Bag-of-components: an online algorithm for batch learning of mixture models

Olivier Schwander   Frank Nielsen

Université Pierre et Marie Curie, Paris, France
École polytechnique, Palaiseau, France

October 29, 2015

Exponential families

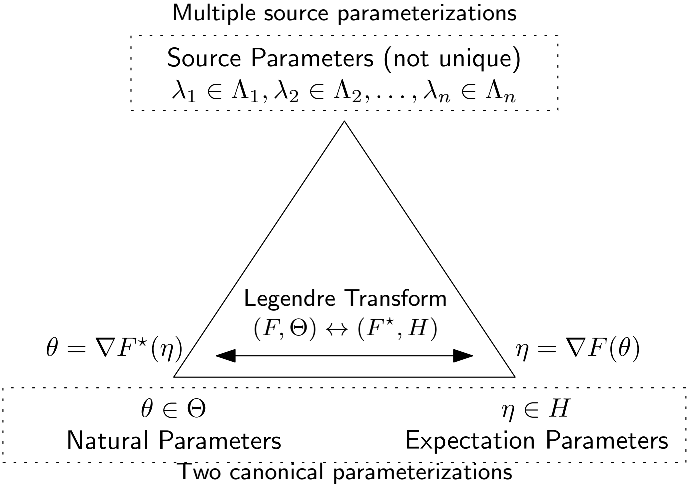
Definition

$$p(x;\lambda) = p_F(x;\theta) = \exp\left(\langle t(x)|\theta\rangle - F(\theta) + k(x)\right)$$

- λ source parameter
- $t(x)$ sufficient statistic
- θ natural parameter
- $F(\theta)$ log-normalizer
- $k(x)$ carrier measure

F is a stricly convex and differentiable function
$\langle\cdot|\cdot\rangle$ is a scalar product

Multiple parameterizations: dual parameter spaces



Bregman divergences

Definition and properties

$$B_F(x \| y) = F(x) - F(y) - \langle x - y, \nabla F(y) \rangle$$

- F is a stricly convex and differentiable function
- No symmetry!

Contains a lot of common divergences

- Squared Euclidean, Mahalanobis, Kullback-Leibler, Itakura-Saito. . .

Bregman centroids

Left-sided centroid

$$\min_c \sum_i \omega_i B_F(c \| x_i)$$

Right-sided centroid

$$\min_c \sum_i \omega_i B_F(x_i \| c)$$

Closed-form

$$c^L = \nabla F^* \left(\sum_i \omega_i \nabla F(x_i) \right)$$

$$c^R = \sum_i \omega_i x_i$$

Link with exponential families

[Banerjee 2005]

Bijection with exponential families

$$\log p_F(x|\theta) = -B_{F^*}\left(t(x) \| \eta\right) + F^*(t(x)) + k(x)$$

Kullback-Leibler between exponential families

- between members of the *same* exponential family

$$KL(p_F(x,\theta_1), p_F(x,\theta_2)) = B_F\left(\theta_2 \| \theta_1\right) = B_{F^\star}\left(\eta_1 \| \eta_2\right)$$

Kullback-Leibler centroids

- In closed-form through the Bregman divergence

Maximum likelihood estimator

A Bregman centroid

$$\begin{aligned}
\hat{\eta} &= \arg\max_{\eta} \sum_i \log p_F(x_i, \eta) \\
&= \arg\min_{\eta} \sum_i B_{F^*}\left(t(x_i) \| \eta\right) \underbrace{-F^*(t(x_i)) - k(x_i)}_{\text{does not depend on } \eta} \\
&= \arg\min_{\eta} \sum_i B_{F^*}\left(t(x_i) \| \eta\right) \\
&= \sum_i t(x_i)
\end{aligned}$$

And $\hat{\theta} = \nabla F^{\star}(\hat{\eta})$

Mixtures of exponential families

$$m(x;\omega,\theta) = \sum_{1\leq i\leq k} \omega_i p_F(x;\theta_i)$$

Fixed

- Family of the components P_F
- Number of components k (model selection techniques to choose)

Parameters

- Weights $\sum_i \omega_i = 1$
- Component parameters θ_i

Learning a mixture

- Input: observations $x_1, \dots, x_N$
- Output: ω_i and θ_i

Bregman Soft Clustering: EM for exponential families

[Banerjee 2005]

E-step

$$p(i,j) = \frac{\omega_j p_F(x_i,\theta_j)}{m(x_i)}$$

M-step

$$\begin{aligned}
\eta_j &= \arg\max_{\eta} \sum_i p(i,j) \log p_F(x_i,\theta_j) \\
&= \arg\min_{\eta} \sum_i p(i,j) \left(B_{F^*}\left(t(x_i) \| \eta\right) \underbrace{- F^*(t(x_i)) - k(x_i)}_{\text{does not depend on } \eta} \right) \\
&= \sum_i \frac{p(i,j)}{\sum_u p(u,j)} t(x_u)
\end{aligned}$$

Joint estimation of mixture models

Exploit shared information between multiple pointsets

- to improve quality
- to improve speed

Inspiration

- Dictionary methods
- Transfer learning

Efficient algorithms

- Building
- Comparing

Co-Mixtures

Sharing components of all the mixtures

$$m_1(x|\omega^{(1)}, \eta) = \sum_{i=1}^{k} \omega_i^{(1)} p_F(x|\,\eta_j)$$

$$\ldots$$

$$m_S(x|\omega^{(S)}, \eta) = \sum_{i=1}^{k} \omega_i^{(S)} p_F(x|\,\eta_j)$$

- Same $\eta_1 \ldots \eta_k$ everywhere
- Different weights $\omega^{(l)}$

co-Expectation-Maximization

Maximize the mean of the likelihoods on each mixtures

E-step

- A posterior matrix for each dataset

$$p^{(l)}(i,j) = \frac{\omega_j^{(l)} p_F(x_i, \theta_j)}{m(x_i^{(l)} | \omega^{(l)}, \eta)}$$

M-step

- Maximization on each dataset

$$\eta_j^{(l)} = \sum_i \frac{p(i,j)}{\sum_u p^{(l)}(u,j)} t(x_u^{(l)})$$

- Aggregation

$$\eta_j = \frac{1}{S} \sum_{l=1}^{S} \eta_j^{(l)}$$

Variational approximation of Kullback-Leibler

[Hershey Olsen 2007]

$$\widetilde{\mathrm{KL}}_{\mathrm{Variationnal}}(m_1, m_2) = \sum_{i=1}^{K} \omega_i^{(1)} \log \frac{\sum_j \omega_j^{(1)} e^{-\mathrm{KL}(p_F(\cdot;\theta_i)\|p_F(\cdot;\theta_j))}}{\sum_j \omega_j^{(2)} e^{-\mathrm{KL}(p_F(\cdot;\theta_i)\|p_F(\cdot;\theta_j))}}$$

With shared parameters

- Precompute $D_{ij} = e^{-\mathrm{KL}(p_F(\cdot|\eta_i), p_F(\cdot|\eta_j))}$

Fast version

$$\mathrm{KL}_{\mathrm{var}}(m_1 \| m_2) = \sum_i \omega_i^{(1)} \log \frac{\sum_j \omega_j^{(1)} e^{-D_{ij}}}{\sum_j \omega_j^{(2)} e^{-D_{ij}}}$$

co-Segmentation

Segmentation from 5D RGBxy mixtures

Original

EM

Co-EM

Transfer learning

Increase the quality of one particular mixture of interest

- First image: only 1% of the points
- Two other images: full set of points

- Not enough points for EM

Bag of Components

Training step

- Comix on some training set
- Keep the parameters
- Costly but offline

$$\mathcal{D} = \{\theta_1, \ldots, \theta_K\}$$

Online learning of mixtures

- For a new pointset
- For each observation arriving:

$$\arg\max_{\theta \in \mathcal{D}} p_F(x_j, \theta) \quad \text{or} \quad \arg\min_{\theta \in \mathcal{D}} B_F(t(x_j), \theta)$$

Nearest neighbor search

Naive version

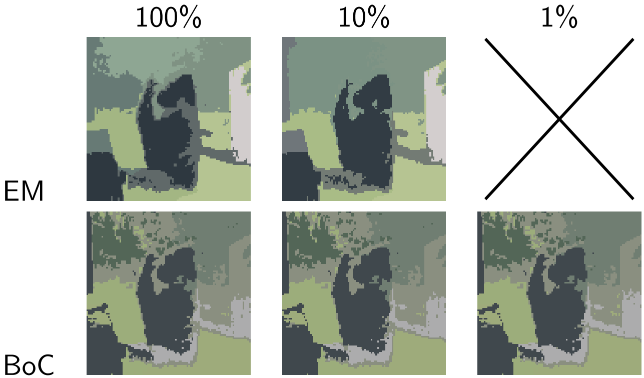
- Linear search
- $O(\textit{number of samples} \times \textit{number of components})$
- Same order of magnitude as one step of EM

Improvement

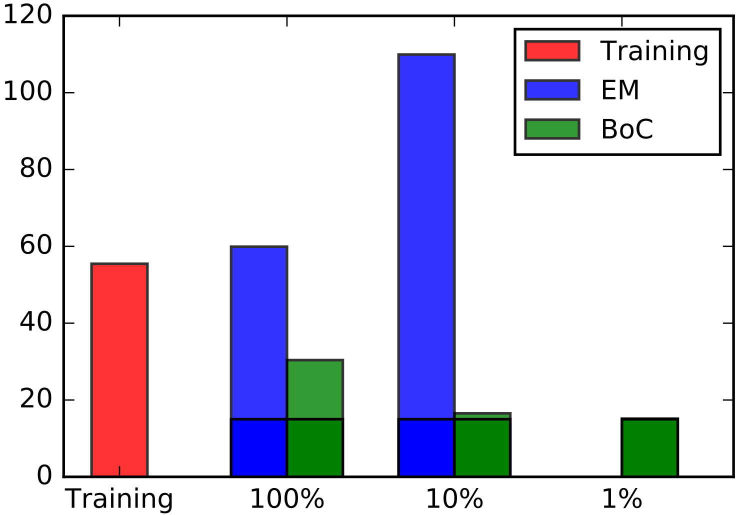
- Computational Bregman Geometry to speed-up the search
- Bregman Ball Trees
- Hierarchical clustering
- Approximate nearest neighbor

Image segmentation

Segmentation on a random subset of the pixels



Computation times



Summary

Comix

- Mixtures with shared components
- Compact description of a lot of mixtures
- Fast KL approximations
- Dictionary-like methods

Bag of Components

- Online method
- Predictable time (no iteration)
- Works with only a few points
- Fast