Multimodal Density Estimation from Random Linear Compressive Projections

Jian Vora

Department of Electrical Engineering, IIT Bombay

jianvora@iitb.ac.in

December 16, 2020

Overview

1. Introduction
2. Preliminaries
    - Mixture of Log-Concave Densities
    - Random Linear Projections
3. Random Projections of Log-Concave Densities
4. Problem Statement
    - Theoretical Guarantees
    - Extension to Non-Isotropic Vectors
5. Subspace Clustering
6. Learning Mixture of Gaussians
7. Empirical Results
    - Synthetic Data
    - Real World Data
8. Johnson Lindenstrauss Lemma

Problem Statement

The ever classical problem of given N random variables $X_1, X_2, ..., X_N$, and their realisations as samples, we need to estimate $p(X_1, X_2, ..., X_N)$

Lots of work in this domain like Gaussian mixture models, Bayesian Nets, Markov Random Fields, Variational Autoencoders ...

Can we do some transformation of the data to try and fit some simple estimators even though the density in the higher dimension is much more generic and expressive? If yes, what can be this tranformation?

Issues with Current Approaches

Using simple histogramming for discrete random variable is clearly not scalable to high-dimensional data as the number of free parameters increases exponential in N

Learning structure in graphical models is crucial and not trivial. AI they have to rely on approximation algorithms for inference

GMMs are not that expressive (say for modeling heavy tails) and the number of components increases very rapidly even for commonly occuring probability densities

Mixture Log-Concave Densities

Log of the density function is concave

Examples? Gaussian, Laplacian, Beta, uniform distribution defined over a convex set ...

Reasonable to assume data drawn from a mixture of these rich distributions which increases expressiveness such a heavier tails.

Mixture Log-Concave Densities

Use the model of mixture of log-concave densities throughout this talk

Consider a random vector $X \in \mathbb{R}^N$ which follows a mixture of log-concave densities $f_i(x)$ with K (finite) components subject to $\sum_i w_i = 1,\ w_i \geq 0$ and each f_i being log concave

$$f_X(x) = \sum_{i=1}^{i=K} w_i f_i(x)$$

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.

Used extensively in compressed sensing, dimensionality reduction, GMM learning

Random Projections of Log-Concave Densities

Random Projections of Log-Concave Densities are close to a Gaussian in a total variation sense!

Lemma 1: *Let X be an isotropic random vector $\in \mathbb{R}^n$ with a log-concave density. There exists a subset $\Theta \subseteq S^{n-1}$, with $\sigma_{n-1}(\Theta) \geq 1 - exp(-\sqrt{n})$ such that for any $\theta \in \Theta$ and any measurable set $A \subseteq \mathbb{R}$,*

$$|\mathbb{P}(X.\theta \in A) - \frac{1}{\sqrt{2\pi}} \int_A \exp(-s^2/2) ds| \leq \frac{C}{n^\alpha}$$

where $C, \alpha > 0$ are universal constants.

`[B. Klartag, Invent. Math.]`

Random Projections of Log-Concave Densities

A stronger result exists for pointwise closeness to a normal gaussian

Lemma 2: *Let X be an isotropic random vector in $\mathbb{R}^n$ with a log-concave density. Let $1 \leq l \leq n^{c_1}$ be an integer. Then there exists a subset $\epsilon \subseteq G_{n,l}$, with $\mu_{n,l}(E) \geq 1 - Cexp(-n^{c_2})$ such that for any $E \in \epsilon$, the following holds: Denote by f_E the density of the random vector $Proj_E(X)$. Then, for any $x \in E$ with $|x| \leq cn^{\alpha}$,*

$$|\frac{f_E(x)}{\phi(x)} - 1| \leq \frac{C}{n^{c_3}}$$

where, $C, c_1, c_2, c_3, \alpha > 0$ are universal constants. Here, $\phi(x) = (2\pi)^{-l/2} exp(-|x|^2/2)$ is the standard gaussian density in E.

`[B. Klartag, R. Eldan, J. Functional Analysis]`

Problem Statement

- As assumed earlier, data is drawn from a mixture of log-concave densities. Perform random projections and the density should be close to a Gaussian Mixture Model with a high probability
- Learn the parameters of the GMM on the subspace
- Comment on the higher dimensional estimates by invoking the Johnson Lindenstrauss Lemma

Done? There are certain problems we need to address...

- Proving theoretically that the projections are close to a gaussian mixture model
- Klartag's error bounds are only isotropic random vectors

We shall look at each of them one after the other

Theoretical Guarantees

Let $Y_1, Y_2, ..Y_K$ be random variables from the K component distributions respectively. Consider $\varepsilon \subseteq G_{D,d}$, then for some $E \subseteq \varepsilon$, $Proj_E(X) = \phi X$ for a fixed ϕ. Then for all $A \subseteq E$ which are measurable we have the following :

$$\mathbb{P}_X(\phi X \in A) = \mathbb{E}_X[\mathbb{I}(\phi X \in A)] = \int \mathbb{I}(\phi X \in A) f_X(x) dx$$

n the above $\mathbb{I}$ denotes the indicator function. Consider a random variable I which takes values $1, 2, ...K$ with $\mathbb{P}(I = i) = w_i$

$$f_X(x) = \sum_{i=1}^{i=K} P(I = i) f_X(x|I = i)$$

$$\mathbb{P}_X(\phi X \in A) = \int \mathbb{I}(\phi X \in A) \sum_{i=1}^{i=K} P(I = i) f_X(x|I = i) dx$$

$$= \sum_{i=1}^{i=K} P(I = i) \int \mathbb{I}(\phi X \in A) f_X(x|I = i) dx$$

Theoretical Guarantees

$$\int \mathbb{I}(\phi X \in A) f_X(x|I = i) = \mathbb{P}(\phi Y_i \in A)$$

$$\mathbb{P}_X(\phi X \in A) = \sum_{i=1}^{i=K} w_i \mathbb{P}(\phi Y_i \in A)$$

If $dim(E) < D^c$, for a certain vector Y following a log concave density, then for a certain gaussian random vector Z in the subspace E, and for some constant C, we have the following results:

$$\sup_{A \subseteq E} |\mathbb{P}(Proj_E(Y) \in A) - P(Z \in A)| \leq \frac{C}{D^c}$$

We can use the above for each Y_i as they are drawn for f_i which is a log concave density. Hence for gaussian vectors $Z_1, Z_2, ...Z_K$(with $Z_i \sim \mathcal{N}(\mu_i, \Sigma_i)$), then considering the following gaussian mixture density and a random vector $\Gamma \sim GMM$

Theoretical Guarantees

$$GMM(x) = \sum_{i=1}^{i=K} w_i \mathcal{N}(x; \mu_i, \Sigma_i)$$

$$\mathbb{P}(\Gamma \in A) = \int_A \sum_{i=1}^{i=K} w_i \mathcal{N}(x; \mu_i, \Sigma_i) = \sum_{i=1}^{i=K} w_i \int_A \mathcal{N}(x; \mu_i, \Sigma_i)$$

$$= \sum_{i=1}^{i=K} w_i \mathbb{P}(Z_i \in A)$$

$$\sup_{A \subseteq E} |\mathbb{P}_X(\phi X \in A) - \mathbb{P}(\Gamma \in A)| = \sup_{A \subseteq E} |\sum_{i=1}^{i=K} w_i \mathbb{P}(\phi Y_i \in A) - \sum_{i=1}^{i=K} w_i \mathbb{P}(Z_i \in A)|$$

$$\leq \sum_{i=1}^{i=K} w_i \sup_{A \subseteq E} |\mathbb{P}(\phi Y_i \in A) - \mathbb{P}(Z_i \in A)| \leq \sum_{i=1}^{i=K} w_i \frac{C_i}{D^c} = \frac{C^{'}}{D^c}$$

Extension to Non-Isotropic Vectors

Most data is clearly non isotropic

However, we can use an affine transformation for the same. Assuming 0 mean, if A is the covariance matrix and P is the projection operator, then our effective new projection operator $P' := A^{-1/2}PA^{1/2}$

The above transformation can blow up errors if we project along directions where the densities don't concentrate

Solution? Subspace Clustering
Generally high dimensional distributions are concentrated around low dimensional subspaces or manifolds. Cluster points and project only along directions where they concentrate.

Sparse Subspace Clustering(SSC)

Reasonable to assume that high-dimensional data is drawn from a union of subspaces or manifolds

The key idea is that, among the infinitely many possible representations of a data point in terms of other points, a sparse representation corresponds to selecting a few points from the same subspace.

Look into some algorithms to do the same

Algorithms for SSC

Consider we have a data matrix $X \in \mathbb{R}^{D\times N}$ where N is the number of samples and D is the dimensionality of the data. We use self-representation, i.e., for a particular datapoint x_i, we can write it as a linear combination of points belonging in the same subspace.

$$C^* = \text{argmin } ||C||_1, \text{s.t. } X = XC \text{ and } \text{diag}(C) = 0$$

where $C \in \mathbb{R}^{N\times N}$ is the representation matrix

1. **Spectral Clustering**:
   $W = C + C^T$ and then use spectral clustering techniques on W to get clusters of the data
2. **Elastic Net Subspace Clustering**: Solves the following optimization problem to find the representation matrix -

$$c_i^* = \text{argmin } \lambda||c_i||_1 + (1-\lambda)||c_i||_2^2, \text{s.t. } x_i = Xc_i \text{ and } c_{i,i} = 0$$

Algorithms for SSC

Orthogonal Matching Pursuit for SSC: Solves the same optimization problem as spectral clustering but used OMP for updates - computationally efficient and scalable to large datasets

Empirical Results:

Ambient Dimension : 100

Subspace Dimension : 15

Number of Subspaces in the Union : 10

Number of Samples : 10,000

Algorithm	Clustering Accuracy
k-means	33.2
Spectral Clustering	65.6
Elastic Net	89.3
SSC-OMP	92.6

Table: Basic comparision of various subspace clustering algorithms

Learning Mixture of Gaussians

Maximize the likelihood of having seen the data until convergence while **not converged**:

1. **E-Step**:

$$\gamma_k = p(z=k|x) = \frac{p(z=k)p(x|z=k)}{p(x)} = \frac{w_k \mathcal{N}(x;\mu_k,\Sigma_k)}{\sum_{i=1}^{i=K} w_i \mathcal{N}(x;\mu_i,\Sigma_i)}$$

1. **M-Step**:

$$\mu_k = \frac{\sum_{i=1}^{i=N} \gamma_k^{(n)} x^{(n)}}{N_k}, \Sigma_k = \frac{\sum_{i=1}^{i=N} \gamma_k^{(n)} (x^{(n)}-\mu_k)(x^{(n)}-\mu_k)^T}{N_k}$$

$$w_k = \frac{N_k}{N}, N_k = \sum_{n=1}^{n=N} \gamma_k^{(n)}$$

3. Evaluate the log-likelihood to check for convergence:

$$\log\ p(x|w,\mu,\Sigma) = \sum_{n=1}^{n=N} \log \sum_{i=1}^{i=K} w_i \mathcal{N}(x;\mu_i,\Sigma_i)$$

Synthetic Data

Example 1 :

Ambient dimension D : 50

Projected dimension d : 20

Mixture Components k : 4

The 4 components were a Gaussian, Laplacian, Beta and Uniform distribution. The elements of ϕ were randomly sampled from a standard normal gaussian. Number of samples were 10k. For each row of the table below, the parameters of the above distributions and the entries of ϕ were generated randomly.

Original Mixture Weights	Weights estimated by the GMM Algorithm
0.25, 0.25, 0.25, 0.25	0.249, 0.248, 0.249, 0.252
0.15, 0.5, 0.14, 0.2	0.156, 0.498, 0.146, 0.199
0.4, 0.3, 0.2, 0.1	0.401, 0.295, 0.202, 0.1
0.1, 0.7, 0.1, 0.1	0.1, 0.699, 0.1, 0.1

Table: Comparison of weights of the mixture with those predicted from the GMM

Synthetic Data

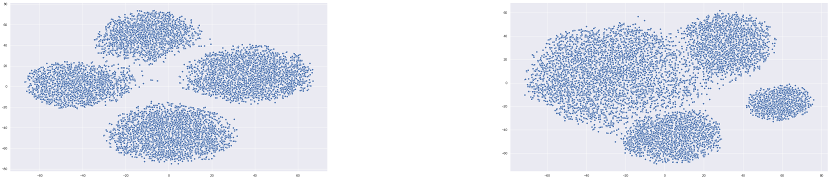

Figure: t-SNE embeddings on 2d for the 15d space which indicates 4 clusters which should be present in the higher dimension as well

Synthetic Data

Example 2:
Ambient dimension D : 150
Projected dimension d : 50
Mixture Components k : 3/6
The 3/6 components were a Gaussian, Laplacian, Uniform distribution.
Number of samples were 10k.

Original Mixture Weights	Weights estimated by the GMM Algorithm
0.173, 0.48, 0.346	0.172, 0.364, 0.46
0.658, 0.163, 0.179	0.655, 0.165, 0.179
0.218, 0.042, 0.262,	0.217, 0.051, 0.305,
0.176, 0.194, 0.107	0.176, 0.193, 0.057

Table: Comparison of weights of the mixture with those predicted from the GMM

Real World Data - MNIST

Dimensionality - 784
Projected Space Dimension - 500

Train Log Likelihood: -206.12
Test Log Likelihood: -211.47

We *decode* the latent space by sampling from the GMM fitted by using the basis pursuit algorithm as mnist images are sparse in their canonical basis

$$y = \Phi x + \eta$$

$$\text{recovery using the following : } \min \|x\|_1 \text{ s.t. } \|y - \Phi x\|_2 \leq \epsilon$$

Real World Data - MNIST

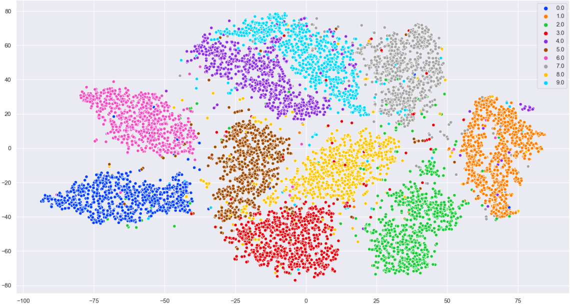

Figure: t-SNE embeddings of the projected space of the test samples where each color code indicates a digit class

Real World Data - MNIST

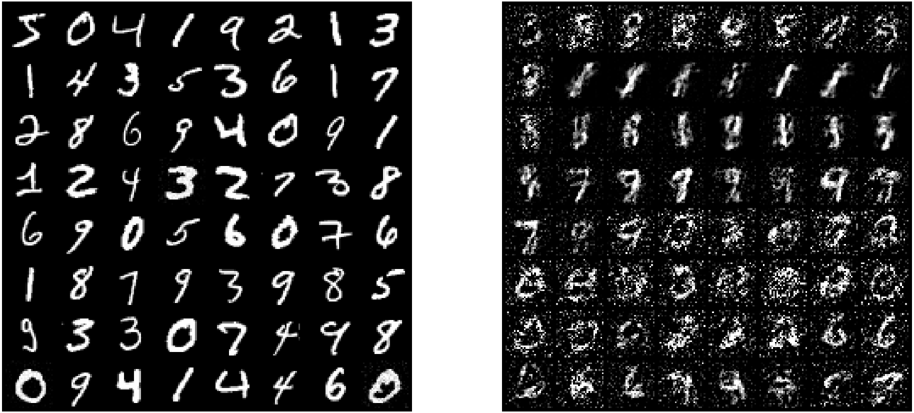

Figure: Groundtruth train image after CS recovery (left) and samples drawn from the fitted GMM after decoding (right)

MNIST Q-Q Plots

To compare whether the samples in the lower dimensional subspace are indeed Gaussian, we use the help of Q-Q plots. A straight line with a slope of 1 indicates that the distributions plotted along the axes being exactly equal to each other.

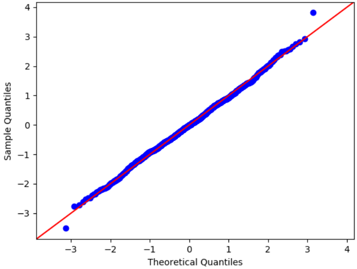

Figure: Q-Q plot for label 0

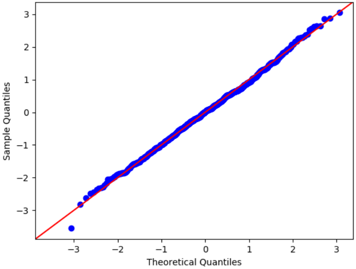

Figure: Q-Q plot for label 4

Real World Data - LFW

First transformation was to convert 5828 dimensional images to 700 dimensions using PCA. The final dimension of random projection was 300.

Train Log Likelihood: -2451.13
Test Log Likelihood: -2457.64

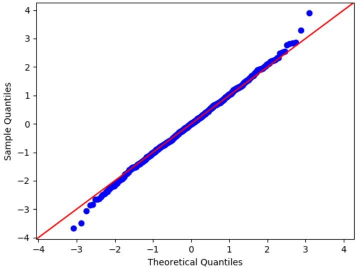

Figure: Q-Q plot for label 0

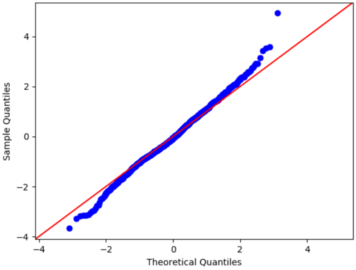

Figure: Q-Q plot for label 1

SSC After Projections

Saves computational cost, as subspaces will be preserved anyways

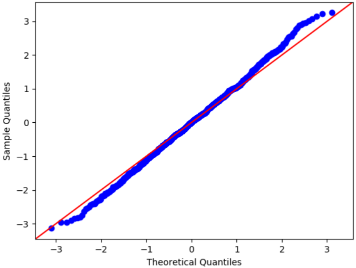

Figure: Q-Q plot for faces label 6

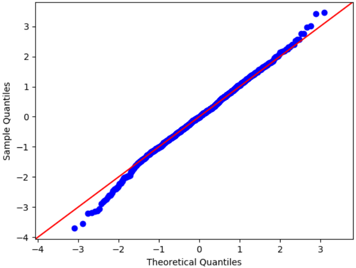

Figure: Q-Q plot for MNIST label 2

JL Lemma

Formally, given $0 < \epsilon < 1$, a set X of m points in $\mathbb{R}^N$ and a number $n > 8ln(m)/\epsilon^2$, there exists a linear map $f : \mathbb{R}^N \mapsto \mathbb{R}^n$ such that

$$(1-\epsilon)||u-v||^2 \leq ||f(u) - f(v)||^2 \leq (1+\epsilon)||u-v||^2$$

for all $u, v \in X$. Note that the dimension of the projected subspace n depends only on m which is the number of datapoints and not the actual dimension of the data N. Let f be a linear projection matrix Φ, and taking $v = 0$, we can restate the JL Lemma in the following manner:

$$(1+\epsilon)^{-1}\|\Phi u\|^2 \leq \|u\|^2 \leq (1-\epsilon)^{-1}\|\Phi u\|^2$$

with the probability of

$$Pr(\|\Phi u\|_2^2 \in [(1-\epsilon)\|u\|_2^2, (1+\epsilon)\|u\|_2^2]) \geq 1 - n^{-2}$$

Invoking the JL Lemma

Using the JL lemma we can estimate the clusters in the higher dimension with high probability and estimate their mean using empirical average. The covariance matrix shall follow as -

$$\Sigma_k = \sum_{i \in A} \frac{(X_i - \mu_k)(X_i - \mu_k)^T}{N}$$

Given that the lower dimensional projections are gaussian, the higher moments carry no extra information in the sense that they will only be functions of the first two moments.

Thus using the method of random projections for density estimation, we get the best fit gaussian for each log-concave density in the higher dimension using the maximum entropy principle.

The component weights can be retrieved exactly with high probability.

Possible Future Work

- Whether the assumption of mixture of log-concave densities can be relaxed to a more general distribution involving a combination of sum and product nodes?
- Whether a non-linear transformation $f(.)$ which preserves distances like a random matrix multiplication also induces a structure like a GMM on the projections? This would be helpful in analysing the latent spaces of current deep neural networks.

References

1. A central limit theorem for convex sets - B. Klartag, Invent. Math., Vol. 168, 91-131
2. Pointwise Estimates for Marginals of Convex Bodies - B. Klartag, R. Eldan, J. Functional Analysis, Vol. 254, Issue 8, (2008), 2275-2293
3. S. Dasgupta and A. Gupta. An elementary proof of a theorem of Johnson and Lindenstrauss. Random Structures and Algorithms, 22(1):60-65, 2003.
4. S. Dasgupta. Learning mixtures of Gaussians. Fortieth Annual IEEE Symposium on Foundations of Computer Science (FOCS), 1999.
5. Sparse Subspace Clustering: Algorithm, Theory, and Applications - Ehsan Elhamifar and René Vidal, IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(11):2765–2781, 2013.
6. Scalable Sparse Subspace Clustering by Orthogonal Matching Pursuit - C. You, D. Robinson, and R. Vidal, Arxiv, abs/1507.01238, 2015.

Questions?