1 EM with stochastic sampling

Note that while the following derivation is in $\mathbb{R}^1$, everything can be generalized to $\mathbb{R}^d$. The governing equation for the problem in $\mathbb{R}^1$ is:

$$\dot{x} = f(x; \beta) \tag{1}$$

Consider an additive model for $f(x)$:

$$f(x) = \sum_{i=1}^{N} \beta_i \phi_i(x) \tag{2}$$

The data is given in the form of a time series, $\mathbf{x}$ at discrete time points. For simplicity, let us assume the observations are collected at equispaced times, jh for $0 \leq j \leq J$. Thus the observed data can be represented as $\mathbf{x} = x_0, x_1, \cdots, x_J$.

We start by introducing data at intermediate time points. In between 2 observed data points, x_i and x_{i+1}, we introduce F sampled values, $z_{i1}, z_{i2}, \cdots, z_{iF}$. The sampled values are created using a diffusion bridge. The i^{th} diffusion bridge sample depends on the observed data, x and current estimate of β, which we call $\beta^{(k)}$:

$$z^{(i)} \sim z \,|\, x, \beta^{(k)} \tag{3}$$

The observed data and sampled data can be combined together, where $M = J + FJ$

$$x_1, z_{1,1}, \cdots, z_{1,F}, x_2, z_{2,1}, \cdots, z_{2,F}, x_3, \cdots, x_J \to y_1^{(i)}, y_2^{(i)}, \cdots, y_M^{(i)} \tag{4}$$

The Expectation Maximization algorithm consists of two steps, computing the expected log likelihood function, Q and maximizing this function with respect to the parameters, β. Using the complete data, y, allows us to write the expected log likelihood as:

$$Q(\beta, \beta^{(k)}) = \sum_{y^{(i)}} \text{complete log likelihood}$$

$$Q = -\sum_i \sum_j \frac{(y_{j+1}^{(i)} - y_j^{(i)} - f(y_j^{(i)})h)^2}{2\sigma^2 h}$$

The above is the E step, and relies upon recent advances in sampling from diffusion bridges. Though the above derivation is in $\mathbb{R}^1$, there are now open-source codes to reliably sample from diffusion bridges in $\mathbb{R}^d$.

For the M step of the Expectation-Maximization algorithm, the complete log likelihood, $Q(\beta, \beta^{(k)})$ is maximized with respect to β:

$$\max_\beta -\frac{1}{2\sigma^2 h} \sum_i \sum_j (y_{j+1}^{(i)} - y_j^{(i)} - h \sum_{k=1}^{M} \beta_k \phi_k(y_j^{(i)}))^2$$

This is a least squares problem, i.e., a simple linear regression problem, but we include the details for the sake of completeness. Differentiating with respect to each component of parameter, β_ℓ and setting the value to zero gives the maximum value, thus providing a direct way to evaluate the M step:

$$\frac{\partial(\cdot)}{\partial\beta_\ell} = -\frac{1}{2\sigma^2 h}\sum_i\sum_j(y_{j+1}^{(i)} - y_j^{(i)} - h\sum_{k=1}^{M}\beta_k\phi_k(y_j^{(i)}))\cdot h\phi_\ell(y_j^{(i)}) = 0$$

Rearranging the terms gives

$$\sum_i\sum_j(\frac{y_{j+1}^{(i)} - y_j^{(i)}}{h})\phi_\ell(y_j^{(i)}) = \sum_k\beta_k\phi_k(y_j^{(i)})\phi_\ell(y_j^{(i)})$$

This system can be written as $r = A\cdot\beta$ where r and A are known. We solve for β via numerical linear algebra.

The resulting value of β is used in the next iteration of the EM algorithm:

$$\beta^{(k+1)} = \arg\max_\beta Q(\beta, \beta^{(k)})$$

We then repeat the E and M steps as described above. The algorithm is guaranteed to monotonically increase both the complete and incomplete log likelihoods.

For the general case in $\mathbb{R}^d$, the model consists of a vector field $f:\mathbb{R}^d\to\mathbb{R}^d$. Each component of this vector field can be developed in additive model form as above. Hence we will obtain d coefficient vectors β. With this modification, the analysis proceeds as above.