CS777 Presentation - RREM

Archit Sharma and Amur Ghose

Analysis of the EM algorithm

April 28, 2018

Introduction - EM's general properties

Expectation Maximization is a pervasively used algorithm in probabilistic ML for doing point estimation in presence of latent variables. The monotonicity guarantee comes from its minorization-maximization nature. However, EM suffers from the general intialization troubles of non-convex problems.

- Convergence proof for Generalized EM (GEM) has been know since Wu (1983), but only in case of an unimodal distributions
- Balakrishnan et al. (2014) show convergence to local maxima under much more general conditions
- Not much work on convergence of EM under noisy data

Most "Robust EM" work is just robustness to initialization. We will qualitatively look at the hardness of doing EM, and also look at how to construct EM algorithms robust to noise (possibly adversarial).

Adversarial EM - a difficult task

To demonstrate how difficult adversarial EM is, we consider the case of recovering a mixture of Gaussians in one dimension. Assume that the generating Gaussians have zero variance (the support of the distribution is still $\mathbb{R}$). Suppose, we generate N points generated (with an unknown apriori distribution). We hand them over to an adversary who can change m points, and we wish to recover at least L clusters accurately. Since the variance is zero, recovering a cluster is equivalent to finding its centroid.

A simple algorithm

- Sort clusters by frequency.
- Choose top K clusters.

The case of L=K

Consider doing the above for $L = K$, and suppose that cluster i was generated with c_i points. If the smallest cluster had size c', the adversary would need $c'/2$ (in the worst case where our algorithm chooses the adversarially generated cluster) points to throw off our clustering by at least one cluster, and we would not be able to do anything about it. This already is very prohibitive.

The Algorithm for Adversary in Zero-Variance Case

Given the algorithm is based on sorting clusters based on frequency, the adversary would like to maximize the number of generated clusters in the top K sorted clusters of the algorithm.

- Sort clusters by frequency. Binary search for the number of clusters we can corrupt with the allocated budget.
- Choose the smallest $K/2$ clusters in the sorted array. If the minimum number of points to "overpower" them is $\leq m$, search between $K/2+1$ and K, else 1 and $K/2-1$. Repeat till convergence.
- A "histogram cutting algorithm" for determining the minimum number of points to required by the adversary to "overpower" smallest k algorithms.

Problem Formulation

Given a data matrix $X \in \mathbb{R}^{N \times d}$, with $\leq \eta N$ adversarially corrupted points, we want to create K clusters by using Gaussian Mixture Modelling. In particular, we are interested in recovering the cluster means as accurately as possible. We will proceed to obtain them using maximum likelihood maximization, with some modification to the EM algorithm to make them robust.

Methods to boost EM - Likelihood Weighing/Thresholding

General Intuition: Outliers will have low likelihoods.

- In addition to posterior probability weight for every data point in the likelihood function, we additionally added likelihood based weighing of the data points. The weights are some function of the likelihood, such as the sigmoid of the likelihood.
- Somewhat better than this method is likelihood thresholding. In every M-step we allow the EM algorithm to discard up to ηN number of points based on the likelihood so far, after a "warm start"-like phase where we calculate our model based on all points at once.

Some rough analogies between robust EM and other optimization

- Heuristic EM methods that weigh points by their likelihoods are common. They usually lack proofs of convergence but perform well in practice. Can be see as the equivalent of IRLS
- Similarly, any method of excluding outliers can be seen as an analogue to thresholding, such as IHT

Our chosen method - Ridiculously Robust EM

General Intuition: Treat the set of outliers as an extra cluster of point.
In this method, we have an extra cluster, with a uniform probability likelihood over any points in its support. For example, this could be the uniform distribution over a large ball B. This is meant to trap outliers and we will output only the "real" K clusters.

Solutions

Index the extra cluster by 0. The expected complete likelihood to be maximized is

$$\mathcal{L}(\pi, \mu, \sigma) = \sum_{i=1}^{N} \sum_{j=0}^{K} \mathbb{E}[z_{ij} = 1] \Big(\log p(x_i | \mu_j, \sigma_j) + \log p(z_{ij} = 1) \Big) \tag{1}$$

Here, $\log p(x_i | \mu_0, \sigma_0) = 0$ for all x_i. Hence, the only additional parameter to be learnt is π_0. This will fail, unless we introduce some constraint on π_0. A natural constraint is $\pi_0 \leq \eta$ (in addition to $\sum_{j=0}^{K} \pi_j = 1$).

Empirical results

Our metric for comparison is the sorted $L2$ loss between vectors. That is, given two vectors a, b we define a' as the vector with indices of a permuted, and b' as the same for b. It is trivial to show that if we seek to minimize

$$||a' - b'||$$

Then a' is the sorted permutation of a and b' the sorted permutation (by the rearrangement inequality). This metric is a natural one to check the convergence since the centroids may be recovered correctly but come out permuted (in one dimension).

L2 loss values for different EM varieties

Our framework uses $K = 3$ GMMs with means of $1, 10, -5$. For all clusters we use the same σ. There is a modest improvement with likelihood thresholding in terms of Gaussian noise of unit variance, that grows rapidly as we choose $\eta = 0.05$ fraction of points at random and corrupt them with uniform noise in a high range, such as $[16, 17]$. The robust algorithm is seen to discard these points correctly.

Table 1: L2 results on 3-GMM

Algo	Zero mean	Uniform noise	Non-zero-mean Gaussian
EM	6.78	18.17	14.72
Threshold	3.84	16.52	11.68
RREM	7.45	6.822	8.87

The mean for the Gaussian used above was 4.