

Nonparametric covariance estimation for longitudinal data via tensor product smoothing

Tayler Blake ¹ Dr. Yoonkyung Lee ²

¹Information Control Company

²The Ohio State University, Department of Statistics

July 11, 2017

The data:

$$Y_i = (Y_{i1}, Y_{i2}, \dots, Y_{iM_i})', \quad i = 1, \dots, N$$

associated with measurement times

$$t_1 < t_2 < \dots < t_{M_i}.$$

Goal: estimate

$$Cov(Y) = \Sigma$$

The flaming hoops:

- ▶ Covariance matrices (and their estimates) should be positive definite.
 - Constrained optimization is a headache.
- ▶ The $\{t_{ij}\}$ may be suboptimal.
 - Observation times may not fall on a regular grid, may vary across subjects.
- ▶ More dimensions, more problems (maybe.)
 - Sample covariance matrix falls apart when m is large.

The flaming hoops:

- ▶ Covariance matrices (and their estimates) should be positive definite.
 - Constrained optimization is a headache.
- ▶ The $\{t_{ij}\}$ may be suboptimal.
 - Observation times may not fall on a regular grid, may vary across subjects.
- ▶ More dimensions, more problems (maybe.)
 - Sample covariance matrix falls apart when m is large.

The flaming hoops:

- ▶ Covariance matrices (and their estimates) should be positive definite.
 - Constrained optimization is a headache.
- ▶ The $\{t_{ij}\}$ may be suboptimal.
 - Observation times may not fall on a regular grid, may vary across subjects.
- ▶ More dimensions, more problems (maybe.)
 - Sample covariance matrix falls apart when m is large.

The flaming hoops:

- ▶ Covariance matrices (and their estimates) should be positive definite. A cute little reparameterization \implies unconstrained estimation, meaningful interpretation
- ▶ The $\{t_{ij}\}$ may be messy.
Frame covariance estimation as function estimation.
- ▶ More dimensions, more problems (maybe.)



Figure: Regulate like Nate Dogg.

Covariance dress-up: the modified Cholesky decomposition

$$Y = (Y_1, \dots, Y_M)' \sim \mathcal{N}(0, \Sigma).$$

For any positive definite Σ , we can find T which diagonalizes Σ :

$$D = T\Sigma T', \quad T = \begin{bmatrix} 1 & 0 & \dots & & \\ -\phi_{21} & 1 & & & \\ -\phi_{31} & -\phi_{32} & 1 & & \\ \vdots & & & \ddots & \\ -\phi_{M1} & -\phi_{M2} & \dots & -\phi_{M,M-1} & 1 \end{bmatrix}$$

Now, for the cutest part:



Okay, really:

Regress Y_j on $Y_{(1:j-1)} = (Y_1, \dots, Y_{j-1})'$:

$$y_j = \begin{cases} e_1 & j = 1, \\ \sum_{k=1}^{j-1} \phi_{jk} y_k + \sigma_j e_j & j = 2, \dots, M \end{cases} \quad (1)$$

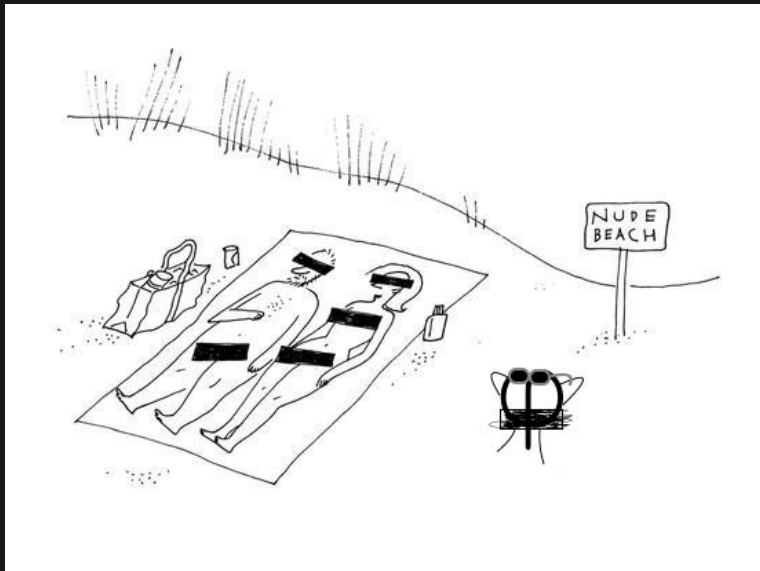
In matrix form:

$$e = TY, \quad (2)$$

and taking covariances on both sides:

$$D = \text{diag}(\sigma_1^2, \dots, \sigma_M^2) = T\Sigma T'. \quad (3)$$

No constraints on the ϕ_{jk} s!



The regression model tool box is a deep, luxurious toolbox.

$$\begin{array}{ll} Y_j \longrightarrow Y(t_j) & e_j \longrightarrow e(t_j) \\ \phi_{jk} \longrightarrow \phi(t_j, t_k) & \sigma_j^2 \longrightarrow \sigma^2(t_j) \end{array}$$

$$y(t_j) = \sum_{k=1}^{j-1} \phi(t_j, t_k) y(t_k) + \sigma(t_j) e(t_j), \quad (4)$$

where

$$e(s) \sim \mathcal{WN}(0, 1)$$

Regularization of $\phi(s, t)$ is more intuitive if we transform the s - t axis.

$$\begin{aligned}l &= s - t \\ m &= \frac{1}{2} (s + t)\end{aligned}$$

Reparameterize ϕ :

$$\phi(s, t) = \phi^*(l, m) = \phi^*\left(s - t, \frac{1}{2}(s + t)\right)$$

Take $\hat{\phi}^*$ to be the minimizer of

$$-2L_{\phi}(\phi, y_1, \dots, y_N) = \sum_{i=1}^N \sum_{j=2}^{m_i} \sigma_{ij0}^{-2} \left(y_{ij} - \sum_{k=1}^{j-1} \phi(t_{ij}, t_{ik}) y_{ik} \right)^2$$

Smooth ANOVA models

Decompose

$$\phi^* (l, m) = \mu + \phi_1 (l) + \phi_2 (m) + \phi_{12} (l, m), \quad (6)$$

so Model 4 becomes

$$y(t_j) = \sum_{k=1}^{j-1} \left[\mu + \phi_1(l_{jk}) + \phi_2(m_{jk}) + \phi_{12}(l_{jk}, m_{jk}) \right] y(t_k) + \sigma(t_j) e(t_j)$$

Approximate $\phi_1, \phi_2, \phi_{12}$ with B-splines.

$$\begin{aligned}
 \phi_1(l) &= \sum_{c=1}^{c_l} B_c(l; q_l) \theta_{lc} = B_l \theta_l, \\
 \phi_2(m) &= \sum_{c'=1}^{c_m} B_{c'}(m; q_m) \theta_{mc'} = B_m \theta_m \\
 \phi_{12}(l, m) &= \sum_{c=1}^{c_l} \sum_{c'=1}^{c_m} B_c(l; q_l) B_{c'}(m; q_m) \theta_{cc'} = B_{lm} \theta_{lm}
 \end{aligned} \tag{7}$$

$$\begin{aligned}
 B_{lm} &= B_m \square B_l \\
 &\equiv (B_m \otimes 1'_{c_l}) \odot (1'_{c_m} \otimes B_l)
 \end{aligned}$$

Difference penalty had to regulate.

For $f(x) = \sum_{i=1}^p B_i(x) \theta_i$, approximate

$$\begin{aligned} \int_0^1 (f''(x))^2 dx &= \int_0^1 \left\{ \sum_{i=1}^p B_i''(x) \theta_i \right\}^2 dx \\ &= k_1 \sum_i (\Delta^2 \theta_i)^2 + k_2, \end{aligned} \tag{8}$$

by

$$\|D_2 \theta\|^2, \quad D_2 \theta = (\Delta^2 \theta_1, \dots, \Delta^2 \theta_{p-2})'$$

In general, **approximate** $\int_0^1 (f^{(d)})^2 dx$ **with** $\|D_d \theta\|^2$

PS-ANOVA Penalty

$$B \equiv [1_p \mid B_l \mid B_m \mid B_{lm}], \quad \theta \equiv (\mu, \theta_l, \theta_m, \theta_{lm})'$$

Find θ minimizing

$$(Y - WB\theta)' D^{-1} (Y - WB\theta) + \theta' P \theta$$

$$P = \begin{bmatrix} 0 & & \\ \underbrace{\lambda_l D'_{d_l} D_{d_l}}_{P_l} & & \\ & \underbrace{\lambda_l D'_{d_l} D_{d_l}}_{P_m} & \\ & & \underbrace{\tau_l D'_{d_l} D_{d_l} \otimes I_{c_m} + \tau_m I_{c_l} \otimes D'_{d_m} D_{d_m}}_{P_{lm}} \end{bmatrix}$$

PS-ANOVA Penalty

$$B \equiv [1_p \mid B_l \mid B_m \mid B_{lm}], \quad \theta \equiv (\mu, \theta_l, \theta_m, \theta_{lm})'$$

Find θ minimizing

$$(Y - WB\theta)' D^{-1} (Y - WB\theta) + \theta' P \theta \quad (9)$$

$$P = \begin{bmatrix} 0 & & \\ \underbrace{\lambda_l D'_{d_l} D_{d_l}}_{P_l} & & \\ & \underbrace{\lambda_l D'_{d_l} D_{d_l}}_{P_m} & \\ & & \underbrace{\tau_l D'_{d_l} D_{d_l} \otimes I_{c_m} + \tau_m I_{c_l} \otimes D'_{d_m} D_{d_m}}_{P_{lm}} \end{bmatrix}$$

Mixed model representation

Find transformation $Q = [Q_n \mid Q_s]$ to map

$$\begin{aligned} BQ &= [BQ_n \mid BQ_s] & Q'\theta &= (\beta', \alpha')' \\ &= [X \mid Z] & &= ((Q_n'\theta)', (Q_s'\theta'))' \end{aligned}$$

to reparameterize the ill-posed Model 9 as

$$\begin{aligned} Y &= W(X\beta + Z\alpha) + e \\ \alpha &\sim \mathcal{N}(0, G), \\ e &\sim \mathcal{N}(0, D) \end{aligned} \tag{10}$$

Mixed model representation

$$\begin{bmatrix} \lambda_l \tilde{\Delta}_l & & & \\ & \lambda_m \tilde{\Delta}_m & & \\ & & \begin{bmatrix} \tau_l \tilde{\Delta}_l & & \\ & \tau_m \tilde{\Delta}_m & \\ & & \tau_m \tilde{\Delta}_m \otimes I_{c_l-d_l} + I_{c_m-d_m} \otimes \tau_l \tilde{\Delta}_l \end{bmatrix} & \end{bmatrix}$$

Decomposition of ϕ^* for $d_l = d_m = 2$

	$\{1\}$	$\{m\}$	$\{B_{j'}(m)\}$
$\{1\}$	$\{1\}$	$\{m\}$	$\{B_{j'}(m)\}$
$\{l\}$	$\{l\}$	$l \times m$	$l \times \{B_{j'}(m)\}$
$\{B_j(l)\}$	$\{B_j(l)\}$	$m \times \{B_j(l)\}$	$\{B_j(l) B_{j'}(m)\}$

Nested PS-ANOVA

Re-express ϕ_{12} :

$$\phi_{12}(l, m) = g_1(l) \left[\sum_{r=1}^{d_m-1} m^r \right] + \left[\sum_{r=1}^{d_l-1} l^r \right] g_2(m) + h(l, m),$$

For $d_l = d_m = 2$,

$$\phi_{12}(l, m) = g_1(l) \ m + l \ g_2(m) + h(l, m)$$

with basis:

$$B = \left[1_p \mid B_1 \mid B_2 \mid B_3 \mid B_4 \mid B_5 \right], \quad (11)$$

where

$$B_3 = m \square B_1$$

$$B_5 = B_2 \square B_1$$

$$B_4 = B_2 \square l$$

Nested PS-ANOVA

$$P = \text{blockdiag}(0, P_1, P_2, P_3, P_4, P_5), \quad (12)$$

Nested PS-ANOVA

Re-express ϕ_{12} :

$$\phi_{12}(l, m) = g_{lr}(l) \left[\sum_{r=1}^{d_m-1} m^r \right] + \left[\sum_{r=1}^{d_l-1} l^{r'} \right] g_{mr'}(m) + h(l, m),$$

For $d_l = d_m = 2$,

$$\phi_{12}(l, m) = g_{l1}(l) m + l g_{m1}(m) + h(l, m)$$

with basis:

$$B = [1_p \mid B_1 \mid B_2 \mid B_3 \mid B_4 \mid B_5], \quad (13)$$

where

$$B_3 = m \square B_1$$

$$B_5 = B_2 \square B_1$$

$$B_4 = B_2 \square l$$

Nested PS-ANOVA

$$P = \text{blockdiag}(0, P_1, P_2, P_3, P_4, P_5)$$

Remove the redundant columns, and give each penalized component its own random effect:

$$B$$