An EM-free gradient descent approach for modeling Mixture of Factor Analyzers : A Short Paper

Siva Rajesh Kasa

DISA, School of Computing, NUS

E-mail: kasa@u.nus.edu

Reproducible code available @: upon email request

Abstract. In this paper, we show how Automatic Differentiation can be used for inference with Mixture of Factor Analyzers and advantages of doing so. We also discuss how this method also has all the properties such as monotonic increase in likelihood and convergence to a local optimum.

Keywords: Constrained Optimization, Mixture of Factor Analyzers, Automatic Differentiation

1. Introduction

Finite mixture distributions are well-known in statistical modeling because they bring the flexibility of non-parametric models while preserving the strong mathematical proprieties of parametric models. In this paper, we extend automatic differentiation to mixture factor analyzers (MFA), a classic high-dimensional modeling tool. According to MFA model, we assume that a sample of observations has been drawn from different populations, whose latent structures are modeled using low-dimensional individual factors. The aim is to decompose the sample into its mixture components, which are usually modeled using a multivariate Gaussian distribution, and to estimate parameters. The assumption of component-wise normality, besides its convenient expression in a closed-form for multi-variate distributions, also allows to employ the EM algorithm for the ML estimation of the parameters. However, in the recent past, the rise in automatic differentiation tools available allows us to do inference using a gradient based

approach (Auto-MFA). Our contributions in this paper, **a) we show how our Auto-MFA maximizes the likelihood better compared to EM-based MFA, because of availability of second order derivatives (such as hessian matrix) b) we show our method Auto-MFA is robust to high-dimensional settings (n $\leq$ p) whereas EM-based MFA fails.**

2. Existing Methods

For multivariate data of a continuous nature, a major chunk of past literature has focussed on the use of multivariate normal components, because of their closed form expression. Within the Gaussian Mixture Model (GMM) -based approach to density estimation and clustering, the density of the p-dimensional random variable $\mathbf{X}$ of interest is modelled as a mixture of a number, say G, of multivariate normal densities in some unknown proportions $\pi_1, \ldots \pi_G$. That is, each data point is taken to be a realization of the mixture probability density function,

$$f(\mathbf{x};\theta) = \sum_{g=1}^{G} \pi_g \phi_p(\mathbf{x};\mu_g,\Sigma_g) \tag{1}$$

where $\phi_d(\mathbf{x};\mu,\Sigma)$ denotes the d-variate normal density function with mean μ and covariance matrix Σ. Here the vector $\theta_{GM}(p,G)$ of unknown parameters consists of the $(G-1)$ mixing proportions π_g, the $G \times p$ elements of the component means μ_g, and the $\frac{1}{2}Gp(p+1)$ distinct elements of the component-covariance matrices Σ_g. Therefore, the G-component normal mixture model (1) with unrestricted component-covariance matrices is a highly parameterized model. We need some way to parsimoniously specify the matrices Σ_g, because they requires $O(p^2)$ parameters. Among the various proposals for dimensionality reduction, we demonstrate our method here by considering Mixtures of Factor Analyzers (MFA), proposed by Ghahramani and Hilton (1997) and developed further that by McLachlan and Peel (2000).

This MFA model allows to explain data by explicitly modeling correlations between variables in multivariate observations. It postulates a finite mixture of linear sub-models for the distribution of the full observation vector $\mathbf{X}$, given the (unobservable) factors $\mathbf{U}$. That is one can provide a local dimensionality reduction method by assuming that the distribution of the observation $\mathbf{X}_i$ can be given as

$$\mathbf{X}_i = \mu_g + \Lambda_g \mathbf{U}_{ig} + \mathbf{e}_{ig} \quad \text{with probability} \quad \pi_g \, (g = 1, \ldots, G) \quad \text{for } i = 1, \ldots, n, \tag{2}$$

where Λ_g is a $p \times q$ matrix of *factor loadings*, the *factors* $\mathbf{U}_{1g}, \ldots, \mathbf{U}_{ng}$ are $\mathcal{N}(\mathbf{0}, \mathbf{I}_q)$ distributed independently of the *errors* $\mathbf{e}_{ig}$, which are independently $\mathcal{N}(\mathbf{0}, \Psi_g)$ distributed, and Ψ_g is a $p \times p$ diagonal matrix $(g = 1, \ldots, G)$. We suppose that $q < p$, which means that q unobservable factors are jointly explaining the p observable features of the statistical units. Under these assumptions, the mixture of factor analyzers model is given by (1), where the g-th component-covariance matrix Σ_g has the form

$$\Sigma_g = \Lambda_g \Lambda_g' + \Psi_g \quad (g = 1, \ldots, G). \tag{3}$$

Note that this model is a superset of Gaussian Mixture Model and single latent factor analyzer model, essentially bridging dimensionality reduction and mixture models. **Therefore, our method can be applied to special cases such as - a) probabilistic principal component analysis (PPCA) model (Tipping and Bishop, 1999) which is a special case of the factor analysis model because it assumes that the distribution of the error term is isotropic and b) parsimonious Gaussian mixture model proposed by McNicholas and Murphy (2008).**

Given, $x_1, x_2, .., x_n$ i.i.d observations, the likelihood of the MFA model is given by

$$\mathscr{L}(x) = \sum_{i}^{n} \sum_{g=1}^{G} \frac{\pi_g}{(2\pi)^{p/2} |\Lambda_g \Lambda_g' + \Psi_g|^{\frac{1}{2}}} \mathrm{x} \exp\{-\frac{1}{2} (\mathbf{x}_i - \mu_g)' (\Lambda_g \Lambda_g' + \Psi_g)^{-1} (\mathbf{x}_i - \mu_g)\} \tag{4}$$

Maximizing the above likelihood (eqn 4) has been done by Expectation Maximization (EM) (Ghahramani and Hilton, 1997; McLachlan and Peel, 2000) or its variants such as Alternate Expectation Conditional Maximization (AECM) algorithm (McNicholas and Murphy, 2008). There are two advantages in using algorithms based on EM or its variants - a) Positive Semi-Definiteness (PSD) of the estimates of Σ_g will be maintained by construction b) the estimates of mixture-proportions π_g will add up to 1 without any need for Lagrange Multipliers.

3. Our Method and results

Automatic Differentiators can provide the exact gradients of equation 4. Please refer to (Baydin et al., 2018) for more details on the Automatic Differentiators. In order to preserve the PSD of Σ_g and constraint on $\sum \pi_g = 1$, we use the following simple tricks (Maclaurin, 2016):

(a) We write the variance term of error $\Psi_g = \psi_g \psi_g^T$ where ψ is a diagonal matrix that contains non-zero terms as $\sqrt{\Psi_{g_{ii}}}$. Now we take the gradients of the loglikelihood $\mathscr{L}(x)$ with

respect to ψ_g and update according $\psi_g := \psi_g + \alpha \frac{\partial \mathscr{L}}{\partial \psi_g}$. Here α is the learning rate. This way the PSD of $\Sigma_g = \Lambda_g \Lambda_g' + \Psi_g$ is always preserved through out all the updated values.

(b) The constraint $\sum \pi_g = 1$ is tackled considering the log proportions trick, using the log-sumexp trick (Robert, 2014). We start with unbounded α_g's as the log-proportions i.e. $\log \pi_g = \alpha_g - \log(\sum_i e^{\alpha_i})$. Note that, we need not impose any constraints on α_g as final computation of π_g automatically leads to normalization, because $\pi_g = \frac{\alpha_g}{\sum_i e^{\alpha_i}}$. Therefore, we can update $\alpha_g := \alpha_g + \frac{\partial l}{\partial \alpha_g}$ without any further need for Lagrange multipliers.

3.1. Data and Results

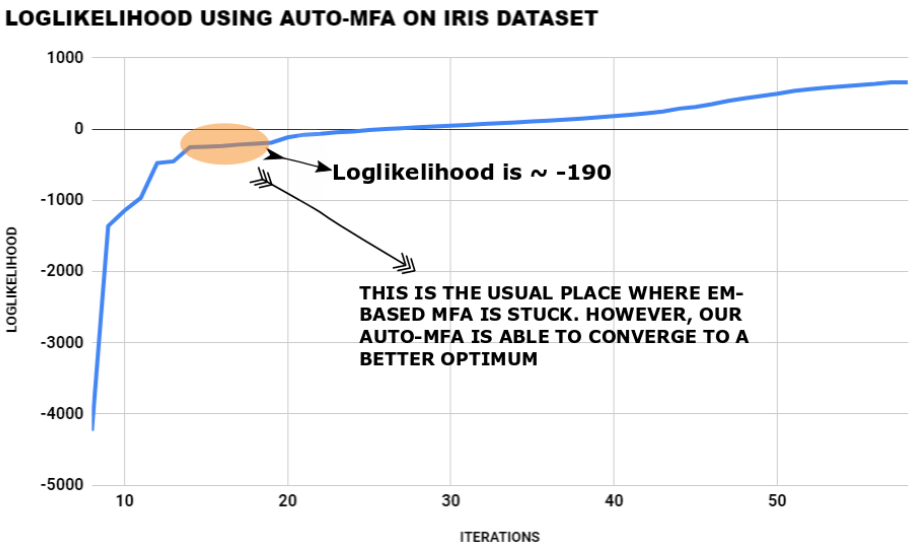

Figure 1. On the IRIS dataset, Auto-MFA converges to a better optimum compared an EM-based MFA

We considered the IRIS dataset from Fisher (1936). It contains sepal length, sepal width, petal length, petal width and class of the 150 different flowers. For the MFA inference using EM, we use the `EMMixmfa` and `FactMixAnalysis` R packages. `EMMixmfa` package contains an option for implementing the case considered in our equation 4 i.e. all the covariance matrices Σ_b and Ψ_g are different for each of the component. Moreover, we consider the most generalized case where Ψ_g is not isotropic.

We use `iris` dataset to compare our algorithm Auto-MFA vs EM-based MFA. Because we have access to second order information such as the hessian matrix, we can use Newton-CG method in Auto-MFA. We run both the algorithms with ten different random initializations. The best of loglikelihood of Auto-MFA was 656 while that of EM-MFA is -180 (Figure 1). However, as expected, the average runtime execution was slower using Auto-MFA (20 seconds) compared to EM-based MFA (2 seconds). This is because run time of gradient-based approach usually depend on learning rate and lower learning rate can lead to higher run time. Moreover, higher runtime can also be attributed to computing the second-order derivatives (hessian matrix). The trade-off for higher runtime is convergence to a better local optimum as evident in Figure 1. Further more, the lower learning rate ensures that the likelihood increases monotonically at every step.

Moreover, Auto-MFA can even work on high dimensional data ($n \leq p$) whereas a traditional EM-based MFA fails in this case because EM-based MFA involves matrix inversion steps and in high-dimensional settings, this matrix is not full rank. To illustrate this, we compare Auto-MFA with EM-based MFA on the first 30 entries in `iris` dataset. While EM-based MFA fails to run on this dataset, our Auto-MFA converges to a local optimum (loglikelihood = -77, best among 10 random initializations).

References

Baydin, A. G., Pearlmutter, B. A., Radul, A. A. and Siskind, J. M. (2018) Automatic differentiation in machine learning: a survey. *Journal of Marchine Learning Research*, **18**, 1–43.

Fisher, R. A. (1936) The use of multiple measurements in taxonomic problems. *Annals of eugenics*, **7**, 179–188.

Ghahramani, Z. and Hilton, G. (1997) The em algorithm for mixture of factor analyzers. *Technical Report CRG-TR-96-1*.

Maclaurin, D. (2016) *Modeling, inference and optimization with composable differentiable procedures*. Ph.D. thesis.

McLachlan, G. J. and Peel, D. (2000) *Finite Mixture Models*. New York: John Wiley & Sons.

McNicholas, P. and Murphy, T. (2008) Parsimonious gaussian mixture models. *Statistics and Computing*, **18**, 285–296.

Robert, C. (2014) Machine learning, a probabilistic perspective.

Tipping, M. E. and Bishop, C. M. (1999) Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **61**, 611–622.