1 EM Algorithm

1.1 Problem Definition

Given the signal $y \in \mathrm{R}^n$ for a given predictor vector $x \in \mathrm{R}^n$ and a signal model $f(x, \theta)$ find the optimal parameters θ_{opt} minimizing some loss L, in other words $\theta_{\mathrm{opt}} = \arg\min_{\theta \in \Theta} \mathrm{L}(y, f(x, \theta))$. The signal model is either an superposition of Voigt profiles or pseudo Voigt profiles.

1.1.1 Signal Model

Through domain knowledge the true model of y is assumed to be a superposition of Voigt profiles with an linear background. For computational reasons a super position of Pseudo Voigt Profiles is assumed:

$$f(x, \theta) = b_\theta + a_\theta x + \sum_{i=1}^{k_p} \mathrm{amp}_\theta \texttt{PseudoVoigt}(x, \mathrm{pos}_\theta, \eta_\theta, \mathrm{f}_\theta) \tag{1}$$

Definition of a pseudo Voigt and conversion between a Voigt and pseudo Voigt profile is done as described on wikipedia (todo replace with proper source).

1.2 Connection to EM

An unnatural connection between gaussian EM and this problem can be forced by interpreting $\tilde{y} = y - \min_i y_i \geq 0$ as an normalized histogram. Let M be an arbitrary integer, interpreted as the number of total samples then $M\tilde{y}_i$ is the number of fictive samples at x_i. Thus if we model the probability of sampling x_i to be $p(x_i) = n(\theta) f_+(x_i, \theta) = n(\theta) \max(0, f(x_i, \theta))$, where $n(\theta)$ is some normalizing constant that depends on θ the log likelihood of sampling the histogram y_i under the model $p(x_i)$ becomes

$$\begin{aligned} \log(\prod_{i=1}^{N} p(x_i)^{M\tilde{y}_i}) &= \sum_{i=1}^{N} M\tilde{y}_i \log(p(x_i)) && (2) \\ &= \sum_{i=1}^{N} M\tilde{y}_i \log(n(\theta) f_+(x_i, \theta)) && (3) \\ &= M \sum_{i=1}^{N} \tilde{y}_i \log\left(\frac{f_+(x_i, \theta)}{\sum_{i=1}^{N} f_+(x_i, \theta)}\right) && (4) \end{aligned}$$

maximizing this loglikelihood is thus equivalent to minimizing

$$L = -\sum_{i=1}^{N} \tilde{y}_i \log\left(\frac{f_+(x_i, \theta)}{\sum_{i=1}^{N} f_+(x_i, \theta)}\right) \tag{5}$$

The EM-Algorithm introduced below will maximize the loglikelihood in (2). Or equivalently minimize (5). This is clearly not the least squares loss minimized by nlr, however it is much easier and faster to minimize. This also defines the loss refereed abstractly in subsection 1.1.

Notes: Only works for uniform sampling of x right now.

2 Minimization of the Loss

For this we will first rewrite $f(x,\theta)$ as a finite mixture of some pdfs of some components $c_j(x)$.

$$n(\theta)f(x,\theta) = \sum_{j=1}^{k_p+1} \pi_{\theta,j} c_{\theta,j}(x) \tag{6}$$

in particular, $c_{\theta,1}(x_i) = \frac{1}{\sum_{i=j}^{N} \max(0, b+ax_j)} \max(0, b+ax_i)$ and for $j > 1$ one finds $c_{\theta,j}(x_i) = \texttt{PseudoVoigt}(x_i, \mathrm{pos}_{\theta,j}, \eta_{\theta,j}, \mathrm{f}_{\theta,j})$ and $\pi_{\theta,j} \in \mathrm{R}$. Inserting this into the loss Equation 2 yields

$$\log(\prod_{i=1}^{N} p(x_i)^{M\tilde{y}_i}) = \log\left(\prod_{i=1}^{N}\left(\sum_{j=1}^{k_p+1} \pi_{\theta,j} c_{\theta,j}(x)\right)^{M\tilde{y}_i}\right) \tag{7}$$

Ignoring the exponent $M\tilde{y}_i$, we can instantly identify this as the incomplete data likelihood (todo better source). We identify j as the latent variable. The exponent causes no further issues as it can be easily modeled as a weight of the data points.

2.1 E-Step

The E-Step is exactly the same as it is for the gaussian mixture model. Simply calculate the conditional probability of the latent variable which is usually called the responsibility (proof omitted):

$$r_{\theta,j}(x_i) = \frac{\pi_{\theta,j} c_{\theta,j}(x_i)}{\sum_{j=1}^{k_p+1} \pi_{\theta,j} c_{\theta,j}(x_i)} \tag{8}$$

2.2 M-Step

For the M-Step we need to maximize:

$$\begin{aligned}
Q(\theta|\theta^t) &= \sum_{j_1=1}^{k_p+1} r_{\theta^t,j_1}(x_1) \sum_{j_2=1}^{k_p+1} r_{\theta^t,j_2}(x_2) \cdots \sum_{j_N=1}^{k_p+1} r_{\theta^t,j_N}(x_N) \left(\log \prod_{i=1}^{N} (\pi_{\theta,j_i} c_{\theta,j_i}(x_i))^{M\tilde{y}_i}\right) && (9)\\
&= \sum_{i=1}^{N} \sum_{j_1=1}^{k_p+1} r_{\theta^t,j_1}(x_1) \sum_{j_2=1}^{k_p+1} r_{\theta^t,j_2}(x_2) \cdots \sum_{j_N=1}^{k_p+1} r_{\theta^t,j_N}(x_N) \left(M\tilde{y}_i \log\left(\pi_{\theta,j_i} c_{\theta,j_i}(x_i)\right)\right) && (10)\\
&= \sum_{i=1}^{N} \sum_{j_i=1}^{k_p+1} r_{\theta^t,j_i}(x_i) \left(M\tilde{y}_i \log\left(\pi_{\theta,j_i} c_{\theta,j_i}(x_i)\right)\right) && (11)\\
&= M \sum_{j=1}^{k_p+1} \sum_{i=1}^{N} r_{\theta^t,j}(x_i) \tilde{y}_i \log\left(\pi_{\theta,j} c_{\theta,j}(x_i)\right) && (12)\\
&= M \sum_{j=1}^{k_p+1} \sum_{i=1}^{N} r_{\theta^t,j}(x_i) \tilde{y}_i \log\left(c_{\theta,j}(x_i)\right) + \sum_{j=1}^{k_p+1} \log\left(\pi_{\theta,j}\right) \sum_{i=1}^{N} M r_{\theta^t,j}(x_i) \tilde{y}_i && (13)
\end{aligned}$$

As $\sum_{j=1}^{k_p+1} \pi_{\theta,j} = 1$ the second term can be maximised with the Lagrange method yielding $\pi_{\theta,j} = \sum_{i=1}^{N} \tilde{q}_j(x_i) / \left(\sum_{j=1}^{k_p+1} \sum_{i=1}^{N} \tilde{q}_j(x_i) \right)$ where $\tilde{q}_j(x_i) \coloneqq r_{\theta^t,j}(x_i)\tilde{y}_i$ will be called the responsibility field of component j. Due to the independence of the parameters of each component we can equivalently maximize the first term in (13) for each j

$$Q_j(\theta|\theta^t) = \sum_{i=1}^{N} \tilde{q}_j(x_i) \log\left(c_{\theta,j}(x_i)\right) \tag{14}$$

which is to be maximised for each j.

3 Peak Birth Strategy: Max Background Responsibility Field

One of the crucial reasons this algorithm performs well is that it first fits 1 peak then 2 peak, adding one peak after another until the required number of peaks have been added. This behaviour is very similar to RJMCMC. The reason for the success of this strategy lies in the placement / birth of the new peaks. A simple approach might add a peak where the residuals (y - fit) is maximal. This approach however leads to poor performance as the residuals usually oscillate strongly around the biggest peaks due to the imperfect fits between pseudo voigts and true voigt profiles, resulting in to many peaks placed near the largest peaks. A much better proposal for the placement is through maxima in the responsibility field $\tilde{q}_0(x_i)$ of the linear background. The reason is that it heavily dampens residual oscillations. This is still imperfect as large a can cause an systematic error.

4 Code and more heuristics

4.0.1 Maximizing $Q_0(\theta|\theta^t)$

As c_0 models the linear background it is a special component. Maximizing $Q_0(\theta|\theta^t)$ is done as an approximation with an l2-loss instead (i.e. $\sum_{i=1}^{N} \left(\tilde{q}_j(x_i) - \gamma c_{\theta,j}(x_i)\right)^2$ for some constant γ). Furthermore a quadratic program is solved to bound the absolute value of a to reasonable values.

4.0.2 Maximizing $Q_{j>0}(\theta|\theta^t)$

These are the pseudo Voigt components. Maximizing $Q_{j>0}(\theta|\theta^t)$ is not trivial as the Lorentz contribution to the pseudo Voigt prevents simple estimation of the peak position. Instead first the maximum of the $\tilde{q}_j(x_i)$ field is found and then a quadratic function is fit around that maximum. The maxima of this quadratic function is used as the final peak position. For the FWHM and η a l2-loss is minimized instead of Q_j, this enables usage of a 1d optimizer for the FWHM and as subroutines a quadratic program determines η and the amplitude of the component.

5 Extensions and unfixed Bugs

- Use voigt profile instead of pseudo voigt.
  - Small Bug should also renormalize PseudoVoigts when they are cut off by x
  - Fit doesn't minimize Q_j either but a least squares like error
  - If a big Birt stra: respo field goes wrong