p}$. Note that in classical high dimension statistics (e.g., $k = 1$), the ratios $\frac{n}{p}$ and $\frac{P}{p}$ are constant as $n \to \infty$. While in the actual tensor setting for $k \geq 2$, the dimension $P$ has a polynomial growth in terms of $n$. Therefore, Theorem 3.1 is more general since it captures the behavior of both regimes. Moreover, since the data are mean-wise centered as per (2), the optimal classification is obtained by taking the sign of the decision function which is also suggested theoretically since the optimal threshold is $\frac{m_1+m_2}{2} = 0$. Figure 1 (left) provides a histogram of the decision function of the $\infty$-Ridge classifier and its theoretical density.

**Tensor-based approach:** To improve classification accuracy, ones needs to retrieve the rank-one structure $\mathbf{M}$ from the data. This can be performed by denoising $\mathbf{W}$, specifically by replacing it with a low-rank tensor approximation, since it is a noisy version of $\mathbf{M}$. Precisely, from the definition of $\mathbf{W}$ in (6) and $\mathbf{X}$ in (3),

$$\mathbf{W} = \sqrt{\frac{n}{p}} \bigotimes_{j=1}^{k} \boldsymbol{\mu}_j + \frac{1}{\sqrt{p}} \tilde{\mathbf{Z}}, \tag{7}$$

where $\tilde{\mathbf{Z}} = \frac{1}{\sqrt{n}} \mathbf{Z} \times_{k+1} \boldsymbol{y} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} y_i \mathbf{Z}_i$. Since $\tilde{\mathbf{Z}}$ is a sum of $n$ i.i.d. random tensors normalized by $\sqrt{n}$, then $\tilde{\mathbf{Z}}$ is also a random tensor with i.i.d. standard Gaussian entries.

**Remark 3.2** (On the data distribution)**.** *Note that for the above supervised learning setting, the Gaussianity assumption on the $\mathbf{Z}_i$ might be relaxed to any distribution with zero mean and unit variance, for which $\tilde{\mathbf{Z}}$ remains a random tensor with i.i.d. standard Gaussian entries by the central limit theorem.*

$\mathbf{W}$ has the form of a *spiked random tensor model* which has been studied in [Seddik et al., 2021a]. In order to extract the hidden rank-one component of $\mathbf{W}$, we consider the best rank-one approximation of $\mathbf{W}$ which yields estimates of the means components $\boldsymbol{\mu}_j$'s (if the classes are separable, i.e., $\|\mathbf{M}\|$ is large enough) and then replace the weights $\mathbf{W}$ in the decision function by such rank-one approximation. Precisely, the best rank-one approximation of $\mathbf{W}$ can be obtained by solving the following objective

$$(\lambda^*, \{\boldsymbol{u}_i^*\}_{i=1}^k) = \underset{\lambda \in \mathbb{R}^+, \boldsymbol{u}_i \in \mathbb{S}^{p_i - 1}}{\arg\min} \|\mathbf{W} - \lambda \bigotimes_{i=1}^{k} \boldsymbol{u}_i\|_{\mathrm{F}}^2, \tag{8}$$

which corresponds to the maximum likelihood estimator (MLE). Computing the above MLE is NP-hard in the worst case [Hillar and Lim, 2013]. However, it is possible to compute consistent estimates of the rank-one components of $\mathbf{W}$ in polynomial time, using tensor SVD[4] [Ben Arous et al., 2021, Seddik et al., 2021a] or tensor power iteration (Algorithm 1) initialized with tensor SVD [Auddy and Yuan, 2022] which was shown to yield more accurate estimation of the rank-one tensor $\mathbf{M}$, provided that the difference between the class-wise means $\|\mathbf{M}\|$ is larger than $\mathcal{O}\left(P^{\frac{1}{4}}/p^{\frac{1}{2}}\right)$ as proved in [Seddik et al., 2021a, Auddy and Yuan, 2022].

In essence, extracting the rank-one component of $\mathbf{W}$ consists of a denoising scheme which allows to considerably reduce the variance of the decision function, thereby providing better classification accuracy. Given the above MLE which we denote $\lambda^* \bigotimes_{i=1}^k \boldsymbol{u}_i^*$, the Tensor-based $\infty$-Ridge classifier, which we refer to as *Tensor-Ridge* (TR), is defined for a new datum $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$ as

$$f_{\mathrm{TR}}(\tilde{\mathbf{X}}_i) = \left\langle \lambda^* \bigotimes_{i=1}^{k} \boldsymbol{u}_i^*, \tilde{\mathbf{X}}_i \right\rangle \underset{\mathcal{C}_2}{\overset{\mathcal{C}_1}{\lessgtr}} 0. \tag{9}$$

We introduce the following quantities in (10) from [Seddik et al., 2021a] which describe the behavior of a $k$-order

[4] SVD applied to the unfolded tensor.

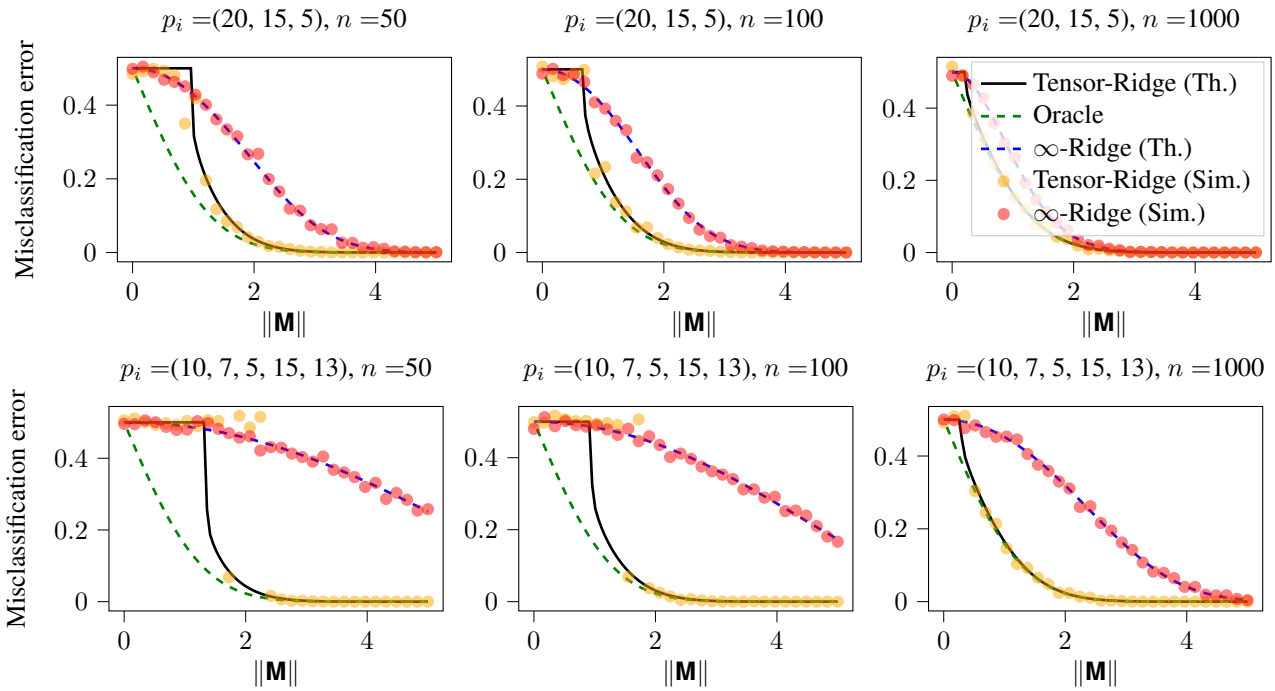


Figure 2: Theoretical versus empirical misclassification error of both $\infty$-Ridge classifier and Tensor-Ridge classifier classifiers. We considered $n$ training data as order $k$ tensors of dimensions $p_1 \times \cdots \times p_k$ with $k \in \{3, 5\}$ having a rank-one structure as in (2) with the $\boldsymbol{\mu}_j$'s being randomly sampled vectors.

**Algorithm 1** Tensor Power Iteration [Anandkumar et al., 2014]

**Require:** An order $k$ tensor $\mathbf{W} \in \mathbb{R}^{p_1 \times \cdots \times p_k}$ and initialization components $\boldsymbol{u}_1^0, \cdots, \boldsymbol{u}_k^0$.
**Output:** Rank-one approximation of $\mathbf{W}$.
$(\boldsymbol{u}_1, \cdots, \boldsymbol{u}_k) \leftarrow (\boldsymbol{u}_1^0, \cdots, \boldsymbol{u}_k^0)$
**while** Not convergence **do**
  **for** $i \in [k]$ **do**
    $\boldsymbol{u}_i \leftarrow \frac{\mathbf{W}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}, :, \boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_k)}{\|\mathbf{W}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}, :, \boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_k)\|}$
  **end for**
**end while**

spiked random tensor model and shall be used subsequently.

$$\begin{cases} f(z, \beta) = z + g(z) - \beta \prod_{i=1}^k q_i(z), \\ q_i(z) = \sqrt{1 - \frac{g_i^2(z)}{c_i}}, \end{cases} \tag{10}$$

where $c_i = \lim_{p_i \to \infty} \frac{p_i}{\sum_{j=1}^k p_j}$ and $(g(z), g_i(z))$ are solutions to the following system

$$\begin{cases} g(z) = \sum_{i=1}^k g_i(z), \\ g_i^2(z) - (g(z) + z) g_i(z) - c_i = 0. \end{cases} \tag{11}$$

Essentially, it was proved in [Seddik et al., 2021a] that the above equations are well defined for $\beta$ greater than some threshold $\beta_s = \mathcal{O}(1)$. The latter corresponds basically to the classes separability condition on $\|\mathbf{M}\|$ above which the MLE in (8) starts to correlate with $\mathbf{M}$.

Therefore, our following result characterizes the theoretical performance of the Tensor-Ridge classifier based on the above random tensor tools.

**Theorem 3.3** (Performance of the Tensor-Ridge classifier)**.** *Under Assumption 2.2, for $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$ with $a \in \{1, 2\}$ independent from the training set $\mathbf{X}$,*

$$\frac{1}{\sigma} \left( f_{TR}(\tilde{\mathbf{X}}_i) - m_a \right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, 1),$$

*where $m_a = (-1)^a \sigma \|\mathbf{M}\| \prod_{j=1}^k q_j(\sigma)$ and $\sigma$ satisfies $f\left(\sigma, \|\mathbf{M}\| \sqrt{\frac{n}{p}}\right) = 0$ where $q_j$ and $f$ are defined in* (10)*. Furthermore, the misclassification error verifies with probability one $\mathbb{P}\left((-1)^a f_{TR}(\tilde{\mathbf{X}}_i) < 0 \mid \tilde{\mathbf{X}}_i \in \mathcal{C}_a\right) - Q\left(\frac{|m_a|}{\sigma}\right) \to 0$.*

*Sketch of proof.* The proof relies on estimating the expectation and the variance of the decision function $f_{\text{TR}}(\tilde{\mathbf{X}}_i)$ for some $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$ with $a \in \{1, 2\}$ independent from the training set $\mathbf{X}$. Indeed, one finds that $\mathbb{E} f_{\text{TR}}(\tilde{\mathbf{X}}_i) = \mathbb{E}\left[(-1)^a \|\mathbf{M}\| \lambda^* \prod_{j=1}^k \langle \frac{\boldsymbol{\mu}_j}{\|\boldsymbol{\mu}_j\|}, \boldsymbol{u}_j^* \rangle\right]$ where the quantities $\lambda^*$ and $\langle \frac{\boldsymbol{\mu}_j}{\|\boldsymbol{\mu}_j\|}, \boldsymbol{u}_j^* \rangle$ are estimated using (10) where $\lambda^* \to \sigma$ with $\sigma$ satisfying $f(\sigma, \|\mathbf{M}\| \sqrt{\frac{n}{p}}) = 0$ and $\langle \frac{\boldsymbol{\mu}_j}{\|\boldsymbol{\mu}_j\|}, \boldsymbol{u}_j^* \rangle \to q_j(\sigma)$.

The variance of $f_{\text{TR}}(\tilde{\mathbf{X}}_i)$ is computed similarly and we find $\text{Var}[f_{\text{TR}}(\tilde{\mathbf{X}}_i)] = \sigma^2$. See supplementary material for detailed proof. □

**Remark 3.4** (On the assumptions)**.** *Theorem 3.3 requires additional assumptions (e.g., Assumption 3 from [Seddik et al., 2021a]). We highlight that this assumption is rather technical and needs the introduction of various notions (e.g., defining the block-wise contracted matrix introduced by [Seddik et al., 2021a]). However, note that Assumption 3 therein is always satisfied by the maximum likelihood estimator when the SNR is larger than some $\mathcal{O}(1)$ constant. In our notation the SNR corresponds to the quantity $\|\mathbf{M}\|$ which controls the difficulty of the classification problem.*

Theorem 3.3 states that the performance of the Tensor-Ridge classifier depends solely on $\|\mathbf{M}\|$ and the ratio $\frac{p}{n}$, and does not depend on the dimension $P$ as was the case for the $\infty$-Ridge classifier in Theorem 3.1. This highlights that the variance $\sigma^2$ for the Tensor-Ridge classifier remains constant as $n \to \infty$. We can further observe this from Figure 1 which shows that Tensor-Ridge yields a lower variance.

Figure 2 depicts the theoretical versus empirical misclassification error for both methods. It particularly shows that the Tensor-Ridge classifier yields drastically better performances (almost closer to the oracle which assumes perfect knowledge of $\mathbf{M}$) when $n$ is small, or alternatively when the dimension of data is high. Note that the empirical curves for the Tensor-Ridge classifier are obtained with tensor power iteration initialized with tensor SVD, and thus converges in polynomial time if $\|\mathbf{M}\|$ is larger than $\mathcal{O}\left(P^{\frac{1}{4}}/p^{\frac{1}{2}}\right)$ as discussed previously. In particular, the last line in Figure 2 highlights this phenomenon, where we can see that the power iteration does not always converge when we increase the tensors order.

Moreover, Figure 3 depicts the misclassification error of both methods varying the ratio $p/n$ and $\|\mathbf{M}\|$. It shows that the Tensor-Ridge classifier performs better for large values of $p/n$ in theory (second plot from the left). More interestingly, the third plot depicts the computationally possible performance which corresponds to the algorithmic threshold $\|\mathbf{M}\| \geq \mathcal{O}\left(P^{\frac{1}{4}}/p^{\frac{1}{2}}\right)$, thereby highlighting the superiority of the tensor-based approach even algorithmically. The last plot corresponds to perfect knowledge of $\mathbf{M}$ and provides an insight about the effect of the noise component in the considered data model.

**Generalization to higher-rank data:** Our results generalize to a more complex model of the following form. Suppose that the $\mathbf{X}_i$'s are distributed in two classes $\mathcal{C}_1$ and $\mathcal{C}_2$ (of cardinality $n_1$ and $n_2$ respectively), such that for $\mathbf{X}_i \in \mathcal{C}_a$ with $a \in 1, 2$,

$$\mathbf{X}_i = \sum_{\ell=1}^{r_a} \bigotimes_{j=1}^{k} \boldsymbol{\mu}_{j,\ell}^{(a)} + \mathbf{Z}_i \in \mathbb{R}^{p_1 \times \cdots \times p_k}, \tag{12}$$

where $\mathbf{Z}_i$ is a random tensor with i.i.d. standard Gaussian entries, $\boldsymbol{\mu}_{j,\ell}^{(a)} \in \mathbb{R}^{p_j}$ are independent from $\mathbf{Z}_i$ such that $\langle \boldsymbol{\mu}_{j,\ell_1}^{(a)}, \boldsymbol{\mu}_{j,\ell_2}^{(a)} \rangle = \delta_{\ell_1 \ell_2}$. That is, the data tensors $\mathbf{X}_i$ have a rank-$r_a$ (with $r_a$ being independent of the dimensions $p_i$) structure with orthogonal components.

Let us denote $\mathbf{M}_a = \sum_{\ell=1}^{r_a} \bigotimes_{j=1}^{k} \boldsymbol{\mu}_{j,\ell}^{(a)}$ the mean tensor of class $\mathcal{C}_a$. In a supervised setting, it is convenient to center the data by subtracting[5] $\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2)$ from each data sample which yields tensors of the form

$$\mathbf{X}_i = (-1)^a \left(\mathbf{M}_1 - \mathbf{M}_2\right) + \mathbf{Z}_i, \tag{13}$$

where $\mathbf{M}_1 - \mathbf{M}_2$ is clearly a low-rank tensor (of rank $r_1 + r_2$) with orthogonal components. Stacking all the data samples $\mathbf{X}_i$ in a data tensor $\mathbf{X} \in \mathbb{R}^{p_1 \times \cdots \times p_k \times n}$, the $\infty$-Ridge classifier has weights tensor of the form

$$\mathbf{W} = \frac{1}{\sqrt{np}} \mathbf{X} \times_{k+1} \boldsymbol{y} = \sqrt{\frac{n}{p}} \mathbf{M} + \frac{1}{\sqrt{p}} \tilde{\mathbf{Z}}, \tag{14}$$

where $\tilde{\mathbf{Z}} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} y_i \mathbf{Z}_i$ and $\mathbf{M} = \mathbf{M}_1 - \mathbf{M}_2 = \sum_{\ell=1}^{r_1+r_2} \bigotimes_{j=1}^{k} \boldsymbol{\mu}_{j,\ell}$ is a rank-$(r_1 + r_2)$ tensor. Therefore, the Tensor-Ridge classifier for this case relies on a low-rank approximation of $\mathbf{W}$ of rank $r_1 + r_2$ which can be obtained through standard tensor decomposition methods (e.g., tensor deflation [Ge et al., 2021]). We, therefore, have the following theorem characterizing the performance of the Tensor-Ridge classifier in this more general case.

**Theorem 3.5** (Performance of the Tensor-Ridge classifier for data model in (13))**.** *Under Assumption 2.2, for $\tilde{\mathbf{X}}_i \in \mathcal{C}_a$ with $a \in \{1, 2\}$ independent from the training set $\mathbf{X}$,*

$$\frac{1}{\sqrt{\sum_{\ell=1}^{r_1+r_2} \sigma_\ell^2}} \left(f_{TR}(\tilde{\mathbf{X}}_i) - m_a\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, 1),$$

*where $m_a = (-1)^a \sum_{\ell=1}^{r_1+r_2} \sigma_\ell \mu_\ell \prod_{j=1}^{k} q_j(\sigma_\ell)$ where $\mu_\ell = \| \bigotimes_{j=1}^{k} \boldsymbol{\mu}_{j,\ell} \|$ and $\sigma_\ell$ satisfies $f(\sigma_\ell, \mu_\ell \sqrt{\frac{n}{p}}) = 0$. $q_j$ and $f$ are defined in* (10)*. Furthermore, the misclassification error verifies with probability one $\mathbb{P}\left((-1)^a f_{TR}(\tilde{\mathbf{X}}_i) < 0 \mid \tilde{\mathbf{X}}_i \in \mathcal{C}_a\right) - Q\left(\frac{|m_a|}{\sqrt{\sum_{\ell=1}^{r_1+r_2} \sigma_\ell^2}}\right) \to 0$.*

*Proof.* The proof strategy is the same as for Theorem 3.3. □

[5] In real scenarios one would first estimate the $\mathbf{M}_a$'s with their empirical estimates through tensor decomposition.

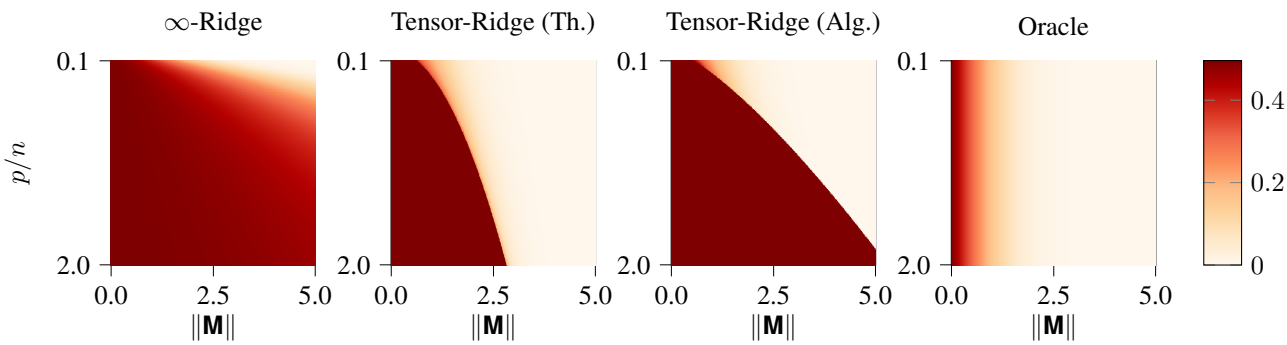


Figure 3: Theoretical misclassification error in terms of the signal strength $\|\mathbf{M}\|$ and the ratio $p/n$ for three order tensors of size $(p, p, p)$. For both $\infty$-Ridge and Tensor-Ridge as per Theorems 3.1 and 3.3 respectively. The third plot from the left corresponds to polynomial time Tensor-Ridge which is possible for $\|\mathbf{M}\|$ larger than $\mathcal{O}(p^{\frac{1}{4}})$ while the last plot corresponds to oracle classifier which assumes perfect knowledge of $\mathbf{M}$.

### 3.2 UNSUPERVISED SETTING

In a setting where only $n$ training samples $\mathbf{X}_1, \dots, \mathbf{X}_n$ are provided without their corresponding labels, one would rely on unsupervised learning to cluster them into classes. Given the data model in (3), a simple unsupervised learning approach [Ng et al., 2002] consists in unfolding $\mathbf{X}$ into a $n \times P$ matrix as

$$\boldsymbol{X} = \mathrm{Mat}_{k+1}(\mathbf{X}) = \boldsymbol{y}\,\mathrm{vec}(\mathbf{M})^\top + \mathrm{Mat}_{k+1}(\mathbf{Z}), \qquad (15)$$

then estimating the labels $\boldsymbol{y}$ through the dominant eigenvector of the Gram matrix $\boldsymbol{X}\boldsymbol{X}^\top$ denoted $\hat{\boldsymbol{y}}$, which coincides with the dominant left singular vector of $\boldsymbol{X}$. The theoretical performance of this *linear spectral method* is given by the following theorem.

**Theorem 3.6** (Performance of linear spectral clustering)**.** *Let $\hat{\boldsymbol{y}}$ be the right singular vector of $\boldsymbol{X}$ corresponding to its largest singular value. The estimated class for the datum $\mathbf{X}_i$ is given as $\hat{\mathcal{C}}_i = \mathrm{sign}(\hat{y}_i)$. Then under Assumption 2.2,*

$$\frac{1}{\sigma}\left(\sqrt{n}\hat{y}_i - \alpha y_i\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0,1),$$

*where $\alpha = \kappa\left(\|\mathbf{M}\|\sqrt{\frac{n}{P+n}}, \frac{n}{P+n}\right)^{-1}$, $\sigma = \sqrt{1-\alpha^2}$ and $\kappa(\beta, c) = \beta\sqrt{\frac{\beta^2(\beta^2+1)-c(c-1)}{(\beta^4+c(c-1))(\beta^2+1-c)}}$ defined for $\beta > (c(1-c))^{\frac{1}{4}}$. Furthermore, the misclassification error is given with probability one by $Q\left(\frac{\alpha}{\sqrt{1-\alpha^2}}\right)$.*

*Proof.* See supplementary material. □

Theorem 3.6 states that the entries of the estimated left singular vector corresponding to the largest singular value of $\boldsymbol{X}$ is a Gaussian random variable, whose mean and variance depend on $\|\mathbf{M}\|$ and the ratio $c = \frac{n}{P+n}$. Essentially, in order to obtain a non-zero correlation between $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$, the signal strength $\|\mathbf{M}\|$ must be greater than $\frac{\sqrt[4]{c(1-c)}}{\sqrt{c}}$. However, under Assumption 2.2, the ratio $\frac{n}{P+n} \to 0$ if $n \to \infty$, thereby yielding a high misclassification error. Indeed, Figure 4 (left) depicts the $2D$ projection space corresponding to the two largest eigenvectors of $\boldsymbol{X}\boldsymbol{X}^\top$ along with its theoretical mean and fluctuations as per Theorem 3.6. Note that the second largest eigenvector of $\boldsymbol{X}\boldsymbol{X}^\top$ is not informative about the classes. In fact, its entries have zero mean and variance $1/n$, which is a classical result from random matrix theory [O'Rourke et al., 2016].

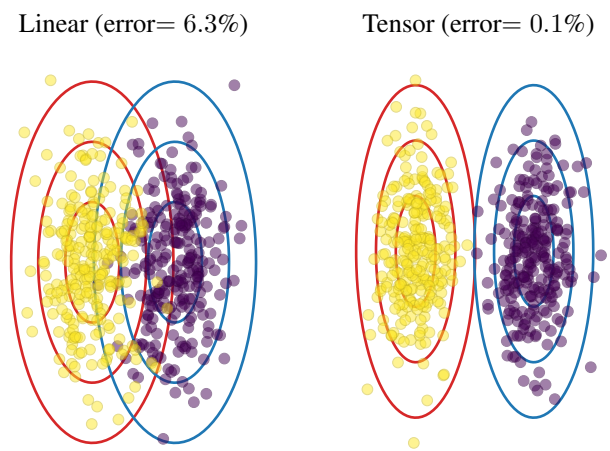


Figure 4: Left: the $2D$ projection space obtained by linear clustering. Right: the $2D$ projection space by Tensor-based clustering obtained through a rank-two CP decomposition of $\mathbf{X}$. We considered $k = 2$ and $n_1 = n_2 = 500$ and data are tensors $\mathbf{X}_i$ of shape $(15, 30, 20)$ generated as the model in (2) with $\|\mathbf{M}\| = 3$. The ellipses correspond to the theoretical means and fluctuations according to Theorems 3.6 and 3.7 respectively.

In contrast, extracting the rank-one structure of the data tensor allows us to improve the classification performance. Indeed, given the data model in (3), computing a rank-1 approximation of $\mathbf{X}$ and extracting the corresponding $(k+1)$-th mode component yields a better estimation of the labels vector $\boldsymbol{y}$. We precisely have the following theorem characterizing the performance of *the Tensor-based clustering*.

**Theorem 3.7** (Performance of Tensor-based clustering)**.** *Let $\hat{\boldsymbol{y}}$ be the $(k+1)$-th mode component of the rank-1 tensor approximation of $\mathbf{X}$. The estimated class for the datum $\mathbf{X}_i$ is given as $\hat{\mathcal{C}}_i = \operatorname{sign}(\hat{y}_i)$. Then under Assumption 2.2,*

$$\frac{1}{\sigma}\left(\sqrt{n}\hat{y}_i - \alpha y_i\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0,1),$$

*where $\alpha = q_{k+1}(\lambda^*)$, $\sigma = \sqrt{1-\alpha^2}$ with $q_{k+1}(\cdot)$ defined by* (10) *for a tensor of order $k+1$ and $\lambda^*$ is the unique solution to $f\left(\lambda^*, \|\mathbf{M}\|\sqrt{\frac{n}{p+n}}\right) = 0$. Furthermore, the misclassification error is given with probability one by $Q\left(\frac{\alpha}{\sqrt{1-\alpha^2}}\right)$.*

*Proof.* See supplementary material. □

**Remark 3.8.** *The generalization of the unsupervised setting to the data model in* (12) *is more challenging since the data tensor $\mathbf{X}$, in this case, does not follow a CP decomposition but rather a block-term decomposition [Rontogiannis et al., 2021] which is more challenging to analyze theoretically and is therefore left for a future investigation.*

As for the linear clustering approach, the estimated labels vector $\hat{\boldsymbol{y}}$ with tensor clustering has Gaussian entries centered on the scaled labels $\boldsymbol{y}$ with a scaling factor $\alpha$ and fluctuations depending on such $\alpha$. However, now the clustering performance depends on $\|\mathbf{M}\|$ and the ratio $\frac{n}{p+n}$, thereby yielding the same clustering performance as $n$ and $p$ increase at the same rate. Figure 4 (right) depicts the $2D$ projection space obtained by a rank-two CP decomposition of $\mathbf{X}$ with its theoretical mean and fluctuations as per Theorem 3.7. From Figure 4 we clearly note that tensor-based clustering yields lower variance compared to the classical linear approach, thereby allowing better clustering performance. This improvement is relatively trivial given the knowledge of the underlying rank-one structure, but our results allow the exact characterization of the performance gap between both methods.

To best illustrate the comparison between linear and Tensor-based clustering, we depict the misclassification errors of both methods in terms of $\|\mathbf{M}\|$ in Figure 5. Essentially, in order to have a correlation between $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$, the signal strength $\|\mathbf{M}\|$ must be greater than some $\mathcal{O}(1)$ threshold in theory. However, in order to estimate the label signal in practice in polynomial time, $\|\mathbf{M}\|$ must be greater than $\mathcal{O}\left((P\times n)^{\frac{1}{4}}/(p+n)^{\frac{1}{2}}\right)$, which coincides with the phase

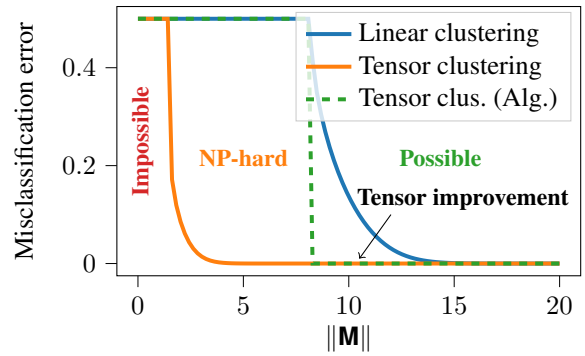


Figure 5: Theoretical misclassification errors in terms of the signal strength $\|\mathbf{M}\|$ for both linear and Tensor-based clustering as per Theorems 3.6 and 3.7 respectively. We considered $n_1 = n_2 = 100$ and data are tensors of shape $(15, 30, 10)$.

transition of linear clustering from Theorem 3.6. Importantly, Figure 5 depicts three different regions: (i) impossible: it is information-theoretically impossible to recover the clusters or even detect them, in the sense that any clustering method output is provably independent of the true classes; (ii) NP-hard: where there is no polynomial time algorithm that can recover the labels signal, and (iii) possible: where recovery is possible in polynomial time (e.g., using tensor power iteration initialized with tensor SVD as discussed previously). Figure 5 clearly highlights the benefit of Tensor-based clustering upon linear clustering if the data has an underlying low-rank structure. Notably, the performances of the different approaches are accurately estimated by Theorems 3.6 and 3.7.

## 4 CONCLUDING REMARKS

This paper has brought a theoretical analysis of learning from tensor data that have a hidden low-rank structure. Both analytical and empirical assessments suggest that a considerable performance gain can be achieved by exploiting such low-rank tensor structure when few training samples are available and such gain is accurately quantified for the considered statistical model in (2).

As such, the paper explicitly demonstrates the application of *random tensor theory* to evaluate the performance of simple learning methods (such as the considered Tensor-Ridge classifier), whose behavior was not so far theoretically understood. This paves the way for more systematic theoretical analysis and improvement of sophisticated machine learning algorithms when dealing with tensor-structured data. In particular, our present analysis can be extended for the understanding of the CP-regressor [Zhou et al., 2013] which basically consists of a Ridge regressor with low-rank tensor prior, which is more adapted for low-rank tensor data with covariance structure.

## References


Hafiz Tiomoko Ali and Romain Couillet. Improved spectral community detection in large heterogeneous networks. *The Journal of Machine Learning Research*, 18(1):8344–8392, 2017.

Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *Journal of machine learning research*, 15:2773–2832, 2014.

Arnab Auddy and Ming Yuan. On estimating rank-one spiked tensors in the presence of heavy tailed errors. *IEEE Transactions on Information Theory*, 68(12):8053–8075, 2022.

Gérard Ben Arous, Daniel Zhengyu Huang, and Jiaoyang Huang. Long random matrices and tensor unfolding. *arXiv preprint arXiv:2110.10210*, 2021.

Wanli Chen, Xinge Zhu, Ruoqi Sun, Junjun He, Ruiyu Li, Xiaoyong Shen, and Bei Yu. Tensor low-rank reconstruction for semantic segmentation. In *European Conference on Computer Vision*, pages 52–69. Springer, 2020.

Rong Ge, Yunwei Ren, Xiang Wang, and Mo Zhou. Understanding deflation process in over-parametrized tensor decomposition. *Advances in Neural Information Processing Systems*, 34:1299–1311, 2021.

Christopher J Hillar and Lek-Heng Lim. Most tensor problems are np-hard. *Journal of the ACM (JACM)*, 60(6): 1–39, 2013.

Frank L Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1-4):164–189, 1927.

Jonathan Kadmon and Surya Ganguli. Statistical mechanics of low-rank tensor decomposition. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124016, 2019.

T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51:455–500, 2019.

Jean Kossaifi, Zachary C Lipton, Arinbjörn Kolbeinsson, Aran Khanna, Tommaso Furlanello, and Anima Anandkumar. Tensor regression networks. *Journal of Machine Learning Research*, 21:1–21, 2020.

Joseph M Landsberg. Tensors: geometry and applications. *Representation theory*, 381(402):3, 2012.

Paul Pu Liang, Zhun Liu, Yao-Hung Hubert Tsai, Qibin Zhao, Ruslan Salakhutdinov, and Louis-Philippe Morency. Learning representations from imperfect time series data via tensor rank regularization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1569–1576, 2019.

Cosme Louart, Zhenyu Liao, and Romain Couillet. A random matrix approach to neural networks. *The Annals of Applied Probability*, 28(2):1190–1248, 2018.

Xiaoyi Mai and Romain Couillet. A random matrix analysis and improvement of semi-supervised learning for large dimensional data. *The Journal of Machine Learning Research*, 19(1):3074–3100, 2018.

Francesca Mignacco, Florent Krzakala, Yue Lu, Pierfrancesco Urbani, and Lenka Zdeborova. The role of regularization in classification of high-dimensional noisy gaussian mixture. In *International conference on machine learning*, pages 6874–6883. PMLR, 2020.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

Sean O'Rourke, Van Vu, and Ke Wang. Eigenvectors of random matrices: a survey. *Journal of Combinatorial Theory, Series A*, 144:361–442, 2016.

Jeffrey Pennington and Pratik Worah. Nonlinear random matrix theory for deep learning. In *Advances in neural information processing systems*, volume 30, 2017.

Athanasios A Rontogiannis, Eleftherios Kofidis, and Paris V Giampouras. Block-term tensor decomposition: Model selection and computation. *IEEE Journal of Selected Topics in Signal Processing*, 15(3):464–475, 2021.

Mohamed El Amine Seddik, Maxime Guillaud, and Romain Couillet. When random tensors meet random matrices. *arXiv preprint arXiv:2112.12348*, 2021a.

Mohamed El Amine Seddik, Cosme Louart, Romain Couillet, and Mohamed Tamaazousti. The unexpected deterministic and universal behavior of large softmax classifiers. In *International Conference on Artificial Intelligence and Statistics*, pages 1045–1053. PMLR, 2021b.

W. Sun and L. Li. Dynamic tensor clustering. *Journal of the American Statistical Association*, 114:1894–1907, 2019.

Will Wei Sun, Botao Hao, and Lexin Li. Tensors in modern statistical learning. *Wiley StatsRef: Statistics Reference Online*, pages 1–25, 2014.

Malik Tiomoko, Romain Couillet, and Hafiz Tiomoko. Large dimensional analysis and improvement of multi task learning. *arXiv preprint arXiv:2009.01591*, 2020.

Malik Tiomoko, Romain Couillet, and Frédéric Pascal. Pca-based multi task learning: a random matrix approach. *arXiv preprint arXiv:2111.00924*, 2021.

Ke Wang and Christos Thrampoulidis. Binary classification of gaussian mixtures: Abundance of support vectors, benign overfitting, and regularization. *SIAM Journal on Mathematics of Data Science*, 4(1):260–284, 2022.

H. Zhou, L. Li, and H. Zhu. Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108, 2013.