Estimation of Phase-Amplitude coupling in SOMATA

Ran Liu

August 8, 2024

Phase-amplitude coupling

- Two signals: one high-frequency, one low-frequency
- Amplitude of the high-frequency signal modulates with the phase of the low-frequency signal
- This phenomenon plays an important role in the coordination and interaction of neural oscillations, often across different regions of the brain.
- We wish to quantify this phenomenon using state-space models

Oscillator model

Let $x_t \in \mathbb{R}^2$ be latent state, $y_t \in \mathbb{R}$ observations, $f \in \mathbb{R}^+$ frequency of oscillator, $Fs \in \mathbb{R}^+$ sampling frequency:

$$x_t = a \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} x_{t-1} + w_t \tag{1}$$

$$y_t = \begin{bmatrix} 1 & 0 \end{bmatrix} x_t + v_t \tag{2}$$

$$\theta = \frac{2\pi f}{Fs} \tag{3}$$

$$w_t \sim \mathcal{N}(0, Q) \tag{4}$$

$$v_t \sim \mathcal{N}(0, R) \tag{5}$$

A generative model of PAC

Let k be peak amplitude of modulation, and Φ be peak phase of modulation.

$$\theta_{slow} = \arccos \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} x_{slow}}{||x_{slow}||} \tag{6}$$

$$y_{fast} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(x_{fast} * \left(1 + k \cos \left(\theta_{slow} - \Phi \right) \right) \right) + v_{fast} \tag{7}$$

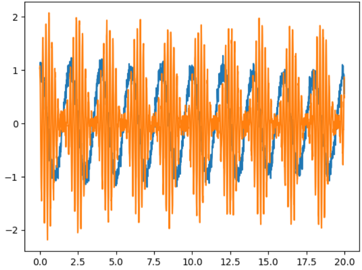

Figure 1: Oscillator-generated signals with phase-amplitude coupling

Method overview

1. Use Oscillator Model Learning or Oscillator Search Algorithms within SOMATA to compute phase $\Phi(t)$ and amplitude $y(t)$.
2. Fit a constrained regression on each window of data:

$$y(t) = \beta_0 + \beta_1 \cos\Phi(t) + \beta_2 \sin\Phi(t) + \epsilon \tag{8}$$

   such that $\beta_0^2 \geq \beta_1^2 + \beta_2^2$, $\epsilon \sim \mathcal{N}(0, \sigma_\beta^2)$
3. Fit a multivariate AR(p) model, and use Kalman filter across windows to compute smoothed $\beta(i) = [\beta_0(i), \beta_1(i), \beta_2(i)]$.
4. Compute modulation index and modulation phase.

$$k^{mod}(i) = \frac{\sqrt{\beta_1^2 + \beta_2^2}}{\beta_0} \tag{9}$$

$$\phi^{mod}(i) = \arctan\frac{\beta_2(i)}{\beta_1(i)} \tag{10}$$

Implementation overview

1. **fit_pac_regression(phase, amplitude)**
   Takes phase vector of shape (n,), amplitude vector of shape (n,), returns MCMC draws from distribution of parameters $(\beta_0, \beta_1, \beta_2)$ given observed data. These parameters follow a multivariate t-distribution; therefore, posterior mean is a reasonable MAP estimator.
2. **optimize_arp(y, p)** Takes y, an array of n observations of d variables, of shape (d, n), and p, order of AR model, and returns A, Q, R, the parameters of the fitted multivariate AR(p) model.
3. **mvar_ssm(y, A, Q, R)** Take data and parameters of AR(p) model, and initializes a StateSpaceModel object, which can be used to perform Kalman smoothing.

Method specifics: constrained regression

We fit the constrained regression using Stan's implementation of NUTS (No-U-Turn Sampler), an efficient MCMC method.

$$y(t) = \beta_0 + \beta_1 \cos \Phi(t) + \beta_2 \sin \Phi(t) + \epsilon \tag{8}$$

such that $\beta_0^2 \geq \beta_1^2 + \beta_2^2$, $\epsilon \sim \mathcal{N}(0, \sigma_\beta^2)$

We obtain samples from $\beta, \sigma_\beta^2 | \phi(t), y(t)$.

Linear regression coefficients are distributed according to a multivariate t-distribution; posterior mean in this case would be a reasonable MAP estimator.

Method specifics: fitting AR(p) model

Our system is specified by:

$$y_i = \beta(i) = x_i + w_i, w_i \sim \mathcal{N}(0, R) \tag{11}$$

$$x_i = \sum_{k=1}^{p} A_k x_{i-k} + v_i, v_i \sim \mathcal{N}(0, Q) \tag{12}$$

We compute the following sequence of autocovariance matrices, for $j \in 0, ..., p$:

$$C_j = \mathrm{Cov}(y_{i-j}, y_i) \tag{13}$$

And construct the block-Toeplitz matrix:

$$C = \begin{bmatrix} C_0 & C_1^T & C_2^T & \dots & C_p^T \\ C_1 & C_0 & C_1^T & \dots & C_{p-1}^T \\ C_2 & C_1 & C_0 & \dots & C_{p-2}^T \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_p & C_{p-1} & C_{p-2} & \dots & C_0 \end{bmatrix} \tag{14}$$

Model specifics: fitting AR(p) model, pt. 2

We have then that $R \in (0, \lambda_{min})$, where λ_{min} is the smallest eigenvalue of C. For a given candidate value of R, AR(p) coefficients are then given by:

$$\begin{bmatrix} C_0 - I_d R & C_1^T & \dots & C_{p-1}^T \\ C_1 & C_0 - I_d R & \dots & C_{p-2}^T \\ \vdots & \vdots & \ddots & \vdots \\ C_{p-1} & C_{p-2} & \dots & C_0 - I_d R \end{bmatrix} \begin{bmatrix} A_1^T \\ A_2^T \\ \vdots \\ A_p^T \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_p \end{bmatrix} \tag{15}$$

$$Q = C_0 - I_d R - \sum_{k=1}^{p} A_k C_k \tag{16}$$

We then pick R through numerical maximization of the model log likelihood, compute A_k by inverting the block-Toeplitz matrix defined in (15), and compute Q using (16).

References and further reading

- Soulat et al. "State space methods for phase amplitude coupling analysis." *Scientific Reports*, **2022**.
- Matsuda and Komaki. "Time Series Decomposition into Oscillation Components and Phase Estimation." *Neural Computation*, **2017**.
- Shumway and Stoffer. "Time Series Analysis and its Applications." 4th edition, **2017**.
- McLachlan. "The EM Algorithm and Extensions." 2nd edition, **2007**.