RECOVERY OF JOINT PROBABILITY DISTRIBUTION FROM ONE-WAY MARGINALS: LOW RANK TENSORS AND RANDOM PROJECTIONS

Jian Vora[1], Karthik S. Gurumoorthy[2], Ajit Rajwade[3]

Dept. of EE., IIT Bombay, India[1]; India Machine Learning, Amazon, India[2];
Dept. of CSE., IIT Bombay, India[3]

ABSTRACT

Joint probability density or joint probability mass function (PDF/PMF) estimation is a fundamental machine learning problem. The number of free parameters scales exponentially with respect to the number of random variables. Hence, most work on nonparametric joint distribution estimation is based on some structural assumptions such as clique factorization adopted by probabilistic graphical models, imposition of low rank on the joint probability tensor and reconstruction from 3-way or 2-way marginals, etc. In the current work, we link random projections of data to the problem of PMF estimation using techniques from tomography. Using it alongside low-rank tensor decomposition, we present an approach to estimate joint distribution from just one-way marginals in a transformed space. We provide a novel algorithm for recovering factors of the tensor from one-way marginals, test it across synthetic and real-world datasets, and also perform MAP inference on the estimated model for classification.

Index Terms— density estimation, low rank tensors, random projections, Radon transform, statistical learning

1. INTRODUCTION

Learning the joint distribution of N random variables (RVs) is a fundamental problem in machine learning, signal processing and statistics [1], [2]. More specifically, given N RVs $\{X_1, X_2, .., X_N\}$, the goal is to estimate their joint probability mass function (PMF) from their samples with potentially missing entries. The joint PMF is used in many applications including, (1) as a generative model to produce new datapoints by sampling, (2) for conditional inferences as having the joint implies we can find any conditional PMFs. For discrete variables, the most common solution is to use empirical frequency of a tuple as a proxy for the joint distribution [3]. However, this simple method suffers from poor sample complexity $\gg \Omega(I^N)$ where I is the number of distinct values that each variable X_i can acquire. One popular route is to make use of problem-specific structural assumptions such as

in graphical models [4] to help reduce the model complexity. However, besides the graphical structure being unknown, inference in these models is difficult due to intractable posterior integrals, requiring randomization (e.g. MCMC [5]) or approximation (e.g. variational Bayes methods [4]). Recently, imposing a low rank assumption on the underlying probability tensor has gained popularity [6] where the N-way low-rank joint PMFs are estimated from only 3-way marginals. This is further extended to using only 2-way marginals in [7].

Our contributions: In this work, we present a novel algorithm that combines the strengths of low-rank tensor approximation and techniques from computed tomography to obtain N-way joint PMFs from *just one-way marginals*.

2. BACKGROUND

2.1. Canonical Polyadic Decomposition of PMF Tensors

An $N-$way tensor $\mathcal{Z} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times ... \times I_N}$ representing the joint PMF of N discrete RVs where $\mathcal{Z}(x_1, x_2, \ldots, x_N) = p(X_1 = x_1, X_2 = x_2, \ldots, X_N = x_N)$, admits a 'Canonical Polyadic Decomposition' (CPD) if it can be decomposed as a sum of F rank-1 tensors [8]. Denoting $a \otimes b$ as the outer-product of two vectors, the CPD model is:

$$\mathcal{Z} = \sum_{f=1}^{f=F} \boldsymbol{\lambda}(f) \boldsymbol{A_1}(:, f) \otimes \boldsymbol{A_2}(:, f) \otimes ... \otimes \boldsymbol{A_N}(:, f). \quad (1)$$

Here F is the smallest number for which such a decomposition exists and is called the rank of the tensor, $\boldsymbol{\lambda} \in \mathbb{R}_+^F$ and the mode latent factors $\boldsymbol{A_i} \in \mathbb{R}_+^{I_i \times F}$. The entries of $\boldsymbol{\lambda}$ and each column of the mode latent factor are non-negative and sum to one. Recovering the PMF tensor $\mathcal{Z}$ is equivalent of estimating the factors $\boldsymbol{\lambda}, \boldsymbol{A_1}, \boldsymbol{A_2}, ..., \boldsymbol{A_N}$ [9].

2.2. Joint PMF Estimation from 3-way or 2-way Marginals

The work in [6] showed the benefits of modelling joint PMFs as low-rank tensors via the CPD decomposition. The low tensor rank is motivated by the fact that in many real world datasets, the RVs are not fully dependent (nor fully independent). They proposed recovering N-way joint PMFs from 3-way marginals of the form $\boldsymbol{Z}_{i,j,k} \triangleq p(X_i, X_j, X_k)$ which are obtained from the data using standard histogramming. From

these marginals, the latent mode factors are recovered from the following coupled tensor factorization (CTF) estimator:

$$\min_{\{A_n\}_{n=1}^N, \lambda} \sum_j \sum_{k>j} \sum_{l>k} \|Z_{j,k,l} - [\lambda, A_j, A_k, A_l]\|_F^2,$$

$$\text{s.t. } \forall n, f, \|A_n(:,f)\|_1 = \|\lambda\|_1 = 1, A_n \succeq 0; \lambda \succeq 0,$$

where $\succeq$ represents the element-wise and $[\lambda, A_j, A_k, A_l] \triangleq \sum_{f=1}^F \lambda(f) A_j(:,f) \otimes A_k(:,f) \otimes A_l(:,f)$. The problem with this approach is two fold: (a) for large N, the number of 3-way marginals increases rapidly as $O(N^3)$, and (b) accurately estimating the 3-way marginals is difficult when the number of samples is small or if the samples have missing entries.

The work in [7] recovers the joint PMF from only 2-way marginals $Z_{j,k} = p(X_j, X_k)$ computed from the samples via histogramming. The mode latent factors are obtained using the relation $Z_{j,k} = A_j D(\lambda) A_k^T$ where $D(\lambda)$ is a diagonal matrix with λ on its diagonal. As the tensor rank $F \gg \min(I_j, I_k)$ [10], methods like non-negative matrix factorization (NMF) [11] cannot be directly applied to $Z_{j,k}$ to determine A_j and A_k. Instead, the work in [7] splits the indices of N variables into sets $S_1 = \{l_1, l_2, ..., l_M\}$ and $S_2 = \{l_{M+1}, l_{M+2}, ..., l_N\}$ and constructs the matrix $\widetilde{Z}$ by row and column-concatenation of $\{Z_{j,k}\}$ (see [7, Eqn. 3]) from indices in S_1, S_2. Then, $\widetilde{Z}$ is then decomposed as $\widetilde{Z} = W H^T$ using the successive projection algorithm (SPA) based on NMF literature [12]. The mode factors are extracted from W and H using the relation $W = [A_{l_1}, A_{l_2}, ..., A_{l_M}]^T$ and $H^T = D(\lambda)[A_{l_{M+1}}, A_{l_{M+2}}, .., A_{l_N}]$. A few expectation maximization (EM) iterations are subsequently executed using the output of the SPA as the initial condition [13], to further boost the performance.

2.3. Random Projections and PMF Estimation

In this subsection (and further in Sec. 3) we show the link between random projections, tomography, and tensor recovery. **Radon Transform:** The Radon Transform of a N-D function $p(x) : \mathbb{R}^N \to \mathbb{R}$, including the case where $p(.)$ is the joint PDF of x, in a direction ϕ ($\|\phi\|_2 = 1$) is defined as follows:

$$\Re_\phi(p)(t) = \int p(x)\delta(t - \phi^t x)dx, \quad (2)$$

where δ is the Dirac delta function and $t \in \mathbb{R}$ is an offset. The Radon Transform can be inverted to reconstruct $p(x)$ using methods like filtered back projection (FBP) [14], which are computationally efficient in 2D or 3D.

We are interested in estimating $p(x)$ from random linear projections of the form $y = \Phi x$ where $\Phi \in \mathbb{R}^{M \times N}$ with entries drawn i.i.d. from $\mathcal{N}(0, 1)$ and then normalized so that each row has unit magnitude. We demonstrate that such data transformation aids in PDF/PMF estimation. For this, consider a vector $x \in \mathbb{R}^N$ and a row ϕ of Φ. The 1D PMF of the projection $\phi^t x$ is:

$$p(\phi^t x = t) = \sum_x p(x)\delta_{kron}\left(\phi^t x - t\right). \quad (3)$$

Comparing Eqns. 2 and 3, we note that the PMF of $\phi^t x$ is in fact the (discretized) Radon Transform of the joint PMF tensor taken in the direction ϕ. Thus, if we collect 1D PMFs of the form $p(\phi_m^t x)$ for various direction vectors $\{\phi_m\}$, then we can use a Radon inversion method to reconstruct the joint PMF tensor. This was first shown in [15] for 2D PDFs/PMFs and later in [16], [17]. For PDFs with exponentially decreasing Fourier transforms, the Radon-based technique has superior convergence rates compared to approaches like kernel density estimation (KDE) [18], [19], [20].

3. PROBLEM STATEMENT AND ALGORITHM

Consider N discrete RVs $\{X_i\}_{i=1}^{i=N}$ each existing in I different states[1]. Our aim is to estimate the joint PMF $p(X_1, X_2, .., X_N)$ where $p(.)$ is a low-rank tensor which follows the CPD from Eqn. 1. However, recovering the tensor p from its Radon projections taken in general directions $\{\phi_m \in \mathbb{R}^N\}$ is an immensely costly operation. Hence we consider only sparse direction vectors $\{\phi_m\}$ with just two non-zero entries (i.e. we consider linear combinations of just two RVs at a time), which facilitates speedy FBP implementation. Define the set $\mathcal{B} \triangleq \{(j,k) : 1 \le j < k \le N\}$. For M random vectors $\phi_m \in \mathbb{R}^2, 1 \le m \le M$, and for $X_{j,k} \triangleq [X_j, X_k]^t$ for $(j,k) \in \mathcal{B}$, we obtain $M|\mathcal{B}|$ random projections of the form $\phi_m^t X_{j,k} \in \mathbb{R}$. We wish to recover the PMF tensor from the 1D PMFs of the form $p(\phi_m^t X_{j,k})$. Define $Z_{j,k}$ as the joint probability $p(X_j, X_k)$ for the two RVs X_j and X_k. On fixing j and k and stacking M one-dimensional PMFs $p(\phi_m^t X_{j,k})$ row-wise, we obtain a matrix $Y_{j,k} \in \mathbb{R}^{M \times I}$. In case of infinite samples, from Eqn. 3:

$$Y_{j,k} = \Re(Z_{j,k}) = \Re(A_j D(\lambda) A_k^T), \quad (4)$$

where $D(.)$ is the diagonal operator as defined in Sec. 2.2, and $\Re$ stands for the Radon transform in multiple directions $\{\phi_m\}_{m=1}^M$. However the 1D PMFs, and hence each $Y_{j,k}$ can only be *estimated*. For each element in set $\mathcal{B}$, we perform random projections and estimate *one dimensional* PMFs empirically from the data using histogramming, and thus assemble $Y_{j,k}$. The goal is now to determine the underlying mode factors $\lambda, A_1, A_2, ..., A_N$ (and thus the joint PMF of $\{X_i\}_{i=1}^N$), which represent these stacked 1D PMF estimates $\{Y_{j,k}\}_{(j,k) \in \mathcal{B}}$ as faithfully as possible. To this end, we formulate the following objective function:

$$J(\{A_n\}_{n=1}^N, \lambda) = \sum_{j,k>j} \|Y_{j,k} - \Re(A_j D(\lambda) A_k^T)\|_F^2. \quad (5)$$

We cannot directly optimise $J(.)$, say via gradient descent updates, as the mode factors will not be identifiable when

---

[1]This work can be easily to the case where different RVs had different number of states. Our work can also handle continuous RVs - see Sec. 4.

$F > I$ as argued in [7]. To circumvent, we introduce an auxiliary variable $\boldsymbol{Z}_{j,k}$ with the constraint $\boldsymbol{Z}_{j,k} = \boldsymbol{A_j}D(\boldsymbol{\lambda})\boldsymbol{A_k}^T$ and transform it into an unconstrained problem by adding a penalty term, namely

$$J_1(\{\boldsymbol{A_n}\}_{n=1}^N, \boldsymbol{\lambda}, \{\boldsymbol{Z}_{j,k}\}) \triangleq \sum_{j,k>j} \|\boldsymbol{Y}_{j,k} - \mathfrak{R}(\boldsymbol{Z}_{j,k})\|_F^2 +$$
$$\rho\|\boldsymbol{Z}_{j,k} - \boldsymbol{A_j}D(\boldsymbol{\lambda})\boldsymbol{A_k}^T\|_F^2. \quad (6)$$

We perform the following three steps to obtain identifiable mode factors: (i) setting the hyper-parameter $\rho = 0$, compute $\boldsymbol{Z}_{j,k}$ from the 1D densities $\boldsymbol{Y}_{j,k}$ using least-squares, (ii) construct $\widetilde{\boldsymbol{Z}}$ from $\boldsymbol{Z}_{j,k}$ as explained in Sec. 2.2, and (iii) determine the matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ as outputs of the function SPA(.) which are the factors of $\widetilde{\boldsymbol{Z}}$. The mode factors $\{\boldsymbol{A_j}\}$ which are sub-matrices of $\boldsymbol{W}$ and $\boldsymbol{H}$ will now be identifiable as shown in [7]. We further refine our estimates by choosing ρ via cross-validation on the likelihood of having observed the validation data, and updating the mode factors and $\boldsymbol{Z_{j,k}}$ till convergence as described in Alg. 1. The resultant mode factors obtained from optimizing $J_1(.)$ can then be used as a good initial condition for a projected gradient descent on the original objective function $J(.)$ with adaptive step-size. In Alg. 1, the operators $\mathfrak{R}$ and $\mathfrak{R}^T$ are implemented as function handles using the `radon` and `iradon` functions of MATLAB respectively. *It is important to note that a procedure which stops at the end of step 4 in Alg. 1 (equivalent to FBP to obtaining $\boldsymbol{Z}_{j,k}$ from $\boldsymbol{Y}_{j,k}$, followed by the algorithm from [7]), would necessarily ignore the fact that the 2-way marginals $\boldsymbol{Z}_{j,k}$ are inter-dependent due to common mode factors.* This motivates the further steps in Alg. 1 which account for such dependencies, and *the empirical results from Sec. 4 further support their inclusion.* We additionally note that the cost function $J_1(.)$ in Eqn. 6 can be easily modified to include further prior information about the density such as it being piece-wise flat (true for PMFs) or smooth. Our algorithm can be viewed as 3 sequential processing blocks which refine the estimates produced by the previous steps in the pipeline. The three blocks are: G1 (lines 2-4), G2 (lines 6-13), and G3 (lines 15-20). We refer to our method as JUROR: **J**oint distribution recovery **U**sing **R**andom projections to **O**ne dimensional **R**egion.

4. NUMERICAL RESULTS

In this section, we present several PMF estimation results on both synthetic and real-world datasets. For synthetic data, we present results for both discrete and continuous RVs. The synthetic data are created from their mode factors, in the following manner: (1) For PMFs of discrete RVs are created, each entry in $\boldsymbol{A_i}$ is generated i.i.d. from Uniform$[0, 1]$ followed by normalizing the columns to have unit ℓ_1 norm. Elements of $\boldsymbol{\lambda}$ are generated in the same manner. (2) For continuous RVs, we consider cumulative interval measures (CIMs) instead of PDFs and represent them as tensors. For the CIM,

each column of $\boldsymbol{A_i}$ is generated by sampling from sinusoidal waves of varying amplitudes, phase and frequency, at regular intervals. For synthetic data, F is known to the algorithm before hand and it tries to find the mode factors given just the data sampled from the underlying distribution. For all the experiments the number of random projections $M = 200$ and $|\mathcal{B}| = \binom{N}{2}$. Let $\boldsymbol{\lambda}, \{\boldsymbol{A_n}\}_{n=1}^N$ and $\widehat{\boldsymbol{\lambda}}, \{\widehat{\boldsymbol{A_n}}\}_{n=1}^N$ be the mode factors of the true and estimated PMFs/CIMs respectively. Let $\kappa \in (0, 1]$ be the fraction of observed (as opposed to missing) entries in the data-points. Since storing the tensor with I^N entries may be infeasible, we compute the mean squared estimation error in terms of the mode factors: MSE $\triangleq \sum_{i=1}^{i=N} \frac{\|\widehat{\boldsymbol{A}_i} - \boldsymbol{A}_i\|_F^2}{N} + \|\widehat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}\|_2^2$ as is the norm in [6, 7].

Algorithm 1 Recovering Mode Latent Factors from 1D Densities of Random Projections

---

1: **procedure** JUROR
2:     Set $\{\boldsymbol{Z}_{j,k}\}_{(j,k)\in B}$ to   be   equal   to $\text{argmin} \sum_{j,k>j} \|\boldsymbol{Y}_{j,k} - \mathfrak{R}(\boldsymbol{Z}_{j,k})\|_F^2$
3:     Assemble $\widetilde{\boldsymbol{Z}}$ using $\{\boldsymbol{Z}_{j,k}\}_{(j,k)\in B}$
4:     $\boldsymbol{W}^{(0)}, \boldsymbol{H}^{(0)} \leftarrow \text{SPA}(\widetilde{\boldsymbol{Z}})$
5:     converged $\leftarrow$ False, $\rho \leftarrow \rho_0$, $q \leftarrow 1$
6:     **while** converged $==$ False **do**
7:         Fetch $\{\boldsymbol{A_n}\}_{n=1}^{n=N}, \boldsymbol{\lambda}$ from $\boldsymbol{W}^{(q-1)}, \boldsymbol{H}^{(q-1)}$
8:         $\boldsymbol{Z}_{j,k}^{(n)} = (\mathfrak{R}^T\mathfrak{R} + \rho I)^{-1}(\mathfrak{R}^T\boldsymbol{Y}_{j,k} + \rho \boldsymbol{A}_j D(\boldsymbol{\lambda})\boldsymbol{A}_k^T)$
9:         Assemble $\widetilde{\boldsymbol{Z}}$ using $\{\boldsymbol{Z}_{j,k}^{(n)}\}_{(j,k)\in B}$
10:       $\boldsymbol{W}^{(q)}, \boldsymbol{H}^{(q)} \leftarrow \text{SPA}(\widetilde{\boldsymbol{Z}})$
11:       $q = q + 1$
12:       **if** $J_1(.) < \epsilon$ **then** converged $\leftarrow$ True
13:     Fetch $\{\boldsymbol{A}_n^0\}_{n=1}^{n=N}, \boldsymbol{\lambda}^0$ from $\boldsymbol{W}^{(q)}, \boldsymbol{H}^{(q)}$
14:     converged $\leftarrow$ False, $q \leftarrow 1$
15:     **while** converged $==$ False **do**
16:       **for** k in 1 to N **do**
17:         $\boldsymbol{A}_k^{(q)} \leftarrow \text{ProjectOnSimplex}(\boldsymbol{A}_k^{(q-1)} - \eta_q \frac{\partial J_1}{\partial \boldsymbol{A}_k})$
18:       $\boldsymbol{\lambda}^{(q)} \leftarrow \text{ProjectOnSimplex}(\boldsymbol{\lambda}^{(q-1)} - \eta_q \frac{\partial J_1}{\partial \boldsymbol{\lambda}})$
19:       $q = q + 1$
20:       **if** $J(.) < \epsilon$ **then** converged $\leftarrow$ True
21:     **return** $\{\boldsymbol{A_n}\}_{n=1}^{n=N}, \boldsymbol{\lambda}$

The results are reported in Tables 1 and 2 for different number of samples (denoted by N_s) for PMF/CIM estimation. In both cases $F > I$ and $\kappa = 1$. The methods compared are (1) CTF using 3-way marginals [6]; (2) the SPA method using 2-way marginals [7]; (3) our JUROR method from Alg. 1 with various combinations of the algorithm stages for the purpose of ablation study defined as A: G1 + G2 + G3, B: G1 + G3, C: G1 + G2; and (4) the EM technique from [13] with random initial conditions referred to as RAND-EM. In Table 3 we consider the real-world scenario where not all features are observed for every data-point and present results for $\kappa = 0.8$ as the probability of observing each feature of a sample.

N_s	100	1000	5000	10000	50000
CTF	0.339	0.294	0.233	0.172	0.103
SPA	0.295	0.264	0.196	0.143	0.084
JUROR-A	**0.253**	**0.205**	**0.174**	0.138	0.092
JUROR-B	0.267	0.229	0.182	**0.131**	**0.087**
JUROR-C	0.262	0.217	0.185	0.140	0.098
RAND-EM	0.284	0.246	0.204	0.149	0.106

Table 1. MSE for PMFs for $F = 25, I = 10, N = 6, \kappa = 1$

N_s	1000	5000	10000	50000	100000
CTF	0.318	0.289	0.227	0.191	0.164
SPA	0.296	0.274	0.238	0.182	0.158
JUROR-A	**0.271**	**0.259**	**0.194**	**0.162**	0.149
JUROR-B	0.283	0.264	0.209	0.175	**0.137**
JUROR-B	0.276	0.268	0.204	0.179	0.155
RAND-EM	0.281	0.268	0.198	0.187	0.157

Table 2. MSE for CIMs for $F = 90, I = 50, N = 5, \kappa = 1$

N_s	100	1000	10000	50000
CTF	0.316	0.151	0.142	**0.092**
SPA	0.238	0.162	0.126	0.117
JUROR-A	**0.205**	**0.147**	0.131	0.103
JUROR-B	0.225	0.159	**0.118**	0.107
JUROR-C	0.218	0.154	0.126	0.114
RAND-EM	0.269	0.173	0.152	0.136

Table 3. MSE for PMFs for $F = 20, I = 15, N = 5, \kappa = 0.8$

The MSE results in Tables 1, 2, 3 are obtained after averaging over 5 random runs, and with $M = 200$ Radon projections per pair of RVs. As expected, our algorithm performs better in the low-sample regime where estimating 1D marginals is much more reliable than higher-D marginals. All the methods start converging in the high-sample regime where out method isn't far from the lowest MSE. The results for high-D PMFs with $N = 15$ are presented in Table 4. From the above tests it is evident that G2 of the algorithm is crucial when we have less samples and the final step G3 is more effective in the high-sample regime. We also study the effect of number of projections M on density estimation using the algorithm JUROR-A. The results are shown in Table 5.

In real-world applications where the true underlying PMF is unknown, we compared the different methods in terms of classification accuracy computed after estimating the joint PMFs of the form $p(\boldsymbol{x}, y)$, where y is the label associated with data-point $\boldsymbol{x} \in \mathbb{R}^N$. The classification is done by assigning class label $\hat{y} = \text{argmax}_y p(y|\boldsymbol{x}) = \text{argmax}_y p(\boldsymbol{x}, y)/p(\boldsymbol{x})$ to $\boldsymbol{x}$. We test our algorithms against some common classification methods on two datasets from the UCI repository [21] - the Car (6D) and the Mushroom (22D) datasets. The value of F was set by cross-validating with the accuracy obtained on

the validation split. Commonly used discriminative classifiers such as SVM-RBF and neural networks were trained using MATLAB's classification toolbox [22]. The train-val-test split was 70-10-20 for the car dataset and 50-20-30 for the mushroom dataset. As seen in Table 6, our method JUROR outperforms other techniques. More results including MSE losses compared on actual tensor values can be found at [23].

N_s	1000	5000	10000	50000	100000
CTF	0.268	0.242	0.201	0.154	**0.104**
SPA	0.245	0.237	0.226	0.176	0.126
JUROR-A	**0.229**	**0.208**	**0.184**	0.162	0.115
JUROR-B	0.258	0.217	0.205	**0.151**	0.109
JUROR-C	0.241	0.225	0.197	0.171	0.119
RAND-EM	0.248	0.226	0.217	0.197	0.143

Table 4. MSE for PMFs for $F = 20, I = 10, N = 15, \kappa = 1$

$M(\downarrow)$	100	1000	5000	10000	50000
200	0.253	0.205	0.174	0.138	0.092
150	0.264	0.214	0.188	0.151	0.106
100	0.289	0.267	0.223	0.184	0.129
50	0.384	0.357	0.305	0.264	0.213

Table 5. MSE vs. M for $F = 25, I = 10, N = 6, \kappa = 1$

Algorithm	**Car**	**Mushroom**
CTF	84.92	95.13
SPA	86.45	96.01
JUROR-A	87.59	**96.72**
JUROR-B	86.37	95.12
JUROR-C	**88.38**	95.79
RAND-EM	82.68	94.65
LOGISTIC REGRESSION	82.37	95.86
SVM-RBF	78.32	95.74
NAIVE BAYES	84.39	89.67
NEURAL NET	85.16	96.37

Table 6. Classification Accuracies on real-world datasets

5. CONCLUSION AND FUTURE WORK

Combining ideas from tomography and low-rank tensors, we presented a novel algorithm to recover the joint PMF from one-way marginals obtained from random projections of the data. Empirical results suggest that our JUROR method is particularly useful in the low-sample regime where most other estimators under-perform. Our framework opens avenues for newer studies at the intersection of tomography, tensors, and density estimation. Some future work may include a theoretical analysis of JUROR from the point of view of sample complexity and estimating continuous PDFs directly using appropriate basis functions for the mode latent factors.

6. REFERENCES

[1] K. Murphy, *Machine learning: a probabilistic perspective*, The MIT Press, 2012.

[2] M. J. Wainwright, *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*, Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.

[3] A. Tsybakov, *Introduction to Nonparametric Estimation*, Springer Publishing Company, Incorporated, 1st edition, 2008.

[4] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models.," in *Machine Learning*, 1999, vol. 37, pp. 183–233.

[5] D. MacKay, "Introduction to monte carlo methods," in *Learning in graphical models*, pp. 175–204. Springer, 1998.

[6] 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, 2018.

[7] S. Ibrahim and X. Fu, "Recovering joint probability of discrete random variables from pairwise marginals," https://arxiv.org/abs/2006.16912.

[8] Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M. Kakade, and Matus Telgarsky, "Tensor decompositions for learning latent variable models," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2773–2832, Jan. 2014.

[9] Nicholas D. Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E. Papalexakis, and Christos Faloutsos, "Tensor decomposition for signal processing and machine learning," *Trans. Sig. Proc.*, vol. 65, no. 13, pp. 3551–3582, July 2017.

[10] X. Fu, K. Huang, and N. D. Sidiropoulos, "On identifiability of nonnegative matrix factorization," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 328–332, 2018.

[11] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *NIPS*, 2001, vol. 13.

[12] N. Gillis and S. Vavasis, "Fast and robust recursive algorithms for separable nonnegative matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 698–714, 2014.

[13] A. Yeredor and M. Haardt, "Maximum likelihood estimation of a low-rank probability mass tensor from partial observations," *IEEE Signal Processing Letters*, vol. 26, pp. 1551–1555, 2019.

[14] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*, SIAM, 2001.

[15] F. O'Sullivan and Y. Pawitan, "Multidimensional density estimation by tomography," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 55, no. 2, pp. 509–521, 1993.

[16] S. Kolouri, G. K. Rohde, and H. Hoffmann, "Sliced wasserstein distance for learning gaussian mixture models," in *CVPR*, 2018.

[17] J. Webber, E. Hussey, E. Miller, and S. Aeron, "On non-parametric density estimation on linear and non-linear manifolds using generalized radon transforms," https://arxiv.org/abs/1901.03780.

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

[19] H. Jiang, "Uniform convergence rates for kernel density estimation," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. 2017, ICML'17, p. 1694–1703, JMLR.org.

[20] L. Cavalier, "Efficient estimation of a density in a problem of tomography," *The Annals of Statistics*, vol. 28, no. 2, pp. 630–647, 2000.

[21] "UCI machine learning repository," https://archive.ics.uci.edu/ml/datasets.php.

[22] D. Ballabio and V. Consonni, "Classification tools in chemistry. part 1: linear models. PLS-DA," *Analytical Methods*, vol. 5, no. 16, pp. 3790, 2013.

[23] J. Vora, K. S. Gurumoorthy, and A. Rajwade, "Recovery of joint probability distribution from one-way marginals: Low rank tensors and random projections," https://arxiv.org/abs/2103.11864.