Computational and Algorithmic Supplement A

"Latent Spatial Regimes in Spatio-Temporal State-Space Models: Evidence on Local Price Leadership in Italian Fuel Markets"

Paolo Maranzano

November 2, 2026

This Supplementary Material collects the computational and algorithmic details of the SC-STEM model, with their derivations.

A1 Estimation: the Kalman filter, the EM algorithm and its accelerations

A1.1 The model in state-space form

Within a given regime, or in the pooled fit, STEM is defined by the state-space model

$$z_t = X_t\beta + A\,y_t + e_t, \qquad e_t \sim N(0,\Sigma_e), \quad \Sigma_e = \sigma_\omega^2 C_\theta + \sigma_\epsilon^2 I_d, \tag{A1}$$

$$y_t = G\,y_{t-1} + \eta_t, \qquad \eta_t \sim N(0,\Sigma_\eta), \quad y_0 \sim N(m_0, C_0), \tag{A2}$$

for $t = 1,\dots,T$, with z_t the d observations at time t, A the $d \times p$ loading matrix and C_0 known; in the paper $p = 1$ and $A = \mathbf{1}_d$.

We write $\Sigma_e = \sigma_\omega^2\Gamma$ with $\Gamma = C_\theta + \gamma I_d$ and $\gamma = \sigma_\epsilon^2/\sigma_\omega^2$, and estimate $\psi = (\beta, \sigma_\omega^2, \log\gamma, \log\theta, G, \Sigma_\eta, m_0)$ by the EM algorithm of Shumway and Stoffer (1982): the E-step is the Kalman smoother, the M-step updates G, Σ_η, m_0, σ_ω^2 and β in closed form and $(\log\theta, \log\gamma)$ by Newton–Raphson. The log-likelihood $\ell(\psi)$ is the by-product of the Kalman filter (the prediction-error decomposition).

Treating the states $y_{0:T}$ as missing data, the complete-data log-likelihood is, up to a constant,

$$\begin{aligned}\ell_c(\psi) = &-\tfrac{1}{2}\log|C_0| - \tfrac{1}{2}(y_0 - m_0)^\top C_0^{-1}(y_0 - m_0)\\ &-\tfrac{T}{2}\log|\Sigma_\eta| - \tfrac{1}{2}\sum_{t=1}^{T}(y_t - Gy_{t-1})^\top\Sigma_\eta^{-1}(y_t - Gy_{t-1})\\ &-\tfrac{T}{2}\log|\Sigma_e| - \tfrac{1}{2}\sum_{t=1}^{T}(z_t - X_t\beta - Ay_t)^\top\Sigma_e^{-1}(z_t - X_t\beta - Ay_t).\end{aligned} \tag{A3}$$

The EM algorithm (Dempster et al., 1977) maximizes the observed log-likelihood $\ell(\psi)$ by iterating an E-step, which computes $Q(\psi \mid \psi^{(j)}) = \mathrm{E}\{\ell_c(\psi) \mid z_{1:T}; \psi^{(j)}\}$, and an M-step, which increases Q over ψ.

A1.2 The Kalman filter and smoother

Every algorithm rests on three recursions, run at given parameters.

The filter. Starting from initial values $y_{0|0} = m_0$ and $P_{0|0} = C_0$, for $t = 1, \dots, T$, the quantities are updated as follows:

$$\begin{aligned} y_{t|t-1} &= G\, y_{t-1|t-1}, & P_{t|t-1} &= G P_{t-1|t-1} G^\top + \Sigma_\eta, \\ v_t &= z_t - X_t\beta - A\, y_{t|t-1}, & F_t &= A P_{t|t-1} A^\top + \Sigma_e, \\ y_{t|t} &= y_{t|t-1} + K_t v_t, & P_{t|t} &= P_{t|t-1} - K_t A P_{t|t-1}, \qquad K_t = P_{t|t-1} A^\top F_t^{-1}, \end{aligned}$$

and the log-likelihood follows from the prediction-error decomposition,

$$\ell(\psi) = -\tfrac{1}{2} \sum_{t=1}^{T} \big\{ d \log(2\pi) + \log |F_t| + v_t^\top F_t^{-1} v_t \big\}. \tag{A4}$$

The $d \times d$ matrix F_t is never formed. Σ_e does not depend on t and $A P_{t|t-1} A^\top$ has rank p, so the Woodbury identity and the matrix determinant lemma give

$$\begin{aligned} F_t^{-1} &= \Sigma_e^{-1} - \Sigma_e^{-1} A \big(P_{t|t-1}^{-1} + A^\top \Sigma_e^{-1} A \big)^{-1} A^\top \Sigma_e^{-1}, \\ \log |F_t| &= \log |\Sigma_e| + \log |P_{t|t-1}| + \log \big| P_{t|t-1}^{-1} + A^\top \Sigma_e^{-1} A \big|, \end{aligned}$$

and the Cholesky factor of Σ_e, $\Sigma_e^{-1} A$ and $A^\top \Sigma_e^{-1} A$ are computed once per pass: the cost of a pass is $O(d^3 + T d^2 p)$ instead of $O(T d^3)$. The log-density is evaluated directly, since on a few hundred locations the density itself underflows. When some responses are missing at time t, the rows of the measurement equation that are not observed are dropped (Durbin and Koopman, 2012, Sect. 4.10), and the constants above are cached by pattern of missingness.

The smoother. Let us define the gains $J_t = P_{t|t} G^\top P_{t+1|t}^{-1}$. The fixed-interval smoother runs backwards, for $t = T-1, \dots, 0$,

$$y_{t|T} = y_{t|t} + J_t (y_{t+1|T} - y_{t+1|t}), \qquad P_{t|T} = P_{t|t} + J_t (P_{t+1|T} - P_{t+1|t}) J_t^\top,$$

the last step giving the smoothed initial state $y_{0|T}$, $P_{0|T}$, with $J_0 = C_0 G^\top P_{1|0}^{-1}$.

The lag-one covariance smoother. The covariances $P_{t,t-1|T} = \mathrm{Cov}(y_t, y_{t-1} \mid z_{1:T})$ follow the backward recursion of Shumway and Stoffer (1982):

$$\begin{aligned} P_{T,T-1|T} &= (I - K_T A)\, G\, P_{T-1|T-1}, \\ P_{t-1,t-2|T} &= P_{t-1|t-1} J_{t-2}^\top + J_{t-1} \big(P_{t,t-1|T} - G P_{t-1|t-1} \big) J_{t-2}^\top, \qquad t = T, \dots, 2. \end{aligned}$$

With missing responses at time T, K_T is built on the observed rows, and it is zero when nothing is observed.

A1.3 The EM algorithm for STEM

E-step. Taking the expectation of Equation ((A3)) given the data, Q depends on the states only through the smoothed initial state and

$$\begin{aligned} S_{11} &= \sum_{t=1}^{T} \big(y_{t|T}\, y_{t|T}^\top + P_{t|T} \big), & S_{00} &= \sum_{t=1}^{T} \big(y_{t-1|T}\, y_{t-1|T}^\top + P_{t-1|T} \big), \\ S_{10} &= \sum_{t=1}^{T} \big(y_{t|T}\, y_{t-1|T}^\top + P_{t,t-1|T} \big), & W &= \sum_{t=1}^{T} \big(A P_{t|T} A^\top + r_t r_t^\top \big), \quad r_t = z_t - X_t \beta^{(j)} - A\, y_{t|T}, \end{aligned}$$

so that

$$
\begin{aligned}
-2Q(\psi \mid \psi^{(j)}) = {} & \log|C_0| + \operatorname{tr}\big[C_0^{-1}\{P_{0|T} + (y_{0|T} - m_0)(y_{0|T} - m_0)^\top\}\big] \\
& + T\log|\Sigma_\eta| + \operatorname{tr}\big[\Sigma_\eta^{-1}\big(S_{11} - S_{10}G^\top - GS_{10}^\top + GS_{00}G^\top\big)\big] \\
& + T\log|\Sigma_e| + \operatorname{tr}\big(\Sigma_e^{-1}W\big),
\end{aligned}
\tag{A5}
$$

up to a constant and with β entering the last term through W. With missing responses, W and the response in the update of β are completed by their conditional expectations given the observed data (Section A2).

M-step. The parameters are updated in turn, each given the updates that precede it (an ECM algorithm, Meng and Rubin, 1993): the initial state, $m_0 \leftarrow y_{0|T}$; the state equation, with Σ_η and G diagonal (the default),

$$
[\Sigma_\eta]_{ii} \leftarrow \tfrac{1}{T}\big[S_{11} - S_{10}G^\top - GS_{10}^\top + GS_{00}G^\top\big]_{ii}, \qquad G_{ii} \leftarrow \frac{[\Sigma_\eta^{-1}S_{10}]_{ii}}{[\Sigma_\eta^{-1}S_{00}]_{ii}},
$$

and otherwise $\Sigma_\eta \leftarrow T^{-1}(S_{11} - S_{10}S_{00}^{-1}S_{10}^\top)$ and $G \leftarrow S_{10}S_{00}^{-1}$; the scale of the measurement error, $\sigma_\omega^2 \leftarrow \operatorname{tr}(\Gamma^{-1}W)/(dT)$, at the current (θ, γ); the regression coefficients, by generalized least squares given the smoothed states,

$$
\beta \leftarrow \Big(\sum_{t=1}^{T} X_t^\top \Sigma_e^{-1} X_t\Big)^{-1} \sum_{t=1}^{T} X_t^\top \Sigma_e^{-1}\big(\hat{z}_t - A\, y_{t|T}\big),
$$

with $\hat{z}_t$ the completed response and $\Sigma_e = \sigma_\omega^2\Gamma$ at the updated σ_ω^2; and the spatial parameters, by Newton–Raphson on $(\log\theta, \log\gamma)$ for

$$
Q_e(\theta, \gamma) = T\log\big|\sigma_\omega^2\Gamma\big| + \operatorname{tr}\big\{(\sigma_\omega^2\Gamma)^{-1}W\big\},
$$

minus twice the part of Q that depends on them, safeguarded as described in Section A1.10. Finally $\sigma_\epsilon^2 = \gamma\,\sigma_\omega^2$. The statistic W is computed once per iteration, at $\beta^{(j)}$, so that the updates are conditional maximizations of Q, each in its own block at the current values of the others. Every update maximizes Q in its own block or, for the Newton step, does not decrease it: an iteration does not maximize Q jointly, but none of its steps decreases it, so the algorithm is a generalized EM algorithm and ℓ does not decrease along the iterations.

A1.4 Stopping rule

The EM algorithm of Section A1.3, and each of the accelerated algorithms described below, stops by the same rule. Two criteria are computed at every iteration: the largest relative change of a free parameter, taken one at a time,

$$
\max_l \frac{|\psi_l^{(j+1)} - \psi_l^{(j)}|}{\max(|\psi_l^{(j)}|, 10^{-3})},
$$

on the scale on which the algorithm updates it and with the loading matrix and C_0 excluded, and the absolute change of the log-likelihood, $|\ell(\psi^{(j+1)}) - \ell(\psi^{(j)})|$. The algorithm stops as soon as either falls below its tolerance, or after a maximum number of iterations: 10^{-3} for both criteria and 500 iterations in the pooled fit and in the final refits of SC-STEM; 10^{-2} on the parameters, 1 on the log-likelihood and 50 iterations for the fits inside the alternation of SC-STEM, described next. The criterion on ℓ is absolute because the differences that matter, between fits and in the information

criteria, are absolute, whereas a relative criterion loosens as the log-likelihood grows with the size of the data. Tolerances of 10^{-3} on both criteria left the estimates within 1 to 2% of those at 10^{-4} and the selected models unchanged in the simulation study, while 10^{-2} moved the range and the variances by tens of percent.

A1.5 The EM algorithm within SC-STEM

SC-STEM fits one STEM model per regime, on the locations $\mathcal{C}_k$ with d_k, the rows A_k of the loading matrix and the parameters ψ_k, and alternates between the parameters and the labels (Algorithm 1 of the manuscript). The EM algorithm enters in two places:

1. In the parameter step of the alternation, where it only has to rank the regimes for the label step: a few plain EM (or ECME) iterations per regime, with tolerances 10^{-2} and 1 and at most 50 iterations, without acceleration. The first time, a regime starts from the least-squares coefficients of its own locations (an unpenalized fit) or from the final refit of the unpenalized solution the fit starts from (a penalized one); every later time, from its own estimates of the previous iteration. The label step scores each location with the smoothed states and the measurement parameters of each regime.

2. In the final refit, where the regimes are re-estimated once on the final partition, with the settings of the pooled fit and the algorithm chosen, starting from the estimates of the last iteration of the alternation. The reported estimates and log-likelihoods come from this refit. Fits that end at the same partition share one refit (Section 3.1 of the manuscript).

A1.6 Where the EM algorithm is slow

This subsection explains why the EM algorithm can need many iterations, and what this implies for the stopping rule of Section A1.4; it motivates the accelerations described in the following subsections.

The rate of convergence. Each iteration of the EM algorithm maps the current parameters into new ones, $\psi^{(j+1)} = M(\psi^{(j)})$, and the maximum $\hat{\psi}$ is the point that this map leaves unchanged. Near the maximum the map is approximately linear,

$$\psi^{(j+1)} - \hat{\psi} \approx J(\psi^{(j)} - \hat{\psi}), \qquad J = I_c^{-1} I_m,$$

with I_c the complete-data information and $I_m = I_c - I_o$ the information that is missing because the states are not observed (Dempster et al., 1977; Meng and Rubin, 1994). The matrix J says how much of the current distance from the maximum survives one iteration. Its eigenvalues lie in $[0, 1)$, and the largest of them, λ, is the *rate of convergence*: along the corresponding direction of the parameter space, every iteration removes only the fraction $1 - \lambda$ of the distance that remains, so that reducing it by a factor ε takes about $\log \varepsilon / \log \lambda$ iterations. When the unobserved states carry most of the information about a direction, the fraction of missing information is large, λ is close to one, and the algorithm is slow along that direction.

What a slow direction does to the stopping rule. The stopping rule of Section A1.4 stops the iterations when the change between two successive iterations, of the parameters or of the

log-likelihood, falls below its tolerance. Along a direction with rate λ the change of one iteration is only a fraction of the distance still to go:

$$\|\psi^{(j)} - \hat{\psi}\| \approx \frac{\|\psi^{(j+1)} - \psi^{(j)}\|}{1-\lambda}, \qquad \ell(\hat{\psi}) - \ell(\psi^{(j)}) \approx \frac{\ell(\psi^{(j+1)}) - \ell(\psi^{(j)})}{1-\lambda^2}.$$

The first relation concerns the criterion on the parameters, the second the criterion on the log-likelihood; they are not further rules, but the error that each criterion leaves when it stops the iterations. A tolerance τ on the relative change of the parameters leaves a relative error of about $\tau/(1-\lambda)$, and a tolerance on the change of the log-likelihood leaves a shortfall of about that tolerance divided by $1-\lambda^2$. The rule is the same in every fit, but the accuracy it buys is not: where λ is close to one, the iterations stop farther from the maximum. Removing the slow directions, which is what the accelerations below do, therefore makes the fits both faster and closer to the maximum at the same tolerance.

On the regimes of SC-STEM four situations make λ close to one or the objective flat:

1. *The split of the common variation between the latent process and the field.* The latent process is common to the locations of a regime; the field, with a range comparable to the size of the regime, is nearly so at every time. Only their dynamics tell them apart, the latent process being persistent and the field white in time, and the unobserved states carry most of the information about the split. The direction along which $(\sigma^2_\omega, \sigma^2_\epsilon, \theta, \Sigma_\eta)$ move together is slow, and it sets the number of iterations.

2. *The intercept and the initial level of the latent process.* The mean of the data is $X_t\beta + AG^t m_0$: a change of m_0 moves the first periods, which the intercept and the smoothed states partly absorb. The E-step attributes a level error mostly to the smoothed states, the M-step then updates m_0 and β given them and corrects only a small part of the error at every iteration. This direction is the slowest, but it carries little likelihood; it is where the error at the stop sits. When G is close to zero, m_0 enters the likelihood only through Gm_0, and G and m_0 trade along a ridge of the likelihood.

3. *The range at the boundary of what the distances identify.* On a small set of locations the likelihood is flat in θ beyond two limits (Section A1.10); the Newton–Raphson step on a flat objective took steps of any size, and every later iteration paid repeated grid searches.

4. *A maximum on the boundary of the parameter space.* When the data do not support a latent process, as when a pooled model is fitted to regimes whose latent processes differ, the maximum lies at $\Sigma_\eta = 0$. Near a boundary the EM algorithm converges sublinearly, every EM-type algorithm is slow, and on the logarithmic scale on which the accelerations extrapolate the boundary is never reached: the iterations stop close to it, with a large relative error on Σ_η and a small absolute one.

A1.7 Estimation algorithms

The first two slow directions of Section A1.6 can be addressed in two ways: by changing what the M-step maximizes, or by combining successive iterations. Together with the EM algorithm of Section A1.3, this gives four algorithms: ECME changes the M-step, SQUAREM combines the iterations, and SQUAREM-ECME does both. This section compares them; Sections A1.8 and A1.9 build ECME and SQUAREM step by step. The four algorithms maximize the same likelihood and share the same E-step: the Kalman filter and smoother compute the exact conditional moments of

the latent states given the data, which in a linear Gaussian state-space model is exact posterior inference. They differ in what the M-step maximizes and in how successive iterations are combined.

- **EM** maximizes the expected complete-data log-likelihood block by block. In the language of machine learning it is coordinate ascent on the evidence lower bound, made tight by the E-step. It is monotone and simple, and it converges linearly, slowly wherever the unobserved states carry most of the information about a direction of the parameters.
- **ECME** updates the regression coefficients and the initial level of the latent process on the observed log-likelihood, with the states integrated out (a "collapsed" update, computed as a generalized least-squares step by the Kalman filter run on the covariates). It removes the slow direction in which the intercept and the initial level move together, at about 25% more time per iteration, and leaves the other directions as they are.
- **SQUAREM** extrapolates two successive EM iterations along the direction in which they converge most slowly, an extrapolation of the fixed-point map in the family of Aitken's and Anderson's accelerations, with a monotonicity safeguard that rejects any extrapolation that would lower the likelihood. It changes neither the E-step nor the M-step; Section A1.9 builds it step by step.
- **SQUAREM-ECME** extrapolates ECME iterations in the same way.

Table A1 sets the four algorithms side by side showing what they share, and where each departs from the EM algorithm.

Table A2 compares the computational performances of the four algorithms using simulated data. Accordingly, SQUAREM is keep as the default algorithm: it took about a third of the time of EM on single fits and about half on whole SC-STEM grids, stopped closer to the maximum, and selected the same models. The four algorithms converge to the same maximum, so the choice changes the cost and the error left at a given tolerance, not the estimator. ECME buys accuracy rather than speed: it removes the error left on the intercepts and on m_0, and on G when G is near zero, but not the slow split of the variance, which sets the number of iterations. SQUAREM buys speed, and with it accuracy at a given tolerance; on the ECME map it combines the two. No algorithm cures a maximum on the boundary of the parameter space, where every EM-type algorithm is slow, and only ECME addresses the ridge between G and m_0.

A1.8 ECME: expectation/conditional maximization either

The "expectation / conditional maximization either" (ECME) algorithm (Liu and Rubin, 1994), keeps the E-step of the EM algorithm and almost all of its M-step: it changes one block of the M-step, the update of the regression coefficients and of the initial level of the latent process, $\delta = (\beta^\top, m_0^\top)^\top$, which it takes on the observed log-likelihood ℓ instead of on the expected complete-data log-likelihood Q. The construction takes five steps.

Step 1: why the EM algorithm is slow on δ. The EM algorithm updates β and m_0 given the smoothed states (Section A1.3). Suppose that the current intercept is too low by some amount. The E-step attributes most of that error to the latent process, whose smoothed path moves up to absorb it; the M-step then updates β and m_0 given that path, and corrects only the small part of the error that the path did not absorb. Every iteration removes a small fraction of the error: this is the second slow direction of Section A1.6, with a rate close to one. The cause is that Q sees δ through the states, and the states and δ explain the same level of the data.

Table A1: The four algorithms at a glance: what they share and where they differ. M is one iteration (an E-step followed by an M-step); Sections A1.3, A1.8 and A1.9 give the details.

	EM	ECME	SQUAREM	SQUAREM-ECME
E-step	the same in all four: the Kalman filter and smoother, exact conditional moments of the latent states			
Σ_η, G, σ^2_ω, (θ, γ)	the same in all four: conditional maximizations of Q, with the safeguarded Newton–Raphson step and the limits of Section A1.10 for (θ, γ)			
(β, m_0)	maximize Q given the smoothed states	maximize ℓ: a GLS step by the augmented Kalman filter	as EM	as ECME
one iteration M	the EM map	the ECME map	the EM map	the ECME map
how the iterations are combined	each starts from the last, $\psi^{(j+1)} = M(\psi^{(j)})$	as EM	cycles: two iterations, an extrapolation, a stabilizing iteration (Algorithm 2)	as SQUAREM, on the ECME map
cost of one M	one filter and smoother pass	one more filter pass (about +25%)	as EM	as ECME
monotone because	no block decreases Q (generalized EM)	the steps on ℓ follow those on Q	an extrapolation that lowers ℓ is rejected	as SQUAREM
slow directions removed (Section A1.6)	none	intercept and m_0, and the ridge between G and m_0	the slowest one, in practice the split of the variance	both
not cured	by none of the four: a maximum on the boundary of the parameter space; by EM and SQUAREM: the ridge between G and m_0			
inside the alternation of SC-STEM	EM	ECME	EM, not accelerated	ECME, not accelerated
limit	the same maximum: the four have the same fixed points			

Step 2: on the observed likelihood the problem is a regression. Given all the other parameters, ℓ depends on δ only through the mean of the data,

$$\mathrm{E}(z_t) = X_t\beta + AG^t m_0,$$

which is linear in δ, while the covariance of the data does not depend on δ at all. As a function of δ, ℓ is therefore the log-likelihood of a Gaussian linear regression with a known covariance, the one the model implies, latent process and spatial field included: a quadratic in δ, maximized by the generalized least-squares (GLS) estimator. Integrating the states out is what removes the trade between the states and the level, and the GLS step finds the level at once.

Step 3: GLS without the $dT \times dT$ covariance. The covariance of all the data is $dT \times dT$ and cannot be formed on a real network. The augmented Kalman filter computes the same estimator from the filter (de Jong, 1991); see also Durbin and Koopman (2012, Ch. 6). Two facts make it work: the gains and the innovation covariances F_t do not depend on the data, and the filter is linear in its data and in its initial mean. The filter of the current parameters is run on every column of the $d \times (1 + r + p)$ matrix

$$\left[\, z_t,\ X_t,\ AG^t \,\right],$$

the first column with the current m_0 as initial mean and the others with zero, the last block standing for a change Δm_0 of m_0. All the columns share the gains and F_t; with $[\, e_t,\ E_t \,]$ their innovations,

Table A2: The four algorithms: time and iterations relative to EM on whole SC-STEM grids (six cells of the simulation study, two replications each); shortfall: median shortfall of the log-likelihood at the stop on single fits. All at the default tolerances. All four are monotone; the two accelerated ones through a safeguard on the extrapolation.

algorithm	M-step	iterations combined	time	iterations	shortfall
EM	conditional maximizations of Q	plain	1	1	0.020
ECME	as EM, with (β, m_0) on ℓ	plain	1.17	0.96	0.017
SQUAREM	as EM	two extrapolated	0.53	0.41	0.0027
SQUAREM-ECME	as ECME	two extrapolated	0.62	0.39	0.0023

linearity gives the innovations of z_t at $(\beta, m_0 + \Delta m_0)$ as $e_t - E_t\delta$, with $\delta = (\beta^\top, \Delta m_0^\top)^\top$, and the prediction-error decomposition gives

$$\ell(\delta) = c - \tfrac{1}{2}\sum_{t=1}^{T}(e_t - E_t\delta)^\top F_t^{-1}(e_t - E_t\delta). \tag{A6}$$

Step 4: the update. Setting the gradient of (A6) to zero gives the GLS normal equations,

$$\hat\delta = M^{-1}v, \qquad M = \sum_{t=1}^{T} E_t^\top F_t^{-1} E_t, \qquad v = \sum_{t=1}^{T} E_t^\top F_t^{-1} e_t,$$

with F_t^{-1} applied through the Woodbury identity and the missing rows dropped, as in the filter; then $m_0 \leftarrow m_0 + \Delta m_0$. Without a latent process there is no m_0 column.

Step 5: the order keeps it monotone. Within an iteration the updates of Σ_η, G, σ_ω^2 and (θ, γ) on Q come first, exactly as in the EM algorithm, and the update of δ on ℓ comes last. The steps on Q do not decrease ℓ, by the argument of the EM algorithm, and the last step maximizes ℓ over δ given the rest, so it does not decrease it either: the algorithm is monotone (Liu and Rubin, 1994). Had the step on ℓ come before steps on Q, the E-step would have to be recomputed in between, as in the AECM algorithm of Meng and van Dyk (1997).
Algorithm 1 puts the steps together.

Algorithm 1 One iteration of ECME, from ψ. The lines marked $\star$ are the ones that differ from an iteration of the EM algorithm, which instead sets $m_0 \leftarrow y_{0|T}$ and updates β by GLS given the smoothed states.

1: E-step: the Kalman filter and smoother at ψ, giving S_{11}, S_{10}, S_{00} and W ▷ as EM
2: update Σ_η and G from S_{11}, S_{10}, S_{00} ▷ as EM
3: update σ_ω^2 from W ▷ as EM
4: update (θ, γ) by the safeguarded Newton–Raphson step on Q_e ▷ as EM
5: run the filter at the new parameters on $[\, z_t,\ X_t,\ AG^t \,]$, giving e_t, E_t and F_t ▷ $\star$
6: $M \leftarrow \sum_t E_t^\top F_t^{-1} E_t$; $\quad v \leftarrow \sum_t E_t^\top F_t^{-1} e_t$ ▷ $\star$
7: $(\beta, \Delta m_0) \leftarrow M^{-1}v$; $\quad m_0 \leftarrow m_0 + \Delta m_0$ ▷ $\star$ maximizes ℓ over δ

Pros and cons. The step removes the second slow direction of Section A1.6 at once, and with it the ridge between G and m_0; it does not change the first, the split of the variance, which sets the number of iterations. An ECME iteration costs one more filter pass than an EM iteration, about 25% more time. ECME therefore buys accuracy rather than speed: at the same tolerance it stops closer to the maximum on the intercepts, on m_0 and on G near zero (Section A1.7).

A1.9 SQUAREM: squared extrapolation methods

SQUAREM (Roland and Varadhan, 2005; Varadhan and Roland, 2008) does not change the E-step or the M-step: it changes only how successive EM iterations are combined. Write M for one whole EM iteration, the E-step followed by the M-step, so that the EM algorithm is the repeated application $\psi^{(j+1)} = M(\psi^{(j)})$ and its maximum $\hat{\psi}$ is a fixed point, $M(\hat{\psi}) = \hat{\psi}$. SQUAREM uses the same map M (an EM iteration, or an ECME iteration for SQUAREM-ECME), but instead of taking the last iterate as the next starting point it combines the last iterates into a point farther along the path the iterations follow. The construction takes six steps.

Step 1: the error of EM shrinks geometrically. Near the maximum the error $e_j = \psi^{(j)} - \hat{\psi}$ obeys $e_{j+1} \approx J e_j$ (Section A1.6). Along one direction with rate λ, $e_j = \lambda^j e_0$: every iteration removes the same fraction $1 - \lambda$ of the error that is left. Reducing the error by a factor ε takes $\log \varepsilon / \log \lambda$ iterations, for instance 135 iterations to divide it by 1000 when $\lambda = 0.95$. This is the cost that SQUAREM attacks.

Step 2: three successive iterates reveal the rate and the limit. Start from the current point ψ_0 and run two plain iterations, $\psi_1 = M(\psi_0)$ and $\psi_2 = M(\psi_1)$, as EM would. Their first and second differences are observable, and along a direction with rate λ they are

$$r = \psi_1 - \psi_0 = (\lambda - 1)\, e_0, \qquad w = \psi_2 - 2\psi_1 + \psi_0 = (\lambda - 1)^2\, e_0.$$

The first is the EM step, the second says how the step shrinks from one iteration to the next. Their ratio gives the unknown rate, $\|r\|/\|w\| = 1/(1 - \lambda)$, and in one dimension $e_0 = r^2/w$, so the limit is $\hat{\psi} = \psi_0 - r^2/w$: this is Aitken's Δ^2 extrapolation. In words, EM computes ψ_1 and ψ_2 anyway, and SQUAREM uses them to estimate where the sequence is going instead of following it one step at a time.

Step 3: the extrapolated point. SQUAREM moves to

$$\psi' = \psi_0 + 2\alpha\, r + \alpha^2 w, \tag{A7}$$

with a steplength $\alpha \geq 1$. Substituting r and w from Step 2,

$$\psi' - \hat{\psi} = \big\{1 - \alpha(1 - \lambda)\big\}^2 e_0.$$

Two values of α explain the formula. With $\alpha = 1$, $\psi' = \psi_0 + 2r + w = \psi_2$: two plain EM iterations, so EM is the special case $\alpha = 1$ of SQUAREM. With $\alpha = 1/(1 - \lambda)$ the error is zero and ψ' is the limit itself, reached in one cycle instead of in $\log \varepsilon / \log \lambda$ iterations. Since $1/(1 - \lambda) = \|r\|/\|w\|$, SQUAREM takes

$$\alpha = \min\Big\{\alpha_{\max},\ \max\Big(1, \frac{\|r\|}{\|w\|}\Big)\Big\},$$

the scheme S3 of Varadhan and Roland (2008); with $\lambda = 0.95$, $\alpha = 20$, and in the exactly linear case one cycle reaches the point that EM approaches only after more than a hundred iterations.

The name comes from the error factor, which is the square of the factor $1 - \alpha(1 - \lambda)$ of a single extrapolated step: the method squares a one-step extrapolation.

Step 4: several directions at once. The parameters move in many directions with different rates, and one α cannot be exact for all of them. Because α is computed from the norms of r and w, it is dominated by the slowest direction, whose components of r and w are the largest after a few iterations; that is the direction that costs EM its iterations (Section A1.6), and the extrapolation removes most of it. A faster direction may be overshot, but its error was already small, and the plain iteration that follows the extrapolation (Step 6) reduces it again quickly.

Step 5: the scale on which to extrapolate. A straight move such as (A7) could give a negative variance or a range outside its limits. The extrapolation is therefore computed on a scale on which every value is admissible,

$$u(\psi) = \big(\beta,\ m_0,\ \operatorname{vec} G,\ \log \operatorname{diag} \Sigma_\eta,\ \log \sigma^2_\omega,\ \log \gamma,\ \log \theta\big)$$

(the logarithm of the diagonal of the Cholesky factor of Σ_η and its off-diagonal elements when Σ_η is not diagonal): r, w and (A7) are computed on $u_i = u(\psi_i)$, the result is mapped back to ψ', and the extrapolated θ is kept within the limits of Section A1.10.

Step 6: safeguards to keep monotonicity as the EM. An EM iteration never decreases the likelihood; an extrapolation can. Four safeguards keep SQUAREM monotone. (a) $\alpha \geq 1$: a cycle never does less than two plain iterations. (b) $\alpha \leq \alpha_{\max}$: the bound starts at 1, grows by a factor 4 every time it binds and shrinks by the same factor when an extrapolation at the bound is rejected, as in the implementation of Du and Varadhan (2020), so the method trusts the extrapolation more as it keeps succeeding. (c) Stabilization: one more plain iteration is run from the extrapolated point, $\psi_3 = M(\psi')$, which also returns $\ell(\psi')$, since every iteration evaluates the log-likelihood at its input by the Kalman filter. (d) The check: the cycle ends at ψ_3 if $\ell(\psi') \geq \ell(\psi_0)$, and at ψ_2 otherwise, whose increase over ψ_0 the EM algorithm guarantees. The check costs nothing, since $\ell(\psi')$ comes with the stabilizing iteration. Algorithm 2 puts the steps together.

What SQUAREM does not change. The E-step and the M-step are those of the EM algorithm (or of ECME, for SQUAREM-ECME). A point at which a cycle stops moving has $r = 0$, that is $M(\psi_0) = \psi_0$, so SQUAREM has the same fixed points as EM and converges to the same maximum: it changes the cost, not the estimator. The stopping rule (Section A1.4) is applied at the start of every cycle, to the step from ψ_0 to ψ_1, and the number of iterations reported counts every call of M, three per extrapolated cycle. A cycle costs three EM iterations and pays off when λ is close to one, which is the case on the regimes of SC-STEM (Section A1.6). SQUAREM is not used inside the alternation of SC-STEM: there the fits stop after a few iterations, where the cycles cost more than they save, and extrapolations from points far from convergence were seen to change the partitions the alternation reaches, so the plain iterations are run.

A1.10 The limits of the range

The distances between the locations identify the range only within limits. When the correlation between the two closest locations falls below 0.05, the field is white within the set of locations and cannot be told from the nugget; when the correlation between the two farthest exceeds 0.95, the field is constant over the locations at every time and cannot be told from an effect common to

Algorithm 2 One cycle of SQUAREM, starting from ψ_0. The lines marked $\star$ are the ones that differ from two iterations of EM; with $\alpha = 1$ the cycle is exactly two EM iterations.

$\psi_1 \leftarrow M(\psi_0)$ ▷ an EM iteration; it also returns $\ell(\psi_0)$
apply the stopping rule of Section A1.4 to the step from ψ_0 to ψ_1
$\psi_2 \leftarrow M(\psi_1)$ ▷ a second EM iteration
$u_i \leftarrow u(\psi_i)$, $i = 0, 1, 2$; $\quad r \leftarrow u_1 - u_0$; $\quad w \leftarrow u_2 - 2u_1 + u_0$ ▷ $\star$ the differences
$\alpha \leftarrow \min\{\alpha_{\max}, \max(1, \|r\|/\|w\|)\}$ ▷ $\star$ the steplength
if $\alpha > 1$ **then**
  $\psi' \leftarrow u^{-1}(u_0 + 2\alpha\, r + \alpha^2 w)$, with θ within its limits ▷ $\star$ the extrapolation
  $\psi_3 \leftarrow M(\psi')$ ▷ $\star$ stabilization; it also returns $\ell(\psi')$
  if $\ell(\psi') \geq \ell(\psi_0)$ **then**
    return ψ_3 ▷ $\star$ accepted
  else
    return ψ_2, and shrink $\alpha_{\max}$ if $\alpha = \alpha_{\max}$ ▷ $\star$ rejected: two EM iterations
  end if
end if
return ψ_2 ▷ $\alpha = 1$: two EM iterations

them, the latent process included. Beyond either limit Q_e and ℓ are flat in θ. The SC-STEM model therefore keeps

$$\theta_{\min} = \frac{-\log 0.95}{h_{\max}} \;\leq\; \theta \;\leq\; \theta_{\max} = \frac{-\log 0.05}{h_{\min}}, \tag{A8}$$

with $h_{\min}$ and $h_{\max}$ the smallest and largest distance between two locations of the fit (of the regime, in SC-STEM), in the Newton–Raphson step and in the extrapolation of SQUAREM. The Newton–Raphson step on $(\log\theta, \log\gamma)$ is safeguarded in two further ways. A Newton step $-H^{-1}g$ is taken only when the Hessian H of Q_e is positive definite ($H_{11} > 0$, $\det H > 10^{-3}$), and is accepted only if it does not increase Q_e, halving it otherwise up to ten times. When H is not positive definite, the point moves to the best of a grid of ten by ten values, from 0.01 to 10 times the current θ and γ within the limits, if that lowers Q_e, and the iterations resume from there; a second Hessian that is not positive definite ends them, since Q_e is then flat. Otherwise the Newton–Raphson iterations stop when the relative change of $(\log\theta, \log\gamma)$ falls below the tolerance of the EM algorithm, or after 50 iterations. A fit whose range ends at a limit reports it; the range is then not estimated freely, though it is still counted among the parameters of the information criteria. The limits and the safeguards apply to all four algorithms: they act in the Newton–Raphson step of the M-step, which the four algorithms share, and, under SQUAREM and SQUAREM-ECME, in the extrapolation.

Without the limits and the safeguards, the Newton–Raphson step on a flat Q_e took steps of any size (θ was seen to move from 0.02 to e^{-700} in a single M-step), every later iteration paid up to thirty grid searches of a hundred evaluations each, and SQUAREM, extrapolating along the flat direction, ran to the limit on the iterations. Where neither the limits nor the safeguards come into play, the iterations are those of the unconstrained algorithm.

A2 Missing observations

Section 3.1 of the manuscript summarizes the treatment of the gaps in the response. This section gives fully describes the filtering and smoothing, and the expected complete-data sufficient statistic of the measurement equation when part of the response is unobserved. Throughout, the regime

index k is fixed and suppressed, A denotes the $d \times p$ loading matrix, Σ_e the measurement covariance, and $\mathcal{Z}$ the whole of the observed data.

A2.1 Filtering and smoothing

The state-space form naturally accommodates the presence of missing data in the response variable (Durbin and Koopman, 2012, Sections 2.7 and 4.10). Suppose that at time t only some of the elements of z_t are observed. Let W_t be the selection matrix whose rows are the corresponding rows of the identity, and write $z_t^* = W_t z_t$. The measurement equation of the regime is replaced at that time point by

$$z_t^* = W_t X_t \beta + W_t A\, y_t + e_t^*, \qquad e_t^* \sim N\Big(0,\ W_t \Sigma_e W_t^\top\Big)\,, \tag{A9}$$

and the Kalman recursions proceed exactly as in the complete case on an observation vector whose dimension varies over time. When nothing is observed at t the update is skipped altogether (which Durbin and Koopman (2012) obtain by setting the observation matrix to zero) so the filtered moments equal the predicted ones and the time point contributes no term to the likelihood, which therefore remains the exact likelihood of the observed data by the prediction-error decomposition. The backward smoothing recursions require no modification whatsoever, since they read the filtered moments and the transition matrix and never the data. The point that needs care is the M-step, and it is where a naive implementation goes wrong: the EM algorithm maximizes the expected *complete-data* log-likelihood, so the sufficient statistics of the measurement equation must be *completed*, not truncated. The rest of this section derives them.

A2.2 Setting

The measurement equation is $z_t = X_t\beta + Ay_t + e_t$ with $e_t \sim N(0, \Sigma_e)$, independent across t and of the latent process. Partition the locations at time t into the observed set O and the missing set M, and write the corresponding blocks of any vector or matrix with those subscripts. The EM algorithm treats both the latent states *and* the unobserved responses as missing data, so the E-step must return

$$\mathrm{E}\Big[e_t e_t^\top \,\Big|\, \mathcal{Z}\Big]\,, \qquad \text{and} \qquad \mathrm{E}[z_t \,|\, \mathcal{Z}]\,, \tag{A10}$$

the first entering the updates of the variance components and of the spatial covariance parameters, the second the update of β.

The Kalman smoother, run on the selected measurement equation of Durbin and Koopman (2012, Sect. 4.10), supplies

$$\hat{y}_t = \mathrm{E}[y_t \mid \mathcal{Z}]\,, \qquad C_t^s = \mathrm{Var}[y_t \mid \mathcal{Z}]\,. \tag{A11}$$

A2.3 The conditional law of the measurement error

Conditionally on the latent state y_t, the measurement error is $e_t = z_t - X_t\beta - Ay_t$, so its observed block $e_{O,t}$ is a known linear function of y_t once $z_{O,t}$ is observed, whereas its missing block is not observed at all. Because e_t is Gaussian with covariance Σ_e, the standard conditioning formula gives

$$e_{M,t} \mid e_{O,t} \ \sim\ N(P\, e_{O,t},\ \Omega)\,, \qquad P = [\Sigma_e]_{MO}\,([\Sigma_e]_{OO})^{-1}\,, \qquad \Omega = [\Sigma_e]_{MM} - P\,[\Sigma_e]_{OM}\,, \tag{A12}$$

and this conditional law does not depend on y_t: the state enters $e_{O,t}$ but not the regression of $e_{M,t}$ on it. Durbin and Koopman (2012) note that a missing element of the observation vector can be estimated by the corresponding element of $A_t\hat{y}_t$; that is exact when Σ_e is diagonal, in which case

$P = 0$, and it is not exact here, where the exponential spatial covariance couples the locations and $P \neq 0$.

Introduce the $d \times |O|$ matrix S that lifts a vector defined on the observed rows to the full vector,

$$S_{O\cdot} = I_{|O|}, \qquad S_{M\cdot} = P, \tag{A13}$$

and let $\tilde{\Omega}$ be the $d \times d$ matrix with Ω in the (M, M) block and zeros elsewhere. Then (A12) can be written compactly as

$$e_t \mid e_{O,t} \;\sim\; N\Big(S\, e_{O,t},\; \tilde{\Omega}\Big)\,. \tag{A14}$$

A2.4 The completed second moment

Write $r_{O,t} = z_{O,t} - X_{O,t}\beta - A_O\hat{y}_t$ for the residual on the observed rows, evaluated at the smoothed state. Applying the tower property with respect to $y_t \mid \mathcal{Z}$,

$$\mathrm{E}[e_t \mid \mathcal{Z}] = \mathrm{E}[\,\mathrm{E}[e_t \mid y_t, \mathcal{Z}]\,|\,\mathcal{Z}] = S\, r_{O,t}, \tag{A15}$$

$$\mathrm{Var}[e_t \mid \mathcal{Z}] = \underbrace{\mathrm{E}[\mathrm{Var}(e_t \mid y_t, \mathcal{Z})|\mathcal{Z}]}_{=\,\tilde{\Omega}} + \underbrace{\mathrm{Var}[\mathrm{E}(e_t \mid y_t, \mathcal{Z})|\mathcal{Z}]}_{=\,SA_OC_t^sA_O^\top S^\top}, \tag{A16}$$

where the second term follows because, given y_t, the conditional mean of e_t is $S\,(z_{O,t} - X_{O,t}\beta - A_O y_t)$, an affine function of y_t with matrix $-SA_O$, whose variance under the smoothing distribution is $SA_OC_t^sA_O^\top S^\top$. Combining (A15) and (A16), we obtain

$$\mathrm{E}\Big[e_te_t^\top \,\Big|\, \mathcal{Z}\Big] \;=\; S\,A_O\,C_t^s\,A_O^\top S^\top \;+\; \tilde{\Omega} \;+\; (Sr_{O,t})\,(Sr_{O,t})^\top\,. \tag{A17}$$

- If nothing is missing, then $M = \emptyset$, $S = I_d$ and $\tilde{\Omega} = 0$, and (A17) reduces to $AC_t^sA^\top + r_tr_t^\top$, the complete-data statistic.
- If nothing is observed, then $O = \emptyset$, S has no columns and $\tilde{\Omega} = \Sigma_e$, so (A17) returns Σ_e: with no information at that time point the measurement error keeps its unconditional variance, and contributes nothing else.

A2.5 The completed observation and the update of β

From (A15), the conditional expectation of the observation vector is

$$\hat{z}_t = \mathrm{E}[z_t \mid \mathcal{Z}] = X_t\beta + A\hat{y}_t + S\,r_{O,t}, \tag{A18}$$

which equals z_t on the observed rows and equals $X_{M,t}\beta + A_M\hat{y}_t + Pr_{O,t}$ on the missing ones. Differentiating the expected complete-data log-likelihood with respect to β and setting the derivative to zero gives

$$\hat{\beta} = \left(\sum_{t=1}^{T} X_t^\top\Sigma_e^{-1}X_t\right)^{-1} \sum_{t=1}^{T} X_t^\top\Sigma_e^{-1}\,(\hat{z}_t - A\hat{y}_t)\,, \tag{A19}$$

so the only change with respect to the complete-data update is the replacement of z_t by $\hat{z}_t$: the design matrix and the covariance keep their full dimension. For the same reason the divisor of the variance update remains the complete-data count dT and not the number of observed values, since (A17) already carries the conditional variance of whatever was not observed.

A2.6 Why the spatial term matters

The conditional law (A12) is the point at which the spatio-temporal case departs from the obvious. Durbin and Koopman (2012) remark that a missing element can be estimated by the corresponding element of the fitted signal; that is exact when the measurement covariance is diagonal, and it is not our case. Here Σ_e couples the locations, so the conditional expectation of a missing observation is its signal *plus* the part of the measurement error predicted from the neighbors observed at the same instant — the same algebra as kriging, applied at a fixed time point. Ignoring that term would bias the variance components downward, by attributing to noise a residual that the model can in fact explain spatially.

A2.7 Consequences elsewhere

The rest of the apparatus follows. The kriging predictor conditions on the sub-vector observed at the chosen time point rather than on the whole of z_t. The assignment score of Section 2.3 of the manuscript is evaluated on the time points at which the location was observed, with T replaced by the count T_i of those points; since the missingness pattern of a location does not depend on the regime it is being scored against, the scores stay comparable across regimes, which is all the label step requires. The parametric bootstrap of Section 3.3 of the manuscript reimposes the observed pattern of gaps on every replicate, since a replicate with a complete response would understate the uncertainty of a fit obtained from an incomplete one. The only structural requirement is that every location retain at least one observation, since a location observed nowhere carries no information about which regime it belongs to.

A2.8 Cost

The matrices P and Ω depend on the time point only through its missingness pattern, so they are computed once per distinct pattern and cached. In a monitoring network a station is typically out of service for a stretch of consecutive days, so the number of distinct patterns is far smaller than T, and the additional cost of the treatment is negligible relative to the filtering pass.

A3 Initialization and the scale of the penalty

This section treats the start of a fit and the scale of the Potts penalty together, because the scale depends on the start. The scale c is computed once, at the first sweep of a fit, from the scores of the locations under the regimes of its first parameter step (Equation (8) of the manuscript), and those regimes depend on the partition from which the fit starts. Algorithm 1 of the manuscript states two rules of initialization. An unpenalized fit starts from the k-means of the departures of the locations from the pooled model. A fit with $\phi > 0$ starts from the partition to which the fit with $\phi = 0$ at the same K has converged. The second rule is what ties the two topics: it makes every penalized fit at a given K measure c on the same fitted unpenalized regimes, so that the penalties of a grid are expressed on one common scale and can be compared (Proposition A3.1). This section defines the departures and says why they, and not the covariate means, carry information on the regimes; then it states what the second rule does to the scale of the penalty, why a penalized fit cannot be started like an unpenalized one, and what follows for the grid of ϕ.

A3.1 The start of an unpenalized fit: departures from the pooled model

Let $\hat{\mu}_{it} = x'_{it}\hat{\beta} + A_i\hat{y}_t$ be the signal of the pooled fit and $r_{it} = z_{it} - \hat{\mu}_{it}$ the residual of location i. With $\tilde{x}_{it}$ the covariates that vary, regress r_{it} on $(1, \tilde{x}_{it})$ over the periods at which i is observed, with slopes b_i and residuals u_{it}. The features of location i are

$$f_i = \Big(\bar{r}_i,\ b_i,\ \operatorname{corr}(u_{it}, u_{i,t-1}),\ \log\widehat{\operatorname{Var}}(u_{it})\Big), \tag{A20}$$

with $\bar{r}_i$ the time mean of r_{it}. The partition is the k-means of the f_i, standardized and compressed by principal components, with restarts and an admissibility filter. The principal components retain 90% of the cumulative variance, and the filter discards the starting partitions whose smallest regime could not support a fit. Two other starts are available: the k-means of the location-wise means of the covariates, which carries information on the regimes only when the covariates differ between them (see below), and the coordinates. Any partition computed elsewhere can also be supplied. Neither the departures nor the covariate means use the coordinates, so spatial contiguity is left entirely to the Potts penalty, and ϕ reads as the price paid for spatial cohesion rather than as a constraint built into the starting point.

Why departures and not covariate means. Under the SC-STEM model, for a location of regime g,

$$r_{it} = x'_{it}(\beta_g - \hat{\beta}) + A_i\big(y_t^{(g)} - \hat{y}_t\big) + e_{it},$$

so the mean and the slopes in (A20) estimate how β_g departs from the pooled coefficients, the autocorrelation of u_{it} reflects the dynamics of $y_t^{(g)} - \hat{y}_t$, and its variance the amplitude of that path together with the error variance of the regime. Every term is common to the locations of one regime and differs between regimes whenever their parameters, or their latent paths, differ. The covariate mean $\bar{x}_i$, which the earlier default clustered, depends on the regime only if the covariates do. When the covariates are exogenous to the regimes — the same distribution in every regime, which is what a regression on them usually presumes — $\bar{x}_i$ has the same law in every regime, and its k-means returns a partition unrelated to the regimes, from which the alternation has to find its way on its own.

A3.2 The two rules, formally

The two rules can be written compactly. Fix K and write $\mathcal{A}(g^{(0)}, \phi)$ for the output of Algorithm 1 of the manuscript, the estimated partition and parameters, when the algorithm is started from the partition $g^{(0)}$ with penalty ϕ. Let g^{init} be the partition given by the initialization above (the k-means of the departures (A20), or a partition supplied by the user). The estimates are

$$(\hat{g}^{\,0}, \hat{\Psi}^0) = \mathcal{A}(g^{\text{init}}, 0), \qquad (\hat{g}^{\,\phi}, \hat{\Psi}^\phi) = \mathcal{A}(\hat{g}^{\,0}, \phi) \quad (\phi > 0).$$

In words: the unpenalized fit starts from the departures and converges to the partition $\hat{g}^{\,0}$, and every penalized fit at the same K starts from $\hat{g}^{\,0}$, not from the departures. A single penalized fit therefore computes $\hat{g}^{\,0}$ first; over a grid of (K, ϕ), the unpenalized fit is computed once at every K and serves as the start of all the penalized fits at that K.

A3.3 Why the penalty needs a scale

In a cross-sectional spatially clustered model each unit contributes a single observation to the likelihood, so the differences $\ell_i(k) - \ell_i(k')$ are of order one and a penalty ϕ of order one balances fit

against cohesion. This is what makes a grid such as $\phi \in [0, 1]$ meaningful across applications in that literature. In the spatio-temporal case the same grid means nothing: each location contributes T observations, so the log-likelihood differences grow with the length of the series and with the scale of the response, and a penalty that reorganizes the partition for $T = 100$ is invisible for $T = 2000$. The factor c of the penalized objective restores comparability. The default is data-driven: with Δ_i the spread of the scores of location i across the candidate regimes, that is the likelihood value of choosing the best regime rather than the worst for that location, c is the median spread divided by the average number of neighbors, (A21) below, computed once, at the first sweep, and held fixed thereafter so that the objective does not change between iterations. With this normalization, $\phi = 1$ is the point at which full agreement with one's entire neighborhood is worth about as much as the typical gain from picking the best-fitting regime. Where the spread is measured matters as much as how, and the rest of this section shows that the rule of initialization fixes it on the fitted unpenalized regimes.

A3.4 The scale is the one of the unpenalized fit

The scale of the penalty under the default normalization is computed at the first sweep,

$$c = \frac{1}{\bar{m}} \operatorname{med}_i \Delta_i, \qquad \Delta_i = \max_h \ell_i(h; \Psi_h^{(1)}) - \min_h \ell_i(h; \Psi_h^{(1)}), \tag{A21}$$

with $\Psi^{(1)}$ the estimates of the first parameter step and $\bar{m}$ the average number of neighbors.

Proposition A3.1. *Under the rule, $\Psi^{(1)} = \hat{\Psi}^0$, so c in (A21) is computed on the fitted unpenalized regimes. Consequently: (i) c is the same for every $\phi > 0$ at the same K; (ii) c and the penalized fits depend on the initialization only through $\hat{g}^0$, so two initializations whose unpenalized fits converge to the same partition give identical penalized fits; (iii) a single penalized fit coincides with the member of a grid at the same (K, ϕ).*

Proof. The first parameter step of $\mathcal{A}(\hat{g}^0, \phi)$ estimates each regime by the EM algorithm on the locations of $\hat{g}^0$ in that regime, from the same starting values and with the same tolerance as the final refit of $\mathcal{A}(g^{\text{init}}, 0)$, which is the same computation on the same partition. The EM algorithm is a deterministic map of the data, the partition and the starting values, so $\Psi^{(1)} = \hat{\Psi}^0$, and (A21) depends on $(\hat{g}^0, \hat{\Psi}^0)$ only. Neither depends on ϕ, which gives (i); both depend on g^{init} only through $\hat{g}^0$, which gives (ii); and a grid passes exactly $\hat{g}^0$, which gives (iii). □

What the scale means. At fixed parameters, location i leaves its best unpenalized regime g^* for a regime h only if

$$\ell_i(g^*) - \ell_i(h) \;<\; \phi\, c \left\{ n_i(h) - n_i(g^*) \right\} \;=\; \phi \operatorname{med}_j \Delta_j \, \frac{n_i(h) - n_i(g^*)}{\bar{m}}, \tag{A22}$$

with $n_i(h)$ the number of neighbors of i in regime h. With all its neighbors in h the right-hand side is about $\phi \operatorname{med}_j \Delta_j$: the location moves only if its own advantage for g^* is below the fraction ϕ of the typical best-against-worst advantage. On fitted regimes Δ_j is of the order of the whole log-likelihood advantage of the right regime over the wrong ones accumulated over the T periods, so values of ϕ of a few hundredths already move the locations whose assignment is ambiguous, and values of order one override the likelihood for most of those near a border. This is why the default grid is $\phi \in \{0, 0.025, 0.05, 0.1, 0.2, 0.5, 1\}$ and the moderate-penalty band in which the selection rule chooses K is $[0.025, 0.2]$.

A3.5 Why a penalized fit cannot be started like an unpenalized one

Two things go wrong when the penalized fit is started from g^{init}.

The scale depends on the start. If g^{init} is unrelated to the regimes, every starting regime is a mixture of the true ones, the estimates $\Psi_h^{(1)}$ are nearly alike, and the spreads Δ_i in (A21) are small; if g^{init} is close to the regimes, the spreads are those of distinct regimes and several times larger. The same ϕ then stands for penalties of different strength under different initializations, and a better initialization receives a stronger penalty.

A strong penalty freezes a poor start. Suppose instead that c is fixed on fitted regimes but the fit starts from a poor g^{init}. At the first sweep the scores of a location barely differ across the nearly alike starting regimes, while the penalty term is of the order of $\phi \operatorname{med}_j \Delta_j$. When

$$\phi\, c \;>\; \max_i \max_{h,h'} \big|\ell_i(h) - \ell_i(h')\big|,$$

every location whose neighbors are not tied between two labels takes the label held by most of its neighbors, whatever its scores: the label step reduces to the iterated conditional modes of the Potts model alone, a majority vote whose fixed points are the configurations in which every location sits in the regime of most of its neighbors. The start is smoothed into compact blocks, the parameters re-estimated on those blocks are again mixtures, and the alternation stops at a fixed point near the start. From $\hat{g}^{\,0}$ the regimes are already as distinct as the data make them, the scores separate the locations, and the penalty only arbitrates the ambiguous ones through (A22).

A3.6 Cost

On a grid of (K, ϕ) that contains $\phi = 0$ the rule costs nothing: the unpenalized fit at every K is a member of the grid and its partition is passed on. A single penalized fit costs one unpenalized fit more. In the parametric bootstrap every refit at $\hat{\phi} > 0$ fits $\phi = 0$ first on the resampled data, which doubles its cost; that is the estimator reproduced, since the start and the scale are part of it.

A4 Degeneracy, the minimum-size constraint and the swap pass

Section 3.1 of the manuscript discusses how the algorithm keeps the partition from degenerating. This section gives the argument and the implementation in full.

A4.1 The degeneracy

Within-cluster homogeneity is exactly what the assignment step pursues. As locations are reallocated to the regime that fits them best, the residual variability within each regime falls, and with it the estimated variance components $\sigma^2_{\epsilon k}$ and $\sigma^2_{\omega k}$. But the assignment score of Section 2.3 of the manuscript rewards a small total variance $v_k = \sigma^2_{\epsilon k} + \sigma^2_{\omega k}$: for a location whose residuals under two regimes are comparable, the regime with the smaller total variance wins. Left unconstrained, the process is self-reinforcing (i.e., the regime with the smallest residual variance attracts locations, which makes it more homogeneous, which lowers its variance further) and the partition collapses onto a single regime. This is the clusterwise counterpart of the degenerate-likelihood problem of Gaussian mixtures, where a component can concentrate on a few observations and drive its variance to zero. The collapse was observed empirically even at $\phi = 0$, that is with no spatial forcing at all, which confirms that it originates in the likelihood and not in the penalty.

A4.2 The minimum-size constraint

To adjust for this issue, we impose a constraint on the size of the regimes: a location may leave its current regime only if that regime remains at or above a minimum size $n_{\min}$, set by default to $r + 2$, with r the number of covariates. The constraint acts on the feasible set rather than on the objective: every configuration visited by the algorithm is admissible by construction, and the monotonicity of the sweep of iterated conditional modes is preserved within the feasible set, since each label still maximizes its own penalized contribution over the labels that remain available to it.

A4.3 The swap pass

The constraint has a cost of its own, and it is severe on small networks. A location sitting in a regime that is exactly at the minimum size can never move, whatever its fit, because moving it would violate feasibility. With d small relative to $K \cdot n_{\min}$, a large share of the partition can therefore stay frozen at its initial value, and the algorithm reports convergence on a configuration it was never free to leave. The remedy is to add a class of moves that the constraint cannot block. Each sweep is followed by a *size-preserving swap pass*: the labels of two locations belonging to different regimes are exchanged whenever the exchange strictly increases the penalized objective Q. A swap leaves every regime size unchanged, so feasibility holds automatically, and it is accepted only on strict improvement, so the monotonicity of the sweep is preserved. Swaps let the algorithm explore configurations that the constraint makes unreachable through single moves, and in practice they are what allows a well-separated partition to be recovered on a small network.

A4.4 Pruning the swaps

Evaluating every candidate swap is quadratic in d, which would dominate the cost of a sweep. The implementation therefore vectorizes the pass over pairs and prunes it with an exact upper bound on the gain achievable by any swap between two given regimes: since exchanging the labels of i and j can alter the penalty only through the edges incident to i and j, the penalty gain cannot exceed $\phi c\,(m_i + m_j + 1)$, where m_i is the number of neighbors of location i. Pairs whose likelihood gain is below the negative of that bound cannot improve Q and are discarded without further computation. The bound is exact, so the pruning changes the running time and not the result.

A4.5 Empty and undersized regimes

Along the iterations a regime can shrink below the number of locations required to estimate its parameters. Rather than aborting, the algorithm freezes such a regime at its last valid parameter values and keeps it available to the assignment step, so that locations can flow back into it; a regime that never obtained a valid fit is excluded from the assignment altogether. A configuration whose final partition still contains empty or undersized regimes is flagged as inadmissible and excluded from model selection, since its likelihood is not comparable with that of a configuration with K genuine regimes.

A5 Monotonicity of the alternation

Section 3.1 of the manuscript states that the alternation of Algorithm 1 is not globally monotone. This section gives the argument and its consequences.

A5.1 Two monotone steps

Each of the two steps of Algorithm 1 is monotone in its own right. The sweep of iterated conditional modes cannot decrease Q at fixed parameters, the swap pass accepts only strict improvements, and the EM algorithm cannot decrease the exact likelihood of a regime at fixed labels. It does not follow that the alternation is monotone, and in general it is not: the trace of the objective may well show a decrease from one iteration to the next.

A5.2 Why the alternation is not

The reason follows from the pseudo-likelihood score of Section 2.3 of the manuscript. The label step maximizes the penalized *pseudo*-likelihood Q, which is built on the conditional scores; the parameter step maximizes the *exact* within-regime likelihood through the EM algorithm. These are two different functions of the same parameters. They agree on what a good partition looks like (i.e., both reward regimes whose members are well described by common parameters) but they are not related by an inequality, so an EM update that raises the exact likelihood of a regime can lower the pseudo-likelihood contributions of some of its members, and hence Q. Had the exact likelihood factorized across locations, the two objectives would coincide and the usual monotonicity argument would apply; it does not factorize precisely because the measurement covariance $\Sigma_{e,k}$ is not diagonal. Non-monotonicity is therefore a property of the spatio-temporal specification, not a defect of the implementation and not a numerical tolerance issue, and it has no counterpart in the cross-sectional models from which the algorithmic apparatus is borrowed.

A5.3 Consequences

Three practical consequences follow. First, the algorithm keeps track of the best configuration visited and returns that one, rather than the last: the final refit is performed on the best partition, not on the terminal one. Second, convergence cannot be declared on the basis of a monotone objective, so the stopping rule combines several criteria — an unchanged partition, an improvement below tolerance, the reappearance of a previously visited partition, which detects cycling, and an iteration budget. Third, the trace of the objective is reported as a diagnostic rather than as a guarantee, and a decreasing trace is informative about the tension between the two objectives rather than a sign of failure.

A6 Behavior of the information criteria as T and d grow

A6.1 Notation and the two quantities to compare

Write $q = r + 3 + 3p$ for the number of parameters of one regime, so that a K-regime configuration has $\mathrm{df}(K) = Kq$ continuous parameters, and $n = dT$ for the number of observations entering the likelihood. Let $\hat{\ell}(K)$ be the maximized log-likelihood at K regimes, the maximum being taken over the parameters *and* over the partitions. A criterion moves from K to $K+1$ when

$$2\,\Delta\hat{\ell}(K) \;>\; \Delta\mathrm{pen}(K), \qquad \Delta\hat{\ell}(K) = \hat{\ell}(K+1) - \hat{\ell}(K). \tag{A23}$$

Everything follows from the orders of the two sides.

A6.2 The likelihood gain

Under the alternative. If the truth has $K_0 > K$ regimes, the best K-regime approximation sits at a strictly positive Kullback–Leibler distance δ_K per observation from the truth, and

$$2\,\Delta\hat{\ell}(K) \asymp 2\,n\,\delta_K = O_p(dT). \tag{A24}$$

The gain is proportional to the number of observations the extra regime describes: it grows in *both* d and T.

Under the null. If the truth has K regimes and we fit $K+1$, the extra regime is superfluous. For a *fixed* refinement of the true partition the likelihood ratio is asymptotically χ^2_q, hence $O_p(1)$ in T: lengthening the series does not help a spurious regime. What does help it is the search. The maximum is taken over the partitions of d locations into $K+1$ groups, a set of cardinality $S(d, K+1)$, the Stirling number of the second kind, and $\log S(d,K) = d\log K - \log K! + o(d)$. The standard extreme-value heuristic for the maximum of many χ^2_q variables — the same heuristic that underlies the extended BIC — gives

$$2\,\Delta\hat{\ell}(K) \asymp 2\,[\log S(d,K+1) - \log S(d,K)] \approx 2\,d\log\!\left(\tfrac{K+1}{K}\right) = O_p(d). \tag{A25}$$

Under the null the spurious gain grows linearly in d and not at all in T: the variables being maximized over are strongly dependent, since neighboring partitions share most of their labels, so (A25) is an upper bound on the order rather than an equality; it is the conservative and the relevant one.

A6.3 The penalty increments

Writing $\mathrm{df} = Kq$ and taking the increment from K to $K+1$,

$$\Delta\mathrm{AIC} = 2q, \tag{A26}$$

$$\Delta\mathrm{AICc} = 2q + \frac{2q^2(2K+1)}{n} + O(n^{-2}), \tag{A27}$$

$$\Delta\mathrm{BIC} = q\log(dT) = q(\log d + \log T), \tag{A28}$$

$$\Delta\mathrm{EBIC}_\gamma = q\log(dT) + 2\gamma\,[\log S(d,K+1) - \log S(d,K)] \approx q\log(dT) + 2\gamma\,d\log\!\left(\tfrac{K+1}{K}\right). \tag{A29}$$

A6.4 Consistency, direction by direction

Combining Equation (A25) with (A26)–(A29), a criterion avoids a spurious regime when its increment exceeds $c\,d$ with $c = 2\log\{(K+1)/K\}$.

AIC and AICc. The increment is $2q$, a constant. The condition $2q > c\,d$ holds only for $d < 2q/c$, so **both are inconsistent, and the failure worsens linearly in** d. Neither depends on T at all. The corrected version adds nothing: by (A27) the correction is $O(\mathrm{df}^2/n) = O(K^2q^2/(dT))$ and vanishes in *both* directions. The AICc is built for the regime in which df/n is not negligible; in a spatio-temporal panel $\mathrm{df}/n = Kq/(dT) \to 0$ however the sample grows, because df is fixed by the specification and n multiplies d by T.

BIC. The condition is $q(\log d + \log T) > c\,d$. Hence

- $T \to \infty$ *with* d *fixed*: the left side diverges, the right is fixed. **The BIC is consistent in** T**.**
- $d \to \infty$ *with* T *fixed*: the left side grows like $q \log d$, the right like $c\,d$. **The BIC over-selects, and does so for every large enough network.**
- *both growing*: consistency requires $d/\log(dT) \to 0$, that is $d = o(\log T)$ — the network must grow only logarithmically in the length of the series.

Lengthening the record therefore helps the BIC and widening the network hurts it.

Extended BIC. Here the penalty carries a term of the same order as the null gain. The condition becomes

$$q \log(dT) + 2\gamma\, d \log\!\left(\tfrac{K+1}{K}\right) \;>\; 2\, d \log\!\left(\tfrac{K+1}{K}\right),$$

which holds for all large d as soon as $\gamma \geq 1$. The extended BIC is the only member of the family whose penalty has the right order in d, and $\gamma = 1$ is exactly the calibration point, since it charges precisely the multiplicity that Equation (A25) extracts. It follows that when the extended BIC does not change the answer on a long panel, it is because the observed gain is far above anything the null can produce.

Notice that, with $T \to \infty$ and d fixed, every label is consistently estimated, since each is identified by an unboundedly informative series, and the BIC is consistent by the argument above. But d is fixed: the model is fitted to a closed set of locations and there is no sense in which its partition extends beyond them.

Moreover, with $d \to \infty$, which is the frame in which spatial prediction is meaningful, the number of labels grows with the sample. The partition is then an *incidental parameter* in the sense of Neyman and Scott (1948): its dimension is proportional to the sample size in the spatial direction, and it is not consistently estimated at fixed T. No penalty that is a function of df and n alone controls the error made in assigning a location to a regime, because that error does not vanish in this frame.

A6.5 Dependence, and the ambiguity of n

A last remark sharpens the point. All of the above uses $n = dT$, that is, the number of *observations* (or the number of observed values when the response has gaps). Under dependence that overstates the information. For a latent process with persistence G, the variance of a temporal average over T points is inflated by $(1+G)/(1-G)$, so an effective count is

$$T_{\text{eff}} \;\approx\; T\,\frac{1-G}{1+G}, \tag{A30}$$

which at a persistence of $G = 0.97$, for instance, equals $T \times 0.015$: a series of 1,826 periods carries the information of roughly 28 independent ones. Substituting n_{eff} for n in (A28) lowers the BIC increment further, from $q \log(dT)$ to $q \log(d\,T_{\text{eff}})$, and makes the criterion weaker still.

The conclusion is not that one of these choices of n is correct and the others are wrong. It is that the margin of the BIC over the null gain (A25) depends on a quantity the criterion does not determine: $q \log(dT)$ with the number of observations, less with the effective count. On a persistent process and a wide network the extended BIC with $\gamma = 1$, whose multiplicity term does not depend on n, is the safer check.

A7 Inference under the refit-with-clustering bootstrap

Section A7.2 describes how the draws of the bootstrap of Section 3.3 of the manuscript are aligned with the estimated regimes and summarized by intervals and tests of one parameter at a time. Many questions concern several coefficients at once: a Granger non-causality test asks whether a block of lags of one variable has zero coefficients, as in the application of Section 5 of the manuscript. This section shows how the Wald test of such a hypothesis is built on the refit-with-clustering bootstrap, and what it takes from it.

A7.1 The procedure

The partition of SC-STEM is estimated on the same data as the parameters, so intervals and tests that treat it as known are too optimistic wherever locations lie near the border between two regimes. The refit-with-clustering bootstrap of Section 3.3 of the manuscript draws new data from the fitted model and re-runs the whole procedure on every draw, clustering included: every draw has its own partition, and the spread of the draws carries the uncertainty of the memberships. Algorithm 3 describes this procedure. Each draw is generated regime by regime, so that the spatial covariance and the latent dynamics of every regime are reproduced, on the observed covariates, coordinates and loading rows, and with the observed pattern of missing values; the locations keep their original order, so every location keeps its neighbors in the graph of the Potts penalty. Every draw is evaluated under its own seed, so that a draw does not depend on the others.

Algorithm 3 The refit-with-clustering bootstrap.

Require: the selected fit at $(\hat{K}, \hat{\phi})$: the partition $\hat{g}$ and the parameters $\hat{\Psi}_k$ of every regime; the number of draws B

for $b = 1, \dots, B$ **do**
  for $k = 1, \dots, \hat{K}$ **do**
    simulate $z^*_{t,\mathcal{C}_k}$, $t = 1, \dots, T$, from the STEM model of regime k at $\hat{\Psi}_k$, with the covariates, coordinates and loading rows of its locations
  end for
  set to missing in z^* the values that are missing in the data
  run Algorithm 1 of the manuscript on z^* at $(\hat{K}, \hat{\phi})$, with the start, the scale of the penalty and the settings of the original fit (for $\hat{\phi} > 0$, from the fit at $\phi = 0$; Section A3), giving $\hat{g}^*_b$ and $\hat{\Psi}^*_{k,b}$
  the draw is usable if the refit succeeded with $\hat{K}$ admissible regimes
end for
for every usable draw b **do**
  map every regime of $\hat{g}^*_b$ onto the regime of $\hat{g}$ with which it shares the most locations, resolving conflicts greedily by decreasing overlap; relabel $\hat{\Psi}^*_{k,b}$ accordingly
end for
from the aligned draws: standard errors, intervals, tests between regimes (Section A7.2), Wald tests (Section A7.4), stability of the partition

A7.2 The aligned draws

Because the labels of a refit are arbitrary, each refit regime is mapped onto the regime of the original fit with which it shares the largest number of locations, resolving conflicts greedily by decreasing

overlap. Draws whose refit failed, or did not recover all $\hat{K}$ regimes, are recorded and excluded from the summaries, and the share of usable draws is reported as a diagnostic in its own right: a low share is evidence that the selected configuration is not reproducible. From the aligned draws the procedure computes standard errors and the normal, basic, percentile and bias-corrected confidence intervals of Davison and Hinkley (1997) for every regime-specific parameter; studentized intervals are not among them, since they would require an asymptotic standard error for every parameter at every draw. It also computes percentile tests for the between-regime differences of the coefficients, which tell whether the regimes differ significantly and in which components of Ψ_k. The same aligned draws serve for tests on several coefficients at once, as in the application of Section 5 of the manuscript and in the rest of this section.

A7.3 Hypothesis on model's parameters

Section 5 of the manuscript introduces a Granger non-causality test asking whether a block of lags of one variable has zero coefficients. We show how the Wald test of such a hypothesis is built on the refit-with-clustering bootstrap. In regime k let $\beta_k \in \mathbb{R}^r$ be the regression coefficients and consider the q linear restrictions

$$H_0^{(k)} : \; R\beta_k = c, \qquad R \in \mathbb{R}^{q \times r} \text{ of rank } q \tag{A31}$$

The hypothesis is post-estimation inference: SC-STEM estimates the partition $\hat{\mathcal{G}}$ and the parameters of every regime jointly, and the test concerns the parameters of a regime of $\hat{\mathcal{G}}$. The difficulty is that $\hat{\mathcal{G}}$ was estimated on the data on which the test is run. A test that treats it as given, such as one based on the model-based covariance of $\hat{\beta}_k$, leaves out the uncertainty of the memberships, and is too optimistic wherever locations lie near the border between two regimes.

The F test and the Wald test. In the classical linear model, with errors independent, homoskedastic and Gaussian and regressors fixed, H_0 is tested exactly by

$$F = \frac{(\mathrm{RSS}_0 - \mathrm{RSS}_1)/q}{\mathrm{RSS}_1/(n-r)} \sim F_{q,\,n-r},$$

with RSS_0 and RSS_1 the residual sums of squares with and without the restrictions. There the Wald statistic is a monotone function of F,

$$W = qF\, n/(n-r)$$

, and the F test is its exact small-sample form. With a single restriction, $q = 1$, the F statistic is the square of the t statistic of the restricted coefficient and the F test is the two-sided t test; in the same way the Wald statistic is the square of the ratio of the estimate to its standard error, and referring it to χ^2_1 is the two-sided z test. A regime of SC-STEM is not that model. Its coefficients are estimated by maximum likelihood together with a latent dynamic process and a spatially correlated measurement error, by the EM algorithm and the Kalman filter, on a response that may have gaps; no ratio of sums of squares has an F distribution, and the test of H_0 is the Wald test itself, asymptotically χ^2_q under H_0 when the partition is known. The partition is not known, and the bootstrap is what accounts for it.

A7.4 A bootstrap-based Wald test

The refit-with-clustering bootstrap draws B samples from the fitted model, regime by regime, re-runs SC-STEM at $(\hat{K}, \hat{\phi})$ on each and maps the regimes of every refit onto the observed ones by the

majority rule, giving the aligned draws $\beta^*_{k,b}$, $b = 1, \dots, B$. Their covariance,

$$\hat{V}_k = \frac{1}{B-1} \sum_{b=1}^{B} (\beta^*_{k,b} - \bar{\beta}^*_k)(\beta^*_{k,b} - \bar{\beta}^*_k)',$$

estimates the sampling covariance of $\hat{\beta}_k$ *including* the variability that comes from re-estimating the partition, since every draw has its own partition. The Wald statistic is

$$W_k = (R\hat{\beta}_k - c)' \, (R\hat{V}_k R')^{-1} \, (R\hat{\beta}_k - c), \tag{A32}$$

referred to χ^2_q. Here the bootstrap supplies the variance only; the reference distribution is the asymptotic one. A bootstrap reference is also available from the same draws: centered at the estimate, a draw reproduces the deviation of $\hat{\beta}_k$ from its true value, which is what W_k measures when H_0 holds, so that

$$W^*_{k,b} = (R\beta^*_{k,b} - R\hat{\beta}_k)' \, (R\hat{V}_k R')^{-1} \, (R\beta^*_{k,b} - R\hat{\beta}_k), \qquad p = \frac{1 + \#\{b : W^*_{k,b} \geq W_k\}}{1 + B}, \tag{A33}$$

the centering that imposes the null on the statistic rather than on the data-generating process. The hypothesis in all the regimes at once stacks the K vectors $R\beta_k - c$, whose joint covariance is estimated from the same draws, since a draw re-estimates all the regimes together; it has qK degrees of freedom. For $q = 1$ and R a unit vector, (A32) is the square of the t statistic reported for every parameter. The aligned draws are part of the output of the bootstrap, so a test of any linear restriction on any subset of the coefficients of a regime, or of several regimes, is computed from them without refitting.

A8 On the use of the Potts penalty on point-referenced data

The Potts term of the penalized objective, Equation (7) of the manuscript, comes from a literature on lattices and areal units. Section 2.2 of the manuscript summarizes why it can be used on point-referenced data; this section gives the full argument. Three points need to be argued rather than assumed.

(a) The graph is a modeling choice, not a datum. The penalty requires an edge set E and nothing else. Neither its definition nor the argument of Besag (1986) for the iterated conditional modes update asks that the edges come from shared boundaries: both hold on any undirected graph. With areal units a shared boundary provides a canonical, if debatable, graph. With points nothing does, and the m-nearest-neighbor graph, a distance threshold and a Delaunay triangulation are all defensible and all different. We use the m-nearest-neighbor graph because the density of a network of locations is often very irregular. The nearest-neighbor graph keeps the degree roughly constant, so that the penalty weighs about the same for every location, while a fixed radius would give an urban station twenty neighbors and an alpine one none. The price is that in sparse areas a nearest neighbor may be far away, imposing continuity where there is no information to support it; a cap on the edge length is the natural remedy. Two details make the construction well posed. The relation is symmetrized, since otherwise a pair would enter the score of one endpoint and not of the other, and the label step would not maximize a well-defined function. The neighbors are measured with the metric of the covariance, geodesic on geographic coordinates, since otherwise the penalty and the spatial covariance of the regimes, Equation (4) of the manuscript, would disagree about which

locations are close: on raw longitude and latitude, at 45° of latitude, a degree of longitude is about 78 km and one of latitude about 111 km. Because the graph is a choice, its effect has to be shown rather than asserted. Its effect is also limited by construction: under the default scale of the penalty (Section A3) the factor c divides by the mean degree, so raising m spreads the same total pull over more and longer edges rather than increasing it.

(b) Spatial proximity enters the model twice. This point has no analog with areal data. The coordinates enter the spatial covariance through the distances h_{ij}, and they enter the objective again through E. A penalty that rewards neighboring locations for sharing a label will therefore produce spatially compact regimes whether or not the data contain any. The two uses are not redundant: the covariance describes the dependence of the errors *within* a regime, which the assignment score, Equation (9) of the manuscript, discards, while the penalty describes the contiguity *of* the regimes. If every regime carries the same parameters, the same latent path and a single error field spans the network, the distribution of the data does not depend on the partition. The scores of a location then differ across regimes only by sampling noise, and the penalty is the only term of the objective that prefers one partition to another: the label step returns compact blocks, the partition that cuts the fewest edges of the graph, for reasons that have nothing to do with the data. Two consequences follow. The spatial compactness of an estimated partition is not evidence of regimes and must not be read as such. And whether regimes exist has to be decided by a criterion that the penalty does not enter, the likelihood of the partition against that of the pooled model (Section 3.2 of the manuscript), rather than by inspecting the partition.

(c) The normalizing constant is intractable on any graph. Read as a prior on the partition, the Potts term is defined only up to its normalizing constant $Z(\phi, K) = \sum_{\text{labelings}} \exp\{\phi c\, P\}$, a sum over K^d configurations with no closed form on a lattice and none on a general graph. The change of support therefore costs nothing that was not already paid in the areal case, and the consequence is the same as there: ϕ is a tuning constant, not a parameter estimated jointly with the others. It is chosen by comparing the likelihoods that the partitions estimated at different penalties reach, a comparison in which the Potts term does not appear (Section 3.2 of the manuscript). A criterion built on the penalized objective would instead have to restore Z, whose omission reverses the sign of the comparison across penalties.

A9 What SC-STEM does and does not cover

Distribution. The model is Gaussian throughout: the measurement errors, the innovations of the latent process and its initial condition are all normal, and the EM algorithm relies on that normality for the closed-form M-steps and on the Kalman filter for the conditional moments of the latent state. Non-Gaussian responses may be brought onto an approximately Gaussian scale by transformation before the model is fitted, with the estimates reported on both scales and the usual back-transformation bias correction applied. A genuinely non-Gaussian formulation would require replacing the Kalman recursions by a filter for nonlinear or non-Gaussian state-space models, and falls outside the model as defined here.

Dimension of the response. The response is univariate at each location and time: z_{it} is a scalar, and the vector z_t collects the same variable measured at different sites, not different variables. What is multivariate in the model is the *latent state*, whose dimension p may exceed one, so that several unobserved temporal processes can act simultaneously on the network through the columns of the

loading matrix A. A genuinely multivariate response is a different specification, and the closest construction available within the present framework is the heterogeneous-network model of Fassò et al. (2007), in which measurements of different type are reconciled through calibration components rather than through a multivariate covariance.

Spatial covariance. The correlation function is exponential, $\exp\{-\theta h\}$, hence isotropic and stationary within a regime, and the analytical first and second derivatives required by the Newton–Raphson step are supplied for that function only. Anisotropy and non-stationarity are therefore not available *within* a regime, although the partition itself introduces a coarse form of non-stationarity *across* the domain, since each regime carries its own range θ_k and its own variance components. Distances may be Euclidean or geodesic, the latter being required when the coordinates are longitude and latitude.

Completeness of the data. Missing values in the *response* are supported, so the panel need not be balanced; Section 3.1 of the manuscript and Section A2 give the treatment. Missing values in the *covariates* or in the *coordinates* are not, since the design matrix enters the closed-form M-step updates directly and the coordinates enter the distance matrix, so that accommodating gaps there would require a genuinely stochastic E-step; they have to be imputed before the model is fitted. Every location must retain at least one observation.

Time and space. The time index is discrete and regularly spaced, since the state equation is a first-order vector autoregression on consecutive time points. The locations are fixed over time and are point-referenced, with the spatial covariance a function of the distance between coordinates. Areal data can be forced into the framework only by taking the centroids of the units as coordinates, which is legitimate only when the units are small relative to the distances between them; when they are not, the range parameter is not identifiable from centroid distances and the construction should be replaced by a neighborhood-based spatial covariance.

Panel structure. The data form a panel in the locations: every location carries a series over the same time window, so that z is a $T \times d$ matrix, which need not be fully observed. A network whose composition changes within the window is represented by marking as missing the periods in which a station was not operating.

Latent dynamics. The transition matrix G and the innovation covariance Σ_η are diagonal by default, so that with $p > 1$ the latent processes evolve independently of one another; both can be made full, at the cost of p^2 and $p(p+1)/2$ parameters per regime respectively. The initial covariance C_0 is held fixed rather than estimated, whereas the initial mean m_0 is a free parameter, and both are regime-specific in the clustered model.

Additional restrictions specific to the clustered layer. Two constraints have no counterpart in the pooled model. First, each regime must contain enough locations to support its own fit, which sets a lower bound on the size of a regime and, jointly with the number of locations, an upper bound on the number of regimes that can be entertained (see Section A4 for the constraint and for the moves needed to keep it from freezing the partition). Second, the loading matrix is taken as known and common across regimes, so the regimes differ in the parameters of Equation (5) of the manuscript but not in how the latent process is loaded onto the locations.

References

Julian Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society: Series B*, 48(3):259–302, 1986.

Anthony C. Davison and David V. Hinkley. *Bootstrap Methods and their Application*. Cambridge University Press, Cambridge, 1997.

Piet de Jong. The diffuse Kalman filter. *The Annals of Statistics*, 19(2):1073–1083, 1991. doi: 10.1214/aos/1176348139.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1–22, 1977.

Yu Du and Ravi Varadhan. SQUAREM: An R package for off-the-shelf acceleration of EM, MM and other EM-like monotone algorithms. *Journal of Statistical Software*, 92(7):1–41, 2020. doi: 10.18637/jss.v092.i07.

James Durbin and Siem Jan Koopman. *Time Series Analysis by State Space Methods*. Oxford University Press, Oxford, 2nd edition, 2012.

Alessandro Fassò, Michela Cameletti, and Orietta Nicolis. Air quality monitoring using heterogeneous networks. *Environmetrics*, 18(3):245–264, 2007. doi: 10.1002/env.837.

Chuanhai Liu and Donald B. Rubin. The ECME algorithm: a simple extension of EM and ECM with faster monotone convergence. *Biometrika*, 81(4):633–648, 1994. doi: 10.1093/biomet/81.4.633.

Xiao-Li Meng and Donald B. Rubin. Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika*, 80(2):267–278, 1993. doi: 10.1093/biomet/80.2.267.

Xiao-Li Meng and Donald B. Rubin. On the global and componentwise rates of convergence of the EM algorithm. *Linear Algebra and its Applications*, 199:413–425, 1994. doi: 10.1016/0024-3795(94)90363-8.

Xiao-Li Meng and David van Dyk. The EM algorithm – an old folk-song sung to a fast new tune. *Journal of the Royal Statistical Society, Series B*, 59(3):511–567, 1997. doi: 10.1111/1467-9868.00082.

Jerzy Neyman and Elizabeth L. Scott. Consistent estimates based on partially consistent observations. *Econometrica*, 16(1):1–32, 1948. doi: 10.2307/1914288.

Christophe Roland and Ravi Varadhan. New iterative schemes for nonlinear fixed point problems, with applications to problems with bifurcations and incomplete-data problems. *Applied Numerical Mathematics*, 55(2):215–226, 2005. doi: 10.1016/j.apnum.2004.10.003.

Robert H. Shumway and David S. Stoffer. An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4):253–264, 1982. doi: 10.1111/j.1467-9892.1982.tb00349.x.

Ravi Varadhan and Christophe Roland. Simple and globally convergent methods for accelerating the convergence of any EM algorithm. *Scandinavian Journal of Statistics*, 35(2):335–353, 2008. doi: 10.1111/j.1467-9469.2007.00585.x.