This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



# Low-rank Characteristic Tensor Density Estimation Part I: Foundations

Magda Amiridi, Nikos Kargas, and Nicholas D. Sidiropoulos, *Fellow, IEEE*

***Abstract*—Effective non-parametric density estimation is a key challenge in high-dimensional multivariate data analysis. In this paper, we propose a novel approach that builds upon tensor factorization tools. Any multivariate density can be represented by its characteristic function, via the Fourier transform. If the sought density is compactly supported, then its characteristic function can be approximated, within controllable error, by a finite tensor of leading Fourier coefficients, whose size depends on the smoothness of the underlying density. This tensor can be naturally estimated from observed and possibly incomplete realizations of the random vector of interest, via sample averaging. In order to circumvent the curse of dimensionality, we introduce a low-rank model of this *characteristic tensor*, which significantly improves the density estimate especially for high-dimensional data and/or in the sample-starved regime. By virtue of uniqueness of low-rank tensor decomposition, under certain conditions, our method enables learning the true data-generating distribution. We demonstrate the very promising performance of the proposed method using several toy, measured, and image datasets.**

***Index Terms*—Statistical learning, Probability Density Function (PDF), Characteristic Function (CF), Tensors, Rank, Canonical Polyadic Decomposition (CPD).**

## I. Introduction

Density estimation is a fundamental yet challenging problem in statistical signal processing and machine learning. Density estimation is the task of learning the joint Probability Density Function (PDF) from a set of observed data points, sampled from an unknown underlying data-generating distribution. A model of the density function of a continuous random vector provides a complete description of the joint statistical properties of the data and can be used to perform tasks such as computing the most likely value of a subset of elements ("features") conditioned on others, computing any marginal or conditional distribution, and deriving optimal estimators, such as the minimum mean squared error (conditional mean) estimator. Density estimation has a wide range of applications including classification [1], [2], [3], [4], clustering [5], data synthesis [6], data completion [7] and reconstruction related applications [8], as well as learning statistical regularities such as skewness, tail behavior, multimodality or other structures present in the data [9].

Original manuscript submitted Nov. 14, 2021 (first version appeared on arXiv Aug. 27, 2020, https://arxiv.org/abs/2008.12315); revised May 10, 2022; accepted May 16, 2022. This work was supported in part by NSF IIS-1704074. M. Amiridi and N.D. Sidiropoulos are with the Department of ECE, University of Virginia, Charlottesville, VA 22904. Author e-mails: (ma7bx,nikos)@virginia.edu. N. Kargas was with the Department of ECE, University of Minnesota; he is now with Amazon, Cambridge, U.K. Author e-mail: karga005@umn.edu

Existing work on density estimation can be mainly categorized into parametric approaches such as Gaussian Mixture Models (GMM) [10], and non-parametric approaches such as Histogram [11] and Kernel Density Estimation (KDE) [12]. A density model must be expressive – flexible enough to represent a wide class of distributions, and tractable and scalable (computationally and memory-wise) at the same time (expressivity-tractability trade-off). Over the last several years, explicit feed-forward neural network based density estimation methods [13], [14], [15] have gained increasing attention as they provide a tractable way to evaluate high-dimensional densities *point-wise*. On the other hand implicit *generative* models such as generative adversarial networks [16] and variational autoencoders [17] can be used to obtain models which allow effective and efficient sampling. Although neural network based solutions show promise in some high-dimensional applications such as image sampling, they are not currently well-suited for many other real-world applications. They lack the ability to compute expectations, marginalize over arbitrary subsets of variables, and evaluate conditionals, as they rather serve for point-wise density *evaluation*, or *sampling*. Additionally, model identifiability (i.e., recovery of the true data-generating distribution) is a fundamental goal of PDF estimation, which most deep generative models have not yet addressed. Incomplete observations (due to causes such as faulty sensors, corrupt data, cost of acquisition, or privacy concerns) present distinct challenges to the training of these models. The majority of such models are trained on complete data only and are unable to handle missing elements in the input vector, during both training and testing ("showtime").

In this paper, we develop a novel non-parametric method for multivariate PDF estimation based on tensor rank decomposition – known as *Canonical Polyadic Decomposition* (CPD) [18], [19]. CPD is a powerful model that can parsimoniously represent high-order data tensors exactly or approximately, and its distinguishing feature is that under certain reasonable conditions it is unique – see [20] for a recent tutorial overview. We show that **any compactly supported continuous density can be approximated, within controllable error, by a finite *characteristic tensor* of leading complex Fourier coefficients, whose size depends on the smoothness of the density**. This characteristic tensor can be naturally estimated via sample averaging from realizations of the random vector of interest.

The main challenge, however, lies in the fact that the size of this tensor (the number of model parameters in the Fourier domain) grows exponentially with the number of random variables – the length of the random vector of interest. In

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



order to circumvent this "curse of dimensionality" (CoD) and further denoise the naive sample averaging estimates, we introduce a low-rank model of the characteristic tensor, whose **degrees of freedom (for fixed rank) grow linearly with the random vector dimension**. Low-rank modeling significantly improves the density estimate especially for high-dimensional data and/or in the sample-starved regime. By virtue of uniqueness of low-rank tensor decomposition, under certain conditions, **our method enables learning the true data-generating distribution.**

In order to handle incomplete (both training and testing) data (vector realizations with missing entries) as well as scaling up to high-dimensional vectors, we further introduce coupled low-rank decomposition of lower-order characteristic tensors corresponding to smaller subsets of variables that share 'anchor' variables, and show that this still enables recovery of the global density, under certain conditions. As an added benefit, **our approach yields a generative model of the sought density, from which it is very easy to sample from**. This is because our low-rank model of the characteristic tensor admits a latent variable naive Bayes interpretation. A corresponding result for finite-alphabet random vectors was first pointed out in [21]. In contrast to [21], our approach applies to continuous random vectors possessing a compactly supported multivariate density function. From an algorithmic standpoint, we formulate a constrained coupled tensor factorization problem and develop a Block Coordinate Descent (BCD) algorithm.

The main results and contributions of this paper can be summarized as follows:

- We show that **any smooth compactly supported multivariate PDF can be approximated by a finite tensor model, without using any prior or data-driven discretization process**. We also show that truncating the sampled multivariate characteristic function of a random vector is equivalent to using a finite separable mixture model for the underlying distribution. Note that we do not assume a latent variable mixture model; instead the **latent variable factorization falls off from compactness of support and smoothness.** Under these relatively mild assumptions, **the proposed model can approximate any high dimensional PDF with approximation guarantees**. By virtue of uniqueness of CPD, assuming low-rank in the Fourier domain, **the underlying multivariate density is identifiable**.
- We show that **high dimensional joint PDF recovery is possible under low tensor-rank conditions, even if we only observe subsets (triples) of variables**. This is a key point that enables one to handle incomplete realizations of the random vector of interest. To the best of our knowledge, no other generic density estimation approach allows this. To tackle this more challenging version of the problem, we propose an optimization framework based on coupled tensor factorization. Our approach jointly learns lower-order (3-dimensional) characteristic functions, and then assembles tensor factors to synthesize the full characteristic function model.
- The proposed model allows efficient and low-complexity inference, sampling, and density evaluation. In that sense, it a more comprehensive solution that neural density evaluation or neural generative models. **We provide convincing experimental results on sampling, likelihood evaluation, and regression on toy, image, and many real datasets that are often used as benchmarks in our context. Our results corroborate the effectiveness of the proposed method even for datasets that have hundreds of variables.**

This is the first of a two-part paper. The second part builds on this foundation to develop a joint compression (nonlinear dimensionality reduction) and compressed density estimation framework that offers additional flexibility and scalability, but does not provide a density estimate in the original space as the "baseline" method in this first part does. Each approach has its own advantages, but the second builds upon the first. It is therefore natural to present them as Part I and Part II.

## II. Background

### A. Related work

Density estimation has been the subject of extensive research in statistics and the machine learning community. Methods for density estimation can broadly be classified as either parametric or non-parametric. Parametric density estimation assumes that the data are drawn from a known parametric family of distributions, parametrized by a fixed number of tunable parameters. Parameter estimation is usually performed by maximizing the likelihood of the observed data. One of the most widely used parametric models is the Gaussian Mixture Model (GMM). GMMs can approximate any density function if the number of components is large enough [22]. However, a very large number of components may be required for good approximation of the unknown density, especially in high dimensions. Increasing the number of components introduces computational challenges and may require a large amount of data [23]. Misspecification and inconsistent estimation is less likely to occur with nonparametric density estimation.

Nonparametric density estimation is more unassuming and in that sense "universal", but the flip-side is that it does not scale beyond a small number of variables (dimensions). The most widely-used approach for nonparametric density estimation is Kernel Density Estimation (KDE) [11], [12]. The key idea of KDE is to estimate the density by means of a sum of kernel functions centered at the given observations. However, worst-case theoretical results show that its performance worsens exponentially with the dimension of the data vector [24].

Our approach falls under nonparametric methods, and is motivated by Orthogonal Series Density Estimation (OSDE) [25], [26], [27], a powerful non-parametric estimation methodology. OSDE approximates a probability density function using a truncated sum of orthonormal basis functions, which may be trigonometric, polynomial, wavelet etc. However, OSDE becomes computationally intractable even for as few as 10 dimensions, since the number of parameters grows exponentially with the number of dimensions. Unlike OSDE, our approach is able to scale to much higher dimensions.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



The first work using tensor decomposition to establish identifiability of latent variable models was [28], where it was shown that, under certain conditions, a finite mixture of non-parametric product distributions is identifiable. The linear independence conditions mentioned in [28] are in fact not necessary for uniqueness; a milder condition pertaining to the sum of Kruskal-ranks of the latent factor matrices is in fact sufficient [29]. Also, [28] did not provide a companion density estimation procedure, which limits its applicability.

Later on, [30], [31] proposed using whitening-based orthogonal tensor decomposition to recover the parameters of certain latent variable (not general density) models – but this algorithm is not always feasible [32] because the whitening step cannot always find a positive semi-definite matrix via linear combination of tensor slices. This happens with positive probability [32], in which case the orthogonal decomposition algorithm fails altogether. Furthermore, if the rank of this matrix is lower than the tensor rank, then the algorithm has a "soft" failure. We also note that the approach in [30] is a kernel method that involves eigenvalue decomposition of $M$ by $M$ matrices, where $M$ is the training sample size, which is prohibitive for large training sets. Our proposed algorithm is scalable (its complexity is linear in $M$) and it avoids earlier pitfalls. It is also worth re-iterating that, whereas there have been prior works dealing with multivariate density estimation for latent variable models such as [28], [30], our work is the first to use tensor models for high-dimensional densities, where the dimensionality is well above $3-10$. Part of our novelty is showing that low-rank tensor factorization can work, with remarkably low ranks, in these high dimensions (up to $256$ here).

Another method also based on low-rank tensor decompositions with theoretical guarantees of identifiability (for distributions of low enough rank) has been presented in [33]. In contrast to [28], [30] and [33], our approach is "universal" for smooth, compactly supported multivariate densities, i.e., no assumptions regarding a multivariate mixture model of non-parametric product distributions are made in the present paper; we show that a latent variable factorization falls off from compactness of support and smoothness. A similar approach to [33] was considered in [34], where a tensor train model is used to approximate a discretized PDF, followed by interpolation. The authors use a conditioning chain and compute each conditional distribution given the model for the full joint distribution – this computation depends on the ordering of the variables. There are no identifiability guarantees, and the method is geared towards sampling applications. This is natural, since tensor train models do not offer easy marginalization and inference.

Recently, several density evaluation and modeling methods that rely on neural networks have been proposed. The Real-valued Neural Autoregressive Distribution Estimator (RNADE) [35] is among the best-performing neural density estimators and has shown great potential in scaling to high-dimensional distribution settings. These so-called autoregressive models (not to be confused with classical AR models for time-series) decompose the joint density as a product of one-dimensional conditionals of increasing conditioning order, and model each conditional density with a parametric model. Normalizing Flows (NF) [36] models start with a base density e.g., standard Gaussian, and stack a series of invertible transformations with tractable Jacobian to approximate the target density. Masked Autoregressive Flow (MAF) [15] is a type of NF model, where the transformation layer is built as an autoregressive neural network. These methods do not construct a joint PDF model but rather serve for point-wise density *evaluation*. That is, for any given input vector (realization), they output an estimate of the density evaluated at that particular input vector (point). For small vector dimensions, e.g., two or three, it is possible to evaluate all inputs on a dense grid, thereby obtaining a histogram-like density estimate; but the curse of dimensionality kicks in for high vector dimensions, where this is no longer an option. Additionally, these methods cannot impute more than very few missing elements in the input, for the same reason (grid search becomes combinatorial).

Another class of neural network density models are the sum-product networks (SPNs) [37], [38]. SPNs are deep probabilistic models, represented by a directed acyclic graph with univariate distributions at the leaves, that decompose a joint distribution into a hierarchy of mixtures (sums) and factorizations (products). Their extension to continuous variables assumes a model for the one-dimensional marginals, e.g., Gaussian or mixture of Gaussians for each input variable, in which case the overall distribution is a mixture of separable Gaussians. In the discrete (finite-alphabet / categorical) case, our model [21] can be interpreted as a shallow SPN; in the continuous case, as considered in this paper, we do not make any assumption on the one-dimensional marginals at the leaves, so our approach can be viewed as a nonparametric shallow SPN. From the viewpoint of SPNs, we show in this paper that

1) a shallow (one-sum layer) SPN is a universal model for smooth and compactly supported multivariate densities, and the underlying 1-D densities can be recovered (versus prescribed). That is, if the true density has low rank, then it can be pinned down and its components can be 'unraveled' via the CPD.
2) the low-rank assumption works well in various practical applications.

A multi-layer SPN, on the other hand, is akin to a hierarchical Tucker model, and thus no model identifiability claims can be made for deep SPNs – they can always be replaced by a shallow SPN with sufficiently many leaves.

SPNs enjoy a tractable marginalization and inference process, but our proposed model allows for even easier marginalization (by discarding the subset of factor matrices corresponding to the variables we are not interesting in), as well as easier and closed form inference. SPNs require specific structural constraints in order to guarantee exact inference [37]. In contrast to SPNs, our model does not require architecture specification.

### *B. Notation*

In this section we briefly present notation conventions and some tensor algebra preliminaries. We use the symbols $\mathbf{x}$,

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



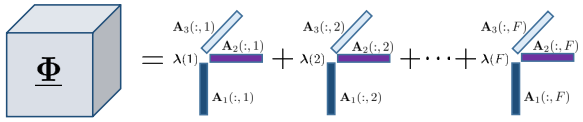


Fig. 1: CPD model of a 3-way tensor.

$\mathbf{X}$, $\underline{\mathbf{X}}$ for vectors, matrices and tensors respectively. We use the notation $\mathbf{x}(n)$, $\mathbf{X}(:,n)$, $\underline{\mathbf{X}}(:,:,n)$ to refer to a particular element of a vector, a column of a matrix and a slab of a tensor. Symbols $\|\mathbf{x}\|_2$, $\|\mathbf{X}\|_F$, and $\|\mathbf{x}\|_\infty$ correspond to $L_2$ norm, Frobenius norm, and infinity norm. Symbols $\circ$, $\circledast$, $\odot$ denote the outer, Hadamard and Khatri-Rao product respectively. The vectorization operator is denoted as $\text{vec}(\mathbf{X})$, $\text{vec}(\underline{\mathbf{X}})$ for a matrix and tensor respectively. Additionally, $\text{diag}(\mathbf{x}) \in \mathbb{C}^{K\times K}$ denotes the diagonal matrix with the elements of vector $\mathbf{x} \in \mathbb{C}^K$ on its diagonal. The set of integers $\{1,\dots,K\}$ is denoted as $[K]$.

### C. Relevant tensor algebra

An $N$-way tensor $\underline{\mathbf{\Phi}} \in \mathbb{C}^{K_1\times K_2\times\cdots\times K_N}$ is a multidimensional array whose elements are indexed by $N$ indices. Any tensor can be decomposed as a sum of $F$ rank-1 tensors

$$\underline{\mathbf{\Phi}} = \sum_{h=1}^{F} \boldsymbol{\lambda}(h)\mathbf{A}_1(:,h)\circ\mathbf{A}_2(:,h)\circ\cdots\circ\mathbf{A}_N(:,h), \qquad (1)$$

where $\mathbf{A}_n \in \mathbb{C}^{K_n\times F}$ and constraining the columns $\mathbf{A}_n(:,h)$ to have unit norm, the real scalar $\boldsymbol{\lambda}(h)$ absorbs the $h$-th rank-one tensor's scaling. A visualization is shown in Figure 1 for the case of $N=3$.

We use $\underline{\mathbf{\Phi}} = [\![\boldsymbol{\lambda}, \mathbf{A}_1,\dots,\mathbf{A}_N]\!]$ to denote the decomposition. When $F$ is minimal, it is called the rank of tensor $\underline{\mathbf{\Phi}}$, and the decomposition is called *Canonical* Polyadic Decomposition (CPD). A particular element of the tensor is given by $\underline{\mathbf{\Phi}}(k_1,k_2,\dots,k_N) = \sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N}\mathbf{A}_n(k_n,h)$. The vectorized form of the tensor can be expressed as $\text{vec}(\underline{\mathbf{\Phi}}) = \left(\odot_{n=1}^{N}\mathbf{A}_n\right)\boldsymbol{\lambda}$. We can express the mode-$n$ matrix unfolding which is a concatenation of all mode-$n$ 'fibers' of the tensor as $\underline{\mathbf{\Phi}}^{(n)} = (\odot_{k\neq n}\mathbf{A}_k)\text{diag}(\boldsymbol{\lambda}){\mathbf{A}_n}^T$, where $(\odot_{k\neq n}\mathbf{A}_k) = \mathbf{A}_N\odot\cdots\odot\mathbf{A}_{n+1}\odot\mathbf{A}_{n-1}\odot\cdots\odot\mathbf{A}_1$.

A key property of the CPD is that the rank-1 components are unique under mild conditions. For learning probabilistic latent variable models and latent representations, the uniqueness of tensor decomposition can be interpreted as model identifiability. A model is identifiable, if and only iff there is a unique set of parameters that is consistent with what we have observed.

*Theorem 1:* [29]: Let $k_{\mathbf{A}}$ be the Kruskal rank of $\mathbf{A}$, defined as the largest integer $k$ such that every $k$ columns of $\mathbf{A}$ are linearly independent. Given $\underline{\mathbf{\Phi}} = [\![\boldsymbol{\lambda}, \mathbf{A}_1,\dots,\mathbf{A}_N]\!]$, if $\sum_{n=1}^{N} k_{\mathbf{A}_n} \geq 2F+N-1$, then the rank of $\underline{\mathbf{\Phi}}$ is $F$ and the decomposition of $\underline{\mathbf{\Phi}}$ in rank-one terms is unique.
Better results allowing for higher tensor rank are available for generic tensors of given rank.

*Theorem 2:* [39]: Given $\underline{\mathbf{\Phi}} = [\![\boldsymbol{\lambda}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\!]$, assume, without loss of generality, that $K_1 \leq K_2 \leq K_3$. Let $\alpha,\beta$ be the largest integers such that $2^\alpha \leq K_1$ and $2^\beta \leq K_2$. If $F \leq 2^{\alpha+\beta-2}$ the decomposition of $\underline{\mathbf{\Phi}}$ in rank-one terms is unique almost surely.

## III. A Characteristic Function Approach

The characteristic function of a random variable $X$ is the Fourier transform of its PDF, and it conveys all information about $X$. The characteristic function can be interpreted as an expectation: the Fourier transform at frequency $\nu \in \mathbb{R}$ is $E\left[e^{j\nu X}\right]$. Similarly, the multivariate characteristic function is the multidimensional Fourier transform of the density of a random vector $\boldsymbol{X}$, which can again be interpreted as the expectation $E[e^{j\boldsymbol{\nu}^T\boldsymbol{X}}]$, where $\boldsymbol{\nu}$ is a vector of frequency variables. The expectation interpretation is crucial, because ensemble averages can be estimated via sample averages; and whereas direct nonparametric density estimation at point $x$ requires samples around $x$, estimating the characteristic function enables reusing all samples globally, thus enabling better sample averaging and generalization. This point is the first key to our approach. The difficulty, however, is that pinning down the characteristic function seemingly requires estimating an uncountable set of parameters. We need to reduce this to a *finite* parameterization with controllable error, and ultimately distill a parsimonious model that can learn from limited data and still generalize well. In order to construct an accurate joint distribution estimate that is scalable to high dimensions without making explicit and restrictive prior assumptions (such as a GMM model) on the nature of the density, and without requiring huge amounts of data, we encode the following key ingredients into our model.

- **Compactness of support**. In most cases, the random variables of interest are bounded, and these bounds are known or can be relatively easily estimated. The assumption of knowing the support of the sought distribution is not limiting in practice. Given that we are aiming to estimate a high-dimensional distribution we should naturally have access to much more data than is needed to estimate the support of any marginal distribution of one of the variables. Also note that we do not need to know the exact support – reasonable upper and lower bounds are enough to compress and shift the range.
- **Continuity of the underlying density and its derivatives**. The joint distribution is assumed to be sufficiently smooth in some sense, which enables the use of explicit or implicit interpolation.
- **Low-rank tensor modeling**. We show that joint characteristic functions can be represented as higher order tensors. In practice these tensor data are not unstructured. Low-rank tensor modeling provides a concise representation that captures the salient characteristics (the *principal components*) of the data distribution in the Fourier domain.

### A. The Univariate Case

Before we delve into the multivariate setting, it is instructive to examine the univariate case. Given a real-valued random

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



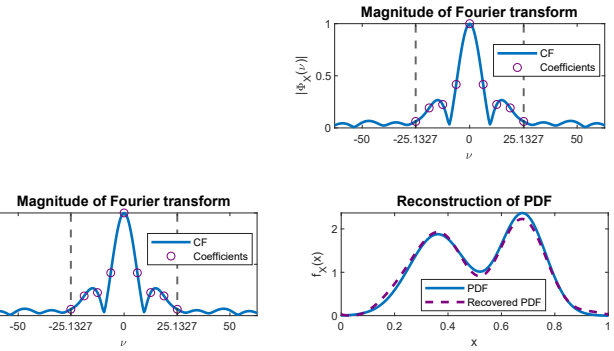


Fig. 2: The Univariate Case: Illustration of the key idea on a univariate Gaussian mixture of two distributions with means $\mu_1 = 0.35$, $\mu_2 = 0.7$ and standard deviations $\sigma_1 = 0.1$, $\sigma_2 = 0.08$. The PDF can be (approximately) recovered from only 9 uniform samples of its Characteristic Function (CF).

variable $X$ with compact support $S_X$, the Probability Density Function (PDF) $f_X$ and its corresponding Characteristic Function (CF) $\Phi_X$ form a Fourier transform pair:

$$\Phi_X(\nu) := \int_{S_X} f_X(x) e^{j\nu x} dx = E[e^{j\nu X}], \tag{2}$$

$$f_X(x) := \frac{1}{2\pi} \int_{-\infty}^{\infty} \Phi_X(\nu) e^{-j\nu x} d\nu. \tag{3}$$

Note that $\Phi_X(0) = \int_{-\infty}^{\infty} f_X(x)dx = 1$. Without loss of generality, we can apply range normalization and mean shifting so that $sX + c \in [0, 1]$ – the transformation is invertible. We may therefore assume that $S_X = [0, 1]$. Every PDF supported in $[0, 1]$ can be uniquely represented over its support by an infinite Fourier series,

$$f_X(x) = \sum_{k=-\infty}^{\infty} \Phi_X[k] e^{-j2\pi kx}, \tag{4}$$

where $\Phi_X[k] = \Phi_X(\nu)\big|_{\nu=2\pi k}$, $k \in \mathbb{Z}$. This shows that *countable* parameterization through samples of the characteristic function suffices for compactly supported densities. But this is still not enough - we need a finite parametrization. Thankfully, if $f_X$ is sufficiently differentiable in the sense that $f_X \in C^p$ i.e., all its derivatives $\frac{\partial f_X}{\partial x}, \frac{\partial^2 f_X}{\partial x^2}, \cdots, \frac{\partial^p f_X}{\partial x^p}$ exist and are continuous we have that

*Lemma 1:* (e.g., see [40]): If $f_X \in C^p$, then

$$|\Phi_X[k]| = \mathcal{O}\left(\frac{1}{1 + |k|^p}\right).$$

It is therefore possible to use a truncated series

$$\widehat{f}_X(x) = \sum_{k=-K}^{K} \Phi_X[k] e^{-j2\pi kx},$$

with proper choice of $K$ that will not incur significant error. Invoking Parseval's Theorem

$$\|f_X - \widehat{f}_X\|_2^2 = \sum_{|k|>K} |\Phi_X[k]|^2,$$

which is controllable by the *smoothing parameter* $K$. The $k$-th Fourier coefficient

$$\Phi_X[k] = \int_0^1 f_X(x) e^{j2\pi kx} dx = E[e^{j2\pi kX}]$$

can be conveniently estimated via the sample mean

$$\widehat{\Phi}_X[k] = \frac{1}{M} \sum_{m=1}^{M} e^{j2\pi k x_m}$$

Here $M$ is the number of available realizations of the random variable $X$.

A toy example to illustrate the idea is shown in Figure 2. For this example, we are given $M = 500$ realizations of a random variable $X$, which is a mixture of two Gaussian distributions with means $\mu_1 = 0.35$, $\mu_2 = 0.7$ and standard deviations $\sigma_1 = 0.1$, $\sigma_2 = 0.08$. The recovered PDF is very close to the true PDF using only 9 coefficients of the CF.

### B. The Multivariate Case

In the multivariate case, we are interested in obtaining an estimate $\widehat{f}_{\boldsymbol{X}}$ of the true density $f_{\boldsymbol{X}}$ of a random vector $\boldsymbol{X} := [X_1, \ldots, X_N]^T$. The *joint* or *multivariate characteristic function* of $\boldsymbol{X}$ is a function $\Phi_{\boldsymbol{X}} : \mathbb{R}^N \to \mathbb{C}$ defined as

$$\Phi_{\boldsymbol{X}}(\boldsymbol{\nu}) := E\left[e^{j\boldsymbol{\nu}^T \boldsymbol{X}}\right], \tag{5}$$

where $\boldsymbol{\nu} := [\nu_1, \ldots, \nu_N]^T$. For any given $\boldsymbol{\nu}$, given a set of realizations $\{\mathbf{x}_m\}_{m=1}^M$, we can estimate the empirical characteristic function of the sequence as

$$\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) = \frac{1}{M} \sum_{m=1}^{M} e^{j\boldsymbol{\nu}^T \mathbf{x}_m}. \tag{6}$$

Under mixing conditions such that sample averages converge to ensemble averages, the corresponding PDF can be uniquely recovered via the multidimensional inverse Fourier transform

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{(2\pi)^N} \int_{\mathbb{R}^N} \Phi_{\boldsymbol{X}}(\boldsymbol{\nu}) e^{-j\boldsymbol{\nu}^T \boldsymbol{x}} d\boldsymbol{\nu}. \tag{7}$$

If the support of the joint PDF $f_{\boldsymbol{X}}(\mathbf{x})$ is contained within the hypercube $S_{\boldsymbol{X}} = [0, 1]^N$, then similar to the univariate case, it can be represented by a multivariate Fourier series

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \sum_{k_1=-\infty}^{\infty} \cdots \sum_{k_N=-\infty}^{\infty} \Phi_{\boldsymbol{X}}[\boldsymbol{k}] e^{-j2\pi \mathbf{k}^T \mathbf{x}}, \tag{8}$$
$$\text{where } \Phi_{\boldsymbol{X}}[\mathbf{k}] = \Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\big|_{\boldsymbol{\nu}=2\pi\mathbf{k}}, \mathbf{k} = [k_1, \ldots, k_N]^T.$$

*Lemma 2:* (see e.g., [40]): *For any $p \in \mathbb{N}$, if the partial derivatives $\frac{\partial^{\theta_1}}{\partial x_1^{\theta_1}} \cdots \frac{\partial^{\theta_N}}{\partial x_N^{\theta_N}} f_{\boldsymbol{X}}(\mathbf{x})$ exist and are absolutely integrable for all $\theta_1, \ldots, \theta_N$ with $\sum_{n=1}^N \theta_n \leq p$ then the rate of decay of the magnitude of the $\mathbf{k}$-th Fourier coefficient $|\Phi_{\boldsymbol{X}}[\mathbf{k}]|$ obeys $|\Phi_{\boldsymbol{X}}[\mathbf{k}]| = \mathcal{O}\left(\frac{1}{1+\|\mathbf{k}\|_2^p}\right)$.*

The smoother the underlying PDF, the faster its Fourier coefficients and the approximation error tend to zero. Thus we can view the joint PDF through the lens of functions with only

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



low frequency harmonics. Specifically, it is known [41], [42, Chapter 23] that the approximation error of the truncated series with absolute cutoffs $\{K_n\}_{n=1}^N$ is upper bounded by

$$\|f_{\boldsymbol{X}} - \widehat{f}_{\boldsymbol{X}}\|_\infty \leq C \sum_{n=1}^{N} \frac{\omega_n\left(\frac{\partial^{\theta_n}}{\partial x_n^{\theta_n}} f_{\boldsymbol{X}}, \frac{1}{1+K_n}\right)}{(1+K_n)^{\theta_n}}, \tag{9}$$

where $\omega_n(f_{\boldsymbol{X}}, \delta) :=$

$$\sup_{|x_j - x_j'| \leq \delta} \left| f_{\boldsymbol{X}}(x_1, \ldots, x_j, \ldots, x_N) - f_{\boldsymbol{X}}(x_1, \ldots, x_j', \ldots, x_N) \right|,$$

and

$$C = C_2 \left(1 + C_1 \prod_{n=1}^{N} \log K_n\right).$$

$C_1, C_2$ are constants independent of $K_n$. The smoother the underlying PDF, the smaller the obtained finite parametrization error. It follows that we can approximate $f_{\boldsymbol{X}}$

$$\widehat{f}_{\boldsymbol{X}}(\mathbf{x}) = \sum_{k=-K_1}^{K_1} \cdots \sum_{k_N=-K_N}^{K_N} \Phi_{\boldsymbol{X}}[\mathbf{k}] e^{-j2\pi \mathbf{k}^T \mathbf{x}}. \tag{10}$$

The truncated Fourier coefficients can be naturally represented using an $N$-way tensor $\underline{\boldsymbol{\Phi}}$ where

$$\underline{\boldsymbol{\Phi}}(k_1, \ldots, k_N) = \Phi_{\boldsymbol{X}}[\mathbf{k}]. \tag{11}$$

## IV. Proposed Approach: Breaking the Curse of Dimensionality

We have obtained a finite parameterization with controllable and bounded error, but the number of parameters $(2K_1+1) \times \cdots \times (2K_N+1)$ obtained by truncating $\Phi_{\boldsymbol{X}}$ as above grows exponentially with $N$. This curse of dimensionality can be circumvented by focusing on the *principal components* of the resulting tensor, i.e., introducing a low-rank parametrization of the *Characteristic Tensor* obtained by truncating the multidimensional Fourier series. Keeping the first $F$ principal components, the number of parameter reduces from order of $K_1 \times \cdots \times K_N$ to order of $(K_1 + \cdots + K_N)F$. Introducing the rank-$F$ CPD in Equation (10), one obtains the approximate model

$$\tilde{f}_{\boldsymbol{X}}(\mathbf{x}) = \sum_{k_1=-K}^{K} \cdots \sum_{k_N=-K}^{K} \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} \Phi_{X_n|H=h}[k_n] \, e^{-j2\pi k_n x_n}, \tag{12}$$

where $H$ can be interpreted as a latent ($H$ for 'hidden') random variable, $\Phi_{X_n|H=h}[k_n]$ is the characteristic function of $X_n$ conditioned on $H=h$

$$\begin{aligned} \Phi_{X_n|H=h}[k_n] &:= \Phi_{X_n|H=h}(\nu|h)\big|_{\nu=2\pi k_n} \\ &= \mathbb{E}_{X_n|H=h}\left[e^{j2\pi k_n X_n}\right], \end{aligned} \tag{13}$$

and we stress that for high-enough $F$, this representation is exact without loss of generality – see, e.g., [20]. For the rest of the paper, we consider $K = K_1 = \cdots = K_N$ for brevity.

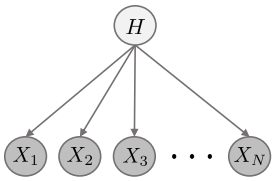


Fig. 3: The proposed generative model $\tilde{f}_{\boldsymbol{X}}(\mathbf{x})$ admits an interpretation as a mixture of $F$ product distributions i.e., a latent variable naive Bayes interpretation.

By linearity and separability of the multidimensional Fourier transformation it follows that

$$\begin{aligned} \tilde{f}_{\boldsymbol{X}}(\mathbf{x}) &= \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} \sum_{k_n=-K}^{K} \Phi_{X_n|H=h}[k_n] \; e^{-j2\pi k_n x_n} \\ &= \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} f_{X_n|H}(x_n|h). \end{aligned} \tag{14}$$

This generative model can be interpreted as mixture of product distributions [33]. The joint PDF $f_{\boldsymbol{X}}$ is a mixture of $F$ separable component PDFs, i.e., there exists a 'hidden' random variable $H$ taking values in $\{1, \ldots, F\}$ that selects the operational component of the mixture, and given $H$ the random variables $X_1, \ldots, X_N$ become independent (See Figure 3 for visualization of this model). We have thus shown the following result:

*Proposition 1:* Truncating the multidimensional Fourier series (sampled multivariate characteristic function) of any compactly supported random vector is equivalent to approximating the corresponding multivariate density by a finite mixture of separable densities.

Thus, by choosing appropriate $K$ and $F$, it is possible to represent and approximate any compactly supported density that it is sufficiently smooth by the proposed model. See Figures 5, 4 where we showcase how each parameter affects the modeling of complex structures in 2D synthetic datasets.

Conversely, if one *assumes* that the sought multivariate density is a finite mixture of separable densities, then it is easy to show that the corresponding characteristic function is likewise a mixture of separable characteristic functions:

$$\begin{aligned} \Phi_{\boldsymbol{X}}(\boldsymbol{\nu}) &= E\left[e^{j\boldsymbol{\nu}^T \boldsymbol{X}}\right] \\ &= E_H\left[E_{\boldsymbol{X}|H}\left[e^{j\nu_1 X_1} \cdots e^{j\nu_N X_N}\right]\right] \\ &= E_H\left[\Phi_{X_1|H}(\nu_1|H) \cdots \Phi_{X_n|H}(\nu_N|H)\right] \\ &= \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} \Phi_{X_n|H}(\nu_n|h). \end{aligned} \tag{15}$$

If we sample the above on any finite $N$-dimensional grid, we obtain an $N$-way tensor and its polyadic decomposition. Such decomposition is unique, under mild conditions [20]. It follows that:

*Proposition 2:* A compactly supported multivariate ($N \geq 3$) mixture of separable densities is identifiable from (samples of) its characteristic function, under mild conditions.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



The main reasons for working in the Fourier / characteristic function domain are that

1) truncation is a "universal" approximation in the sense that it only requires smoothness of the joint PDF;
2) the Fourier transform is "global" allowing us to estimate the PDF in places where there is a scarcity of "local" samples – which is a key problem in high-dimensional cases;
3) since we limit ourselves to regular samples of the characteristic function (multivariate Fourier series), we can invert using the computationally advantageous Fast Fourier Transform; and
4) relative to the moment generating function, the characteristic function always exists.

The above analysis motivates the following course of action. Given a set of realizations $\{\mathbf{x}_m\}_{m=1}^M$,

1) estimate
$$\underline{\mathbf{\Phi}}[\mathbf{k}] = \frac{1}{M}\sum_{m=1}^{M} e^{j2\pi \mathbf{k}^T \mathbf{x}_m}, \tag{16}$$
2) fit a low-rank model
$$\underline{\mathbf{\Phi}}[\mathbf{k}] \approx \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} \Phi_{X_n|H=h}[k_n], \tag{17}$$
3) and invert using
$$f_{\boldsymbol{X}}(\mathbf{x}) = \sum_{h=1}^{F} p_H(h) \prod_{n=1}^{N} f_{X_n|H}(x_n|h), \text{ where}$$
$$f_{X_n|H}(x_n|h) = \sum_{k_n=-K}^{K} \Phi_{X_n|H=h}[k_n] e^{-j2\pi k_n x_n}. \tag{18}$$

When building any statistical model, identifiability is a fundamental question. A statistical model is said to be identifiable when, given a sufficient number of observed data, it is possible to uniquely recover the data-generating distribution. When applying a non-identifiable model, different structures or interpretations may arise from distinct parametrizations that explain the data equally well. Most deep generative models do not address the question of identifiability, and thus may fail to deliver the true latent representations that generate the observations. Our approach is fundamentally different, because it builds on rigorous and controllable Fourier approximation and identifiability of the characteristic tensor.

In the Appendix (Section VIII), we provide additional statistical insights regarding the proposed methodology, including the asymptotic behavior of the empirical characteristic function and the mean squared error reduction afforded by low-rank tensor modeling in the characteristic function domain.

Two issues remain. First, uniqueness of CPD only implies that each rank-one factor is unique, but leaves scaling/counter-scaling freedom in $p_H$ and the conditional characteristic functions. To resolve this, we can use the fact that each conditional characteristic function must be equal to 1 at the origin (zero frequency). Likewise, $p_H$ must be a valid probability mass function. These constraints fix the scaling indeterminacy.

We note here that, under certain rank conditions (see Section II-C) on the Fourier series coefficient tensor, the proposed method ensures that the reconstructed density is positive and integrates to one, as it should. This is due to the uniqueness properties of the Fourier series representation and the CPD: if there exists a density that generates a low-rank characteristic tensor, and that tensor can be uniquely decomposed, the sum of Fourier inverses of its components is unique, and therefore equal to the generating density. Under ideal low-rank conditions, this is true even if we ignore the constraints implied by positivity when we decompose the characteristic tensor in the Fourier domain. This is convenient because strictly enforcing those in the Fourier domain would entail cumbersome spectral factorization-type (positive semidefinite) constraints. We therefore propose the following formulation:

$$\begin{aligned} \min \quad & \|\underline{\mathbf{\Phi}} - [\![\boldsymbol{\lambda}, \mathbf{A}_1, \ldots, \mathbf{A}_N]\!]\|_F^2 \\ \text{subject to} \quad & \boldsymbol{\lambda} \geq \mathbf{0}, \mathbf{1}^T\boldsymbol{\lambda} = 1, \\ & \mathbf{A}_n(K+1,:) = \mathbf{1}^T,\ n = 1 \ldots N, \end{aligned} \tag{19}$$

where $\mathbf{A}_n(K+1+k_n, h)$ holds $\Phi_{X_n|H=h}[k_n]$, and $\boldsymbol{\lambda}(h)$ holds $p_H(h)$.

The second issue is more important. When $N$ is large, instantiating or even allocating memory for the truncated characteristic tensor is a challenge, because its size grows exponentially with $N$. Fortunately, there is a way around this problem. The main idea is that instead of estimating the characteristic tensor of all $N$ variables, we may instead estimate the characteristic tensors of subsets of variables, such as triples, which partially share variables with other triples. The key observation that enables this approach is that the marginal characteristic function of any subset of random variables is also a constrained complex CPD model that inherits parameters from the grand characteristic tensor. Marginalizing with respect to the $n'$-th random variable, we have that

$$\begin{aligned} \underline{\mathbf{\Phi}}(k_1, \ldots, k_{n'} = 0, \ldots, k_N) &= \sum_{h=1}^{F} \prod_{\substack{n=1 \\ n\neq n'}}^{N} \Phi_{X_n|H}[k_n] \underbrace{\Phi_{X_{n'}|H}[0]}_{=1} \\ &= \sum_{h=1}^{F} \prod_{\substack{n=1 \\ n\neq n'}}^{N} \Phi_{X_n|H}[k_n]. \end{aligned} \tag{20}$$

Thus, a characteristic function of any subset of three random variables $X_i, X_j, X_\ell$ (triples) can be written as a third-order tensor, $\underline{\mathbf{\Phi}}_{ij\ell}$, of rank $F$. These sub-tensors can be jointly decomposed in a coupled fashion (see the optimization problem in (22)) to obtain the sought factors that allow synthesizing the big characteristic tensor. In this way, we beat the curse of dimensionality for low-enough model ranks. In addition to affording significant computational and memory reduction, unlike neural network based methods, the above approach allows us to work with fewer and even missing data during the training phase, i.e., only having access to incomplete realizations of the random vector of interest. We estimate lower-order characteristic function values from only those realizations that all three random variables in a given triple appear together. Our method can easily be adapted to work with pairs or quadruples, but reliably estimating fourth-order characteristic functions requires more sample averaging, whereas the $2-$ dimensional

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



case requires stricter identifiability conditions. Hence working with 3-D tensors offers a good compromise between these conflicting considerations.

In earlier work, we proposed a similar approach for the categorical case where every random variable is finite-alphabet and the task is to estimate the joint probability mass function (PMF) [21]. There we showed that every joint PMF of a finite-alphabet random vector can be represented by a naïve Bayes model with a finite number of latent states (rank). If the rank is low, the high dimensional joint PMF is almost surely identifiable from lower-order marginals – which is reminiscent of Kolmogorov extension.

In case of continuous random variables, however, the joint PDF can no longer be directly represented by a tensor. One possible solution could be discretization, but this unavoidably leads to discretization error. In this work, what we show is that we can *approximately* represent any smooth joint PDF (and evaluate it at any point) using a low-rank tensor *in the characteristic function domain*, thereby avoiding discretization loss altogether.

Our joint PDF model enables easy computation of any marginal or conditional density of subsets of variables of $\boldsymbol{X}$. Using the conditional expectation, the response variable, taken without loss of generality to be the last variable $X_N$, can be estimated in the following way (see detailed derivation in Section VIII.).

$$E\left[X_N|X_1,\ldots,X_{N-1}\right] = \frac{1}{c_1}\sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N-1}\sum_{k_n=-K}^{K} \mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n}\sum_{k_N=-K}^{K} c_{2,k_N}\mathbf{A}_N(k_N,h) \tag{21}$$

$$\text{where } c_1 = \sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N-1}\sum_{k_n=-K}^{K}\mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n},$$

$$\text{and } c_{2,k_N} = \frac{e^{-j2\pi k_N}}{-j2\pi k_N} + \frac{1-e^{-j2\pi k_N}}{[-j2\pi k_N]^2}.$$

One of the very appealing properties of the proposed approach is that it is a generative model that affords easy sampling. According to Equation (18), a sample of the multivariate distribution can be generated by first drawing $H$ according to $p_H$ and then independently drawing samples for each variable $X_n$ from the conditional PDF $f_{X_n|H}$. The resulting generative model can be visualized in Figure 3.

### A. Algorithm: Coupled Tensor Factorization

We formulate the problem as a coupled complex tensor factorization problem and propose a Block Coordinate Descent algorithm for recovering the latent factors of the CPD model representing the joint CF. Then, we only need to invert each conditional CF and synthesize the joint PDF. We refer to this approach as Low-Rank Characteristic Function based Density Estimation (LRCF-DE).

**Algorithm 1 Low-Rank Characteristic Function based Density Estimation (LRCF-DE).**

**Input**: A real-valued dataset $D \in \mathbb{R}^{N\times M}$, parameters $F, K$.
**Output**: The joint PDF model $f_{\boldsymbol{X}}$.
Compute $\underline{\boldsymbol{\Phi}}_{ij\ell}\forall i,j,\ell \in \{1,\ldots,N\},\ \ell > j > i$ from training data, using (16).
Initialize $\boldsymbol{\lambda}, \mathbf{A}_1, \ldots, \mathbf{A}_N$ in compliance with their constraints.
**repeat**
  **for all** $n \in \{1,\ldots,N\}$ **do**
    Solve the optimization problem with respect to $\mathbf{A}_n$ defined in (23).
  **end for**
  Update $\boldsymbol{\lambda}$ by solving the optimization problem defined in (25).
**until** convergence criterion satisfied
Assemble the joint PDF as in equation (27).

We begin by defining the following coupled tensor factorization problem

$$\begin{aligned}\min_{\boldsymbol{\lambda},\mathbf{A}_1,\ldots,\mathbf{A}_N} &\sum_i\sum_{j>i}\sum_{\ell>j}\left\|\underline{\boldsymbol{\Phi}}_{ij\ell} - [\![\boldsymbol{\lambda},\mathbf{A}_i,\mathbf{A}_j,\mathbf{A}_\ell]\!]\right\|_F^2 \\ \text{subject to}\quad &\boldsymbol{\lambda}\geq\mathbf{0}, \mathbf{1}^T\boldsymbol{\lambda}=1, \\ &\mathbf{A}_n(K+1,:)=\mathbf{1}^T,\ n=1,\ldots,N.\end{aligned} \tag{22}$$

Each lower-dimensional joint CF of triples, $\underline{\boldsymbol{\Phi}}_{ij\ell}$, can be computed directly from the observed data via sample averaging according to equation (16). The formulated optimization problem (22) is non-convex and NP-hard. However it becomes convex with respect to each variable if we fix the remaining ones and can be handled using alternating optimization. By using the mode-1 matrix unfolding of each tensor $\underline{\boldsymbol{\Phi}}_{ij\ell}$, the optimization problem with respect to $\mathbf{A}_i$ becomes

$$\begin{aligned}\min_{\mathbf{A}_i}\quad &\sum_{j\neq i}\sum_{\ell\neq i,\ell>j}\left\|\underline{\boldsymbol{\Phi}}^{(1)}_{ij\ell} - (\mathbf{A}_\ell\odot\mathbf{A}_j)\text{diag}(\boldsymbol{\lambda})\mathbf{A}_i^T\right\|_F^2 \\ \text{subject to}\quad &\mathbf{A}_i(K+1,:)=\mathbf{1}^T.\end{aligned} \tag{23}$$

The exact update for each factor $\mathbf{A}_i$ can be computed as

$$\mathbf{A}_i \leftarrow {\mathbf{G}_i}^{-1}\mathbf{V}_i, \tag{24}$$

where

$$\begin{aligned}\mathbf{G}_i &= (\boldsymbol{\lambda}\boldsymbol{\lambda}^T)\circledast\sum_{j\neq i}\sum_{\ell\neq i,\ell>j}\mathbf{Q}_{\ell j}^H\mathbf{Q}_{\ell j}, \\ \mathbf{V}_i &= \text{diag}(\boldsymbol{\lambda})\sum_{j\neq i}\sum_{\ell\neq i,\ell>j}\mathbf{Q}_{\ell j}^H\underline{\boldsymbol{\Phi}}^{(1)}_{ij\ell}, \\ \mathbf{Q}_{\ell j} &= \mathbf{A}_\ell\odot\mathbf{A}_j.\end{aligned}$$

For each update, the row of $\mathbf{A}_i$ that corresponds to zero frequency is removed and updating $\mathbf{A}_i$ becomes an unconstrained complex least squares problem. A vector of ones is appended at the same row index after each update $\mathbf{A}_i$. Due to role symmetry the same form holds for each factor $\mathbf{A}_n$.

Now, for the $\boldsymbol{\lambda}$-update we solve the following optimization problem

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



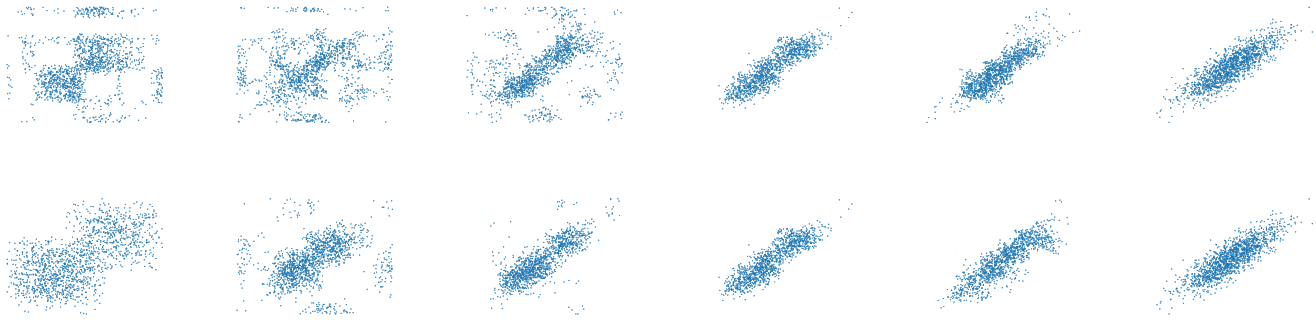

Fig. 4: Visualization of synthetic $M' = 1500$ samples generated from the proposed model trained on the Weight-Height dataset for different $F, K$ parameter combinations – The rightmost figure represents the ground truth. On the first row, we fixed $K$, $K = 4$, and varied $F$, $F \in [2, 4, 6, 8, 10]$ (from left to right). On the second row, we fixed $F$, $F = 8$, and varied $K$, $K \in [1, 2, 3, 4, 5]$ (from left to right).

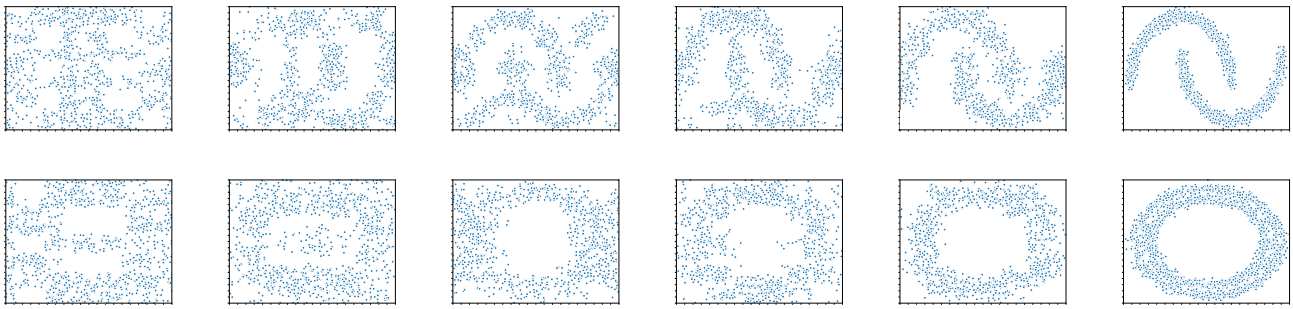

Fig. 5: Qualitative synthetic $M' = 1500$ samples obtained from the proposed model trained on $M = 2000$ samples of a toy 2-D Moons and Circles datasets (for fixed $K$, $K = 11$, from left to right $F \in [1, 2, 3, 4, 6]$ – The rightmost figures represent the ground truth).

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \sum_{i} \sum_{j>i} \sum_{\ell>j} \left\| \text{vec}(\underline{\boldsymbol{\Phi}}_{ij\ell}) - (\mathbf{A}_\ell \odot \mathbf{A}_j \odot \mathbf{A}_i)\boldsymbol{\lambda} \right\|_F^2 \\ \text{subject to} \quad & \boldsymbol{\lambda} \geq \mathbf{0}, \ \mathbf{1}^T\boldsymbol{\lambda} = 1. \end{aligned} \tag{25}$$

The optimization problem (25) is a least squares problem with a probability simplex constraint. We use an ADMM algorithm to tackle it. Towards this end, we reformulate the optimization problem by introducing an auxiliary variable $\hat{\boldsymbol{\lambda}}$ and rewrite the problem equivalently as

$$\begin{aligned} \min_{\boldsymbol{\lambda}, \hat{\boldsymbol{\lambda}}} \quad & f(\hat{\boldsymbol{\lambda}}) + r(\boldsymbol{\lambda}) \\ \text{subject to} \quad & \hat{\boldsymbol{\lambda}}^T = \boldsymbol{\lambda}, \end{aligned}$$

where, $f(\hat{\boldsymbol{\lambda}}) = \sum_{i} \sum_{j>i} \sum_{\ell>j} \|\text{vec}(\underline{\boldsymbol{\Phi}}_{ij\ell}) - (\mathbf{A}_\ell \odot \mathbf{A}_j \odot \mathbf{A}_i)\hat{\boldsymbol{\lambda}}\|_F^2$ and $r(\boldsymbol{\lambda})$ is the indicator function for the probability simplex. $C = \{\boldsymbol{\lambda} | \boldsymbol{\lambda} \geq \mathbf{0}, \mathbf{1}^T\boldsymbol{\lambda} = 1\}$,

$$r(\boldsymbol{\lambda}) = \begin{cases} 0, & \boldsymbol{\lambda} \in C \\ \infty, & \boldsymbol{\lambda} \notin C. \end{cases}$$

At each iteration $\tau$, we perform the following updates

$$\begin{aligned} \hat{\boldsymbol{\lambda}}^{\tau+1} &\leftarrow (\mathbf{G} + \rho\mathbf{I})^{-1}(\mathbf{V} + \rho(\boldsymbol{\lambda}^\tau + \mathbf{u}^\tau)) \\ \boldsymbol{\lambda}^{\tau+1} &\leftarrow \mathcal{P}_C(\boldsymbol{\lambda}^\tau - \hat{\boldsymbol{\lambda}}^{\tau+1} + \mathbf{u}^\tau) \\ \mathbf{u}^{\tau+1} &\leftarrow \mathbf{u}^\tau + \boldsymbol{\lambda}^{\tau+1} - \hat{\boldsymbol{\lambda}}^{\tau+1}, \end{aligned}$$

where

$$\begin{aligned} \mathbf{G} &= \sum_{i} \sum_{j>i} \sum_{\ell>j} \mathbf{Q}_{\ell ji}^H \mathbf{Q}_{\ell ji}, \\ \mathbf{V} &= \sum_{i} \sum_{j>i} \sum_{\ell>j} \mathbf{Q}_{\ell ji}^H \text{vec}(\underline{\boldsymbol{\Phi}}), \\ \mathbf{Q}_{\ell ji} &= \mathbf{A}_\ell \odot \mathbf{A}_j \odot \mathbf{A}_i. \end{aligned} \tag{26}$$

$\mathcal{P}_C(\mathbf{y})$ denotes the projection operator onto the convex set $C$ – it computes the Euclidean projection of the real part of a point $\mathbf{y} = [y_1, \ldots, y_F]^T \in \mathbb{C}^F$ onto the probability simplex

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^F} \quad & \frac{1}{2}\|\mathbf{x} - \Re(\boldsymbol{y})\|_F^2 \\ \text{subject to} \quad & \mathbf{x} \geq \mathbf{0}, \ \mathbf{1}^T\mathbf{x} = 1, \end{aligned}$$

using the method described in [43]. The overall procedure is described in Algorithm 1.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



As the final step, the factors are assembled from the triples and the joint CF over all variables is synthesized as $\underline{\mathbf{\Phi}} = [\![\boldsymbol{\lambda}, \mathbf{A}_1, \ldots, \mathbf{A}_N]\!]$. Given, the model of the joint CF, the corresponding joint PDF model can be recovered at any point as

$$f_{\boldsymbol{X}}(\mathbf{x}) = \sum_{h=1}^{F} \boldsymbol{\lambda}(h) \prod_{n=1}^{N} \sum_{k_n=-K}^{K} \mathbf{A}_n(k_n, h) e^{-j2\pi k_n x_n}. \quad (27)$$

## V. Experiments

### A. Low Dimensional Toy Data

We first show motivating results from modeling low dimensional datasets and showcase the expressivity of the proposed model as well as the significance of each parameter. Our model depends on the degree of smoothness and tensor rank ($K$ and $F$). We pick the number of Fourier series coefficients and the tensor rank through cross-validation for real data. Using these figures, one can see how the model changes when varying one of the two parameters separately. We begin by modeling $M = 2000$ samples from the Weight-Height dataset. In Figure 4, we present $M' = 1500$ synthetic samples obtained from the proposed model for different smoothing parameters $K \in [1, 2, 3, 4, 5]$ and ranks $F \in [2, 4, 6, 8, 10]$. By judiciously selecting the parameter search space, our approach yields an expressive generative model that can well-represent the data.

Following the same procedure, we now visualize $M = 2000$ samples from the 2-D Moons and Circles datasets. We fix the number of smoothing coefficients $K$, $K = 11$, and visualize synthetic $M' = 1500$ samples obtained from the proposed model for different approximation ranks $F \in [1, 2, 3, 4, 6]$ in Figure 5. The results show that our model is able to capture complex structures and properties of the data for an appropriate choice of rank $F$.

### B. Real Data

We test the proposed approach on datasets (see a brief description of the datasets in Table II) obtained from the UCI machine learning repository [44].

For each dataset we randomly hide $20\%$ of data (testing set) and consider the remaining entries as observed information (training set). The parameters, which include the tensor rank $F$ and the smoothing parameter $K$, are chosen using cross-validation. The smoothing parameter $K$ is chosen from the set $\{5, 10, 20, 25, 30\}$ and the rank $F$ from $\{5, 10, 20, 30, 50, 100\}$. We use $20\%$ of the training data as validation data, where we seek to find the optimal parameter values maximizing the average log-likelihood of the validation samples. Once the hyperparameters are chosen, we train the model using all the training data (including the validation data) and measure its performance on the testing set. We compare our approach against standard baselines described in section II-A.

Evaluating the quality of density models is an open and difficult problem [45]. Following the approach in [35], [15], we calculate and report the average log-likelihood of unseen data samples (testing set), further averaged over 5 random data splits. The results are shown in Table I. LRCF-DE has a higher average test sample log likelihood on almost all datasets. Overall, we observe that our method outperforms the baselines in $4$ datasets and is comparable to the winning method in the remaining ones.

Following the derivation in Equation (21), we test the proposed model in several regression tasks. We evaluate and report the Mean Absolute Error (MAE) in estimating $X_N$ for the unseen data samples in Table III and additional results for multi-output regression are presented in Table IV. Overall, we observe that LRCF-DE outperforms the baselines on almost all datasets, and performs comparable to the winning method in the remaining ones.

We have to stress again the fact that neural network based density estimation methods evaluate multivariate densities point-wise. These methods cannot impute more than a few missing elements in the input as grid search becomes combinatorial. Due to the interpretation of the approximation of the sought density as a finite mixture of separable densities and the coupled tensor factorization approach, our method allows us to easily work with missing data during both training and testing. Here, we showcase the results of LRCF-DE against MAF for simultaneously predicting the last two random variables of each dataset given the remaining ones.

| Data set | LRCF-DE | MAF |
|---|---|---|
| Red wine | 0.82 | 0.91 |
| White wine | 0.93 | 0.97 |
| First-order theorem proving (F-O.TP) | 0.69 | 0.72 |
| Polish companies bankruptcy (PCB) | 4.97 | 5.46 |
| Superconductivty | 20.84 | 20.72 |
| Corel Images | 1.36 | 1.59 |
| Gas Sensor Array Drift (Gas Sensor) | 25.7 | 26.1 |

TABLE IV: MAE for multi-output regression tasks.

The reduction in free parameters makes the proposed model particularly beneficial in the low-sample regime. We conducted an additional experiment on the Gas dataset to study how our model performs in terms of test-set log-likelihood when the number of samples is varied in comparison with the best performing neural network based PDF estimator from the baselines considered, namely, MAF. The results in Figure 6 verify that small to moderate training sample sizes result in much better LRCF-DE performance than MAF.

As our last experiment, we train LRCF-DE to learn the joint distribution of grayscale images from the USPS dataset [46], which contains $9298$ images of handwritten digits of size $16 \times 16 \rightarrow N = 256$. The number of examples for each digit is shown in Table VIII. The purpose of this experiment is to show that one can obtain reasonably accurate samples of digit images, by only modeling the distribution of triples of variables, something which has never been done before on images. We sample from the resulting $256$-dimensional model, and provide visualization of the generated data. We fix the tensor rank to $F = 8$ and the smoothing parameter to $K = 15$, and draw $8$ random samples of each digit (class). The resulting samples are shown in Figure 7, and they are very pleasing –

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



| Data set | MoG | KDE | RNADE | MAF | LRCF-DE |
|---|---|---|---|---|---|
| Red wine | $11.9 \pm 0.29$ | $9.9 \pm 0.16$ | $14.41 \pm 0.16$ | $\mathbf{15.2 \pm 0.09}$ | $\mathbf{16.4 \pm 0.67}$ |
| White wine | $16.1 \pm 1.48$ | $14.8 \pm 0.12$ | $17.1 \pm 0.26$ | $\mathbf{17.3 \pm 0.20}$ | $\mathbf{18.4 \pm 0.17}$ |
| F-O.TP | $125.4 \pm 7.79$ | $103.05 \pm 0.84$ | $\mathbf{152.48 \pm 5.62}$ | $149.6 \pm 8.32$ | $\mathbf{154.34 \pm 8.43}$ |
| PCB | $152.9 \pm 3.88$ | $147.6 \pm 1.63$ | $171.7 \pm 2.75$ | $\mathbf{179.6 \pm 1.62}$ | $\mathbf{194.4 \pm 2.43}$ |
| Superconductivty | $134.7 \pm 3.47$ | $127.2 \pm 2.82$ | $140.2 \pm 1.03$ | $\mathbf{143.5 \pm 1.32}$ | $\mathbf{146.1 \pm 2.31}$ |
| Corel Images | $211.7 \pm 1.04$ | $201.4 \pm 1.18$ | $\mathbf{223.6 \pm 0.88}$ | $218.2 \pm 1.35$ | $\mathbf{222.6 \pm 1.25}$ |
| Gas Sensor | $310.3 \pm 3.47$ | $296.48 \pm 1.62$ | $\mathbf{316.3 \pm 3.57}$ | $315.4 \pm 1.458$ | $\mathbf{316.6 \pm 2.35}$ |

TABLE I: Average test-set log-likelihood per datapoint for 5 different models on UCI datasets; **higher is better**.

| Data set | N | M |
|---|---|---|
| Red wine | 11 | 1599 |
| White wine | 11 | 4898 |
| First-order theorem proving (F-O.TP) | 51 | 6118 |
| Polish companies bankruptcy (PCB) | 64 | 10503 |
| Superconductivty | 81 | 21263 |
| Corel Images | 89 | 68040 |
| Gas Sensor Array Drift (Gas Sensor) | 128 | 13910 |

TABLE II: Dataset information.

| Data set | MoG | KDE | RNADE | MAF | LRCF-DE |
|---|---|---|---|---|---|
| Red wine | 1.28 | 1.13 | 0.66 | 0.63 | 0.56 |
| White wine | 1.79 | 1.31 | 0.80 | 0.75 | 0.59 |
| F-O.TP | 1.86 | 1.46 | 0.63 | 0.52 | 0.48 |
| PCB | 5.6 | 7.73 | 4.43 | 4.52 | 3.85 |
| Superconductivty | 18.56 | 19.96 | 16.46 | 16.38 | 16.53 |
| Corel Images | 0.53 | 0.93 | 0.27 | 0.27 | 0.28 |
| Gas Sensor | 29.7 | 35.3 | 26.8 | 26.2 | 26.7 |

TABLE III: MAE for regression tasks.

in light of the fact that our model is "agnostic": designed for general-purpose density estimation, not specifically for realistic-looking image synthesis. It is possible to incorporate image modeling domain knowledge in the design of LRCF-DE (such as correlation between adjacent pixel values), but this is beyond the scope of this paper. This manuscript is the first part containing the foundations of a two-part paper. The

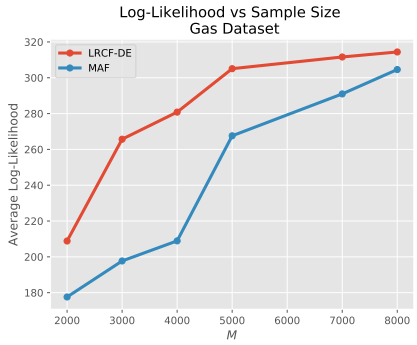


Fig. 6: LRCF-DE is particularly beneficial in the small to moderate training sample regime. Using a limited number of training points from the Gas dataset, highlights the superior performance of LRCF-DE against the state of the art deep learning based PDF estimator (MAF).

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Samples | 1553 | 1269 | 929 | 824 | 852 | 716 | 834 | 792 | 708 | 821 | 9298 |

TABLE V: Images of handwritten digits - USPS dataset information.

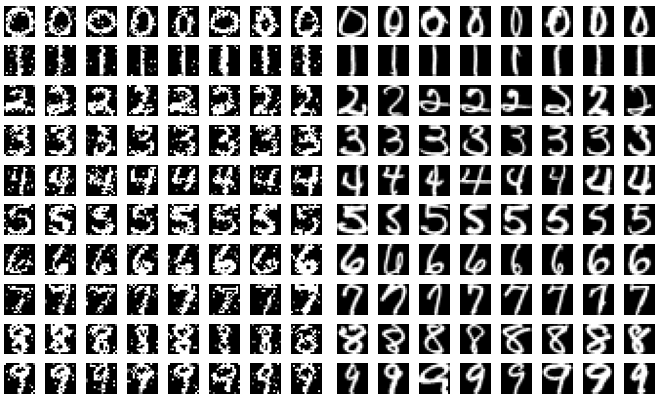

Fig. 7: The first eight columns correspond to class-conditional synthetic samples (generated by LRCF-DE) and the rest correspond to real samples from the USPS dataset.

second part [47] builds on this foundation to develop a joint compression (nonlinear dimensionality reduction) and compressed density estimation framework that offers additional flexibility and scalability and is used to demonstrate improved image sampling performance against well known deep learning models, including autoregressive methods and VAEs.

## VI. History, priority and one final comparison

The first preprint of this paper appeared on Aug. 27, 2020, see V1 of [48], building upon our earlier work in [21] and [49] which dealt with the case of discrete/categorical random variables and multivariate histograms.

While this paper was undergoing multiple rounds of review and revision, we noticed via Google Scholar recommendations that several other papers and preprints were popping up, claiming similar approaches and results. In particular, we saw [50] in December 2021, and realized that the authors were citing Parts I and II of our work as closely related in their response to a reviewer in openreview https://openreview.net/forum?id=uholDBWSVP, without acknowledging and citing our work in the paper itself. Eventually, they uploaded a new version that cites us in a very confusing way. In fact [50]

(see also [51]) uses multivariate histograms to approximate smooth multivariate densities, and derives the convergence rate of a low-rank histogram estimator that exists but is practically intractable. Their result does not tell us what is the performance of any practical algorithm based on tensor decomposition, and we do not know whether this bound is practically attainable.

In July 2021 we discovered [52], which was very similar to Part I of this work – the main difference being the use of a tensor train decomposition instead of CPD. The authors of [52] told us that they were unaware of our work, and offered to cite it in a revised arXiv version; the conference version [52] was already out and they could not update it. The revised arXiv version of [52] citing (but not comparing to) our work was uploaded on Feb. 25, 2022.

During the last round of review of this manuscript, one reviewer acknowledged that we have priority over [50] and [52], but felt that we should still cite these papers and compare with [52] (since [50] did not include any well-developed algorithm).

We decided that it is worth recounting this experience and comparing with [52]. We downloaded the code provided by the authors of [52], and applied it with their own parameter settings on the same datasets that they tried. We included in the comparison both the method in this (Part I of our two-part) paper (LRCF-DE), and one of the methods (HTF-DE without compression) in Part II of this paper, which is now published [47]. As we expected, both our methods outperform [52]. We present our results in Tables VI and VII.

Beyond numerical results, however, there are other important reasons to choose our approach, which is based on CPD, over the tensor train-based approach in [52]. These are as follows.

• **Ease of marginalization:** For CPD, marginalization is trivial – i.e., it incurs zero complexity. This is because if $[\![\boldsymbol{\lambda}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\!]$ is the CPD model of density $f_{X_1,X_2,X_3}$, the lower order marginal $f_{X_1,X_2}$ is simply $[\![\boldsymbol{\lambda}, \mathbf{A}_1, \mathbf{A}_2]\!]$, i.e., we simply drop one factor matrix; see also [21]. That is, there is no cost in obtaining a closed-form expression for the lower-order marginal. Evaluating our density model for $N$ random variables using $K$ frequencies per variable and tensor rank $F$ entails complexity $NKF$ per evaluation point. Thus, when we evaluate lower-order marginals involving $n < N$ variables, the cost per evaluation is only $nKF$. For tensor trains, marginalization according to [52] is carried out by taking the inner product of two $N$-way tensors in tensor train format[1], using Algorithm 1 in [52] which has complexity $O(NKr_{tt}^2)$, where $K$ is the number of basis functions (Fourier coefficients in our context) per variable, and $r_{tt}$ is tensor train rank. Note that this complexity estimate is *per evaluation* of the resulting marginal at a particular point, and it involves $N$ instead of $n$.

• **Ease of sampling:** For our CPD-based model, sampling is very simple: one draws a sample from the known PMF of the latent variable, and then independently draws from each 1-D conditional PDF of the sampled variables via the transformation method. The latter process can be fully parallelized. Tensor trains are inherently sequential models, and as stated in [52], *"the tensor-train format allows fast, exact sampling in the autoregressive fashion."* A well known downside of (sequential) auto-regressive models, however, is their slow sampling time. That is because the sample coordinates are generated one by one, which slows down the process for high-dimensional datasets. The tensor-train sampling process (Algorithm 2 in [52]) has complexity $O(NKr_{tt}^2 + NKL)$, where $L$ is the number of bisection steps per iteration. Again, this is significantly more complicated than sampling from our CPD model.

• **Identifiability:** Last but not least, CPD modeling affords identifiability guarantees which are not available for the tensor train approach in [52].

[1]It is unclear what happens if the integral of one of the basis functions is zero.

## VII. Conclusions

In this work, we have revisited the classic problem of non-parametric density estimation from a fresh perspective – through the lens of complex Fourier series approximation and tensor modeling, leading to a low-rank characteristic function approach. We showed that any compactly supported density can be well-approximated by a finite *characteristic tensor* of leading complex Fourier coefficients as long as the coefficients decay sufficiently fast. We posed density estimation as a constrained (coupled) tensor factorization problem and proposed a Block Coordinate Descent algorithm, which under certain conditions enables learning the true data-generating distribution. Results on real data have demonstrated the utility and promise of this novel approach compared to both standard and recent density estimation techniques.

## VIII. Appendix

Due to the subtleties of tensor rank, the possible non-existence of best low-rank tensor approximation, and other technical issues, no perturbation theory currently exists for tensor decomposition to estimate how close low-rank approximation of a perturbed low-rank tensor is to the unperturbed low-rank tensor. In what follows, we summarize what is known for our method without imposing low-rank structure, and further explain why partially imposing low-rank structure is beneficial, using matrix results.

### A. Bias, variance, consistency of the empirical characteristic function

In this appendix we summarize important properties of empirical characteristic functions as statistical estimators of the corresponding characteristic functions. We refer the reader to [53] for proofs and additional results.

By linearity of expectation, it is easy to see that the empirical characteristic function is an unbiased estimator of the corresponding characteristic function, i.e.,

$$E\left[\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu})\right] = \Phi_{\boldsymbol{X}}(\boldsymbol{\nu}),$$

for all $\boldsymbol{\nu}$ and $M \geq 1$. For the remainder of this section, we assume that $\{\mathbf{x}_m\}_{m=1}^M$ is i.i.d. in $m$. The variance of the

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2022.3175608



| | POWER | GAS | HEPMASS | MINIBOONE | BSDS300 |
|---|---|---|---|---|---|
| Squared TTDE [52] | 0.46 | **8.93** | −21.34 | −28.77 | 143.30 |
| LRCF-DE | **0.47** | 8.44 | −17.53 | −17.05 | **152.30** |
| HTF-DE [47] | 0.44 | 8.51 | **-15.72** | **-15.43** | 151.31 |

TABLE VI: Average test-set log-likelihood per datapoint for different tensor-based models on UCI datasets; **higher is better**.

| | HEPMASS | MINIBOONE | BSDS300 |
|---|---|---|---|
| Squared TTDE [52] | 0.00313 | 0.1793 | 0.621 |
| LRCF-DE | **0.00292** | **0.1591** | **0.533** |
| HTF-DE [47] | 0.00301 | 0.1714 | 0.575 |

TABLE VII: Prediction performance (MSE); **lower is better**. We predict the first variable for each dataset.

| | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Samples | 1553 | 1269 | 929 | 824 | 852 | 716 | 834 | 792 | 708 | 821 | 9298 |

TABLE VIII: Images of handwritten digits - USPS dataset information.

empirical characteristic function estimate can be shown [53] to be

$$\begin{aligned}\mathrm{Var}[\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu})] &= E\left[\left|\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) - E\left[\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu})\right]\right|^2\right] \\ &= E\left[\left|\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) - \Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\right|^2\right] \\ &= \frac{1}{M}\left(1 - |\Phi_{\boldsymbol{X}}(\boldsymbol{\nu})|^2\right).\end{aligned}$$

Note that $0 \leq \left|\Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\right| \leq 1$, and therefore $\mathrm{Var}[\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu})] \leq \frac{1}{M}$. It follows that

$$\lim_{M\to\infty} E\left[\left|\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) - \Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\right|^2\right] = 0,$$

i.e., for any fixed $\boldsymbol{\nu}$, $\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu})$ converges to $\Phi_{\boldsymbol{X}}(\boldsymbol{\nu})$ in the mean-squared sense. By the strong law of large numbers, it also converges almost surely for any fixed $\boldsymbol{\nu}$. Furthermore, for any fixed positive $T < \infty$

$$\lim_{M\to\infty} \sup_{\|\boldsymbol{\nu}\|_2 \leq T} \left|\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) - \Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\right| = 0,$$

almost surely. It can also be shown [53] that for any increasing sequence $T_M$ such that $\lim_{M\to\infty} \frac{\log(T_M)}{M} = 0$, it holds

$$\lim_{M\to\infty} \sup_{\|\boldsymbol{\nu}\|_2 \leq T_M} \left|\widehat{\Phi}_{\boldsymbol{X}}(\boldsymbol{\nu}) - \Phi_{\boldsymbol{X}}(\boldsymbol{\nu})\right| = 0,$$

almost surely. In our context, we only use a sampled and truncated version of the characteristic function (corresponding to a truncated multivariate Fourier series), hence $T$ is always finite – we do not need the latter result.

It is also worth noting that the covariance of different samples of the empirical characteristic function (corresponding to different values of $\boldsymbol{\nu}$) goes to zero $\sim \frac{1}{M}$, and so does the covariance of its real and imaginary parts. As a result, for large $M$, the errors in the different elements of the characteristic tensor are approximately uncorrelated, with uncorrelated real and imaginary parts. This suggests that when we fit a model to the empirical characteristic function, it makes sense to use a least squares approach. Another motivation for this comes from Parseval's theorem: minimizing integrated squared error in the Fourier domain corresponds to minimizing integrated squared error between the corresponding multivariate distributions. This is true in particular when we limit the support of the distribution to a hypercube and use the samples of the characteristic function that correspond to the multivariate Fourier series, thereby replacing the multivariate integral in the Fourier domain by a multivariate sum.

### *B. Low-rank denoising: reduction of the mean squared error*

Our use of a low-rank model in the characteristic tensor domain is primarily motivated by the need to avoid the "curse of dimensionality": using a rank-$F$ model with $2K+1$ Fourier coefficients per mode parametrizes the whole $N$-dimensional multivariate density using just $FN(2K+1)$ coefficients, and avoids instantiating and storing a tensor of size $(2K+1)^N$, which is close to impossible even for moderate $N$. However, there is also a variance benefit that comes from this low-rank parametrization. We know from [54] that for a square $L \times L$ matrix of rank $F$ observed in zero-mean white noise of variance $\sigma^2$, low-rank denoising attains mean squared error $cLF\sigma^2$ asymptotically in $L$, where $c$ is a small constant. In practice the asymptotics kick in even for relatively small $L$ [54]. Contrast this to the raw $L^2\sigma^2$ if one does not use the low-rank property.

For an $N$-way tensor of rank $F$, assume $N$ is even, $K_n = K$, $\forall N$ (for simplicity of exposition), and "unfold" the characteristic tensor into a $K^{N/2} \times K^{N/2}$ matrix. In practice we use $F$ *far* less than $K^{N/2}$, and thus the resulting matrix will be *very* low rank. Invoking [54], low-rank tensor modeling will yield a reduction in mean squared error by a factor of at least $\frac{F}{K^{N/2}}$. We say at least, because this low-rank matrix structure is implied but does not imply low-rank tensor structure, which is much stronger. Note also that mean squared error in the characteristic tensor domain translates to mean squared error between the corresponding distributions, by virtue of Parseval's theorem.

### C. Derivation of (21)

In this appendix we present the derivation of (21), which is used to solve regression tasks.

$$
\begin{aligned}
&E\left[X_N|X_1,\ldots,X_{N-1}\right]\\
&=\int_0^1 x_N f_{X_N|X_1,\ldots,X_{N-1}}(x_N|x_1,\ldots,x_{N-1})dx_N\\
&=\int_0^1 x_N \frac{f_{X_1,\ldots,X_N}(x_1,\ldots,x_N)}{f_{X_1,\ldots,X_{N-1}}(x_1,\ldots,x_{N-1})}dx_N\\
&=\frac{1}{c_1}\int_0^1 x_N f_{X_1,\ldots,X_N}(x_1,\ldots,x_N)dx_N\\
&=\frac{1}{c_1}\int_0^1 x_N \sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N}\sum_{k_n=-K}^{K}\mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n}dx_N\\
&=\frac{1}{c_1}\sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N-1}\sum_{k_n=-K}^{K}\mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n}\\
&\qquad\qquad\sum_{k_N=-K}^{K}\mathbf{A}_N(k_N,h)\int_0^1 x_N e^{-j2\pi k_N x_N}dx_N\\
&=\frac{1}{c_1}\sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N-1}\sum_{k_n=-K}^{K}\mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n}\\
&\qquad\qquad\sum_{k_N=-K}^{K}c_{2,k_N}\mathbf{A}_N(k_N,h),
\end{aligned}
$$

where

$$
\begin{aligned}
c_1 &= f_{X_1,\ldots,X_{N-1}}(x_1,\ldots,x_{N-1})\\
&=\sum_{h=1}^{F}\boldsymbol{\lambda}(h)\prod_{n=1}^{N-1}\sum_{k_n=-K}^{K}\mathbf{A}_n(k_n,h)e^{-j2\pi k_n x_n},
\end{aligned}
$$

and

$$
c_{2,k_N}=\frac{e^{-j2\pi k_N}}{-j2\pi k_N}+\frac{1-e^{-j2\pi k_N}}{[-j2\pi k_N]^2}.
$$

## REFERENCES


[1] T. Schmah, G. E. Hinton, S. L. Small, S. Strother, and R. S. Zemel, "Generative versus discriminative training of RBMs for classification of fMRI images," in *Advances in Neural Information Processing Systems*, 2009, pp. 1409–1416.

[2] E. L. Ray, K. Sakrejda, S. A. Lauer, M. A. Johansson, and N. G. Reich, "Infectious disease prediction with kernel conditional density estimation," *Statistics in Medicine*, vol. 36, no. 30, pp. 4908–4929, 2017.

[3] M. Amiridi, N. Kargas, and N. Sidiropoulos, "Information-theoretic feature selection via tensor decomposition and submodularity," *IEEE Transactions on Signal Processing*, vol. PP, pp. 1–1, 11 2021.

[4] M. Amiridi, N. Kargas, and N. D. Sidiropoulos, "Statistical learning using hierarchical modeling of probability tensors," in *2019 IEEE Data Science Workshop (DSW)*. IEEE, 2019, pp. 290–294.

[5] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *International Conference on Knowledge Discovery and Data Mining*, vol. 96, 1996, pp. 226–231.

[6] H. Zen and A. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 3844–3848.

[7] D. Titterington and J. Sedransk, "Imputation of missing values using density estimation," *Statistics & Probability Letters*, vol. 8, no. 5, pp. 411–418, 1989.

[8] J. Ballé, V. Laparra, and E. P. Simoncelli, "Density modeling of images using a generalized normalization transformation," in *International Conference on Learning Representations*, 2016.

[9] B. W. Silverman, *Density estimation for statistics and data analysis*. Routledge, 2018.

[10] K. Pearson, "Contributions to the mathematical theory of evolution," *Philosophical Transactions of the Royal Society of London. A*, vol. 185, pp. 71–110, 1894.

[11] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Ann. Math. Statist.*, vol. 27, no. 3, pp. 832–837, 09 1956.

[12] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[13] M. Germain, K. Gregor, I. Murray, and H. Larochelle, "Made: Masked autoencoder for distribution estimation," in *International Conference on Machine Learning (ICML)*, 2015, pp. 881–889.

[14] B. Uria, M.-A. Côté, K. Gregor, I. Murray, and H. Larochelle, "Neural autoregressive distribution estimation," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 7184–7220, 2016.

[15] G. Papamakarios, T. Pavlakou, and I. Murray, "Masked autoregressive flow for density estimation," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 2338–2347.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 2672–2680.

[17] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations*, 2014.

[18] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, vol. 6, no. 1-4, pp. 164–189, 1927.

[19] R. A. Harshman *et al.*, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," *UCLA Working Papers Phonetics*, vol. 16, pp. 1–84, 1970.

[20] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, 2017.

[21] N. Kargas, N. D. Sidiropoulos, and X. Fu, "Tensors, learning, and "Kolmogorov extension" for finite-alphabet random vectors," *IEEE Transactions on Signal Processing*, vol. 66, no. 18, pp. 4854–4868, Sep. 2018.

[22] G. J. McLachlan and K. E. Basford, *Mixture models: Inference and applications to clustering*. M. Dekker New York, 1988, vol. 38.

[23] J. Chen, "Optimal rate of convergence for finite mixture models," *The Annals of Statistics*, pp. 221–233, 1995.

[24] D. W. Scott, "Feasibility of multivariate density estimates," *Biometrika*, vol. 78, no. 1, pp. 197–205, 1991.

[25] M. Girolami, "Orthogonal series density estimation and the kernel eigenvalue problem," *Neural computation*, vol. 14, no. 3, pp. 669–688, 2002.

[26] S. Efromovich, "Orthogonal series density estimation," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 467–476, 2010.

[27] A. B. Tsybakov, *Introduction to nonparametric estimation*. Springer Science & Business Media, 2008.

[28] E. S. Allman, C. Matias, and J. A. Rhodes, "Identifiability of parameters in latent structure models with many observed variables," *The Annals of Statistics*, vol. 37, no. 6A, pp. 3099–3132, 2009.

[29] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of N-way arrays," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 14, no. 3, pp. 229–239, 2000.

[30] L. Song, A. Anandkumar, B. Dai, and B. Xie, "Nonparametric estimation of multi-view latent variable models," in *International Conference on Machine Learning*. PMLR, 2014, pp. 640–648.

[31] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky, "Tensor decompositions for learning latent variable models," *Journal of machine learning research*, vol. 15, pp. 2773–2832, 2014.

[32] T. G. Kolda, "Symmetric orthogonal tensor decomposition is trivial," *arXiv preprint arXiv:1503.01375*, 2015.

[33] N. Kargas and N. D. Sidiropoulos, "Learning mixtures of smooth product distributions: Identifiability and algorithm," in *22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019, pp. 388–396.

[34] S. Dolgov, K. Anaya-Izquierdo, C. Fox, and R. Scheichl, "Approximation and sampling of multivariate probability distributions in the tensor train decomposition," *Statistics and Computing*, vol. 30, no. 3, pp. 603–625, 2020.

[35] B. Uria, I. Murray, and H. Larochelle, "Rnade: The real-valued neural autoregressive density-estimator," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 2175–2183.

[36] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, vol. 37, 07–09 Jul 2015, pp. 1530–1538.

[37] H. Poon and P. Domingos, "Sum-product networks: A new deep architecture," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 689–690.

[38] I. París, R. Sánchez-Cauce, and F. J. Díez, "Sum-product networks: A survey," *arXiv preprint arXiv:2004.01167*, 2020.

[39] L. Chiantini and G. Ottaviani, "On generic identifiability of 3-tensors of small rank," *SIAM Journal on Matrix Analysis and Applications*, vol. 33, no. 3, pp. 1018–1037, 2012.

[40] G. Plonka, D. Potts, G. Steidl, and M. Tasche, *Numerical Fourier Analysis*. Springer, 2018.

[41] J. C. Mason, "Near-best multivariate approximation by fourier series, chebyshev series and chebyshev interpolation," *Journal of Approximation Theory*, vol. 28, no. 4, pp. 349–358, 1980.

[42] D. C. Handscomb, *Methods of numerical approximation: lectures delivered at a Summer School held at Oxford University, September 1965*. Elsevier, 2014.

[43] W. Wang and M. A. Carreira-Perpinán, "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application," *arXiv preprint arXiv:1309.1541*, 2013.

[44] M. Lichman *et al.*, "UCI machine learning repository," 2013.

[45] L. Theis, A. v. d. Oord, and M. Bethge, "A note on the evaluation of generative models," in *4th International Conference on Learning Representations (ICLR)*, 2016.

[46] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems*, 1990, pp. 396–404.

[47] M. Amiridi, N. Kargas, and N. D. Sidiropoulos, "Low-rank characteristic tensor density estimation Part II: Compression and latent density estimation," *IEEE Transactions on Signal Processing*, 2022, DOI: 10.1109/TSP.2022.3158422.

[48] M. Amiridi and N. D. S. Nikos Kargas, "Nonparametric multivariate density estimation: A low-rank characteristic function approach," *arXiv*, 2020, arXiv:2008.12315.

[49] N. Kargas and N. D. Sidiropoulos, "Completing a joint pmf from projections: A low-rank coupled tensor factorization approach," in *2017 Information Theory and Applications Workshop (ITA)*. IEEE, 2017, pp. 1–6.

[50] R. A. Vandermeulen and A. Ledent, "Beyond smoothness: Incorporating low-rank analysis into nonparametric density estimation," *Advances in Neural Information Processing Systems*, vol. 34, 2021, arXiv:2204.00930.

[51] R. A. Vandermeulen, "Improving nonparametric density estimation with tensor decompositions," *arXiv preprint arXiv:2010.02425*, 2020.

[52] G. S. Novikov, M. E. Panov, and I. V. Oseledets, "Tensor-train density estimation," in *Uncertainty in Artificial Intelligence*. PMLR, 2021, pp. 1321–1331, arXiv:2108.00089.

[53] N. G. Ushakov, *Selected topics in characteristic functions*. Walter de Gruyter, 2011.

[54] M. Gavish and D. L. Donoho, "Optimal shrinkage of singular values," *IEEE Transactions on Information Theory*, vol. 63, no. 4, pp. 2137–2152, 2017.


**Magda Amiridi** received the Diploma in Electrical and Computer Engineering from the Technical University of Crete (TUC), Chania, Greece, in 2018. Currently, she is working towards the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Virginia, where she is also affiliated with the Signal and Tensor Analytics Research (STAR) group under the supervision of Professor N. D. Sidiropoulos. She is working on theoretical foundations and algorithmic approaches for non-parametric probabilistic modeling using low-rank tensors. She is a student member of the IEEE and student chapter of IEEE's Signal Processing Society at UVA.

**Nikos Kargas** received the Diploma and M.Sc. degree in Electronic and Computer engineering from the Technical University of Crete (TUC), Chania, Greece, in 2013 and 2015, respectively, and Ph.D. degree in Electrical Engineering from University of Minnesota, Minneapolis, in 2020. His interests are in the areas of Machine learning, Statistics and Optimization. A major focus of his work is on tensor methods for high-dimensional distribution and general nonlinear function learning.

**Nicholas D. Sidiropoulos** (Fellow, IEEE) received the Diploma in Electrical Engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, and the M.S. and Ph.D. degrees in Electrical Engineering from the University of Maryland at College Park, College Park, MD, USA, in 1988, 1990, and 1992, respectively. He has served on the faculty of the University of Virginia (UVA), the University of Minnesota (UMN), and the Technical University of Crete (TUC), Greece, prior to his current appointment as Louis T. Rader Professor at UVA. From 2015 to 2017, he was an ADC Chair Professor at UMN. His research interests are in signal processing, communications, optimization, tensor decomposition, and factor analysis, with applications in machine learning and communications. He received the NSF/CAREER award in 1998, the IEEE Signal Processing Society (SPS) Best Paper Award in 2001, 2007, and 2011, served as IEEE SPS Distinguished Lecturer (2008–2009), and as Vice President - Membership of IEEE SPS (2017–2019). He received the 2010 IEEE Signal Processing Society Meritorious Service Award, and the 2013 Distinguished Alumni Award from the University of Maryland, Department of ECE. He is a Fellow of EURASIP (2014).