

Soil Carbon Models

12/9/2016

Century Model

The century model is proposed by Parton et al. (1988). We only focus on the three soil pools in this model which are listed below along with their decomposition rates:

pool 1: Active Soil C, $\kappa_1 \approx 1/1.5$,

pool 2: Slow Soil C, $\kappa_2 \approx 1/25$,

pool 3: Passive Soil C, $\kappa_3 \approx 1/1000$,

where κ denotes the decay rate which is defined as 1 over the turnover.

We denote the transfer rate from pool j to pool i by r_{ij} . The transfer rates are parameterized as a ratio of the decay rate: $r_{ij} = \alpha_{ij}\kappa_j$. For the Century model, we have:

$$\begin{aligned}\alpha_{21} &= 1 - F(T) - 0.004, \\ \alpha_{31} &= 0.004, \\ \alpha_{12} &= 0.42, \\ \alpha_{32} &= 0.03, \\ \alpha_{13} &= 0.45, \\ \alpha_{23} &= 0,\end{aligned}$$

where 'T' is the soil silt + clay content, and $F(T) = 0.85 - 0.68 \times T$. These values are expert-tuned; however, in our Bayesian framework, we estimate them from the data. For each pool, we can write the following differential equation:

$$\frac{dC_i(t)}{dt} = -\kappa_i C_i(t) + \sum_{j \neq i} \alpha_{ij} \kappa_j C_j(t).$$

Combining all these differential equations into a single formula, we get:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & \alpha_{12}\kappa_2 & \alpha_{13}\kappa_3 \\ \alpha_{21}\kappa_1 & -\kappa_2 & 0 \\ \alpha_{31}\kappa_1 & \alpha_{32}\kappa_2 & -\kappa_3 \end{pmatrix} C(t).$$

The total amount of Carbon in the beginning is C_{tot} which is divided among the three pools: $C(0) = (\gamma_1, \gamma_2, \gamma_3) \cdot C_{tot}$.

1 Outline of Simulations

1.1 Data generation and fitting models

We generate simulated data from the century model above, and fit the following six models to the simulated data:

1. Independent Model:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & 0 & 0 \\ 0 & -\kappa_2 & 0 \\ 0 & 0 & -\kappa_3 \end{pmatrix} C(t).$$

2. Cascade Model:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & 0 & 0 \\ \alpha_{21}\kappa_1 & -\kappa_2 & 0 \\ \alpha_{31}\kappa_1 & \alpha_{32}\kappa_2 & -\kappa_3 \end{pmatrix} C(t).$$

3. Feedback Model:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & \alpha_{12}\kappa_2 & \alpha_{13}\kappa_3 \\ \alpha_{21}\kappa_1 & -\kappa_2 & \alpha_{23}\kappa_3 \\ \alpha_{31}\kappa_1 & \alpha_{32}\kappa_2 & -\kappa_3 \end{pmatrix} C(t).$$

4. Century Model:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & \alpha_{12}\kappa_2 & \alpha_{13}\kappa_3 \\ \alpha_{21}\kappa_1 & -\kappa_2 & 0 \\ \alpha_{31}\kappa_1 & \alpha_{32}\kappa_2 & -\kappa_3 \end{pmatrix} C(t).$$

5. Two-pool Model:

$$\frac{dC(t)}{dt} = \begin{pmatrix} -\kappa_1 & \alpha_{12}\kappa_2 \\ \alpha_{21}\kappa_1 & -\kappa_2 \end{pmatrix} C(t).$$

6. One-pool Model:

$$\frac{dC(t)}{dt} = -\kappa_1 C(t).$$

For generating the simulated data, we use the expert-tuned values for decomposition and transfer parameters and also assume $\gamma_1 = \gamma_2 = 0.1$ and $\gamma_3 = 0.8$.

1.2 CO₂ flux

We assume that our observations are in terms of CO₂ fluxes. Theoretically, the CO₂ flux at time t is defined as $\sum_{i=1}^3 |dC_i(t)/dt|$. In experiments, this is calculated by measuring the CO₂ emitted between a cap time, t_{cap} and t : $\Delta CO_2/(t - t_{cap})$.

1.3 Sampling times

Generally, the sampling times are not uniform. The CO₂ flux is sampled more frequently in the beginning and then less frequently towards the end. A common practice is to sample once per day for the first week, then once per week for the first month, and then once per month. Given that the scale of the turnover rates are in years, this amounts to the following sampling times:

$$t_s = (\frac{1}{360}, \frac{2}{360}, \frac{3}{360}, \frac{4}{360}, \frac{5}{360}, \frac{6}{360}, \frac{7}{360}, \frac{14}{360}, \frac{21}{360}, \frac{28}{360}, \frac{60}{360}, \frac{90}{360}, \frac{120}{360}, \dots)$$

1.4 Observation error

From empirical studies we know that the noise standard deviation is approximately half the value of the flux. We simulate this by averaging the values of simulated CO₂ flux and dividing by two to set the value for noise standard deviation.

1.5 Replications

Generally, we have several replications for each experiment (i.e., several CO₂ fluxes). We can fit these replicates in three ways:

1. No pooling: fitting models separately to each model, i.e., estimating the model parameters independently for each replication;
2. Complete pooling: fitting a single model to all the replications, i.e., estimating only a single set of parameters for all models;
3. Partial pooling: fitting a hierarchical Bayesian model which estimates parameters jointly for all replications and allows for variation between replicates.

2 Statistical models

As mentioned in the previous section, we fit the data in three ways: no pooling, complete pooling, and partial pooling using a hierarchical model.

2.1 No pooling and complete pooling

In terms of model specification, no pooling and complete pooling are similar; the only difference is the data provided for likelihood computations. For complete pooling, all the data is used for parameter estimation and for no pooling, each replication is used separately.

- **Turnover rates:** we assume a prior which is centered around the values chosen by experts. We believe that the estimated values should be around these values; so, we set the standard deviations to be 1/10'th of the mean times a Cauchy random variable. Therefore, we enforce the standard

deviation to be small, but let the data specifies the exact value. We can check the sensitivity of the results to these assumptions later.

$$\begin{aligned}\tau_1 &\sim \mathcal{N}(1.5, 0.15\sigma_1) & \tau_2 &\sim \mathcal{N}(25, 2.5\sigma_2) & \tau_3 &\sim \mathcal{N}(1000, 100\sigma_3) \\ \sigma_1 &\sim \text{Cauchy}(0, 1) & \sigma_2 &\sim \text{Cauchy}(0, 1) & \sigma_3 &\sim \text{Cauchy}(0, 1)\end{aligned}$$

The decomposition rates are then $\kappa_1 = 1/\tau_1, \kappa_2 = 1/\tau_2, \kappa_3 = 1/\tau_3$.

- **Initial allocations:** the vector $(\gamma_1, \gamma_2, \gamma_3)$ governs how the Carbon is initially divided between the three pools. We know that $\gamma_1 + \gamma_2 + \gamma_3 = 1$; so, the vector is simplex. We assign a uniform prior to this vector:

$$(\gamma_1, \gamma_2, \gamma_3) \sim \text{Dirichlet}(1, 1, 1)$$

- **Transfer rates:** the transfer rates are between 0 and 1 and are given a uniform prior with the following constraints for each j :

$$\sum_{i \text{ s.t. } i \neq j} a_{ij} < 1, \quad j = 1, 2, 3$$

We can model this inequality constraint as an equality in the form of a simplex vector. For example, we can write:

$$a_{11} + a_{21} + a_{31} = 1,$$

where a_{11} is a nuisance parameter which is not used in the model but enforces the inequality constraint. Thus, we can write:

$$\begin{aligned}(a_{11}, a_{21}, a_{31}) &\sim \text{Dirichlet}(1, 1, 1) \\ (a_{12}, a_{22}, a_{32}) &\sim \text{Dirichlet}(1, 1, 1) \\ (a_{13}, a_{23}, a_{33}) &\sim \text{Dirichlet}(1, 1, 1)\end{aligned}$$

2.2 Partial pooling (hierarchical model)

We assume that we have K replications. For each parameter, we have a global value, denoted by the superscript 0, and local values for each replication, denoted by a superscript k where $k = 1, \dots, K$. The global parameters have priors identical to the ones in the no pooling model. The local parameter are equal to these global parameters plus some variability.

- **Turnover rates:**

$$\begin{aligned}\tau_1^0 &\sim \mathcal{N}(1.5, 0.15\sigma_1) & \tau_2^0 &\sim \mathcal{N}(25, 2.5\sigma_2) & \tau_3^0 &\sim \mathcal{N}(1000, 100\sigma_3) \\ \sigma_1 &\sim \text{Cauchy}(0, 1) & \sigma_2 &\sim \text{Cauchy}(0, 1) & \sigma_3 &\sim \text{Cauchy}(0, 1) \\ \tau_1^k &\sim \mathcal{N}(\tau_1^0, 0.15\sigma_4) & \tau_2^k &\sim \mathcal{N}(\tau_2^0, 2.5\sigma_5) & \tau_3^k &\sim \mathcal{N}(\tau_3^0, 100\sigma_6) \\ \sigma_4 &\sim \text{Cauchy}(0, 1) & \sigma_5 &\sim \text{Cauchy}(0, 1) & \sigma_6 &\sim \text{Cauchy}(0, 1)\end{aligned}$$

- **Initial allocations:**

$$\begin{aligned}(\gamma_1^0, \gamma_2^0, \gamma_3^0) &\sim \text{Dirichlet}(1, 1, 1) \\ (\gamma_1^k, \gamma_2^k, \gamma_3^k) &\sim \text{Dirichlet}(\gamma_1^0, \gamma_2^0, \gamma_3^0)\end{aligned}$$

Another possibility is:

$$\begin{aligned}(\gamma_1^0, \gamma_2^0, \gamma_3^0) &\sim \text{Dirichlet}(1, 1, 1) \\(\lambda_1^k, \lambda_2^k, \lambda_3^k) &\sim \mathcal{N}((\gamma_1^0, \gamma_2^0, \gamma_3^0), \Sigma) \\(\gamma_1^k, \gamma_2^k, \gamma_3^k) &= \frac{(\lambda_1^k, \lambda_2^k, \lambda_3^k)}{\lambda_1^k + \lambda_2^k + \lambda_3^k}\end{aligned}$$

- **Transfer rates:** As described in the previous section, by introducing a nuisance parameter, we can model the transfer rates as simplex vectors. A hierarchical structure can then be imposed in exact same way as explained in the previous paragraph for initial allocations.