Open Problem: Improper Learning of Mixtures of Gaussians

Elad Hazan EHAZAN@CS.PRINCETON.EDU

Roi Livni RLIVNI@CS.PRINCETON.EDU

Princeton University, Princeton, NJ

Editors: Sebastien Bubeck, Vianney Perchet and Philippe Rigollet

Abstract

We ask whether there exists an efficient unsupervised learning algorithm for mixture of Gaussians in the over-complete case (number of mixtures is larger than the dimension). The notion of learning is taken to be worst-case compression-based, to allow for improper learning.

1. Introduction

One of the most prominent and well studied paradigms of unsupervised learning is that of learning Mixture-Of-Gaussians (MOG). In this setting, a given distribution over data is assumed to be from a mixture distribution over normal random variables, and the goal is to identify the Gaussians and in turn provide a good clustering of the data.

The MOG model has long served as a tool for scientific discovery, see e.g. (Hardt, 2014) and references therein. Numerous algorithms have been developed over the years for MOG under various probabilistic assumptions. However, computationally MOG it is known to be a hard problem, requiring exponential sample complexity in the distributional setting (Kalai et al., 2012; Regev and Vijayaraghavan, 2017; Arora et al., 2005). The related worst-case problem of “k-means” is known to be NP-hard (Dasgupta, 2008).

This does not prohibit efficient improper learning. Numerous computationally hard supervised learning problems, such as Max-Cut, are learnable efficiently and without a significant penalty in sample complexity (Hazan et al., 2012). However, this requires a definition of unsupervised learning that is not probabilistic and allows for convex relaxation.

We thus focus on the definition of unsupervised learning in terms of compression with low reconstruction error, see formal definition and references in (Hazan and Ma, 2016).

Henceforth we describe an efficient improper learning algorithm for MOG in the under-complete case, and ask whether such an efficient algorithm exists in general.

2. Problem definition

We consider a pair of a class of compression functions $\mathcal{F} : \mathcal{X} \to \{0,1\}^k$ over a domain $\mathcal{X} = \mathbb{R}^d$, and a fixed reconstruction function $\rho_{\mathcal{F}}$. Given a distribution D over $\mathcal{X}$ we define the reconstruction error of a fixed $f \in \mathcal{F}$ as:

$$R(f; \rho_{\mathcal{F}}) = \mathop{\mathbb{E}}_{x \sim D} \|\rho_{\mathcal{F}}(f(x)) - x\|^2.$$

Throughout, if the reconstruction function is known, we will neglect the dependence of R in $\rho_{\mathcal{F}}$ and simply write $R(f)$ (it is possible to consider a compression scheme in which the reconstruction function is also a function over the sample, although for simplicity we ignore this option here).

Given a new pair $\mathcal{G} : \mathcal{X} \to \{0,1\}^{k'}$ and $\rho_{\mathcal{G}}$, we will say that an algorithm A learns $\mathcal{F}$ through $\mathcal{G}$ with sample complexity $m(\epsilon)$ if given a sample $S = \{x_1, \ldots, x_m\}$ drawn IID from some unknown distribution D, then with probability $2/3$ the algorithm A returns $g \in \mathcal{G}$ such that

$$R(g) \leq \min_{f \in \mathcal{F}} R(f) + \epsilon$$

The quantities of interest are the computational efficiency of the algorithm A, the dependence of the sample complexity m in ϵ and the Rademacher complexity of $\mathcal{F}$, and the compression factor which depends on k'. Note that if we don't consider k' , one can always use $k' = d \log 1/\epsilon$ bits of representation to return an ϵ–net whose reconstruction error is ϵ-competitive with any MOG model. However, we desire a k' parameter which depends polynomially on the hypothesis class learned $k' = \text{poly}(k, \log 1/\epsilon)$.

The Euclidean reconstruction error is related to the log-probability error in the probabilistic interpretation of the generative mixture of Gaussians model.

3. Learning MOG by PCA

In this section we show how to efficient (and improperly) learn the spherically symmetric version of MOG according to the definitions above, but only in the underconstrained case and with exponential blowup in compression size.

Let $X \in \mathbb{R}^{m \times d}$ be a data matrix sampled i.i.d from some arbitrary distribution $x \sim D$. We let $R = \max_{x \in X} \|x\|$.

Given a set of k centers of Gaussians, $k << d$, $\mu_1, ..., \mu_k$ (w.l.o.g. with norm at most one), minimize reconstruction error given by

$$\mathbb{E}_{x \sim D}[\min_i \|\mu_i - x\|^2]$$

We next provide a relaxation of the objective that may consider improper solutions. Specifically we allow several means to contribute to the reconstruction error, and consider the following objective:

$$\mathbb{E}_{x \sim D}[\min_{\|\alpha_x\|_1 \, \leq \, 1} \| \sum_j \alpha_j \mu_j - x\|^2]$$

Let $M \in \mathbb{R}^{d \times k}$ be the matrix of all μ vectors. Then we can further relax and write the optimization problem as

$$\min_M \mathbb{E}_{x \sim D}[\min_{\alpha_x \in \mathbb{R}^k} \|x - M\alpha_x\|^2]$$

This corresponds to an **encoding by vector α rather than by a single coordinate**. We can write the closed form solution to α_x as:

$$\arg\min_{\alpha_x'} \|x - M\alpha_x\|^2 = M^{-\dagger} x$$

where $M^{-\dagger}$ denotes the pseudo inverse, and the objective becomes

$$\min_{\alpha_x} \|x - M\alpha_x\|^2 = \|x - MM^{-\dagger}x\|^2$$

Thus, we're left with the optimization problem of

$$\min_M \mathbb{E}_{x\sim D}[\|x - MM^{\dagger}x\|^2]$$

which amounts to PCA.

Thus, by finding the k-PCA of the data, we can ensure reconstruction error less than the best k-MOG model, which satisfies our definition of improper unsupervised learning. PCA is also an efficient algorithm.

In terms of representation size, however, things are not as satisfying. To represent the encoding by PCA up to ϵ error, one needs $k \log \frac{R}{\epsilon}$ bits. This is in contrast to the $\log k$ bits required by k-MOG. This is also the reason why PCA only works up to $k = d$, i.e. the underconstrained case.

4. The Question

We offer 100$ for the solver of the following question: design a simple and efficient algorithm for k-MOG with only $\text{poly}(\log(k), \log\frac{1}{\epsilon})$ compression size, that works for the overconstrained case. Bi-criteria approximation algorithms for k-means may be a good place to start, such as in (Makarychev et al., 2015). Alternatively, prove an impossibility for the existence of any such *efficient* algorithm.

An additional 100$ is offered for a solution of the analogue of this question to the hypothesis class of general Gaussians, that may have non-identity convariance. Spectral auto-encoders may be a good place to start (Hazan and Ma, 2016).

References

Sanjeev Arora, Ravi Kannan, et al. Learning mixtures of separated nonspherical gaussians. *The Annals of Applied Probability*, 15(1A):69–92, 2005.

Sanjoy Dasgupta. *The hardness of k-means clustering*. Department of Computer Science and Engineering, University of California, San Diego, 2008.

Moritz Hardt. Blog post. http://blog.mrtz.org/2014/04/22/pearsons-polynomial.html, 2014. Published: 2014-04-22.

Elad Hazan and Tengyu Ma. A non-generative framework and convex relaxations for unsupervised learning. In *Advances in Neural Information Processing Systems*, pages 3306–3314, 2016.

Elad Hazan, Satyen Kale, and Shai Shalev-Shwartz. Near-optimal algorithms for online matrix prediction. In *Conference on Learning Theory*, pages 38–1, 2012.

Adam Tauman Kalai, Ankur Moitra, and Gregory Valiant. Disentangling gaussians. *Communications of the ACM*, 55(2):113–120, 2012.

Konstantin Makarychev, Yury Makarychev, Maxim Sviridenko, and Justin Ward. A bi-criteria approximation algorithm for k means. *arXiv preprint arXiv:1507.04227*, 2015.

Oded Regev and Aravindan Vijayaraghavan. On learning mixtures of well-separated gaussians. *arXiv preprint arXiv:1710.11592*, 2017.