Gaussian Mixture Model

Introduction

- For a recent review , see M. Kuhn and E. Feigelsen arXiv:1711.11101

Likelihood of a datum x_i for a Gaussian mixture model is given by:

$$p(x_i|\theta) = \sum_{j=1}^{M} \alpha_j \mathcal{N}(\mu_j \sigma_j) \qquad \sum_{j=1}^{M} \alpha_j = 1$$

Log likelihood given by :

$$\ln L = \sum_{i=1}^{N} \ln \left[\sum_{i=1}^{M} \alpha_j \mathcal{N}(\mu_j, \sigma_j) \right]$$

GMM (Contd)

- Maximization of $\ln L$ can be trivially done using Expectation-Maximization Algorithm (Dempster, Laird, Rubin 1977)

- The optimum number of components can be computed using model comparison techniques.

- Generalization of GMM to account for errors is known as `` Extreme Deconvolution'' in astrophysics literature (arXiv:0905.2979). For an application of this see arXiv:2206.02751

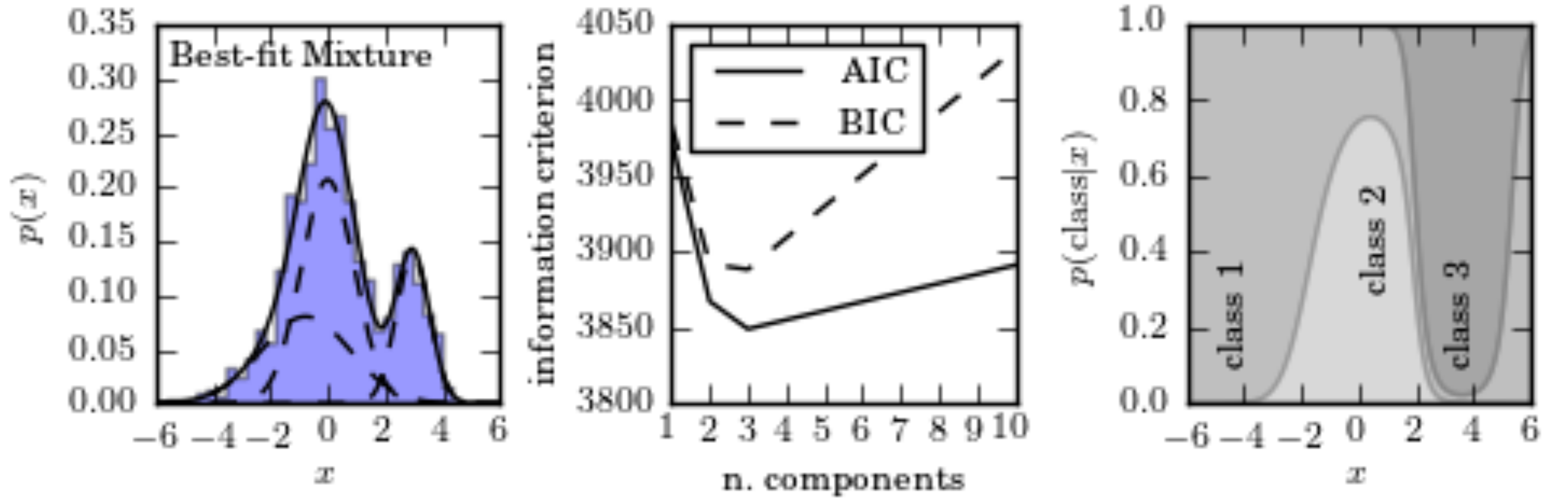
- GMM can be generalized to other mixture models, but you cannot use E-M algorithm for maximization (also other variants , eg. Dirichlet mixture models) Routines available in R to do mixture models of t-distributions etc (see arXiv:1910.08968)

GMM in Python

```
>>> import numpy as np
>>> from sklearn.mixture import GaussianMixture (GMM deprecated after v0.17)
>>> X=np.random.normal(size=(100,1))
>>> model=GaussianMixture(2)
>>> model.fit(X)
   GaussianMixture(covariance_type='diag', init_params='wmc', min_covar=0.001,
  n_components=2, n_init=1, n_iter=100, params='wmc', random_state=None, thresh=None, tol=0.001, verbose=0)
>>> model.means_
array([[-0.44802848],[ 0.53396065]])
```

More detailed documentation in sklearn

Example from AstroML (Fig. 4.2)



Example of a one-dimensional Gaussian mixture model with three components. The left panel shows a histogram of the data, along with the best-fit model for a mixture with three components. The center panel shows the model selection criteria AIC (see Section 4.3) and BIC (see Section 5.4) as a function of the number of components. Both are minimized for a three-component model. The right panel shows the probability that a given point is drawn from each class as a function of its position. For a given x value, the vertical extent of each region is proportional to that probability. Note that extreme values are most likely to belong to class 1.

Example from Astro Literature

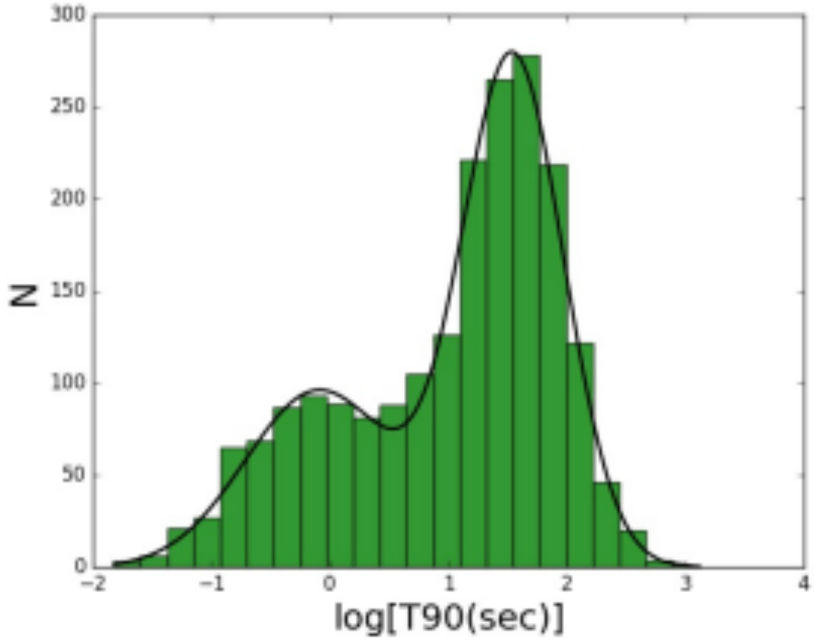


Fig. 1 A fit for the 2-component model for BATSE GRBs. Details of the fits can be found in Table 1.

Soham Kulkarni & SD (arXiv:1612.08235)