MethodsX 9 (2022) 101660

Contents lists available at ScienceDirect

MethodsX

journal homepage: www.elsevier.com/locate/mex

Method Article

Statistical tests for non-independent partitions of large autocorrelated datasets

Anthony R. Ives*, Likai Zhu, Fangfang Wang, Jun Zhu, Clay J. Morrow, Volker C. Radeloff

Integrative Biology, University of Wisconsin-Madison, Madison, WI 53706, USA

A B S T R A C T

Large sets of autocorrelated data are common in fields such as remote sensing and genomics. For example, remote sensing can produce maps of information for millions of pixels, and the information from nearby pixels will likely be spatially autocorrelated. Although there are well-established statistical methods for testing hypotheses using autocorrelated data, these methods become computationally impractical for large datasets.

- The method developed here makes it feasible to perform *F*-tests, likelihood ratio tests, and *t*-tests for large autocorrelated datasets. The method involves subsetting the dataset into partitions, analyzing each partition separately, and then combining the separate tests to give an overall test.
- The separate statistical tests on partitions are non-independent, because the points in different partitions are not independent. Therefore, combining separate analyses of partitions requires accounting for the non-independence of the test statistics among partitions.
- The methods can be applied to a wide range of data, including not only purely spatial data but also spatiotemporal data. For spatiotemporal data, it is possible to estimate coefficients from time-series models at different spatial locations and then analyze the spatial distribution of the estimates. The spatial analysis can be simplified by estimating spatial autocorrelation directly from the spatial autocorrelation among time series.

A R T I C L E I N F O

Method name: Method for performing statistical tests using non-independent data partitions

Keywords: Statistics for big data, Remote sensing, Hypothesis tests on large datasets, Likelihood ratio test, F-test, t-test

Article history: Received 26 November 2021; Accepted 28 February 2022; Available online 12 March 2022

DOI of original article: 10.1016/j.rse.2021.112678

* Corresponding author.

E-mail address: arives@wisc.edu (A.R. Ives).

https://doi.org/10.1016/j.mex.2022.101660

Specifications Table

Subject Area;	Environmental Science
More specific subject area;	*Statistics*
Method name;	*Method for performing statistical tests using non-independent data partitions*
Name and reference of original method;	*These are standard statistical methods documented in most statistical textbooks, for example, Neter et al.* [12] *and Judge et al.* [10]
Resource availability;	*The methods are implemented in the package remotePARTS in the R programming language. This is available at* https://github.com/morrowcj/remotePARTS

Method details

Overview

The method developed here uses a conceptually simple approach to perform statistical estimation and hypothesis tests on large autocorrelated datasets [1,4,8,11,13,14]. The method was developed specifically for remote-sensing datasets, so it will be described in this context, although it could be applied to other types of large datasets such genome samples from different subjects in which base-pair similarity is likely to be greater for base pairs located nearby on a chromosome due to limited recombination. The approach divides the dataset into non-overlapping partitions, and statistical hypothesis tests are conducted on each partition. The results of these tests are not independent, because the data points from different partitions are not independent. Nonetheless, it is possible to calculate the correlation between the statistical test scores from different partitions and combine the scores for an overall test. The selection of sampling schemes for partitions is arbitrary, but here we will focus on random sampling to produce partitions. This approach has the advantage of guaranteeing that each partition gives a representation of the entire dataset. The method is central to the analysis of spatiotemporal data given in Ives et al. [9].

We apply this approach to three common tests used for regression and ANOVA on Gaussian data [12]: the *F*-test, the likelihood ratio test (LRT), and the *t*-test. The *F*-test and LRT involve hypotheses comparing a full statistical model with a reduced model, thereby allowing tests of hypotheses on multiple coefficients in a model. The *t*-test is used for inference about individual coefficients. These tests can be performed for datasets with autocorrelated errors using generalized least squares (GLS) regression [10]. In GLS, a correlation matrix is specified to describe the autocorrelation of errors. We apply the approach to both purely spatial and spatiotemporal data.

Mathematical derivations

The specific problem addressed here involves the regression of a response (dependent) variable y_l on p predictor (independent) variables for $l = 1, 2, ..., N$ locations in a spatial map. The p predictor variables for location l are contained in a 1 x p vector X_l; at the minimum, X_l contains the value 1 to correspond to an intercept. The regression model for location l is

$$y_l = X_l\mathbf{B} + \gamma_l \tag{1}$$

where γ_l is the remaining random error, and $\mathbf{B}$ is a p x 1 vector containing regression coefficients for the p variables in X_l. We assume that the parameters $\mathbf{B}$ are the same for all locations l. We further assume that the correlation between γ_l and γ_k for locations l and k depends on the distance between them. For example, if the correlation diminishes exponentially with distance, then $\text{cor}[\gamma_l, \gamma_k] = \exp(-d_{lk}/r)$ where d_{lk} is the distance between locations l and k, and the parameter r gives the "range" of the correlation, with larger values of r giving correlations over greater spatial distances. Thus, the covariance matrix for the errors, $\mathbf{\Gamma} = (\gamma_1, ..., \gamma_N)'$ (where the apostrophe denotes transpose), is the N x N covariance matrix $\mathbf{V} = \sigma^2\mathbf{C}$, where $\mathbf{C}$ is the correlation matrix containing the values of $\text{cor}[\gamma_l, \gamma_k]$ for all pairs of locations l and k, and σ^2 is a common variance.

GLS analysis

A single dataset, or a subset of a larger dataset, in which errors are autocorrelated can be analyzed by GLS when **V** is known [10]. Thus, let **Y** denote an N x 1 vector of response variables, and **X** denote an N x p matrix of predictor variables including a column of ones for the intercept. The p x 1 vector $\hat{\mathbf{B}}$ containing the estimates of the regression coefficients **B** is

$$\hat{\mathbf{B}} = \left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-1}\left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{Y}\right) \tag{2}$$

The sum-of-squared error is then

$$SSE = \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right)'\mathbf{V}^{-1}\left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right) \tag{3}$$

An F-test is based on the sum-of-squared error of the full model (SSE) and a reduced model (SSE_0) defined by the hypothesis to be tested; for example, the null hypothesis that some of the predictor variables have no effect on the response variable is tested using the reduced model with these predictor variables removed. Under the null hypothesis, and letting $SSR_{1-0} = SSR_1 - SSR_0 = SSE_0 - SSE$ denote the difference in sum-of-squared regression between full and reduced models, we have

$$(SSR_{1-0}/df_1)/(SSE/df_2) \sim F \tag{4}$$

Thus, the ratio of SSR_{1-0}/df_1 to SSE/df_2 follows an F distribution where df_1 is the degrees of freedom equaling the difference in the number of predictor variables between full and reduced models, and $df_2 = N - p$ is the degrees of freedom for the sum-of-squared error in the full model. Strategic selection of the reduced model makes it possible to test a wide range of hypotheses, not only about whether one or more of the coefficients in **B** are different from zero, but also whether some or all of the coefficients are equal by generating orthogonal contrasts [12].

For GLS, a LRT is closely related to an F-test. Specifically, the log-likelihood ratio between the full and reduced models is SSR_{1-0} for the case when the full and reduced models have the same covariance matrix **V**. The LRT uses the asymptotic approximation

$$2\ SSR_{1-0} \sim \chi_{df1} \tag{5}$$

Thus, twice the difference between the SSRs from the full and reduced models is χ^2 distributed with df_1 degrees of freedom. The distribution for the F-test (Eq. 4) will converge to the distribution for the LRT when df_2 approaches infinity. For large datasets, df_2 will be large, and the F-test and LRT will give very similar results.

Like the LRT, the t-test is closely related to the F-test. Letting $MSE = SSE/df_2$ be the mean squared error, the estimated covariance matrix for the estimates $\hat{\mathbf{B}}$ is

$$\widehat{\mathrm{var}}\left[\hat{\mathbf{B}}\right] = MSE\ \left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-1} \tag{6}$$

The standard error of the estimate of a coefficient b_h contained in $\hat{\mathbf{B}}$, $\mathrm{se}[\hat{b}_h]$, is the square-root of the h^{th} diagonal element of $\widehat{\mathrm{var}}\ [\hat{\mathbf{B}}]$, and a test of the null hypothesis that the coefficient is zero is performed with t-distribution:

$$\hat{b}_h/\mathrm{se}\left[\hat{b}_h\right] \sim t_{df2} \tag{7}$$

To simplify the following developments, it is useful to recast the GLS model above by transforming variables. Specifically, let **D** be the matrix such that $\mathbf{DVD}' = \mathbf{I}$. For example, **D** could be the inverse of the Cholesky decomposition of **V**. With this construction of **D**, the covariance matrix for the transformed errors $\mathbf{A} = \mathbf{D\Gamma}$ is $\mathrm{E}[\mathbf{AA}'] = \mathrm{E}[(\mathbf{D\Gamma})(\mathbf{D\Gamma})'] = \mathbf{DVD}' = \mathbf{I}$. If $\mathbf{U} = \mathbf{DX}$ and $\mathbf{Z} = \mathbf{DY}$, then

$$\hat{\mathbf{B}} = (\mathbf{U}'\mathbf{U})^{-1}(\mathbf{U}'\mathbf{Z}) \tag{8}$$

Further, let **H** denote the hat matrix for **U** defined by

$$\mathbf{H} = \mathbf{U}(\mathbf{U}'\mathbf{U})^{-1}\mathbf{U}' \tag{9}$$

Then

$$SSR_{1-0} = \mathbf{Z}'(\mathbf{H} - \mathbf{H}_0)\mathbf{Z} \tag{10}$$

$$SSE = \mathbf{Z}'(\mathbf{I} - \mathbf{H})\mathbf{Z} \tag{11}$$

where $\mathbf{H_0}$ is the hat matrix for the reduced model derived from $\mathbf{U_0} = \mathbf{DX_0}$, and $\mathbf{X_0}$ is the matrix of predictor variables in the reduced model.

Correlations among partitions for SSR_{1-0}, SSE, and $\hat{\mathbf{B}}$

The primary computational burden of GLS is inverting $\mathbf{V}$ (or its Cholesky decomposition). A map with 10^6 pixels would require inverting a 10^6 x 10^6 $\mathbf{V}$ matrix, and the computation time for this inversion scales roughly with N^3. If a map is partitioned into n_p subsets of size m and each subset is analyzed separately, then the computational burden will scale linearly with N for a fixed subset size m. Using the computational advantage of analyzing partitions, the method below makes it feasible to combine the results from the n_p separate analyses.

For the F-test, the method requires calculating the correlation between sums-of-squares computed for the n_p different subsets. Specifically, it is necessary to calculate the correlation between SSR_i and SSR_j, and between SSE_i and SSE_j; for notational convenience, SSR_i denotes SSR_{1-0} (the difference between the SSRs for the full and reduced models) for partition i, and SSE_i denotes the SSE of the full model for partition i. Further, let $\mathbf{Z}_i$, $\mathbf{U}_i$, and $\mathbf{A}_i$ denote the transformed response and predictor variables, and the transformed errors, for partition i. It is possible to show that, for any m x m matrices $\mathbf{S}_i$ and $\mathbf{S}_j$,

$$\mathrm{cov}\left[(\mathbf{A}\prime_i\mathbf{S}_i\mathbf{A}_i)\left(\mathbf{A}\prime_j\mathbf{S}_j\mathbf{A}_j\right)\right] = vec(\mathbf{S}_i)\prime\mathrm{cov}\left[(\mathbf{A}_i\mathbf{A}\prime_i)\otimes\left(\mathbf{A}_j\mathbf{A}\prime_j\right)\right]vec\left(\mathbf{S}_j\right) \tag{12}$$

where vec is the vec operator that stacks columns of a matrix on top of each other to form a vector, and $\otimes$ is the Kronecker product. The matrix $\mathrm{cov}[(\mathbf{A}_i\mathbf{A}_i')\otimes(\mathbf{A}_j\mathbf{A}_j')]$ can be expressed in terms of the matrix $\mathbf{V}_{ij} = \sigma^2\mathbf{C}_{ij}$ containing covariances between errors $\gamma_{i,l}$ and $\gamma_{j,k}$ from partitions i and j, respectively. Specifically, $\mathrm{cov}[(\mathbf{A}_i\mathbf{A}_i')\otimes(\mathbf{A}_j\mathbf{A}_j')] = \mathbf{R}_{ij}\otimes\mathbf{R}_{ij} + \mathbf{P}$, where $\mathbf{R}_{ij} = \mathbf{D}_i\mathbf{C}_{ij}\mathbf{D}_j'$ and $\mathbf{P}$ is the matrix constructed by horizontally joining the matrices $\mathbf{R}_{ij}\otimes\mathbf{R}_{ij[\varphi]}$ where $[\varphi]$ denotes the φ^{th} column of $\mathbf{R}_{ij}$. Letting $\mathbf{H}_i$ and $\mathbf{H_0}_i$ denote the hat matrices $\mathbf{H}_i = \mathbf{U}_i(\mathbf{U}_i'\mathbf{U}_i)^{-1}\mathbf{U}_i'$ and $\mathbf{H_0}_i = \mathbf{U_0}_i(\mathbf{U_0}_i'\mathbf{U_0}_i)^{-1}\mathbf{U_0}_i'$, we have

$$\mathrm{cor}\left[SSR_i, SSR_j\right] = vec(\mathbf{H}_i - \mathbf{H}_{0i})'\left(\mathbf{R}_{ij}\otimes\mathbf{R}_{ij}\right)vec\left(\mathbf{H}_j - \mathbf{H}_{0j}\right)/df_1 \tag{13}$$

where the appearance of df_1 arises by noting that $\mathrm{var}[SSR_i] = vec(\mathbf{H}_i - \mathbf{H_0}_i)'vec(\mathbf{H}_i - \mathbf{H_0}_i) = 2df_1$. Eq. (13) was simplified using the empirically confirmed identity that $vec(\mathbf{H}_i - \mathbf{H_0}_i)'(\mathbf{R}_{ij}\otimes\mathbf{R}_{ij})\,vec(\mathbf{H}_j - \mathbf{H_0}_j) = vec(\mathbf{H}_i - \mathbf{H_0}_i)'\mathbf{P}vec(\mathbf{H}_j - \mathbf{H_0}_j)$ for the matrices $\mathbf{H}_j - \mathbf{H_0}_j$ under the null hypothesis. Using a similar derivation,

$$\mathrm{cor}\left[SSE_i, SSE_j\right] = vec(\mathbf{I} - \mathbf{H}_i)'\left(\mathbf{R}_{ij}\otimes\mathbf{R}_{ij}\right)vec\left(\mathbf{I} - \mathbf{H}_j\right)/df_2 \tag{14}$$

The LRT depends on the SSR_{1-0}, and therefore the method requires calculating the correlations between values of SSR_i from the n_p partitions, as is already given for the F-test. For t-tests on the coefficient values, it is necessary to calculate the correlations between the estimators of the coefficients given by Eq. (8), which are given as

$$\mathrm{cor}\left[\hat{\mathbf{B}}_i, \hat{\mathbf{B}}_j\right] = (\mathbf{U}\prime_i\mathbf{U}_i)^{-1}\left(\mathbf{U}\prime_i\mathbf{R}_{ij}\mathbf{U}_j\right)\left(\mathbf{U}\prime_j\mathbf{U}_j\right)^{-1} \tag{15}$$

where $\hat{\mathbf{B}}_i$ is the estimator of the coefficients from partition i.

Combining tests from the partitions

From the values of SSR_i and SSE_i calculated for each partition, $i = 1, 2, ..., n_p$, and the correlations between SSR_i and SSR_j, and between SSE_i and SSE_j, it is possible to compute an overall F-test for the

data. The procedure for the LRT is similar to that for the *F*-test, and below they are presented together. The procedure for the *t*-test is somewhat different and is described after the *F*-test and LRT.

For a given partition i, $2SSR_i$ follows a χ^2 distribution with df_1 degrees of freedom. A χ^2 distribution with df_1 degrees of freedom is the sum of df_1 squared Gaussian variables with mean 0 and variance 1. Thus, SSR_i can be expressed as $SSR_i = G^2_{i,1} + G^2_{i,2} + ... + G^2_{i,df1}$. The overall test statistic depends on the sum of SSR_i from all partitions $i = 1, 2, ..., n_p$. Let $\mathbf{G}$ denote the (n_p df_1) x 1 vector of values of $G_{i,\mu}$ ($\mu = 1, ..., df_1$) from all partitions. By construction Eqs. 8-(11), the distributions $G_{i,\mu}$ and $G_{i,\nu}$ ($\mu \neq \nu$) within the same partition i are independent. To satisfy the identity in Eq. (13) for *SSR*, the correlations between $G_{i,\mu}$ and $G_{j,\nu}$ from different partitions i and j are

$$\mathrm{cor}\left[G_{i,\mu}, G_{j,\nu}\right] = \left(\rho_{i,j}/df_1\right)^{1/2} \tag{16}$$

for $\rho_{i,j} = vec(\mathbf{S}_i)'(\mathbf{R}_{ij} \otimes \mathbf{R}_{ij})vec(\mathbf{S}_j)/df_1$ and $\mathbf{S}_i = (\mathbf{H}_i - \mathbf{H_0}_i)$. Letting $\mathbf{P}$ be the correlation matrix containing values of $\mathrm{cor}[G_{i,\mu}, G_{j,\nu}]$, the distribution of the sum of SSR_i from all partitions is

$$SSR = \sum_i SSR_i \sim \mathbf{G'PG} \tag{17}$$

where $\mathbf{G'PG}$ follows a quadratic Gaussian distribution. The probability density function of this quadratic Gaussian can be computed directly [5,7] to give the probability of $\mathbf{G'PG}$ being greater than an observed value of *SSR*, which produces the *P*-value of the LRT.

The *F*-test depends on both *SSR* and *SSE*, where *SSE* is defined like *SSR* as $SSE = \Sigma_i SSE_i$ and each SSE_i has $df_{2,i}$ degrees of freedom. Because partitions may differ in size, $df_{2,i}$ may differ among partitions. Because the values of $df_{2,i}$ will be large (certainly >100), the values of $SSE_i/df_{2,i}$ will be approximately Gaussian distributed with mean 1 and variance $2/df_{2,i}$. The correlation between the Gaussian distributions of $SSE_i/df_{2,i}$ and $SSE_i/df_{2,i}$ can be derived from Eq. (14) with $\mathbf{S}_i = (\mathbf{I} - \mathbf{H}_i)$. Thus, the *F*-score is approximated as the quadratic Gaussian distribution for *SSR* divided by the Gaussian distribution for *SSE*. There is no closed-form expression for this distribution, and therefore it is obtained via simulating a large number (e.g., 10^5) values to generate the approximate (parametric bootstrapped) test distribution.

For the *t*-test, the estimator of the coefficient $b_{h,i}$, $\hat{b}_{h,i}$, from partition i follows a Gaussian distribution, and the test statistic for the mean value of $\hat{b}_{h,i}$ from all partitions is $(\Sigma_i \hat{b}_{h,i}/n_p)/\mathrm{se}[\Sigma_i \hat{b}_{h,i}/n_p]$ Eq. (15). gives the correlation between $\hat{b}_{h,i}$ and $\hat{b}_{h,j}$ from partitions i and j, $\mathrm{cor}[\hat{b}_{h,i}, \hat{b}_{h,j}]$. Thus, $\mathrm{se}[\Sigma_i \hat{b}_{h,i}/n_p]^2 = (1/n_p)^2 \Sigma_{ij} \mathrm{cor}[\hat{b}_{h,i}, \hat{b}_{h,j}]\mathrm{se}[\hat{b}_{h,i}]\mathrm{se}[\hat{b}_{h,j}]$ is the sum of n_p random variables each having a χ^2 distribution with $df_{2,i}$ degrees of freedom. From this expression, $(\Sigma_i \hat{b}_{h,i}/n_p)/\mathrm{se}[\Sigma_i \hat{b}_{h,i}/n_p]$ is approximately distributed as a *t*-distribution with $\Sigma_i df_{2,i}$ degrees of freedom; for large degrees of freedom $df_{2,i}$, this will approach a Gaussian distribution with mean zero and variance one.

Spatiotemporal analyses

The spatiotemporal analyses follow the approach presented in Ives et al. [9] for analyzing time trends in remote-sensing data. The approach involves first fitting a time-series model to the time series in each pixel on a map and obtaining the estimate of the time trend. As described below, the correlations between the residuals obtained from the fitted time-series model approximate the spatial autocorrelation of the estimated coefficients of the time trends. Therefore, the spatial autocorrelation matrix required for the GLS spatial analysis can be estimated before the GLS analysis is performed. Although this approach can be used for different time-series models, here we focus on two: least-squares (LS) regression, and regression with AR(1) errors estimated using REML. The former is useful, because it allows analytical solutions, whereas the second has better statistical properties and is therefore preferentially used for the analyses in Ives et al. [9].

To explain the approach, we use a specific spatiotemporal model, and we also use this model in the validation. The model for time series within pixel l is

$$z_l(t) = a_l + c_l t + \varepsilon_l(t)$$

$$\varepsilon_l(t) = \beta_l \varepsilon_l(t-1) + \delta_l(t) \tag{18}$$

Here, $z_l(t)$ is the value of interest in pixel l at time t (t = 1, 2, ..., T), a_l is the intercept, and c_l is the time trend. Random errors $\varepsilon_l(t)$ follow a stationary first-order Gaussian autoregressive process with mean zero generated from the Gaussian random variable $\delta_l(t)$ that has mean zero and variance σ^2, with values independent through time so that $E[\delta_l(t)\ \delta_l(s)] = 0$ for $s \neq t$. Thus, the vector $(\varepsilon_l(1)$, ..., $\varepsilon_l(T))'$,has distribution $N(0, \sigma^2/(1 - \beta_l{}^2)\ \mathbf{\Sigma}_l)$, where the elements of the correlation matrix $\mathbf{\Sigma}_l$ are $\text{cor}[\varepsilon_l(t), \varepsilon_l(s)] = \beta_l{}^{|t-s|}$ for all t and s. To include spatial autocorrelation, we assume that the Gaussian random variables $\delta_l(t)$ and $\delta_k(t)$ at t from pixels l and k are correlated, with parameter $\text{cor}\delta = \text{cor}[\delta_l(t), \delta_k(t)]$, but values of $\delta_l(t)$ and $\delta_k(s)$ are independent when $s \neq t$.

The procedure for spatiotemporal data collapses temporal information from the time series into two quantities: the pixel-specific estimates of the time trend c_l and the correlations between the temporal errors $\varepsilon_l(t)$ and $\varepsilon_k(t)$ from pixels l and k. Note that the estimates of c_l, $\hat{c}_l$, are now the dependent variable in the spatial model, rather than the data $z_l(t)$. For LS regression, the exact relationship between the correlation $\text{cor}[\hat{c}_l, \hat{c}_k]$ and $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ can be derived analytically under the assumption that the temporal autocorrelation coefficients β_l and β_k are known. The T x T covariance matrix $\mathbf{W}_{lk}$ whose t,s-element (t = 1, ..., T; s = 1, ..., T) is the covariance between $\varepsilon_l(t)$ and $\varepsilon_k(s)$ is given by

$$\mathbf{W}_{lk} = \text{cor}_\delta\ (\mathbf{I} - \beta_i \mathbf{\Psi})^{-1}\ \mathbf{\Delta}_{lk}\left((\mathbf{I} - \beta_k \mathbf{\Psi})^{-1}\right)' \tag{19}$$

where $\mathbf{I}$ is the T x T identity matrix, and $\mathbf{\Psi}$ is the backward shift operator [3]. Under the assumption that the time series are sampled from their stationary distributions, $\mathbf{\Delta}_{lk}$ is a diagonal matrix whose first diagonal element is the value of $\text{cov}[\varepsilon_l(t), \varepsilon_k(t)]$ at the stationary distribution, and other diagonal elements are one. The stationary distribution of $\mathbf{E}(t) = (\varepsilon_l(t), \varepsilon_k(t))$, follows a bivariate Gaussian distribution with mean (0, 0), and covariance matrix $\mathbf{\Omega}_\mathbf{E}$ satisfying $\text{vec}(\mathbf{\Omega}_\mathbf{E}) = \sigma^2(\mathbf{I} - \mathbf{\Phi}\otimes\mathbf{\Phi})^{-1} vec(\mathbf{\Omega})$ where $\mathbf{\Phi}$ is the 2 x 2 diagonal matrix containing β_l and β_k, and $\mathbf{\Omega}$ is the 2 x 2 matrix with 1 on the diagonal and $\text{cor}\delta$ on the off-diagonal.

For LS regression, $\text{cor}[\hat{c}_l, \hat{c}_k]$ can be computed as

$$\text{cor}\left[\hat{c}_l, \hat{c}_k\right] = \left(\mathbf{K}\mathbf{W}_{lk}\mathbf{K}'\right)/\left(\left(\mathbf{K}\mathbf{W}_{ll}\mathbf{K}'\right)\left(\mathbf{K}\mathbf{W}_{kk}\mathbf{K}'\right)\right) \tag{20}$$

where $\mathbf{K}$ is the second row of the 2 x T matrix $(\mathbf{J}'\mathbf{J})^{-1}\mathbf{J}$, where $\mathbf{J}$ is the T x 2 matrix containing 1 in the first column and time t = 1, ..., T in the second column; in other words, $\mathbf{J}$ is the matrix of independent variables that would be used to fit Eq. (18) using LS regression.

We used Eqs. (19) and (20) to explore the relationship between $\text{cor}[\hat{c}_l, \hat{c}_k]$ and $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$, thereby investigating the validity of using $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ as an approximation of $\text{cor}[\hat{c}_l, \hat{c}_k]$ in a spatial GLS analysis of $\hat{c}_l$. Eqs. (19) and (20) apply only for LS regression analyses of the time series (Eq. 18), and therefore we performed simulations for regression with AR(1) errors estimated using REML. When $\beta_l = \beta_k$, $\text{cor}[\hat{c}_l, \hat{c}_k] = \text{cor}[\varepsilon_l(t), \varepsilon_k(t)] = \text{cor}\delta$ for LS regression (Table 1). For AR(1) regression, $\text{cor}[\hat{c}_l, \hat{c}_k]$ and $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ are slightly lower than $\text{cor}\delta$, although they are equal (within the uncertainty of the simulation) (Table 2). Therefore, when $\beta_l = \beta_k$, $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ is an excellent approximation for $\text{cor}[\hat{c}_l, \hat{c}_k]$. Keeping $\beta_l = 0.8$, as β_k decreases from the high value of $\beta_k = 0.8$, $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ decreases below $\text{cor}\delta$; $\text{cor}[\hat{c}_l, \hat{c}_k]$ also decreases below $\text{cor}\delta$, but the decrease is not as great as $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ especially for longer time series (larger T). Therefore, $\text{cor}[\hat{c}_l, \hat{c}_k]$ is generally greater than $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$, with the ratio $\text{cor}[\hat{c}_l, \hat{c}_k]/\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ greater when the difference between β_l and β_k is large, and when the time-series are longer. For the example dataset from Alaska analyzed in Ives et al. [9], the time series have length T = 34, and estimates of β_l had a mean of 0.40 and a standard deviation of 0.23. The simulations in Table 2 for AR(1) regression suggest that using $\text{cor}[\varepsilon_l(t), \varepsilon_k(t)]$ to approximate $\text{cor}[\hat{c}_l, \hat{c}_k]$ will lead to a small overestimate of $\text{cor}[\hat{c}_l, \hat{c}_k]$, although this will not change the conclusions; this is discussed in detail in Ives et al. [9].

To build the correlation matrix $\mathbf{C}$ for the GLS spatial analysis, we assume that $\text{cor}[\hat{c}_l, \hat{c}_k]$ decays according to some function $v(d_{lk})$ of the distance d_{lk} between pixels l and k. For example, Ives et al. [9]use an exponential-power function $v(d_{lk}) = \exp(-(d_{lk}/r)^g)$ where the range parameter r and shape

Table 1
Relationship between the correlation for estimates of the time trend parameter, cor[$\hat{\boldsymbol{c}}_l$, $\hat{\boldsymbol{c}}_k$], and the correlation for residuals from least-squares (LS) regression, cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$]. To explore extreme differences in the strength of temporal autocorrelation between time series, $\beta_l = 0.8$ for one time series and $\beta_k = -0.4$, 0, 0.4, and 0.8 for the other time series, with time-series length $T = 10$, 30, or 100. The values of cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$] and cor[$\hat{c}_l$, $\hat{c}_k$] were calculated analytically using Eqs. (19) and (20), respectively. Throughout, the parameter governing the correlation between $\delta_l(t)$ and $\delta_l(s)$, corδ, was 0.5.

β_l	β_k	T	cor$_\delta$	cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$]	cor[$\hat{c}_l$, $\hat{c}_k$]	cor[$\hat{c}_l$, $\hat{c}_k$]/cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$]
0.8	-0.4	10	0.5	0.21	0.25	1.21
0.8	-0.4	30	0.5	0.21	0.38	1.84
0.8	-0.4	100	0.5	0.21	0.46	2.22
0.8	0.0	10	0.5	0.30	0.32	1.06
0.8	0.0	30	0.5	0.30	0.41	1.37
0.8	0.0	100	0.5	0.30	0.47	1.58
0.8	0.4	10	0.5	0.40	0.41	1.00
0.8	0.4	30	0.5	0.40	0.45	1.11
0.8	0.4	100	0.5	0.40	0.48	1.20
0.8	0.8	10	0.5	0.50	0.50	1.00
0.8	0.8	30	0.5	0.50	0.50	1.00
0.8	0.8	100	0.5	0.50	0.50	1.00

Table 2
Relationship between the correlation for estimates of the time trend parameter, cor[$\hat{c}_l$, $\hat{c}_k$], and the correlation for residuals, cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$], from regression with AR(1) errors fit with REML. To explore extreme differences in the strength of temporal autocorrelation between time series, $\beta_l = 0.8$ for one time series and $\beta_k = -0.4$, 0, 0.4, and 0.8 for the other time series, with time-series length $T = 10$, 30, or 100. The values of cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$] and cor[$\hat{c}_l$, $\hat{c}_k$] were calculated by simulating Eq. (18) for 50,000 pairs of time series. Throughout, the parameter governing the correlation between $\delta_l(t)$ and $\delta_l(s)$, cor$_\delta$, was 0.5.

β_l	β_k	T	cor$_\delta$	cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$]	cor[$\hat{c}_l$, $\hat{c}_k$]	cor[$\hat{c}_l$, $\hat{c}_k$]/cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$]
0.8	-0.4	10	0.5	0.23	0.24	1.04
0.8	-0.4	30	0.5	0.22	0.33	1.54
0.8	-0.4	100	0.5	0.21	0.43	2.06
0.8	0.0	10	0.5	0.31	0.30	0.95
0.8	0.0	30	0.5	0.31	0.37	1.19
0.8	0.0	100	0.5	0.30	0.44	1.47
0.8	0.4	10	0.5	0.41	0.40	0.97
0.8	0.4	30	0.5	0.40	0.41	1.01
0.8	0.4	100	0.5	0.40	0.47	1.16
0.8	0.8	10	0.5	0.46	0.49	1.06
0.8	0.8	30	0.5	0.48	0.47	0.98
0.8	0.8	100	0.5	0.49	0.48	0.99

parameter g are estimated from the calculated values of cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$] for a subset of pairs of locations using nonlinear regression of cor[$\varepsilon_l(t)$, $\varepsilon_k(t)$] on d_{lk}. For large datasets, the estimates of parameters in $v(d_{lk})$ have small standard errors, and therefore uncertainty in $v(d_{lk})$ is assumed to be negligible.

Implementation

The partition approach for performing statistical tests can be implemented directly using the derivations above. When using random partitions, for large datasets it is not necessary to calculate the pairwise correlations between SSR_i (or SSE_i) from all partitions Eqs. 13,(14), or the pairwise correlations between the estimates of the coefficients from all partitions (Eq. 15). These correlations vary little between different pairs of partitions, and the number of partitions can be set after examining the variation in correlations for specific datasets. This approach saves considerable computational time for large datasets with many partitions.

Although the methods are presented assuming that the correlation matrices $\mathbf{C_i}$ are known, in many applications $\mathbf{C_i}$ will contain one or more parameters to be estimated. For example, spatial models will often contain a "nugget" to capture local (non-spatial autocorrelation) variation [4,9]. In the spatiotemporal analysis we outline above, other parameters giving the spatial extent of the autocorrelation (e.g., r in $v(d_{lk})$) can be estimated from the residuals of the time-series analyses. However, for the purely spatial model these would be estimated during the GLS spatial analysis. Any parameters of $\mathbf{C_i}$ can be estimated for each partition separately and the formulae applied with the estimated matrices of $\mathbf{C_i}$. The resulting omnibus test statistics are then conditional on the parameter estimates of $\mathbf{C_i}$.

The methods are implemented as a part of the package remotePARTS in the R programming language. It is available at https://github.com/morrowcj/remotePARTS.

Methods validation

Spatial model

To assess type I error rates and power, we performed a simulation study using the regression model

$$y_l = b_0 + bx_l + \gamma_l \tag{21}$$

in which the N spatial errors γ_l follow a multivariate Gaussian distribution with correlation matrix $\mathbf{C}$, N(0, ${\sigma^2}_\gamma\mathbf{C}$). The simulation was performed on a 60 x 60 pixel map (N = 3,600), which was small enough to perform a GLS analysis Eqs. 1-(7) without partitioning the datasets. Spatial autocorrelation was introduced by assuming that the elements of $\mathbf{C}$ equal $\exp(-d_{lk}/r)$. Values of r were standardized to the scale of the map, and values of r = 0.03 and 0.1 correspond to 3% and 10% of the maximum distance on the map (corner to corner). Values of x_l were given by the "latitude" that was defined as the row number in the 60 x 60 map divided by 60. For each value of b = 0, 2, 4, ..., 20, five hundred simulations were performed, and the null hypothesis H_0:b = 0 was tested for each at the significance levels of α = 0.05 and 0.01. We performed the partition analyses with eight partitions, and computed the pairwise correlations between SSR_i (or SSE_i) Eqs. 13,(14) as the average from a random subset of six partitions (a subset of 15 of the 28 total number of pairwise correlations).

Fig. 1 gives the proportion of the simulations for which H_0:b = 0 was rejected, with the significance level α given by the black dotted line. The tests were based on different methods for producing P-values. First, GLS (Eq. 4) was used to perform an F-test (P_{GLS}), which gives the "gold standard" since the GLS is the best linear unbiased estimator (BLUE) [10]. Second, the partition method developed here was applied using eight partitions to give an F-test (P_F), a LRT (P_{LRT}), and a t-test (P_t). Third, the lowest P-value from the eight partitions from F-tests was selected and adjusted for eight multiple comparisons using either the Hochberg adjustment (P_{hoch}) [6] or the False Discovery Rate adjustment (P_{fdr}) [2]. Finally, a single partition was selected at random and its P-value was used (P_{single}).

As expected, all three values using the method developed here (P_F, P_{LRT}, and P_t) gave very similar results. Type I error rates were appropriate, with close to 5% and 1% of simulations rejected at significance levels of α = 0.05 and 0.01, respectively. There was slight loss of power in comparison to the GLS (P_{GLS}). Nonetheless, this loss of power was less than that of either Hochberg or FDR adjustments for multiple comparisons (P_{hoch} and P_{fdr}). In fact, when autocorrelation was high (r = 0.1), the Hochberg or FDR adjustments had lower power than the randomly chosen partition (P_{single}).

Because partitions were created randomly, there is variation in the test statistics depending on the partitions created. To illustrate this, Fig. 2 gives the distributions of P-values for a partition LRT of H_0:b = 0 for two simulated datasets constructed as described above with r = 0.03 (Fig. 2a) and r = 0.1 (Fig. 2b). The variation in the P_{LRT} with eight partitions is created solely by the selection of different partitions, because the datasets were the same for all analyses in the same panel. It is interesting that the variation in P-values is less for the more-highly autocorrelated dataset (r = 0.1, Fig. 2b), which is likely due to the greater correlations between SSR_i from different partitions which makes the test scores from the separate partitions more similar. Finally, note that these results are

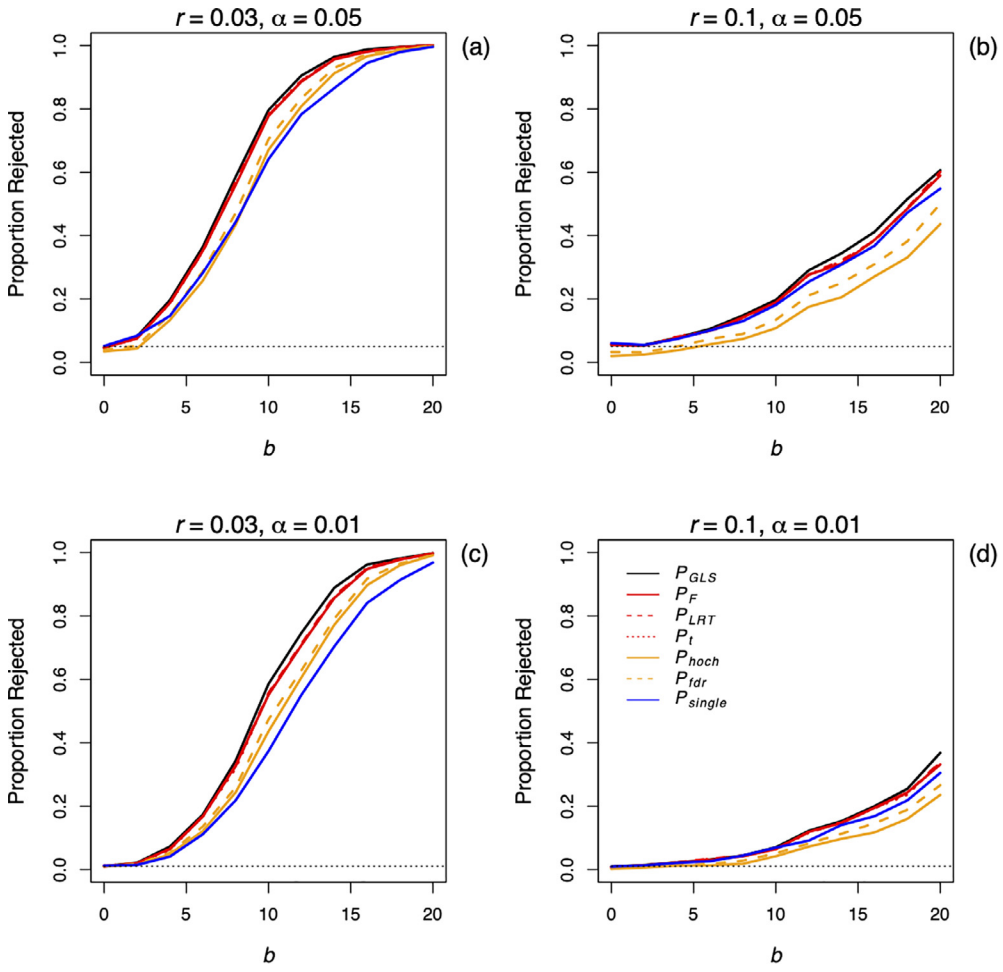

Fig. 1. Type I errors and power for the methods combining correlated tests among partitions for spatial data. Five-hundred simulations were performed on a 60 x 60 map using equation (19) for each value of b = 0, 2, 4, ..., 20, with either moderate (r = 0.03; a,c) or strong (r = 0.1; b,d) spatial autocorrelation. The hypothesis H_0:b = 0 was tested at significance levels of α = 0.05 (a,b) and 0.01 (c,d) using: an GLS F-test (P_{GLS}); the partition method with eight partitions giving an F-test (P_F), a LRT (P_{LRT}), and a t-test (P_t); selecting the lowest P-value and applying a Hochberg (P_{hoch}) or the False Discovery Rate adjustment (P_{fdr}); and randomly selecting one partition (P_{single}).

for "small" datasets compared to those that the method was designed to analyze; the variation in P-values from different random partitions is less for larger remote-sensing datasets [9]. For smaller datasets, the analyses can be run multiple times with different random partitions, and the overall P-value is selected as the median of multiple values computed.

To compare the statistical results for different numbers (and hence sizes) of partitions, we use the same simulation model on a 60 x 60 pixel map to investigate the power to reject H_0:b = 0 using a LRT when the true value is b = 10 Table 3). For 8 and 16 partitions, a subset of six partitions was chosen at random to compute the pairwise correlations between SSR_i among partitions and SSE_i among partitions (Eqs. 13,(14). The proportion of the simulations for which H_0:b = 0 was rejected changed little with the number of partitions, n_p = 1, 2, 4, 8, and 16. This suggests that loss of power when partitioning data is insensitive to the number of partitions.

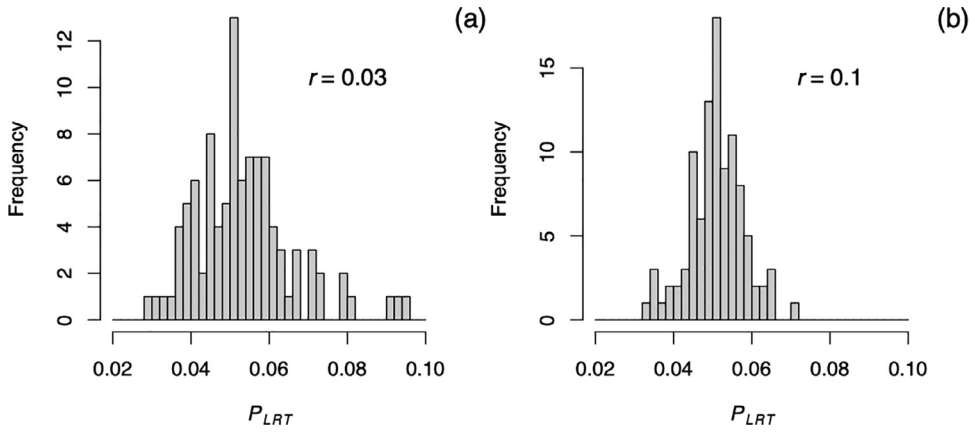

Fig. 2. *P*-values from a LRT of H_0:$b = 0$ applied to the same simulated 60 x 60 pixel dataset but using different random partitions. (a) A simulated dataset with $b = 20$ and the range parameter $r = 0.03$, and (b) a simulated dataset with $b = 5$ and the range parameter $r = 0.1$. The example datasets were selected to give *P*-values near 0.05. For each panel, the same dataset was fit 100 times with different partitions. Eight random partitions each containing 450 pixels were selected for each analysis.

Table 3
Power of the partition method for a given number of partitions. Five-hundred simulations were performed on a 60 x 60 map using Eq. (21) with $b = 10$ and either moderate ($r = 0.03$) or strong ($r = 0.10$) spatial autocorrelation. The hypothesis H_0:$b = 0$ was tested at a significance level of $\alpha = 0.05$ for different numbers of partitions: 1, 2, 4, 8, and 16. Significance was determined using an F-test (for 1 partition, i.e., the entire dataset) and LRTs (for partitions $n_p = 2$, 4, 8, and 16). The reported values are the proportions of the 500 simulations for which H_0:$b = 0$ was rejected.

Number of partitions	$r = 0.03$	$r = 0.10$
1	0.722	0.188
2	0.732	0.196
4	0.724	0.192
8	0.716	0.196
16	0.718	0.200

Spatiotemporal model

We performed a simulation study using the time-series model given in Eq. (18) on a map of 40 x 40 pixels. Each time series was 30 data points long, with moderate temporal autocorrelation ($\beta_l = 0.2$). Spatial autocorrelation in the matrix **C** was assumed to have the form $v(d_{lk}) = (1 - nugget)\exp(-(d_{lk}/r))$ for $l \neq k$; thus, a proportion *nugget* of the error variance is "local" to a pixel. Spatial autocorrelation was assumed to be either moderate or strong ($r = 0.03$ or $r = 0.1$). We estimated r from the correlation among residuals, while *nugget* was estimated during the GLS spatial analysis via maximum likelihood. Thus, unlike the spatial example (Fig. 1), the spatiotemporal example included a parameter in matrix **C** that was estimated.

We assumed that the map was divided into 16 squares (10 x 10 pixels each) and assigned to four land-cover classes (e.g., Fig. 4 in [9]). The time trends in each of the four land-cover classes were given by $c_1 = 0$, $c_2 = c$, $c_3 = 2c$, $c_4 = 3c$, with values of $c = 0$, 0.04, ..., 0.20. We simulated 500 datasets for each parameter combination and tested the hypothesis that there were no differences in

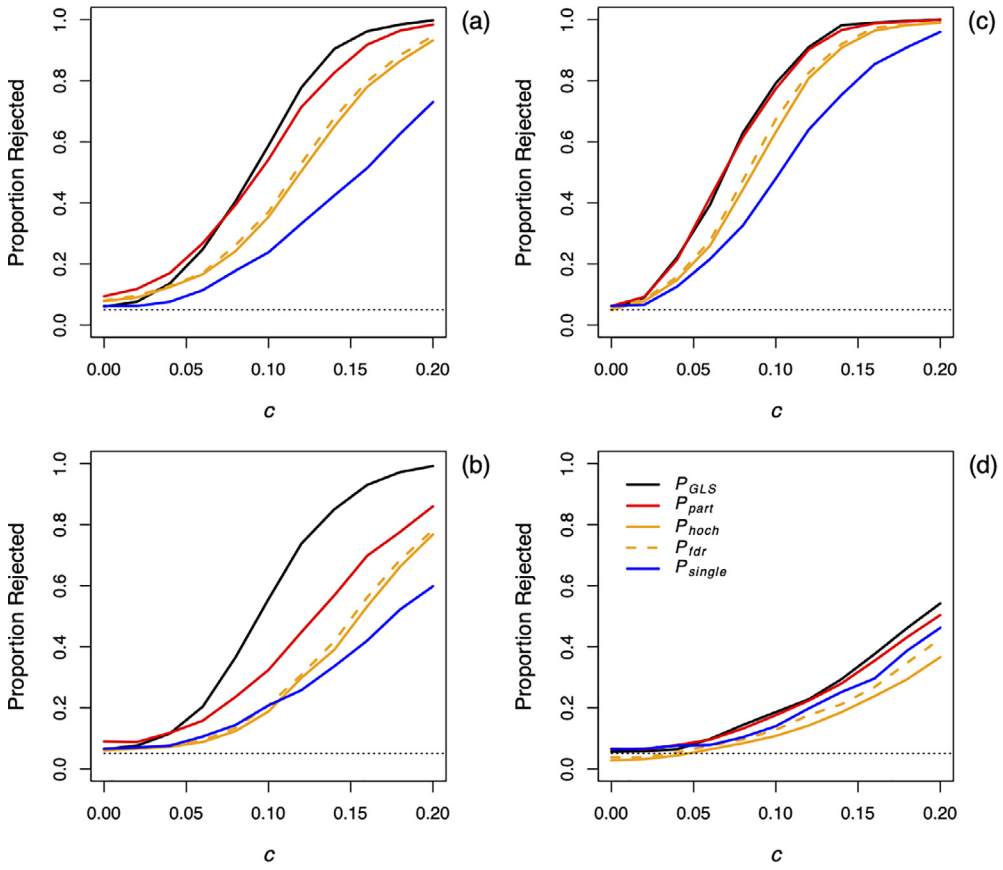

Fig. 3. Type I errors and power for the methods combining correlated tests among partitions for spatiotemporal data. Five-hundred simulations were performed using Eq. (16), and the proportions of simulated datasets for which the null hypotheses of (a,b) no differences in time trends among land-cover classes and (c,d) no overall time trend were rejected at the significance level of $\alpha = 0.05$. Simulations were performed for weak ($r = 0.03$; a,c) and strong ($r = 0.1$; b,d) spatial autocorrelation. The true time trends for the four land-cover classes were $c_1 = 0$, $c_2 = c$, $c_3 = 2c$, $c_4 = 3c$, with values of $c = 0$, 0.02, ..., 0.2. To scale the time trends, the time variable t ranged from 0 to 1 over the 30-year simulated time series, and the standard deviation σ of $\delta_l(t)$ was 1. Therefore, a value of 0.12, for example, represents a change in the mean value of $z_l(t)$ over 30 years of $0.12\sigma(1 - \beta_l^2)^{-0.5}$. To include pixel-scale (non-spatially autocorrelated) variation, we added non-spatial variation in the form of a Gaussian random variable with mean zero and variance 0.16 to c for each pixel. Simulations were performed for a 40 x 40 grid of pixels with weak temporal autocorrelation ($\beta_l = 0.2$). The hypotheses were tested using (i) a GLS analysis applied to all of the data (P_{GLS}), (ii) a GLS applied to eight partitions of the data with the log-likelihood ratios combined among partitions (P_{part}), (iii) the lowest P-value from the GLS analyses of the eight partitions correcting for multiple comparisons using a Hochberg (P_{hoch}) or False Discovery Rate (P_{fdr}) adjustment, and (iv) a randomly selected partition (P_{single}).

trends among land-cover classes, H_0:c_1=c_2=c_3=c_4, and the hypothesis that there was no overall time trend, H_0:c=0. We performed the test using GLS on the entire map. We also partitioned the map into eight random partitions and tested each separately. We then combined the tests either by selecting the partition with the lowest P-value and adjusting for multiple comparisons, or combining the tests as described in the section *Combining tests from the partitions*.

For both tests of differences among land-cover classes (Fig. 3a, b) and tests for an overall trend (Fig. 3c, d), the GLS analyses of the entire map had the highest statistical power to reject the null hypothesis. The method combining statistical results among partitions had the second-highest power (P_{part}), while the method using adjustments for multiple comparisons had lower power (P_{hoch}

and P_{fdr}). Finally, in general picking a single partition at random had the lowest power (P_{single}). Spatial autocorrelation reduced the statistical power of all methods ($r = 0.1$; Fig. 3b, d). The method combining statistical results among partitions (P_{part}) had somewhat inflated type I error rates for the analyses of land-cover classes, rejecting 9% of the simulated datasets when the null hypothesis was true. The inflated type I error rates, however, were the result of the relatively small size of the simulated map (40 x 40 pixels) necessitated by the application of the GLS analysis of the entire map (P_{GLS}). Repeating the same analysis on a 60 x 60 pixel map (P_{part}) gave rejection rates of 5.4% ($r = 0.03$) and 5.2% ($r = 0.1$). The inflated type I errors for the smaller map occurred due to the estimation of the *nugget*, because GLS in which no parameters in the correlation matrix **C** are estimated does not give inflated type I errors [10].

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank the editor and three anonymous reviewers for very helpful comments on the overall project from which this work derives. This work was supported by NASA-AIST [80NSSC20K0282] to ARI, Volker C. Radeloff, Fangfang Wang, and Jun Zhu, and NSF [DEB-1556208] to ARI.

References

[1] S. Banerjee, A.E. Gelfand, A.O. Finley, H. Sang, Stationary process approximation for the analysis of large spatial datasets, J. R. Stat. Soc. Ser. B-Stat. Methodol. 70 (2008) 825–848.
[2] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: a practical and powerful approach to multiple testing, J. R. Stat. Soc. Ser. B-Stat. Methodol. 57 (1995) 289–300.
[3] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, Time series analysis: forecasting and control, Third ed, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1994.
[4] N.A.C. Cressie, Statistics for spatial data, Wiley & Sons, Inc, New York, USA, 1993 *revised edition*.
[5] P. Duchesne, P. Lafaye de Micheaux, Computing the distribution of quadratic forms: further comparisons between the Liu-Tang-Zhang approximation and exact methods, Comput. Stat. Data Anal. 54 (2010) 858–862.
[6] Y. Hochberg, A sharper bonferroni procedure for multiple tests of significance, Biometrika 75 (1988) 800–802.
[7] J.P. Imhof, Computing the distribution of quadratic forms in normal variables, Biometrika 48 (1961) 419–426.
[8] A.R. Ives, J. Zhu, Statistics for correlated data: phylogenies, space, and time, Ecol. Appl. 16 (2006) 20–32.
[9] A.R. Ives, L. Zhu, F. Wang, J. Zhu, C.J. Morrow, V.C. Radeloff, Statistical inference for trends in spatiotemporal data, Remote Sens. Environ. 266 (2021) 112678, doi:10.1016/j.rse.2021.112678.
[10] G.G. Judge, W.E. Griffiths, R.C. Hill, H. Lutkepohl, T.-C. Lee, The theory and practice of econometrics, Second ed, John Wiley and Sons, New York, 1985.
[11] E.T. Krainski, V. Gómez-Rubio, H. Bakka, A. Lenzi, D. Castro-Camilo, D. Simpson, F. Lindgren, H. Rue, Advanced spatial modeling with stochastic partial differential equations using R and INLA, CRC Press/Taylor and Francis Group, 2019.
[12] J. Neter, W. Wasserman, M.H. Kutner, Applied linear regression models, Inc., Homewood, IL, 1989 Richard D. Irwin.
[13] C.K. Wikle, A. Zammit-Mangion, N. Cressie, Spatio-temporal statistics with R, Taylor and Francis Group, Boca Raton, FL, 2019.
[14] Zammit-Mangion, A., & Cressie, N. (2018). FRK: An R package for spatial and spatio-temporal prediction with large datasets. *arXiv*