Department of Electrical Engineering
Indian Institute of Technology, Bombay

EE 491: Bachelor's Thesis Project
Low Rank Joint Probability Tensor Recovery from 1D Marginals

Jian Vora, 170100026

Faculty Mentors
Prof. Ajit Rajwade & Prof. Rajbabu Velmurugan

December, 2020

Contents

1 Abstract **1**

2 Probability Tensor Decomposition **3**

2.1 Low Rank of the Joint Probability Tensor 3

2.2 Canonical Polyadic Decomposition of a Tensor 4

2.3 Uniqueness of the Canonical Polyadic Decomposition 5

3 Tomography in Density Estimation **6**

3.1 Random Linear Projections 6

3.2 Preliminaries from Tomography 6

3.2.1 Radon Transform of a Function 6

3.2.2 Filtered Back Projection(FBP) Algorithm 7

3.3 Random Projections for Density Estimation 7

3.4 Empirical Results of Density Estimation with Tomography 8

4 Literature Review **12**

4.1 Joint Density Estimation from Random Projections 12

4.2 Joint Density Estimation from 3-way Marginals 12

4.3 Joint Density Estimation using Expectation Maximization 13

4.4 Joint Density Estimation from 2-way Marginals 13

5 High-Dimensional Density Estimation **14**

5.1 Problem Statement and Algorithm 14

5.2 Empirical Results on Synthetic Data 14

6 Conclusion and Future Work **19**

7 References **20**

1. Abstract

Given a set of N random variables $X_1, X_2, ..., X_N$, estimating the joint density $p(X_1, X_2, ..., X_N)$ is a fundamental problem in many fields such as statistics and machine learning. This probabilistic interpretation can help us make may inferences to help us aid take better decisions for the problem in hand. For a simple example, take X_1 as the time taken to drive to a particular destination and X_i, $2 \leq i \leq N$ denote a scalar number indicating the traffic on the $N-1$ streets. We would want to make predictions such as the minimum time needed for reaching the destination given we somehow know the traffic profiles of all the streets, i.e. more formally, we would want to find the following : min $p(X_1|X_2, X_3, ..., X_N)$. Maximum a posteriori(MAP) estimation of the joint probability density is used in classification problems. Probabilistic models can also be used for anomaly detection in various fields like law, medicine. Capturing the joint density is helpful as we can infer a variety of queries of such form easily(?). Most the work done can be classified in a few categories namely:

1. If the realisations of these random variables are discrete, then standard histogramming would work but it requires a large number of samples(exponential in N) to make the estimate of the joint close to the actual density and is clearly not scalable for large N. In the big data era, it is very common for data to be high dimensional(an image of size 200 $\times$200 is 40k dimensional), and hence such a naive approach for modelling densities without accounting for the inherent structure which data posseses fails in these scenarios.

2. A parametric model for estimating the underlying distribution. A Gaussian mixture model is the simplest of a capturing the probability density by modelling it as a mixture of gaussians. This is indeed shown to be a universal approximator, i.e. with enough number of components, it can approximate any density perfectly. This however is not very practical to use because the number of components increases very rapidly for even commonly occuring densities. The expressive power of these models is limited.

3. Bayesian networks or Markov random fields which explicitly model some (in)dependencies between the set of random variables and hence scalable to large data however require a lot of approximation techniques of a variety of common queries. These techniques model a directed/undirected acyclic graph with the nodes representing the random variables and edges with weights representing the conditional probabilities. Although these are the most popular methods for current sized datasets, learning the structure of these graphs is arduous coupled with a variety of approximation algorithms to perform inference once these models are learned.

Given the relevance of the problem of density estimation, we would want to find the density of the N variables while still saving us from the curse of dimensionality. Thus, density estimation without assuming some structure in the data is mission impossible. This report shall aim to answer the following questions pertinent to density estimation:

1. What is a good prior on the joint probability density tensor which holds true for a variety of real-world datasets?

2. Given just marginals of the joint density, can we reconstruct the entire tensor using the prior as mentioned in (1)?

3. Can transforming the data to a different space help us in the pursuit of the above goals?

This report is organized as follows - Chapter 2 introduces the canonical polyadic model which we shall use to model the probability tensor and the link between radon tranform with density estimation. Chapter 3 introduces the algorithm to reconstruct the joint tensor given the marginals in the radon space. Chapter 4 includes some preliminary results of our algorithm on recovering 2-way joint densities and Chapter 5 includes empirical results for recovering high dimensional densities. Chapter 6 concludes the discussion and mentions some avenues for future work. Chapter 7 contains the relevant literature references on which this work is built on.

2. Probability Tensor Decomposition

2.1 Low Rank of the Joint Probability Tensor

Going back to the setting of N random variables $X_1, X_2, ..., X_N$, we only consider the setting where each random variable X_i takes a discrete number of, say I_i, states. Thus we want to model $p(X_1, X_2, ..X_N)$ which is an $N-$way tensor(we shall refer to this as $\boldsymbol{X}$ henceforth) which each axis having a length of I. Thus the sum of entries of $\boldsymbol{X}$ is one with $\boldsymbol{X} \in {\mathbb{R}_+}^{I_1 \times I_2 \times I_3 ... \times I_N}$ and our aim is recovering this core pmf tensor from some sample realisations of the set of random variables. More specifically, we have $\boldsymbol{X}(x_1, x_2, ...x_N) = p(X_1 = x_1, X_2 = x_2, ..X_N = x_N)$. One structure which this core tensor $\boldsymbol{X}$ follows is that of a low-rank. This is a rational assumption in the sense that for much real-world datasets, most of the random variables are reasonably (in)dependent. For example, in the case of images, we know neighbouring pixels values are highly correlated while value of far off pixels are almost independent. This is also true for a large variety of temporal data where realisations of current random variables are independent to a large extend to values far off back in time which is incorporates using a Markov model. Thus, our aim is recovering the tensor $\boldsymbol{X}$ from partial entries exploiting the low-rankness of this tensor. To make this assumption explicit, we see how this low-rankness assumption holds for the case of just 2 random variables.

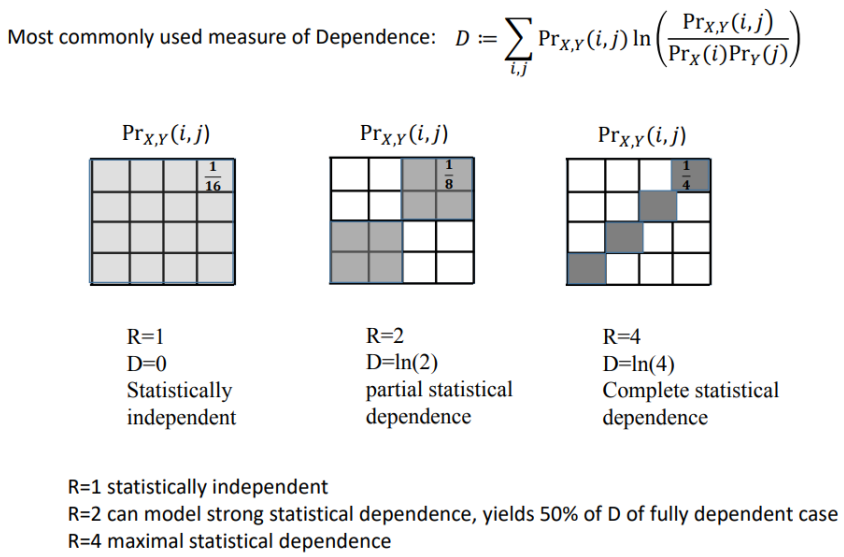

Figure 2.1: Illustration of low-rankness of the core probability tensor, this work is in the regime of $R = 2$, i.e. partial statistical dependence. Credits: [7]

Our problem formulation is that of a low rank tensor recovery from partially observable information about the tensor entries. We employ a Canonical Polyadic Decomposition on $\boldsymbol{X}$ to force low-rankess which is elaborated in the subsequent section.

2.2 Canonical Polyadic Decomposition of a Tensor

$N-$way tensor $\boldsymbol{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times ... \times I_N}$ admits a Canonical Polyadic Decomposition(CPD) if it can be decomposed as a sum of F rank-1 tensors.Let $a \otimes b$ denote the outerproduct of two vectors, then formally the CPD model can be stated as follows:

$$\boldsymbol{X} = \sum_{f=1}^{f=F} \boldsymbol{\lambda}(f) \boldsymbol{A_1}(:, f) \otimes \boldsymbol{A_2}(:, f) \otimes \boldsymbol{A_3}(:, f) \otimes ... \otimes \boldsymbol{A_N}(:, f)$$

F is the smallest number for which such a decomposition exists. Thus whenever we refer to a tensor $\boldsymbol{X}$ having rank F, it implies that $\boldsymbol{X}$ can be decomposed into F rank one tensors using the above CPD model. $\boldsymbol{\lambda} \in {\mathbb{R}_+}^F$ and $\boldsymbol{A_i} \in {\mathbb{R}_+}^{I_i \times F}$ and $[\boldsymbol{\lambda}, \boldsymbol{A_1}, \boldsymbol{A_2}, ..., \boldsymbol{A_N}]$ can uniquely determine the core tensor. $\boldsymbol{A_i}$s are referred to as mode latent factors in the subsequent parts of this report. Hence recovering the core pmf tensor $\boldsymbol{X}$ is equivalent of recovering the mode latent factors. The above model has various forms as follows:

$$\boldsymbol{X} = \sum_{f=1}^{f=F} \boldsymbol{\lambda}(f) \prod_{n=1}^{n=N} \boldsymbol{A_n}(i_n, f)$$

$$\text{vec}(\boldsymbol{X}) = (\boldsymbol{A_N} \otimes \boldsymbol{A_{N-1}} \otimes ... \otimes \boldsymbol{A_1}) \boldsymbol{\lambda}$$

There is link between the CPD model and the naive bayes model. Assume a latent variable H taking F distinct states, then the same CPD model as described above can be formulated in the following manner:

$$Pr(X_1 = i_1, X_2 = i_2, .., X_N = i_n) = \sum_{f=1}^{f=F} Pr(H = f) \prod_{n=1}^{n=N} Pr(X_n = i_n | H = f)$$

Thus the mode latent factors have an interpretation of conditional densities. Thus, comparing with the original CPD model, we get $\boldsymbol{\lambda}(f) = Pr(H = f)$ and $\boldsymbol{A_n}(i_n, f) = Pr(X_n = i_n | H = f)$. Thus, as it is clearly evident, the entries of $\boldsymbol{\lambda}$ and each column of the mode latent factor should sum to one to be valid densities along with the non-negativity constraints. F is an important hyperparameter which explores how much dependency between the variables would we like to model. Typically, we want $F << min_k \prod_{n \neq k} I_n$, for such a decompositon to be meaningful to capture low-rankness of the tensor. This is also reasonable in practice as random variables are not completely dependent. We now see when is the CPD decomposition unique in the next section.

2.3 Uniqueness of the Canonical Polyadic Decomposition

In this section, we see the various constraints on F, the rank of the tensor to see when is the CPD decomposition unique. Here, uniqueness refers to essential uniqueness defined as:

Lemma 1: For a tensor $\boldsymbol{X}$ of rank F, we say that a decomposition $\boldsymbol{X}= [\boldsymbol{\lambda}, \boldsymbol{A_1}, \boldsymbol{A_2}, .., \boldsymbol{A_N}]$ is essentially unique if the factors are unique up to a common permutation and scaling / counter-scaling of columns of the mode latent factors.

We clearly do not have the scaling ambiguity if we the non-negativity and sum to one constraints on the columns of the latent factors. Let $\boldsymbol{X} = [\boldsymbol{A_1}, \boldsymbol{A_2}, \boldsymbol{A_2}]$, where $\boldsymbol{A_1} \in \mathbb{R}^{I_1 \times F}$, $\boldsymbol{A_2} \in \mathbb{R}^{I_2 \times F}$, and $\boldsymbol{A_3} \in \mathbb{R}^{I_3 \times F}$ with $I_1 \leq I_2 \leq I_3$ without loss of generality, then:

Lemma 2: If $\min(I_1, I_2) \geq 3$ and $F \leq I_3$, then, $\text{rank}(\boldsymbol{X}) = F$ and the decomposition of $\boldsymbol{X}$ is essentially unique, almost surely, if and only if $F \leq (I_1 - 1)(I_2 - 1)$

3. Tomography in Density Estimation

3.1 Random Linear Projections

For a given vector $X \in \mathbb{R}^N$, we define the following operation as random linear projections:

$$Y = \Phi X$$

where $\Phi \in \mathbb{R}^{M \times N}$ is composed of entries drawn i.i.d from a standard normal Gaussian or a bernoulli $\{-1, 1\}$ distribution with the columns scaled accordingly. If $M < N$, then we call the projection as compressive. Such a forward model is commonly used for compressive sensing of signals and random projections have also been shown to do dimensionality reduction much more computationally efficiently as compared to some other expensive methods like PCA.

3.2 Preliminaries from Tomography

3.2.1 Radon Transform of a Function

The Radon transform of a 2-D function $f(x, y)$ is defined as:

$$R(r, \theta)[f] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y)\delta(r - xcos\theta - ysin\theta)dxdy$$

where r is the perpendicular distance of a line from the origin and θ is the angle between the line and the y-axis. According to the Fourier slice theorem, this transformation is invertible with minimal reconstruction loss and that transform is called the Inverse Radon operation.

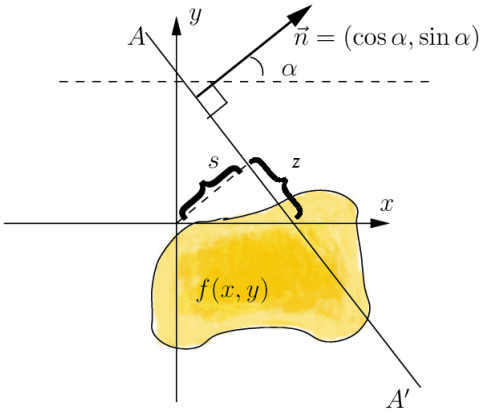

Figure 3.1: Graphical Illustration of the Radon Transform. Source: Wikipedia

The radon transform $R(r,\theta)[f]$ can be inverted to reconstruct the function $f(x,y)$ using many computationally efficient algorithms with the filtered back projection(FBP) algorithm being the most prominent which is described in the next section.

3.2.2 Filtered Back Projection(FBP) Algorithm

Analytical reconstruction methods consider continuous tomography. So the problem is: given the radon transform we want to recover the object described in (x,y) coordinates. The most common type of image analytical reconstruction is Filtered Back Projection (FBP). Fourier slice theorem states that for a $2-D$ function $f(x,y)$, the $1-D$ Fourier transforms of the Radon transform along r, are the $1-D$ radial samples of the $2-D$ Fourier transform of $f(x,y)$ at the corresponding angles. Using the Fourier slice theorem, we can see that if we are given the Fourier transform of a projection at enough angles, the projections could be assembled into a complete estimate of the two-dimensional transform and then simply inverted to arrive at an estimate of the object. This procedure is known as the Filtered Back Projection algorithm. It has been shown to be extremely accurate and amenable to fast implementation. The back projection algorithm mathematically expressed as:

$$\hat{f}(x,y) = \int_0^{\pi} R(r,\theta) d\theta$$

The filtered back projection algorithm is also based on a similar principle but the signal $R(r,\theta)$ is modulated by a filter. Let $R(v,\theta)$ be the $1-D$ fourier transform of $R(r,\theta)$ keeping the θ fixed. Then for a variety of filters with the transfer function $H(v)$, we have:

$$\hat{f}(x,y) = \int_0^{\pi} \int_{-\infty}^{+\infty} R(v,\theta) H(v) dv d\theta$$

There are several famous choices for the filter H such as the Ram-Lak filter, Cosine filter and the Shepp-Logan filter. Thus, whenever we refer to inverse radon tranform, we refer to the filtered back projection algorithm with a filter implicitly assumed.

3.3 Random Projections for Density Estimation

Consider a random vector $X \in \mathbb{R}^N$ and another random vector $\theta \in \mathbb{R}^N$. θ can be considered as one column of the random matrix Φ defined earlier in this chapter. We are interested in estimating $p(X)$ from random linear projections of the data. $\theta^T X$ is a scalar variable and hence we can estimate $p(\theta^T X)$ which is a 1D density estimate and is fairly easy to do using either histogramming or kernel density estimate. Let $\theta = [\theta_1, \theta_2, ..., \theta_N]$ and $X = [X_1, X_2, ..., X_N]$, then

$$p(\theta^T X = t) = p(\sum_i \theta_i X_i = t)$$

$$p(\theta^T X = t) = \sum_{a_1} \sum_{a_2} .. \sum_{a_{N-1}} p(\theta_1 X_1 = a_1, \theta_2 X_2 = a_2, ...\theta_N X_N = t - \sum_{i=1}^{i=N-1} a_i)$$

Thus the 1D density estimate of a linear projection of a data vector is infact the radon transform of the joint probability tensor taken at an angle θ. Thus, if we collect densities of the form $p(\theta_i^T X)$ for various i, then we can use concepts of inverse radon transform to reconstruct the core probability tensor. Furthermore, if we incorporate the low-rankness of the tensor, we can recover the tensor using 1D marginals in the radon space. We now investigate the usage of this method of radon transform for density estimation on just 2 random variables X_1 and X_2.

3.4 Empirical Results of Density Estimation with Tomography

Suppose we have N random variables and want to estimate $p(X_1, X_2, ...X_N)$, then on arranging then as a vector X, we take linear combinations of them through a linear ϕ operator. Let the number of projections be m and the number of samples S.

$$Y = \phi X; Y = [y_1, y_2, ..., y_m]^T$$

From S samples we get the following - $p(y_1), p(y_2), ..., p(y_m)$ and use this for getting the original $p(x_1, x_2, ...x_n)$. Certain notations are as follows:

N : Number of random variables
S : Number of samples of X
m : Number of projections
I : Number of discrete states each random variable takes
$numbins$: Number of values in which each y_i shall be discretized

Some experimental conditions the first preliminary experiment were as follows:

N : 2
S : Variable for analysis
m : Variable for analysis
I : 10
$numbins$: 16

Data Generation Process : We take a truncated 2D gaussian (from -2 to +2 in either directions) with an identity covariance matrix and 0 mean. This is put into 10 bins as cummulative interval measures to get the PDF matrix. PDF of y_i is generated using **histograms** with the $numbins$ parameter.

Implementation details : iradon implemented with Ram-Lak filter and without any filter was studies angles obtained by using 'atand' command for each row of ϕ

Evaluation Metric: Jensen-Shannon Divergence

$$JS(P,Q) = \frac{KL(P||(P+Q)/2) + KL(Q||(P+Q)/2)}{2}$$

```
Number of Projections = [5, 20, 50, 100, 200, 500, 1000]
Number of Samples     = [10, 100, 1000, 5000, 10000, 20000]
```

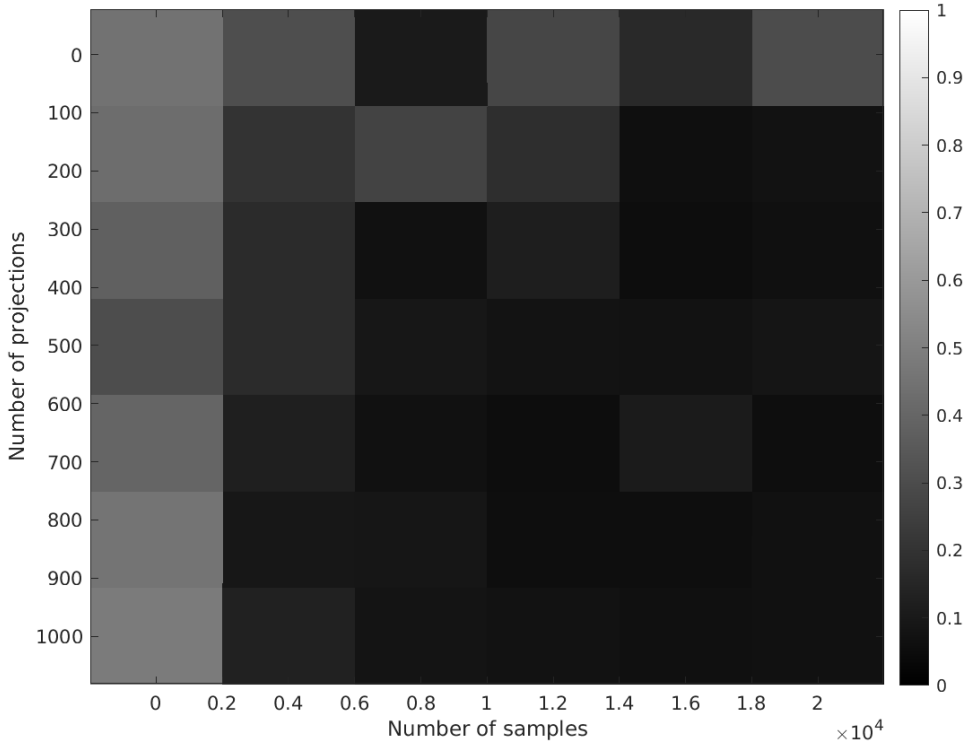

Figure 3.2: JSD (smaller the better) errors using Ram-Lak Filter

Comparision with other 2D density estimators

Number of samples	JSD Radon	JSD Histogramming
100	0.0608	0.1902
500	0.0478	0.0529
1000	0.0221	0.0203
10000	0.0107	0.0019

Table 3.1: Reconstruction Comparisons with 1000 projections for Radon

We conclude that the radon approach for simple density estimation performs better than standard histogramming approach especially when the number of samples are less. Further, we

need the underlying densities to be continuous and should have decaying fourier coefficients in order to be able to apply inverse radon transform efficiently. Thus, most the this work is concerned with the low sample regime with continuous densities. Next, to validate our claim, we look at capturing the densities of a variety of gaussian mixture models.

Data Generation Process : We take a truncated 2D gaussian (from -8 to +8 in either directions) with varying number of components and random means and full covariances while ensuring the positive semi-definite condition. This is put into 500 bins as cummulative interval measures to get the PDF matrix. PDF of y_i is generated using histograms with the *numbins* parameter.

```
Number of Projections = [2,10,40,100,500]
Number of Samples     = [100,500,1000,5000,10000,20000]
```

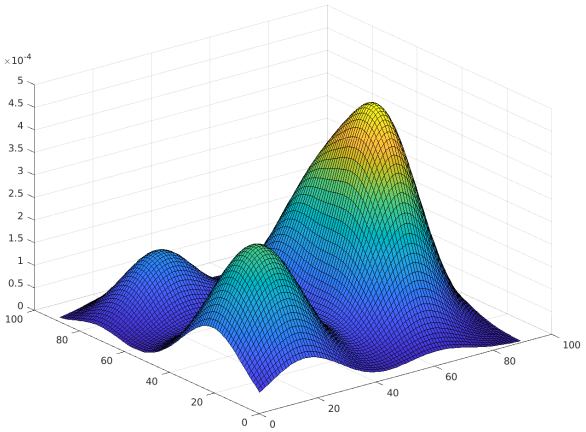

Figure 3.3: GMM 1 (4 components)

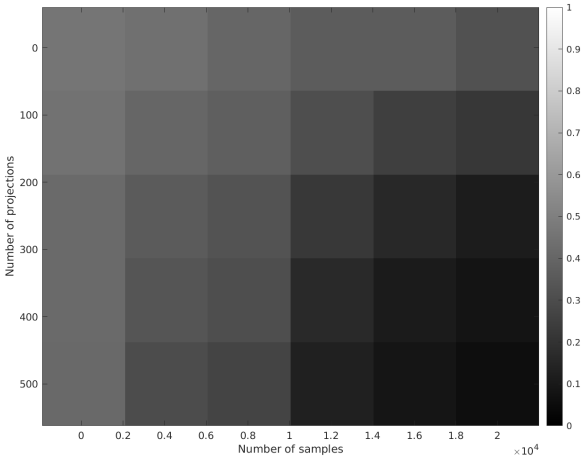

Figure 3.4: GMM 1 Errors

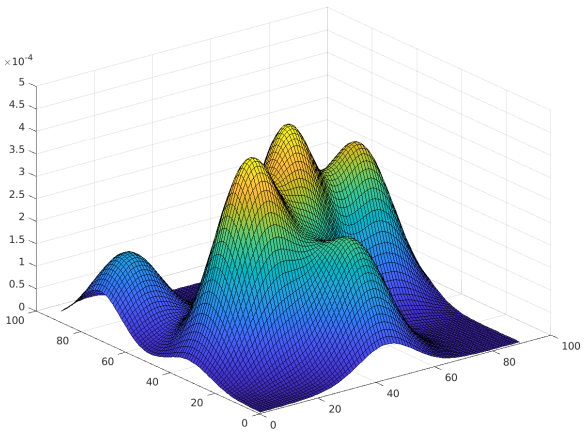

Figure 3.5: GMM 2 (9 components)

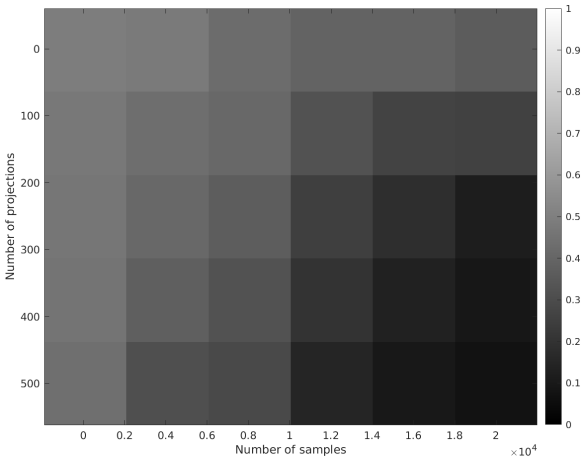

Figure 3.6: GMM 2 Errors

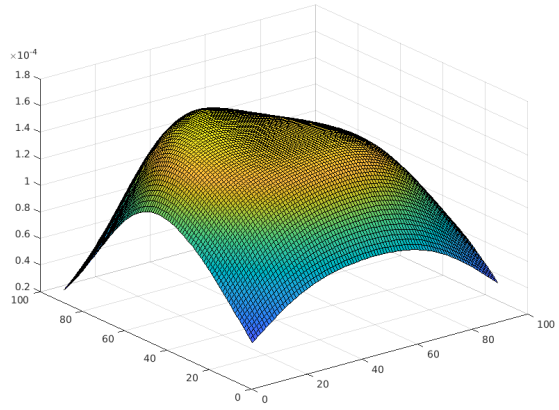

Figure 3.7: GMM 3 (4 components)

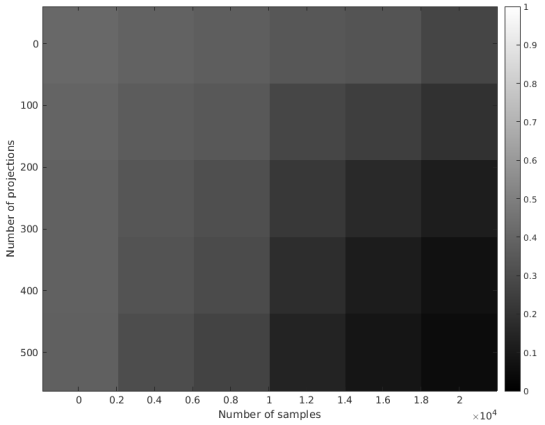

Figure 3.8: GMM 3 Errors

Thus, in this chapter, we introduced the link between tomography and density estimation. The take away message is that by estimating the densities of random projection of data, we can reconstruct the original density using the inverse radon transform. We presented results on simple 2D densities which will be used later to reconstruct the core joint pmf tensor using the low rank canonical polyadic decomposition model which is described in the subsequent sections.

4. Literature Review

4.1 Joint Density Estimation from Random Projections

Multidimensional Density Estimation by Tomography by Yudi Pawitan was the first paper linking random projections of data with density estimation. It formulated the problem of pmf estimation as inverting the radon transform. However, there were two major things lacking in this formulation:

1. Experiments were conducted for only two-dimensional joint densities as inverting the radon transform is costly as the number of dimensions keep on increasing.

2. Did not leverage the property that joint density tensors have a low-rank

4.2 Joint Density Estimation from 3-way Marginals

This was the first work concerning density estimation from marginals by Kargas et.al. In this paper, they first estimate 3-way marginals of the form $p(X_i, X_j, X_k)$ from the data using standard histogramming. Once we have the 3-way marginals, we can decompose that uniquely to get the corresponding mode factors and once we have all the model factors, then we can uniquely determine the core pmf tensor. F is to be such that the decomposition of the 3-way marginals is unique in accordance with the lemma stated in Chapter 2. They use the ADMM to solve the proposed optimization problem of joint factorisation. Their algorithm can be described as follows:

Procedure: Joint PMF Recovery From Triples

1. Estimate $\boldsymbol{X}_{j,k,l}$ from the data
2. Jointly factor $\boldsymbol{X}_{j,k,l} = [\boldsymbol{\lambda}, \boldsymbol{A_j}, \boldsymbol{A_k}, \boldsymbol{A_l}]$ using the CPD model of rank F
3. Once all the latent factors are available using the joint factorisation, reconstruct using

$$\boldsymbol{X} = \sum_{f=1}^{f=F} \boldsymbol{\lambda}(f)\boldsymbol{A_1}(:, f) \otimes \boldsymbol{A_2}(:, f) \otimes \boldsymbol{A_3}(:, f) \otimes ... \otimes \boldsymbol{A_N}(:, f)$$

For solving the joint factorisation problem, we use the following least squares optimization problem using alternating minimization:

$$\min_{\{\boldsymbol{A_n}\},\boldsymbol{\lambda}} \sum_j \sum_{k>j} \sum_{l>k} \|\boldsymbol{X}_{j,k,l} - [\boldsymbol{\lambda}, \boldsymbol{A_j}, \boldsymbol{A_k}, \boldsymbol{A_l}]\|_F^2$$

subject to appropriate normalization of $\boldsymbol{\lambda}$ and $\boldsymbol{A_n}, n = 1, 2, 3, ..., N$

4.3 Joint Density Estimation using Expectation Maximization

The formulation of CPD model is similar to that of a gaussian mixture model where instead of the mixture weights, we have $\boldsymbol{\lambda}$ and instead of estimating the means and covariances of the gaussians, we have to estimate the mode latent factors. Thus, work by Yeredor et.al. tries to find the parameters by maximising the likelihood of observing the data. It uses an EM update like the one used for training the GMMs.

$$\log\ P(y[1], y[2], ..., y[T]) = \sum_{t=1}^{t=T} \log\ \sum_{f=1}^{f=F} \lambda_f \prod_{n=1}^{n=N} \boldsymbol{A_n}(y[t], f)$$

Thus, we formulate the following optimization problem for minimizing the negative log-likelihood:

$$\min_{\{\boldsymbol{A_n}\},\boldsymbol{\lambda}} - \sum_{t=1}^{t=T} \log\ \sum_{f=1}^{f=F} \lambda_f \prod_{n=1}^{n=N} \boldsymbol{A_n}(y[t], f)$$

subject to appropriate normalization of $\boldsymbol{\lambda}$ and $\boldsymbol{A_n}, n = 1, 2, 3, ..., N$

4.4 Joint Density Estimation from 2-way Marginals

In this work by Ibrahim et.al., they try to recover the joint density from 2-way marginals. Given the data matrix, we first estimate the 2-way marginals $\boldsymbol{X}_{j,k}$ from histogramming and then try to recover the mode latent factors in the following manner:

$$\boldsymbol{X}_{j,k} = \boldsymbol{A_j} \boldsymbol{D}(\boldsymbol{\lambda}) \boldsymbol{A_k^T}$$

However, we cannot directly use non-negative matrix factorisation as F is generally greater than I, the number of states each variable takes and hence the matrix factorisation wont't be unique. We however, can split the N variables into 2 sets with M being the index of the split. Thus we have 2 splits namely $\{1, 2, ..., M\}$ and $\{M+1, M+2, ..., N\}$ We then arrange the 2-way marginals in the form a matrix $\boldsymbol{A}$ where the block $\boldsymbol{A_{i,j}} = \boldsymbol{X_{i,M+i}}$ where NMF can be applied to get the mode factors. This matrix $\boldsymbol{A} \in \mathbb{R}^{MI \times (N-M)I}$.

They use the Successive Projection Algorithm(SPA) for carrying out the NMF. After this, they also propose running EM with the initial values of the EM algorithm as the output of the SPA algorithm which would be a better initialisation and the optimization would not get stuck in a local minima which vanilla EM is infamous for.

5. High-Dimensional Density Estimation

5.1 Problem Statement and Algorithm

Using the ideas discussed in the previous chapters of tensor decomposition and tomography, we present a way to reconstruct the joint probability tensor from 1D marginals. Given a data-matrix D of dimensionality N and number of samples K , our algorithm to estimate the density proceeds as follows:

1. For all indices i, j such that $1 \leq i < j \leq N$, take random linear projections of data with variables X_i and X_j. For each such pair, if we take M projections, then in total we will have $M.N(N-1)/2$ scalar variables.

2. Using the above projections, estimate densities of the form $p(X_i, X_j)$ using the inverse radon transform method as discussed in Chapter 2.

3. Once we have all the pair wise marginals, then reconstruct the core probability tensor using the SPA algorithm as discussed in Chapter 4.

4. Refine the estimates of the mode latent factors by running EM on the estimates obtained.

Thus, our contribution mainly is to estimate the $2-D$ marginals using the Radon transform approach rather than histogramming. With those estimates of the $2-D$ marginals, we run the SPA algorithm followed by refining the estimates by using EM. We present some empirical results showing that using the radon approach helps in density estimation particularly when we have smooth densities and the number of samples are small.

5.2 Empirical Results on Synthetic Data

Mode latent factors were generated with values of each column from a sine wave to ensure the underlying core pmf tensor is smooth. The amplitude, frequency and phase were chosen at random for each column of the mode latent factors. The mixing ratios were chosen randomly with no such smoothness prior. 200 projections were taken for the radon approach.

The MSE in this case is the sum of RMSE of λ and the scaled RMSE of A_i. N is the number of random variables for joint density estimation, I is the number of states each variable takes and F is the rank of the tensor in the CPD model.

	1000	5000	10000	30000
SPA	0.0911	0.0543	0.0459	0.0497
SPA-EM	0.0737	0.0721	0.0358	**0.0422**
SPA-R	**0.0607**	**0.0469**	0.0376	0.0433
SPA-R-EM	0.0668	0.0579	**0.0353**	0.0452
RAND-EM	0.0713	0.0473	0.0537	0.0598

Table 5.1: MSE Comparison for F = 10, I = 50, N = 5

	1000	5000	10000	30000
SPA	0.0821	0.0963	0.0654	0.0497
SPA-EM	0.0814	0.0518	0.0452	0.0473
SPA-R	0.0611	0.0543	**0.0442**	0.0475
SPA-R-EM	0.0642	**0.0512**	0.0494	**0.0468**
RAND-EM	**0.0473**	0.0521	0.0489	0.0719

Table 5.2: MSE Comparison for F = 10, I = 200, N = 4

Next, we compare our method to previously mentioned techniques. All the MSE values reported are averaged over 5 randomly chosen densities generated as mentioned above. All the experimental conditions are mentioned in the plots itself. The legend can be explained as follows:

CNMF-SPA: Using the SPA algorithm to predict the mode latent factors

SPA-EM: Running EM after the CNMF-SPA algorithm to refine the mode latent factors

CNMF-SPA-R: Using the SPA algorithm to predict the mode latent factors but with the initial 2-way marginals coming from the radon transform approach

SPA-EM-R: Running EM after the CNMF-SPA-R algorithm to refine the mode latent factors

RAND-EM: Not estimating and marginals, run EM directly with radnom initialiaztions of the mode latent factors and λ

CTD-EM: Estimating the 3-way marginals from histogramming and then pushing the mode latent factors for EM refinement

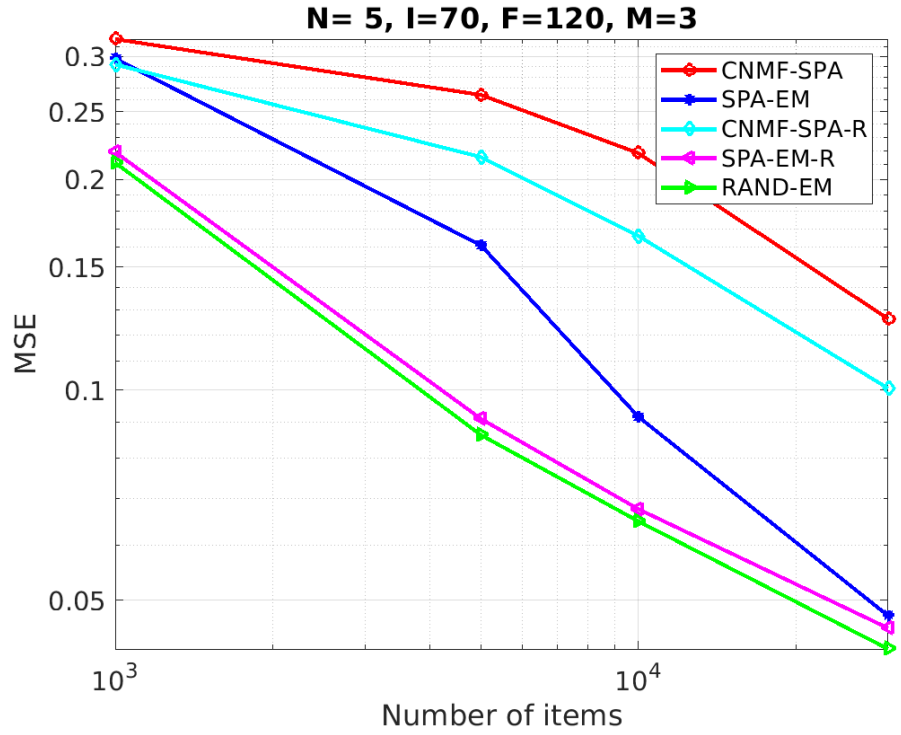

Figure 5.1: Errors averaged over 5 different randomly chosen densities using the Ram-Lak filter

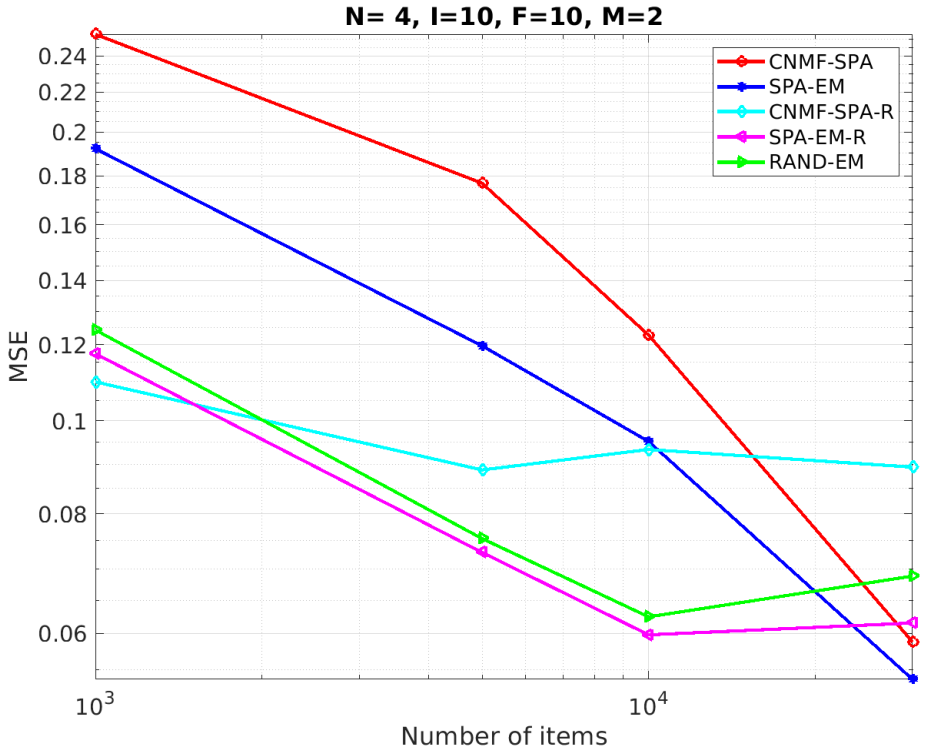

Figure 5.2: Errors averaged over 5 different randomly chosen densities using the Cosine filter

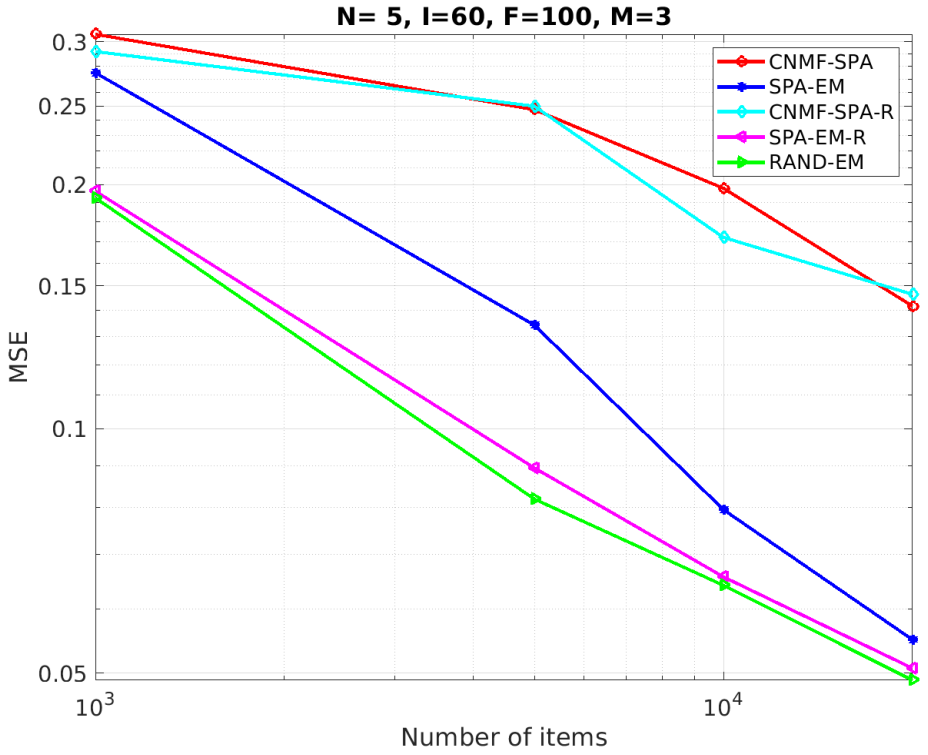

Figure 5.3: Errors compared in different experimental settings

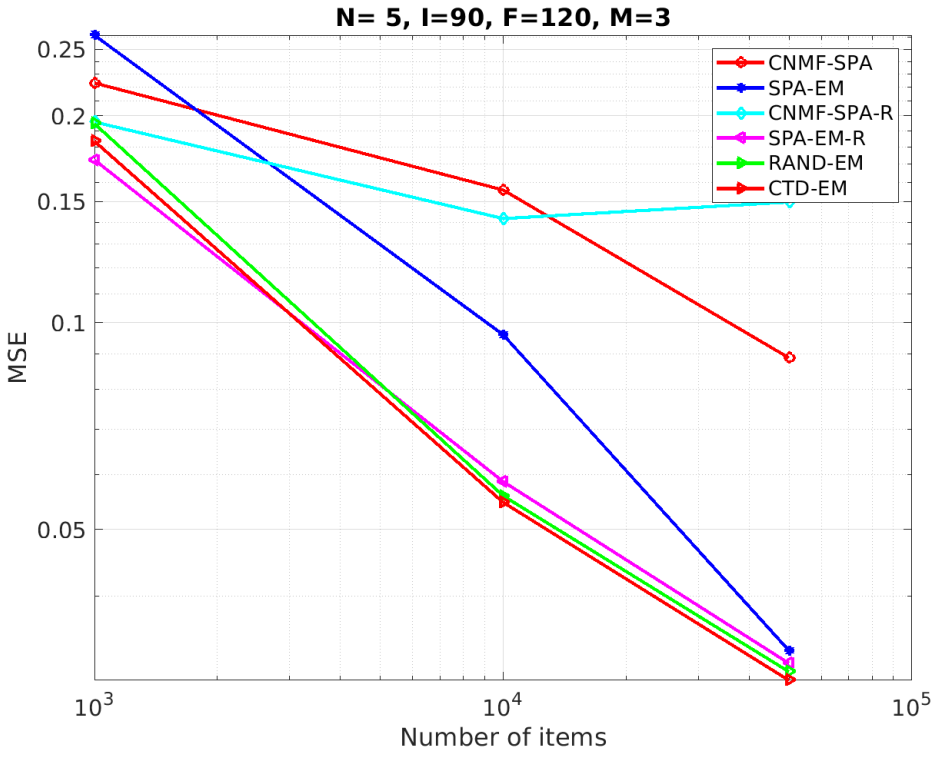

Figure 5.4: Errors compared with the 3-way marginals approach as well

We also try with missing data, which means that realisations of a random variable are

missing. We keep the random variables with a probability $p = 0.8$ and mask otherwise and hence we have an incomplete data matrix. The results are as follows:

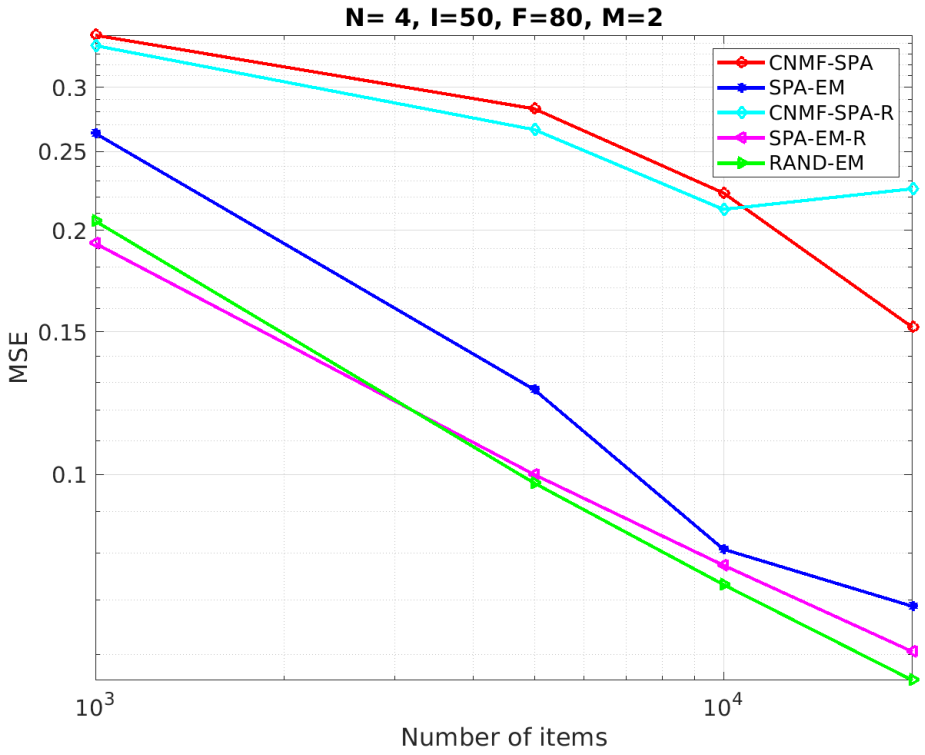

Figure 5.5: Errors with **missing data** with probability 0.2

We even tried a higher 12 dimensional data using all the above approaches. Each column represents the number of samples used to estimate the density.

	1000	10000	50000
SPA	0.182	0.158	0.083
SPA-EM	0.217	0.126	0.085
SPA-R	0.142	0.117	0.103
SPA-R-EM	**0.102**	**0.081**	0.078
RAND-EM	0.113	0.084	**0.076**

Table 5.3: MSE Comparison for F = 200, I = 90, N = 12

Note that our approach is more constrained that all the previous approaches because inverting the radon transform is in itself a lossy operation. For illustration, even if we have infinite samples, then iradon cannot reconstruct the matrix exactly and hence even under these constraints, our approach is performing better than some previously done works in this space.

6. Conclusion and Future Work

We presented a link between density estimation and tomography. We try to solve the problem of density estimation from random projections of data and then use the low-rankness of the pmf tensor to recover it from the marginals. Some future work can be as follows:

1. Try the algorithm on real-world datasets for doing some probabilistic inferences like classification or regression

2. Remove the constraint of the rank of the tensor being strictly F and rather pose the problem as a low-rank tensor recovery where we penalize some metric to penalise the rank of the tensor. This would in general lead to a more expressive model and reduce a tunable paramter from the system.

3. Scale the algorithm to very high dimensional data sets(for example 500) which is not possible in the current framework because of the limitations of the SPA algorithm

7. References

1. K. P. Murphy, Machine learning: a probabilistic perspective. MIT press, 2012.

2. N. Kargas, N. D. Sidiropoulos, and X. Fu, Tensors, learning, and Kolmogorov extensionfor finite-alphabet random vectors, IEEE Trans. Signal Process., vol. 66, no. 18, pp. 48544868, 2018.

3. A. Yeredor and M. Haardt, Maximum likelihood estimation of a lowrank probability mass tensor from partial observations, IEEE Signal Process. Lett., vol. 26, no. 10, pp. 15511555, Oct 2019

4. X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications, IEEE Signal Process. Mag., vol. 36, no. 2, pp. 5980, March 2019.

5. 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 Trans. Signal Process., vol. 65, no. 13, pp. 35513582, 2017.

6. S. Ibrahim, X. Fu, Recovering Joint Probability of Discrete Random Variables from Pairwise Marginals - https://arxiv.org/abs/2006.16912

7. Tensors and Probability: An Intriguing Union - N. Sidiropoulos, N. Kargas, X. Fu, https://sigport.org/sites/default/files/docs/KeynoteGlobalSIP2018-Sidiropoulos.pdf

8. S. Dasgupta. Experiments with random projection. Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI), 2000

9. S. Dasgupta. Learning probability distributions. Ph.D. dissertation, University of California at Berkeley, 2000

10. Finbarr O'Sullivan and Yudi Pawitan, Multidimensional Density Estimation by Tomography, Journal of the Royal Statistical Society. Series B (Methodological) , 1993, Vol. 55, No. 2 (1993), pp. 509-521