Summary: The Probabilistic Model of Age Assignment

February 28th 2017

Taehee Lee

0. Notations:

- $d^i = \{d_1^i, d_2^i, \cdots, d_{\mathrm{L}_i}^i\}$: the depth index of the data points of record i.
- $v^i = \{v_1^i, v_2^i, \cdots, v_{\mathrm{L}_i}^i\}$: $\delta^{18}\mathrm{O}$ data of record i.
- $w^i = \{w_1^i, w_2^i, \cdots, w_{\mathrm{L}_i}^i\}$: radiocarbon data of record i.
- $\mathrm{A}^i = \{\mathrm{A}_1^i, \mathrm{A}_2^i, \cdots, \mathrm{A}_{\mathrm{L}_i}^i\}$: the ages in the stack that the data points of record i are aligned to.
- $\mathrm{R}^i = \{\mathrm{R}_1^i, \mathrm{R}_2^i, \cdots, \mathrm{R}_{\mathrm{L}_i}^i\}$: the inverse sedimentation ratio defined by following:

$$\mathrm{R}_n^i \triangleq \frac{\mathrm{A}_n^i - \mathrm{A}_{n-1}^i}{d_n^i - d_{n-1}^i}$$

- τ^i: the mean shift on the $\delta^{18}\mathrm{O}$ values of record i.
- $\mu = \{\mu_t\}_{1 \le t \le \mathrm{T}}$: $\delta^{18}\mathrm{O}$ values for the ages $1 \le t \le \mathrm{T}$ in the target stack.
- $\sigma = \{\sigma_t\}_{1 \le t \le \mathrm{T}}$: standard deviations of residuals of observed $\delta^{18}\mathrm{O}$ values for the ages $1 \le t \le \mathrm{T}$ in the target stack.
- $\theta = \{\theta_t\}_{1 \le t \le \mathrm{T}}$: radiocarbon parameter for the ages $1 \le t \le \mathrm{T}$ in the target stack.
- $\mathrm{N}(a|b,c)$: the pdf of normal distribution at a with mean b and standard deviation c.
- $f(d|e)$: the pdf of radiocarbon data distribution at d with parameter e.
- $\rho: \{1,2,\cdots,\mathrm{R}\} \times \{1,2,\cdots,\mathrm{R}\} \to [0,1]$: the transition function.

1. What we have observed, assumed, or known: $\{d^i\}_{i=1}^{\mathrm{M}}$, $\{v^i\}_{i=1}^{\mathrm{M}}$, $\{w^i\}_{i=1}^{\mathrm{M}}$, θ, N, f, ρ.

2. What we need to estimate: $\{\mathrm{A}^i\}_{i=1}^{\mathrm{M}}$, $\{\mathrm{R}^i\}_{i=1}^{\mathrm{M}}$, $\{\tau^i\}_{i=1}^{\mathrm{M}}$, μ, σ.

3. Comment:

Because it is reasonable to assume the finite number of sedimentation rates (say, 17) with pseudo-data for the missing time obtained by interpolating the adjacent ones, to make the algorithm faster, in this method we do not assume the continuous domain of ρ. Also, by using radiocarbon data, we can give not only a better estimation but also shorter running time of the algorithm in practice.

Let us assume that each $\delta^{18}\mathrm{O}$ and radiocarbon data are conditionally independent given the time when they were generated. If not, we need to either learn their joint distribution or give a new model on the relationships of those data with some related parameter sets.

The pdf of radiocarbon data f is defined by taking 0 outside of $[\mathrm{mean} - 5\mathrm{std}, \mathrm{mean} + 5\mathrm{std}]$.

4. Model: it consists of two steps. One is the forward algorithm and the other is the backward sampling.

4.1. Forward Algorithm:

$$\mathrm{F}^i\left(n, r_n^i, t_n^i\right) \triangleq p\left(v_{1:n}^i, w_{1:n}^i, \mathrm{R}_n^i = r_n^i, \mathrm{A}_n^i = t_n^i \middle| d_{1:\mathrm{L}_i}^i\right)$$

$$= \mathrm{N}\left(v_n^i \middle| \tau^i + \mu_{t_n^i}, \sigma_{t_n^i}\right) f\left(w_n^i \middle| \theta_{t_n^i}\right) \sum_r \rho\left(r, r_n^i\right) \mathrm{F}^i\left(n-1, r, t_n^i - r_n^i\left(d_n^i - d_{n-1}^i\right)\right)$$

4.2. Backward Sampling:

For each i, get $\left\{\tilde{t}^{i,l}\right\}_{l=1}^{1000} = \left\{\left\{\tilde{t}_n^{i,l}\right\}_{n=1}^{\mathrm{L}_i}\right\}_{l=1}^{1000}$ and $\left\{\tilde{r}^{i,l}\right\}_{l=1}^{1000} = \left\{\left\{\tilde{r}_n^{i,l}\right\}_{n=1}^{\mathrm{L}_i}\right\}_{l=1}^{1000}$ by following:

$$\tilde{t}_n^{i,l} \equiv \tilde{t}_{n+1}^{i,l} - \tilde{r}_{n+1}^{i,l}\left(d_{n+1}^i - d_n^i\right)$$

$$\mathbb{P}\left(\mathrm{R}_n^{i,l} = r \middle| \mathrm{A}_{n+1}^{i,l} = \tilde{t}_{n+1}^{i,l}, \mathrm{R}_{n+1}^{i,l} = \tilde{r}_{n+1}^{i,l}, v_{1:n+1}^i, w_{1:n+1}^i, d_{1:\mathrm{L}_i}^i\right)$$

$$\propto \mathrm{N}\left(v_{n+1}^i \middle| \tau^i + \mu_{\tilde{t}_{n+1}^{i,l}}, \sigma_{\tilde{t}_{n+1}^{i,l}}\right) f\left(w_{n+1}^i \middle| \theta_{\tilde{t}_{n+1}^{i,l}}\right) \rho\left(r, \tilde{r}_{n+1}^{i,l}\right) \mathrm{F}^i\left(n, r, \tilde{t}_{n+1}^{i,l} - \tilde{r}_{n+1}^{i,l}\left(d_{n+1}^i - d_n^i\right)\right)$$

I.e., the very essence of this sampling is to sample ($\tilde{t}_{\mathrm{L}_i}^{i,l}$ and) $\tilde{r}^{i,l}$ iteratively: after that, $\tilde{t}_{1:\mathrm{L}_i-1}^{i,l}$ is automatically assigned just after the corresponding $\tilde{r}_{2:\mathrm{L}_i}^{i,l}$ is sampled.

You may have already noticed, however, that in fact there is a subtle but not negligible problem in this finite sedimentation rates setting: we assumed that ages $1 \le t \le \mathrm{T}$ in the stack is discretized so the terms $t_n^i - r_n^i\left(d_n^i - d_{n-1}^i\right)$ in the forward algorithm or $\tilde{t}_n^{i,l} \equiv \tilde{t}_{n+1}^{i,l} - \tilde{r}_{n+1}^{i,l}\left(d_{n+1}^i - d_n^i\right)$ in the backward sampling may not be in the time set! If so, $\mathrm{F}^i\left(n-1, r, t_n^i - r_n^i\left(d_n^i - d_{n-1}^i\right)\right)$ or $\left(\mu_{\tilde{t}_n^{i,l}}, \sigma_{\tilde{t}_n^{i,l}}, \theta_{\tilde{t}_n^{i,l}}\right)$ may even not be well-defined. To deal with it, if it is possible to assume that the forward terms $\mathrm{F}^i(n, r, t)$ is 'nice' for small perturbations on t for each fixed r and the values $(\mu_t, \sigma_t, \theta_t)$ do not change much as t varies a little, then we can either choose the nearest t in the stack or define the interpolated terms from the nearest parameters as those values.

5. Other Possible Approaches:

5.1. Forward-Backward Algorithm:

$$\mathrm{F}^i\left(n, r_n^i, t_n^i\right) \triangleq p\left(v_{1:n}^i, w_{1:n}^i, \mathrm{R}_n^i = r_n^i, \mathrm{A}_n^i = t_n^i \middle| d_{1:\mathrm{L}_i}^i\right)$$

$$= \mathrm{N}\left(v_n^i \middle| \tau^i + \mu_{t_n^i}, \sigma_{t_n^i}\right) f\left(w_n^i \middle| \theta_{t_n^i}\right) \sum_r \rho\left(r, r_n^i\right) \mathrm{F}^i\left(n-1, r, t_n^i - r_n^i\left(d_n^i - d_{n-1}^i\right)\right)$$

$$\mathrm{B}^i\left(n, r_n^i, t_n^i\right) \triangleq p\left(v_{n+1:\mathrm{L}_i}^i, w_{n+1:\mathrm{L}_i}^i \middle| \mathrm{R}_n^i = r_n^i, \mathrm{A}_n^i = t_n^i, d_{1:\mathrm{L}_i}^i\right)$$

$$= \sum_r \left[\mathrm{N}\left(v_{n+1}^i \middle| \tau^i + \mu_{r\left(d_{n+1}^i - d_n^i\right) + t_n^i}, \sigma_{r\left(d_{n+1}^i - d_n^i\right) + t_n^i}\right)\right.$$

$$\left. \times f\left(w_n^i \middle| \theta_{r\left(d_{n+1}^i - d_n^i\right) + t_n^i}\right) \rho\left(r_n^i, r\right) \mathrm{B}^i\left(n+1, r, r\left(d_{n+1}^i - d_n^i\right) + t_n^i\right)\right]$$

$$\mathrm{P}^i\left(n, r_n^i, t_n^i\right) \triangleq p\left(\mathrm{R}_n^i = r_n^i, \mathrm{A}_n^i = t_n^i \middle| v_{1:\mathrm{L}_i}^i, d_{1:\mathrm{L}_i}^i\right) \propto \mathrm{F}^i\left(n, t_{n-1}^i, t_n^i\right) \mathrm{B}^i\left(n, t_{n-1}^i, t_n^i\right)$$

Then we can get the exact posterior probability on each element of data by following:

$$\mathbb{P}\left(\mathrm{A}_n^i = t \middle| v_{1:\mathrm{L}_i}^i, d_{1:\mathrm{L}_i}^i\right) = \sum_r \mathrm{P}^i(n, r, t)$$

5.2. Blocked Gibbs Sampler: this is one of the other possible ways to get samples. One good aspect of this algorithm is that it does not require any burdensome precomputations, such as the forward algorithm in the above methods. However, it does depend on the initial conditions, so if we do not have any heuristic ways of choosing appropriate initial conditions, good sampling cannot be guaranteed.

$$\mathbb{P}\left(\mathrm{A}_n^{i,l} = t \middle| \mathrm{A}_{n-2}^{i,l} = \tilde{t}_{n-2}^{i,l}, \mathrm{A}_{n-1}^{i,l} = \tilde{t}_{n-1}^{i,l}, \mathrm{A}_{n+1}^{i,l} = \tilde{t}_{n+1}^{i,l}, \mathrm{A}_{n+2}^{i,l} = \tilde{t}_{n+2}^{i,l}, v_n^i, d_{1:\mathrm{L}_i}^i\right)$$

$$\propto \mathrm{N}\left(v_n^i \middle| \tau^i + \mu_t, \sigma_t\right) \rho\left(\frac{\tilde{t}_{n-1}^{i,l} - \tilde{t}_{n-2}^{i,l}}{d_{n-1}^i - d_{n-2}^i}, \frac{t - \tilde{t}_{n-1}^{i,l}}{d_n^i - d_{n-1}^i}\right) \rho\left(\frac{t - \tilde{t}_{n-1}^{i,l}}{d_n^i - d_{n-1}^i}, \frac{\tilde{t}_{n+1}^{i,l} - t}{d_{n+1}^i - d_n^i}\right) \rho\left(\frac{\tilde{t}_{n+1}^{i,l} - t}{d_{n+1}^i - d_n^i}, \frac{\tilde{t}_{n+2}^{i,l} - \tilde{t}_{n+1}^{i,l}}{d_{n+2}^i - d_{n+1}^i}\right)$$

Note that the above formula involves two blocks for the efficient sampling: $\left\{\mathrm{A}_1^{i,l}, \mathrm{A}_3^{i,l}, \mathrm{A}_5^{i,l}, \cdots\right\}$ and $\left\{\mathrm{A}_2^{i,l}, \mathrm{A}_4^{i,l}, \mathrm{A}_6^{i,l}, \cdots\right\}$.

6. How to estimate the parameters: by using the EM algorithm.

First, we assume that we already knew parameters, and infer the hidden states $\left\{\mathrm{A}^i\right\}_{i=1}^{\mathrm{M}}$. Second, we assume that we already knew the hidden states, and infer the parameters with ML estimators. Now go back to the first step if some predetermined convergence criteria have not been achieved yet.

7. Time complexity: $\mathcal{O}(\mathrm{MT}^2)$.

If we assume that the domain of ρ is discretized by R elements, then it is $\mathcal{O}(\mathrm{MTR})$. According to Seonmin's paper, her algorithm took about one month with 200 computing nodes as it takes over 100 hours to complete one iteration: note that the codes in her github only used three sets of data as an example, but in the actual work we will need to deal with a lot of sets of those.