Integrating Data Across Misaligned Spatial Units

Yuri Zhukov[1], Jason S. Byers[2], Marty Davidson[3], and Ken Kollman[4]

[1,3,4]University of Michigan
[2]Duke University

September 12, 2022

Abstract

Theoretical units of interest often do not align with the spatial units at which data are available. This problem is pervasive in political science, particularly in sub-national empirical research that requires integrating data across incompatible geographic units (e.g., administrative areas, electoral constituencies, grid cells). Overcoming this challenge requires researchers not only to align the scale of empirical and theoretical units, but also to understand the consequences of this change of support for measurement error and statistical inference. We show how the accuracy of transformed values and the estimation of regression coefficients depend on the degree of nesting (i.e., whether units fall completely and neatly inside each other) and on the relative scale of source and destination units (i.e., aggregation, disaggregation, hybrid). We introduce simple, nonparametric measures of relative nesting and scale, as *ex ante* indicators of spatial transformation complexity and error susceptibility. Using election data and Monte Carlo simulations, we show that these measures are strongly predictive of transformation quality across multiple change-of-support methods. We propose several validation procedures and provide open-source software to make transformation options more accessible, customizable, and intuitive.

Misaligned units of analysis present challenges for users of geospatial data. Researchers studying legislative elections in the U.S. might observe data for some variables at the electoral district level (e.g. campaign strategies) and data for other variables at the county level (e.g., crime). To understand how, for example, local crime influences campaign strategies, one must integrate these two datasets somehow, using measured values at the county level to estimate levels of crime in each legislative district. Statistically, this represents a change-of-support (CoS) problem: one makes inferences about a variable at one geographic support (*destination units*) using measurements from a different support (*source units*). Changes of support always entail information loss, potentially leading to consequential measurement error and biased estimation of statistical relationships and parameters. Substantively, this is a general problem of mismatch between demand for data at theoretically-relevant levels and the reality that data may be unavailable at those levels.

How prevalent are such problems in social science? Mismatched units are quite common. They routinely appear, for instance, in studies of sub-national (within-country) variation, where data are accessible at disparate levels of analysis, reflecting varying degrees of geographic precision, or different definitions of the same units across data sources. These units and scales (e.g., administrative areas, postal codes, grids) do not always correspond to theoretical quantities of interest. Nor do they always relate to each other in straightforward ways. While some units are perfectly nested (e.g., U.S. counties and states), others overlap only partially (e.g., counties and legislative districts).[1]

If testing the implications of a theory requires data at one geographic support (e.g., electoral constituency), but at least some of the theoretically relevant data are only available at another support (e.g., administrative unit), researchers face a consequential choice: either conduct tests not commensurate with the spatial levels at which a theory is specified, or convert the data to more appropriate units. The latter — transforming the data — can be a messy step, which many researchers make implicitly or explicitly at some point in the data management process. In a review of sub-national empirical research published in top political science journals since 2010, we found that 20% of articles, across all major empir-

[1]While CoS problems can include data transformations involving points, lines and continuous surfaces (rasters), we limit our scope to areal units (polygons), due to their ubiquity in social science.

ical subfields, change the geographic support of key variables.[2] For example, Benmelech, Berrebi and Klor (2015) explore the relationship between house demolitions and suicide terrorist attacks in Israel by aggregating data on places of residence to the district level. Similarly, Branton et al. (2015) examine the impact of protests on public opinion regarding immigration policy, by combining data across multiple geographic units with county-level data on survey responses and protest events. Rozenas and Zhukov (2019) study the effect of famine on elections, protests, and violence in Ukraine, by transforming several historical datasets to 1933-era Soviet administrative units.

It can be tempting to treat changes of support as routine, similar to merging two tables by a common ID. But integrating misaligned geospatial data is an error-prone process (Matheron, 1989; Gotway and Young, 2002, 2007) and the literature currently lacks common standards for evaluating it. Much of the canonical geostatistical literature on this topic relates to geology and environmental science (e.g. Cressie, 1996). While multiple transformation options exist (see Comber and Zeng, 2019), there is no "silver bullet." Despite recent efforts to address CoS problems in political science (Goplerud, 2016; Donnay and Linke, 2016; Lee and Rogers, 2019; Darmofal and Eddy, 2020), economics (Eckert et al., 2020) and public health (Zhu, Waller and Ma, 2013), there are no prevailing best practices.

We make three contributions. First is a framework for diagnosing and addressing CoS problems involving discrete geographic areas, focusing on the relative degree of nesting (i.e. whether one set of units falls completely and neatly inside the other), and the relative scale of source and destination units (i.e. aggregation, disaggregation, hybrid). We propose simple, nonparametric measures of nesting and scale that can help assess *ex ante* the complexity of spatial transformations.

Second, we show that relative nesting and scale are consequential for the performance of common spatial transformation methods, including overlays, interpolation, and kriging. We hone in on the problem using examples with U.S. and Swedish electoral data. Election data present challenges theoretically and empirically; they also offer opportunities for advancements in the study of CoS problems. Electoral units — districts, constituencies, precincts — can be oddly shaped. These shapes can change endogenously, frustrating attempts to measure and explain how the same community's voting behavior evolves over

[2]See Appendix A1 for a full enumeration of articles and journals included in this review.

time. We transform electoral data across spatial units and validate the results with "ground truth" data from precincts. We confirm the intuition that measurement error and biased estimation of regression coefficients are most severe when disaggregating non-nested units, and least severe when aggregating nested units. We generalize the problem in a Monte Carlo analysis with random spatial units. While much of the literature has highlighted the risks of over-aggregation, our analysis shows that dissaggregation can create threats to inference that are also quite severe (Cook and Weidmann, 2022).

Third is practical advice to researchers, recommending (a) the reporting of nesting and scale metrics for all spatial data transformations, (b) checking the face-validity of transformations where possible, and (c) performing sensitivity analyses with alternative transformation methods, particularly if "ground truth" data for validation are not available.

Additionally, we introduce new open-source software (`SUNGEO` R package) to equip researchers with routines and documentation to implement these techniques on their own.

There is no one-size-fits-all solution. Our goals are to make spatial transformation options accessible and intuitive, reveal the conditions under which certain options may be preferable, and make transparent the consequences of these methodological decisions.

1 Problem Setup

The *geographic support* of a variable is the area, shape, size, and orientation of its spatial measurement. *Change-of-support problems* (CoS) emerge when we make statistical inferences about a spatial variable at one support by using data from a different support. One general case occurs when no data for relevant variables are available for desired spatial units. For example, a theory may be specified at the level of one unit (e.g., counties), but available data are either at smaller levels (e.g., neighborhoods), at larger levels (e.g., states) or are otherwise incompatible (e.g., grid cells, legislative districts, police precincts). The second case arises when different data sources define the same units differently. Data on geographic areas vary in the precision and placement of boundaries, and there are few universally-accepted standards for assigning and classifying units (e.g., handling of disputed territories). A third case involves variation in unit geometries over time. Historical changes in the number of units (e.g., splits, consolidations, reapportionment), their bound-

aries (e.g., annexation, partition, redistricting), and their names can all occur.

Changes of support involve potentially complex and interdependent choices, affecting data reliability and substantive inferences about quantities of interest. Yet a CoS is often unavoidable, since the alternative — using data from theoretically inappropriate units — is itself only feasible if all other data are available for those same units.

CoS problems relate to several others: ecological inference (EI) — deducing lower-level variation from aggregate data (Robinson, 1950) — the modifiable areal unit problem (MAUP) — that statistical inferences depend on the geographical regions at which data are observed (Openshaw and Taylor, 1979) — and Simpson's paradox — a more general version of MAUP, where data can be grouped in alternative ways affecting inference. In each case, inferential problems arise primarily due to the segmenting of data into different units (e.g., in geographic terms, "scale effect" in MAUP, or "aggregation bias" in EI), or due to differences in unit shape and the distribution of confounding variables (e.g., "zoning effect" in MAUP, or "specification bias" in EI) (Morgenstern, 1982).

Geostatisticians view EI and MAUP as special cases of CoS problems (Gotway and Young, 2002). In political science, cross-level inference problems have long bedeviled research into micro-level attitudes and behavior. Because information is inevitably lost in aggregation, using aggregate data to infer information about lower-level phenomena nearly always introduces error (see survey in Cho and Manski, 2008). Our focus is on more general transformations from one aggregate unit to another. These involve not only disaggregation (as in EI), but also possible combinations of disaggregation and aggregation across non-nested units. As with EI and MAUP, no general solution to CoS problems exists. But we can identify conditions under which these problems might become more or less severe.

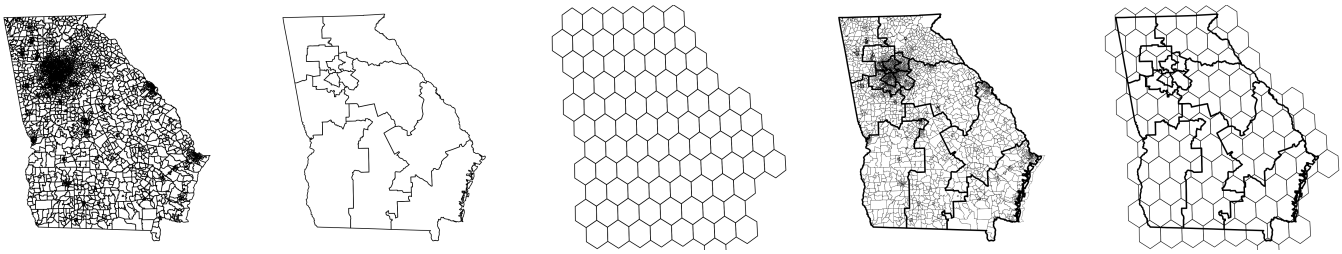
1.1 A General Framework for Changes-of-Support

In what follows, *destination unit* refers to the desired spatial unit given one's theory, and *source unit* refers to the unit at which data are available. We characterize two dimensions. The first, *relative nesting*, captures whether source units fall completely and neatly inside destination units. If perfectly nested, CoS problems become computationally simpler, and can sometimes be implemented without any geospatial transformations (e.g. aggregating

tables by common ID, like postal code). If units are not nested, CoS requires splitting polygons across multiple features, and reallocating or interpolating their values. Nesting is a geometric concept, not a political one: even units that are politically nested (e.g., counties within states) may appear non-nested when rendered as geospatial data features. Such discrepancies may be genuine (e.g., historical boundary changes), or driven by measurement error (e.g., simplified vs. detailed boundaries), or differences across sources.

The second dimension, *relative scale*, adds additional, useful information. It captures whether source units are generally smaller or larger than destination units. If smaller, transformed values will represent an aggregation of measurements taken at source units. If larger, transformed values will entail disaggregation — a more difficult process, posing nontrivial ecological inference challenges (King, 1997; Anselin and Cho, 2002). Many practical applications represent hybrid scenarios, where units are relatively smaller, larger, or of similar size to other units, depending on location. For instance, U.S. congressional districts in large cities are smaller than counties, but in rural areas they are larger than counties.

Figure 1: **Spatial data layers** (U.S. state of Georgia)



(a) Precincts (b) Constituencies (c) Grid cells (d) a ∩ b (e) b ∩ c

To illustrate, consider the degree of nesting and relative scale among three sets of polygons in Figure 1: (a) electoral precincts in the U.S. state of Georgia in 2014, (b) Georgia's electoral constituencies (congressional districts) in 2014, and (c) regular hexagonal grid cells (half-degree in diameter) covering the state of Georgia. Georgia is an interesting case study due to its diverse voting population and the judicial history surrounding its elections. Frequent legal challenges and court rulings have required that the state's data on electoral boundaries be accurate and publicly available at granular resolution (Bullock III, 2018).

We know *a priori* that precincts report to constituencies during the administration

of elections, so the units in Figure 1a should be fully nested within and smaller than those in 1b. The intersection in Figure 1d confirms that every precinct falls inside a larger constituency, and — with the exception of small border misalignments on the Atlantic coast — the transformation does not split precincts into multiple parts. If constituency IDs are available for precincts, a change of support from (a) to (b) could be reduced to calculating group sums, a straightforward procedure.

The hexagonal grid in (c) presents more difficulty. The grid cells were drawn independently of the other two layers, and are not nested — a transformation from (b) to (c) requires splitting many constituencies across multiple grid cells, and vice versa. The overlay in Figure 1e suggests that the cells are smaller than most, but not all, constituencies (e.g., Atlanta metro area). Changes of support to and from (c) therefore require complex spatial operations, accounting for differences in scale *and* misalignment of boundaries.

While visual assessments of nesting and scale can be informative, they introduce subjectivity and are often infeasible. Small geometric differences can be difficult to detect visually, and partial degrees of nesting are hard to characterize consistently. Visual inspections are also slow and not scalable for batch processing, which requires automated sub-routines.

We thus propose two simple, non-parametric measures of relative nesting and scale. Let $\mathcal{G}_S$ be a set of source polygons, indexed $i = 1, \ldots, N_S$, and $\mathcal{G}_D$ be a set of destination polygons, indexed $j = 1, \ldots, N_D$. Let $\mathcal{G}_{S \cap D}$ be the intersection of these polygons, indexed $i \cap j = 1, \ldots, N_{S \cap D} : N_{S \cap D} \geqslant \max(N_S, N_D)$. Let a_i be the area of source polygon i, a_j be the area of destination polygon j, and $a_{i \cap j}$ be the area of $i \cap j : a_{i \cap j} \leqslant \min(a_i, a_j)$.[3] Let $M_{S \cap D}$ be a $N_{S \cap D} \times 3$ matrix of indices mapping each intersection $i \cap j$ to its parent polygons i and j. $M_{i \cap D}$ is a subset of this matrix, indexing the $N_{i \cap D}$ intersections of polygon i (see Appendix A2 for examples). Let $1(\cdot)$ be a Boolean operator, equal to 1 if "$\cdot$" is true.

Our first measure — *relative nesting* (RN) — captures how closely source and destination boundaries align, and whether one set fits neatly into the other:

$$RN = \frac{1}{N_S} \sum_{i}^{N_S} \sum_{i \cap j}^{N_{i \cap D}} \left(\frac{a_{i \cap j}}{a_i} \right)^2 \tag{1}$$

[3] We recommend using an equal area map projection for all area calculations.

which reflects the share of source units that are not split across destination units. Values of 1 indicate full nesting (no source units are split across multiple destination units), and a theoretical lower limit of 0 indicates no nesting (every source unit is split across many destination units). RN has similarities with the Herfindahl-Hirschman Index (Hirschman, 1945) and the Gibbs-Martin index of diversification (Gibbs and Martin, 1962), which the electoral redistricting literature has used to assess whether "communities of interest" remain intact under alternative district maps (Chen, 2010). A value of $RN = 1$, for example, corresponds to a redistricting plan in which every source unit is assigned to exactly one destination unit (e.g., "Constraint 1" in Cho and Liu 2016).

The second measure — *relative scale* (RS) — captures whether a CoS task is generally one of aggregation or disaggregation:

$$RS = \frac{1}{N_{S \cap D}} \sum_{i \cap j}^{N_{S \cap D}} 1(a_i < a_j) \tag{2}$$

which is the share of intersections in which source units are smaller than destination units. Its range is from 0 to 1, where 1 indicates pure aggregation (all source units are smaller than intersecting destination units) and 0 indicates no aggregation (all source units are at least as large as destination units). Values between 0 and 1 indicate a hybrid (i.e. some source units are smaller, others are larger than destination units).

Table 1 reports pairwise RN and RS measures for the polygons in Figure 1, with source units in rows and destination units in columns. The table confirms that precincts are always smaller ($RS = 1$) and almost fully nested within constituencies ($RN = 0.98$), with the small difference likely due to measurement error. Precincts are also smaller than ($RS = 1$) and mostly nested within grid cells ($RN = 0.92$). At the opposite extreme, obtaining precinct-level estimates from constituencies or grid cells would entail disaggregation ($RS = 0$) into non-nested units ($0.01 \leqslant RN \leqslant 0.05$). Other pairings show intermediate values: hybrids of aggregation and disaggregation, where changes of support require splitting many polygons.

Before proceeding, let us briefly consider how RN and RS relate to each other, with more details in the appendix. Note, first, the two measures are not symmetric (i.e., $RS_{S,D} \neq 1 - RS_{D,S}$, $RN_{S,D} \neq 1 - RN_{D,S}$), except in special cases where the outer boundaries of

$\mathcal{G}_S$ and $\mathcal{G}_D$ perfectly align, with no "underlapping" areas as in Figure 1e. Appendix A2 considers extensions of these measures, including (by definition) symmetrical versions of RS and RN, conditional metrics defined for subsets of units, measures of spatial overlap, and metrics that require no area calculations at all.

Table 1: **Relative scale and nesting of polygons in Figure 1.**

(a) Relative scale (RS)

Source	Destination		
	(a)	(b)	(c)
(a) Precincts	–	1.00	1.00
(b) Constituencies	0.00	–	0.12
(c) Grid cells	0.00	0.89	–

(b) Relative nesting (RN)

Source	Destination		
	(a)	(b)	(c)
(a) Precincts	–	0.98	0.92
(b) Constituencies	0.01	–	0.29
(c) Grid cells grid	0.05	0.54	–

Second, as Table 1 suggests (and Appendix A2 shows in more detail), the two measures are positively correlated. Relatively smaller units tend to nest within larger units. Yet because RN is more sensitive to small differences in shape, area, and orientation, unlike RS, it rarely reaches its limits of 1 or 0. Even pure aggregation (e.g. precinct-to-grid, $RS = 1$) may involve integrating units that are technically not fully nested ($RN = 0.92$). One barrier to reaching $RN = 1$ is that polygon intersections often result in small-area "slivers" due to minor border misalignments, but these can be easily excluded from calculation.[4]

There are cases where (within a limited range) RN and RS diverge, as they do in the precinct-to-grid example in Table 1. As we show in Appendix A3 using randomly generated maps, such divergence reflects the fact that the distributions of RN and RS have different shapes: RS is bimodal, with peaks around $RS = 0$ and $RS = 1$, while RN is more normally distributed, with a mode around $RN = 0.5$. The relationship between the two measures resembles a logistic curve, where numerical differences are largest in the tails and smallest in the middle. The differences between the measures tend to be numerically small. We observe no cases, for instance, in which $RS > 0.5$ and $RN < 0.5$ (or vice versa) for the

[4]The `nesting()` function in our `R` package (see below) includes a tolerance parameter, governing the minimum area of intersections used to calculate RN.

same set of destination and source units.

2 How Nesting and Scale Affect Transformation Quality

How do the accuracy and bias of transformed values vary with relative nesting and scale? We evaluated the performance of several CoS algorithms in two applications: (1) transformations of electoral data across the polygons in Figure 1, and (2) a Monte Carlo study of CoS operations across randomly-generated synthetic polygons. A comprehensive review of CoS methods, their assumptions and comparative advantages, is beyond the scope of this paper (see summary in Appendix A4). Instead, we focus on how relative scale and nesting affect the reliability of spatial transformations in general, holding one's choice of CoS algorithm constant. Specifically, we compare transformed values in destination units to their "true" values, across multiple CoS operations.

Let K be a set of CoS algorithms. Each algorithm, indexed $k \in \{1, \ldots, K\}$, specifies a transformation $f_k(\cdot)$ between source units $\mathcal{G}_S$ and destination units $\mathcal{G}_D$. These transformations range from relatively simple operations that require no data beyond two sets of geometries, to more complex operations that incorporate information from covariates. Let $\mathbf{x}_{\mathcal{G}S}$ be a $N_S \times 1$ vector of observed values in source units $\mathcal{G}_S$, and $\mathbf{x}_{\mathcal{G}D}$ be the $N_D \times 1$ vector of "true" values in destination units $\mathcal{G}_D$. Let $\widehat{\mathbf{x}_{\mathcal{G}D}}^{(k)} = f_k(\mathbf{x}_{\mathcal{G}S})$ be a vector of estimated values for $\mathbf{x}_{\mathcal{G}D}$, calculated using CoS algorithm k. These transformed values are typically point estimates, although some methods also provide uncertainty measures.

Consider the following CoS algorithms:

- *Simple overlay.* This method requires no re-weighting or geostatistical modeling, and is standard for the aggregation of event data. For each destination polygon, it identifies source features that overlap with it (polygons) or fall within it (points, polygon centroids), and computes statistics (e.g. sum, mean) for those features. If a source polygon overlaps with multiple destination polygons, it is assigned to the destination unit with the largest areal overlap. Advantages: speed, ease of implementation. Disadvantages: generates missing values, particularly if $N_S \ll N_D$.

- *Area weighted interpolation.* This is a default CoS method in many commercial

and open-source GIS. It intersects source and destination polygons, calculates area weights for each intersection, and computes area-weighted statistics (i.e., weighted mean, sum). It can also handle point-to-polygon transformations through an intermediate tessellation step (Appendix A4). Advantages: no missing values, no ancillary data needed. Disadvantages: assumes uniform distribution in source polygons.

- *Population weighted interpolation.* This method extends area-weighting by utilizing ancillary data on population or any other covariate. It intersects the three layers (source, destination, population), assigns weights to each intersection, and computes population-weighted statistics. Advantages: softens the uniformity assumption. Disadvantages: performance depends on quality and relevance of ancillary data.[5]

- *TPRS-Forest.* This method uses a nonparametric function of geographic coordinates (thin-plate regression spline, TPRS) to estimate a spatial trend, capturing systematic variation or heterogeneity (Davidson, 2022*b*). It uses a Random Forest to model spatial noise, reflecting random, non-systematic variation. For each destination unit, it calculates a linear combination of trend and noise. Advantages: needs no ancillary data, provides estimates of uncertainty. Disadvantages: computationally costly.

- *TPRS-Area weights.* This is a hybrid of TPRS-Forest and areal interpolation. It decomposes source values into a non-stationary geographic trend using TPRS, and performs areal weighting on the spatial residuals from the smooth functional output (Davidson, 2022*a*). Advantages: provides estimates of uncertainty for area weighting; can optionally incorporate ancillary data. Disadvantages: computationally costly.

- *Ordinary (block) kriging.* This model-based approach is widely used as a solution to the CoS problem in the natural and environmental sciences (Gotway and Young, 2007). It uses a variogram model to specify the degree to which nearby locations have similar values, and interpolates values of a random field at unobserved locations (or blocks representing destination polygons) by using data from observed locations.

[5] A more general disadvantage of CoS methods reliant on ancillary data is that using a covariate $\mathbf{z}$ to impute values of $\mathbf{x}$ will bias (predetermine), by construction, any later estimate of association of $\hat{\mathbf{x}}$ with $\mathbf{z}$.

Advantages: provides estimates of uncertainty. Disadvantages: can generate overly-smooth estimates, sensitive to variogram model selection, assumes stationarity.

- *Universal (block) kriging.* This extends ordinary kriging by using ancillary information. It interpolates values of a random field at unobserved locations (or blocks), using data on the outcome of interest from observed locations *and* covariates (e.g., population) at both sets of locations. Advantages: relaxes stationarity assumption. Disadvantages: ancillary data may not adequately capture local spatial variation.

- *Rasterization.* This is a "naive" benchmark against which to compare other methods. It converts source polygons to raster (i.e. two-dimensional array of pixels), and summarizes the values of pixels that fall within each destination polygon. Advantages: no modeling, re-weighting or ancillary data. Disadvantages: assumes uniformity.

We employ two variants of the first three methods, using (a) polygon source geometries, and (b) points corresponding to source polygon centroids. These approaches represent different use cases: a "data-rich" scenario where full source geometries are available, and a "data-poor" scenario with a single address or coordinate pair. We also implement two variants of TPRS-Forest: (a) spatial trend only, and (b) spatial trend plus residuals.

We use three diagnostic measures: (1) root mean squared error, $\sqrt{\sum_j \frac{1}{N_D}(x_{j\mathcal{G}D} - \widehat{x_{j\mathcal{G}D}})^2}$, (2) Spearman's rank correlation for $\mathbf{x}_{\mathcal{G}D}$ and $\widehat{\mathbf{x}_{\mathcal{G}D}}$, and (3) estimation bias, $E[\widehat{\beta}_{(\widehat{\mathbf{x}_{\mathcal{G}D}})}] - \beta_{(\mathbf{x}_{\mathcal{G}D})}$, from an OLS regression of a synthetic variable $\mathbf{y}$ (see below) on transformed values $\hat{\mathbf{x}}$. The first two diagnostics capture how closely the numerical values of the transformed variable align with true values, and the direction of their association. The third captures the downstream impact of the CoS operation for model-based inference. For measures 1 and 3, values closer to zero are preferred. For measure 2, values closer to one are preferred.

2.1 Illustration: Changing the Geographic Support of Electoral Data

Our first illustration transforms electoral data across the polygons in Figure 1. We demonstrate generalizability through a parallel analysis of Swedish electoral data (Appendix A5).[6]

[6]We used data on Georgia's 2014 elections to the U.S. House of Representatives, and Sweden's 2010 elections to the Riksdag, due to the availability of high-precision boundary information and vote tallies.

The variable we transformed was *Top-2 Competitiveness*, scaled from 0 (least competitive) to 1 (most competitive):

$$\text{Top-2 Competitiveness} = 1 - \text{winning party vote share margin} \tag{3}$$

$$= \frac{\text{valid votes} - (\text{votes for winner} - \text{votes for runner-up})}{\text{valid votes}} \tag{4}$$

We obtained "true" values of competitiveness for precincts (Figure 1a) and constituencies (Figure 1b) from official election results, measuring party vote counts and votes received by all parties on the ballot (Kollman et al., 2022). For grid cells (Figure 1c), we constructed aggregates of valid votes and their party breakdown from precinct-level results.

Note that our analysis did not seek to transform Top-2 Competitiveness directly from source to destination units. Rather, we transformed the three constitutive variables in eq. 4 — valid votes, and votes for the top-2 finishers — and reconstructed the variable after the CoS. In Appendix A6 we compare our results against those from direct transformation.

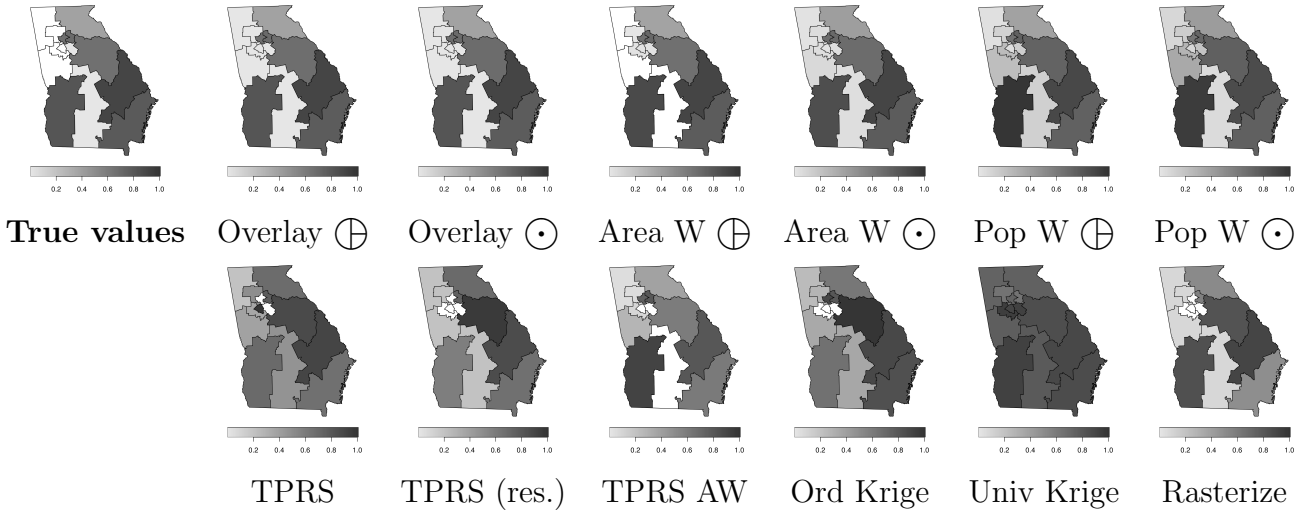
Because the purpose of much applied research is not univariate spatial transformation, but multivariate analysis (e.g., effect of $\mathbf{x}$ on $\mathbf{y}$), we created a synthetic variable, $y_i = \alpha + \beta x_i + \epsilon_i$, where x_i is Top-2 Competitiveness in unit i, $\alpha = 1$, $\beta = 2.5$, $\epsilon \sim N(0, 1)$. We assess how changes of support impact the estimation of regression coefficients, in a situation where the "true" value of that coefficient ($\beta = 2.5$) is known.

Figure 2 shows transformed values of Top-2 Competitiveness ($\widehat{\mathbf{x}_{\mathcal{G}D}}^{(k)}$) alongside true values in destination units ($\mathbf{x}_{\mathcal{G}D}$), where darker areas are more competitive. Figure 2a reports the results of precinct-to-constituency transformations (corresponding to "$a \cap b$" in Figure 1d). Figure 2b reports constituency-to-grid transformations ("$b \cap c$" in Figure 1e). Of these two, the first set of transformations (where $RN = 0.98$, $RS = 1$) more closely resembles true values than the second set ($RN = 0.29$, $RS = 0.12$), with fewer missing values and implausibly smooth or uniform predictions.

Figure 3 reports fit diagnostics for the full set of CoS transformations across spatial units in Georgia (vertical axes), as a function of the transformations' relative nesting and scale (horizontal axes). Each point corresponds the quality of fit for a separate CoS algorithm. The curves represent fitted values from a linear regression of each diagnostic on source-to-destination RN and RS coefficients. Grey regions are 95% confidence intervals.

Figure 2: **Output from change-of-support operations** (Georgia). ⨁: source features are polygons. ⊙: source features are polygon centroids.

(a) Precinct-to-constituency ($RS = 1, RN = 0.98$)



(b) Constituency-to-grid ($RS = 0.12, RN = 0.29$)

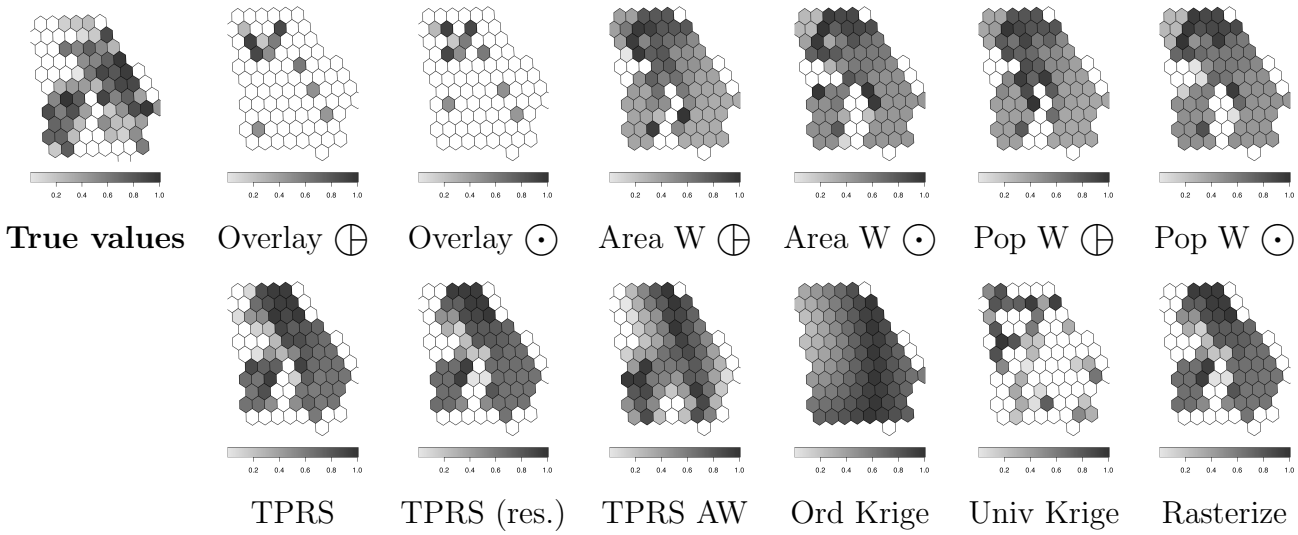
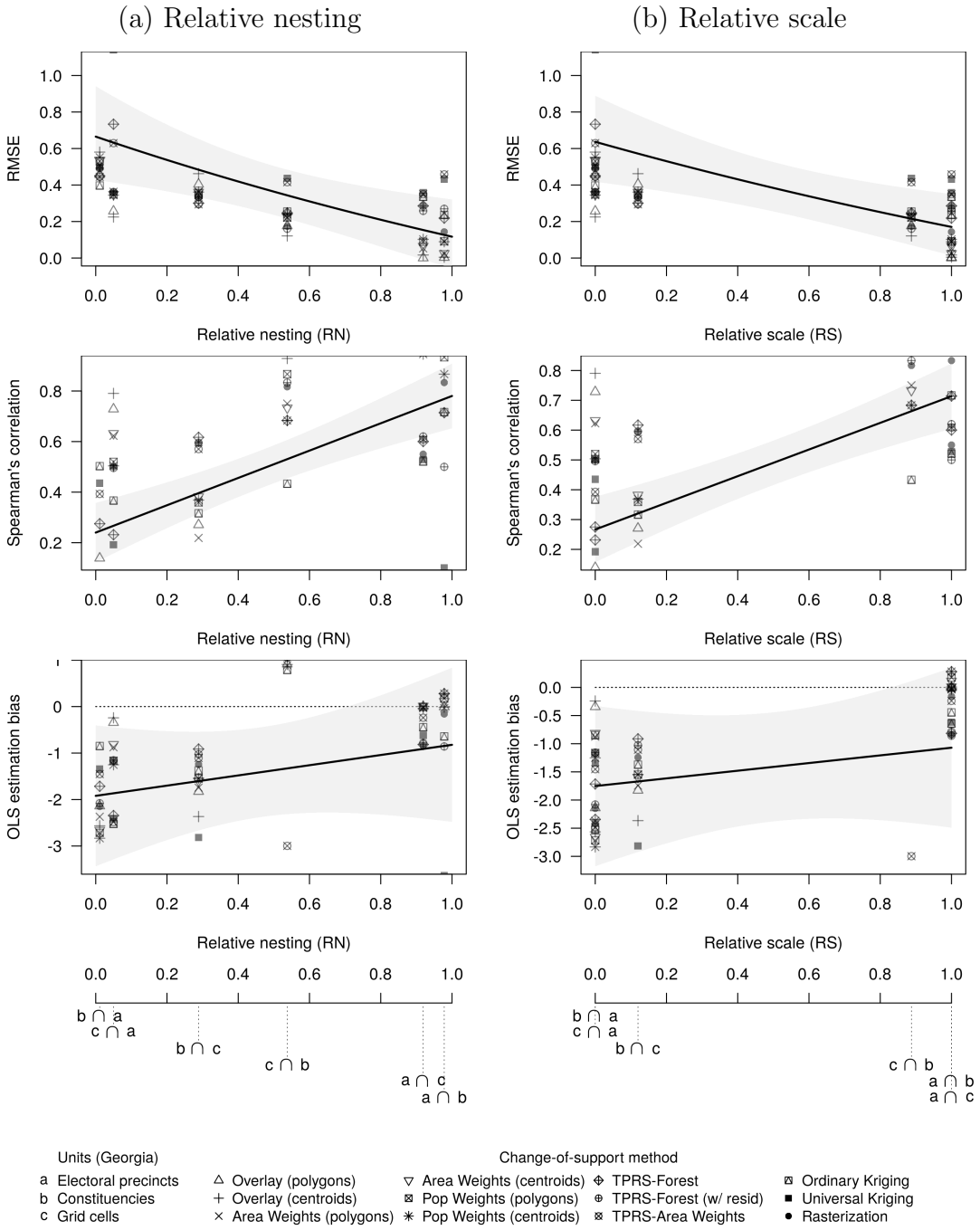


Figure 3: **Relative nesting, scale and transformations of election data** (Georgia)



These results confirm that the accuracy of CoS transformations is increasing in RN and RS. RMSE is lower, and correlation is higher, where source units are relatively smaller (3a) and more fully nested (3b). Results for OLS estimation bias are more uncertain, but there are fewer extreme values and a general convergence toward zero as RN and RS increase.

Our analysis also reveals substantial differences in the relative performance of CoS algorithms. Overall, simpler methods like overlays and areal interpolation produce more reliable results. For example, median RMSE — across all levels of RN and RS — was 0.22 or lower for both types of simple overlays, compared to 0.47 for universal kriging. Median correlation for simple overlays was 0.85 or higher, compared to 0.13 for universal kriging. Simple overlays also returned the smallest OLS bias, with a median of -0.13, compared to a quite severe underestimation of OLS coefficients for universal kriging (-2.4). Similar patterns emerge in our analysis of electoral data from Sweden (Appendix A5).

Because the Top-2 Competitiveness variable in our analysis is a function of several other variables, we considered how transformation quality changes when we transform this variable directly versus reconstructing it from transformed components. The comparative advantages of these approaches depend on the relative nesting and scale of source and destination units: indirect transformations perform better when RN and RS are closer to 1, direct transformations are preferable as RN and RS approach 0 (Appendix A6).

2.2 Illustration: Monte Carlo Study with Synthetic Polygons

To generalize, we performed Monte Carlo simulations with artificial boundaries and variables on a rectangular surface. This analysis compares fit diagnostics from the same CoS algorithms, over a broader set of transformations covering the full range of RN and RS.

We consider two use cases. First, we change the geographic support of an *extensive variable*, like population size or number of crimes. Extensive variables depend on the area and scale of spatial measurement: if areas are split or combined, their values must be split or combined accordingly, such that the sum of the values in destination units equals the total in source units (i.e. satisfying the pycnophylactic, or mass-preserving, property). Second, we change the support of an *intensive variable*, like temperature or elevation. Intensive variables do not depend on the size of spatial units; quantities of interest in destination

units are typically weighted means. Two or more extensive variables can combine to create a new intensive variable, like population density or electoral competitiveness. In Appendix A6, we consider the merits of (re-)constructing these variables before versus after a CoS.

While real-world data rarely conform to a particular known distribution, we designed the simulated geographic patterns to mimic the types of clustering and heterogeneity that are common in data on political violence and elections (see examples in Appendix A7).

At each iteration, our simulations executed these steps:

1. *Draw random source ($\mathcal{G}_S$) and destination ($\mathcal{G}_D$) polygons.* Within a rectangular bounding box $\mathcal{B}$, we sampled a random set of N_S points and created N_S tessellated polygons, such that for any polygon $l_i \in \{1, \ldots, N_S\}$ corresponding to point $i \in \{1, \ldots, N_S\}$, all points inside l_i were closer to i than to any other point $-i$. We did the same for N_D destination polygons. The relative number of source-to-destination polygons (N_S:N_D) ranged from 10:200 to 200:10, totalling 191^2 configurations. Figure 4a shows one realization of $\mathcal{G}_S, \mathcal{G}_D$, with $N_S = 200, N_D = 10$.

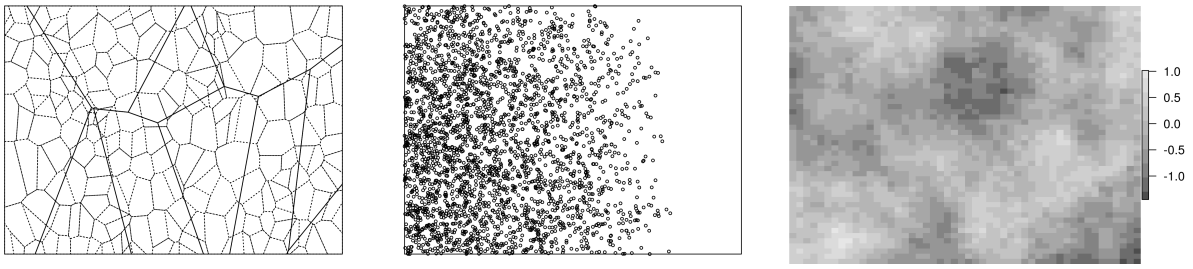
2. *Assign "true" values of a random variable X to units in $\mathcal{G}_S$ and $\mathcal{G}_D$.* For extensive variables, we simulated values from an inhomogeneous Poisson point process (PPP). Figure 4b and 4c illustrate examples of each. For intensive variables, we simulated values from a mean-zero Gaussian Random Field (GRF), implemented with the sequential simulation algorithm (Goovaerts, 1997).[7] In both cases, while we sought to imitate the spatially-autocorrelated distribution of real-world social data, we also simulated spatially-random values as benchmarks (see Appendix A7).[8] As before, we also created a synthetic variable $Y = \alpha + X\beta + \epsilon$, with $\alpha = 1$, $\beta = 2.5$, $\epsilon \sim N(0, 1)$.

3. *Change the geographic support of X from $\mathcal{G}_S$ to $\mathcal{G}_D$*, using the CoS algorithms listed above. We then compared transformed values $\widehat{x_{\mathcal{G}D}}$ to the assigned "true" values $x_{\mathcal{G}D}$, and calculated the same summary diagnostics as before, adding a Normalized RMSE $\left(\sqrt{\sum_j \frac{1}{N_{\mathcal{G}D}}(x_{j\mathcal{G}D} - \widehat{x_{j\mathcal{G}D}})^2}/(\max(x_{\mathcal{G}D}) - \min(x_{\mathcal{G}D}))\right)$ for extensive variables.

[7] We used the (rasterized) PPP and GRF as ancillary data for population-weights and universal kriging.

[8] We modeled autocorrelation in the PPP by allowing intensity to vary as a spherical function of spatial coordinates. We modeled autocorrelation in the GRF with a Matern covariance semi-variogram model.

Figure 4: **Examples of spatial data layers used in Monte Carlo study**. Dotted lines are source units ($\mathcal{G}_S$), solid lines are destination units ($\mathcal{G}_D$)



(a) Polygons (b) Poisson Point Process (c) Gaussian Random Field

We ran this simulation for all $N_S \in [10, \ldots, 200]$ and $N_D \in [10, \ldots, 200]$, totalling 191^2 potential CoS operations, from aggregation ($N_S = 200, N_D = 10$) to disaggregation ($N_S = 10, N_D = 200$). We repeated this process 100 times, with different random seeds.

To facilitate inferences about the relationship between nesting and the diagnostic measures, we estimated a series of semi-parametric regressions of the form:

$$M_{km} = f(RN_{km}) + \text{Method}_k + \epsilon_{km} \tag{5}$$

where k indexes CoS algorithms, and m indexes simulations. M_{km} is a diagnostic measure for CoS operation km (i.e. [N]RMSE, correlation, OLS bias), $f(RN_{km})$ is a cubic spline of RN (or RS), Method_k is a fixed effect for each CoS algorithm, and ϵ_{km} is an i.i.d. error term. This specification restricts our inferences to the effects of nesting and scale *within* groups of similar operations, adjusting for baseline differences in performance across algorithms.

Figure 5 reports predicted values of [N]RMSE, Spearman's correlation, and OLS bias across all methods at different levels of RN, for both extensive and intensive variables. Appendix A8 reports results for the RS coefficient, which generally align with these. As source units become more nested and relatively smaller, [N]RMSE and OLS bias trend toward 0, while correlation approaches 1. The primary difference between extensive and intensive variables is in the estimation of OLS coefficients. For extensive variables, we see attenuation bias, which becomes less severe as RN approaches 1. For intensive variables, we see attenuation bias as RN approaches 1, but inflation bias as RN approaches 0.

Figure 6 breaks the Monte Carlo results down by CoS algorithm. In each matrix, the first, left-most column reports average statistics for each algorithm, pooled over all values of RN. The remaining columns report average statistics in the bottom decile of RN (0% to 10%), the middle decile (45% to 55%), and the top decile (90% to 100%). The bottom row presents median statistics across all algorithms. In all cases, darker colors represent better-fitting transformations (i.e., closer to 0 for [N]RMSE and bias; closer to 1 for correlation).

We draw several conclusions. First, the relative performance of all CoS algorithms depends, strongly, on values of RN (and RS, see Appendix A8). In most cases, the largest improvements in performance occur between the lowest and middling ranges. Where values of RN are low (bottom 10%), most algorithms fare poorly. Performance improves as RN increases, especially in the first half of the range. For example, median RMSE (intensive variables) is 1.02 for CoS operations with low RN scores, 0.64 for intermediate values, and 0.36 for the top decile. In practical terms, this means that most algorithms will perform far better even at middling levels of RN — roughly, 0.4 and up — than at lower levels.

Second, while no CoS algorithm clearly stands out, some perform consistently worse than others. For example, population weighting offers no discernible advantages in transforming intensive variables (but seemingly plenty of disadvantages) relative to simple area weighting. Ancillary data from covariates, these results suggest, do not always improve transformation quality. Centroid-based simple overlays also fare quite poorly throughout.

Third, some CoS algorithms are more sensitive to variation in RN than others. For example, simple overlays produce credible results for extensive variables only when RN is high, while areal and population weighting results are more stable.

Are RN and RS redundant? After we condition on RN, for example, does RS add any explanatory value in characterizing the quality of CoS operations (i.e., measurement error and bias of transformed values)? As we show in Appendix A3, RN is more strongly predictive of transformation quality than RS when we consider the two metrics separately. But RS is certainly not redundant; including information about both RN and RS accounts for more variation in transformation quality than does information about RN alone.[9]

[9]Because the purpose of our analysis is not to maximize model fit, but to illuminate how each measure relates to transformation quality, we used the simpler specifications here.

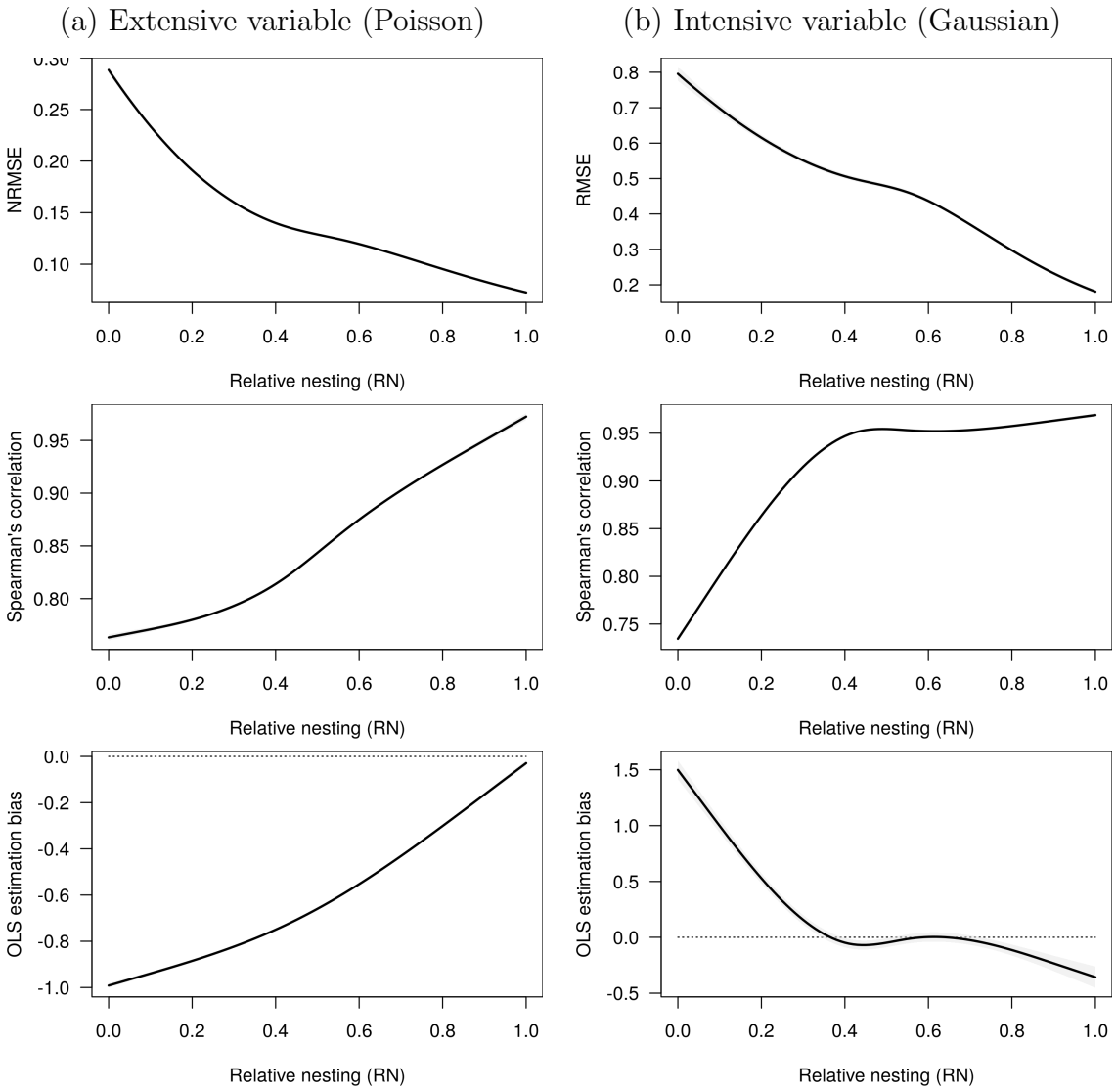
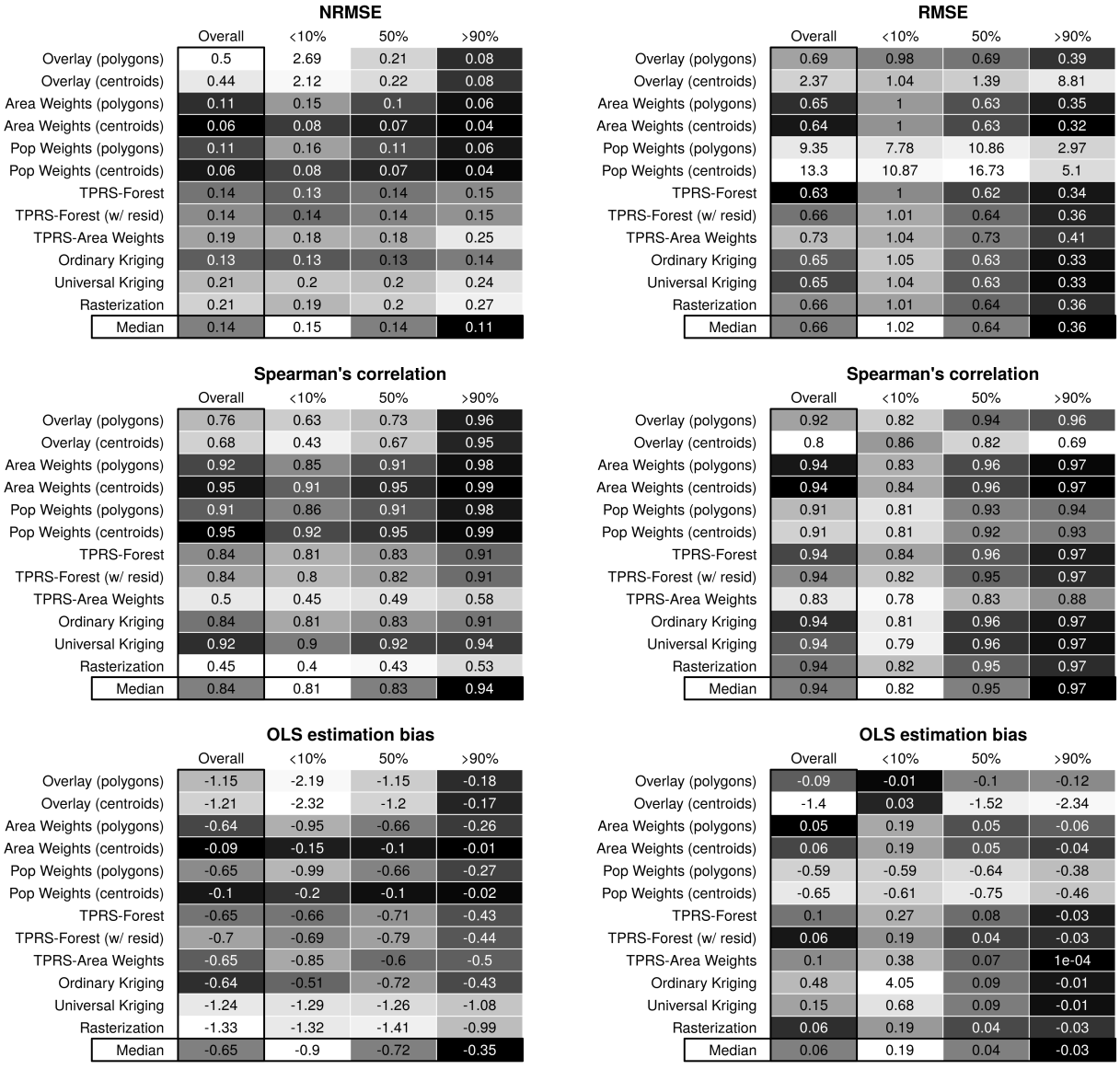


Figure 5: **Relative nesting and transformations of synthetic data**



(a) Extensive variable (Poisson) (b) Intensive variable (Gaussian)

Figure 6: **Transformation quality at different percentiles of relative nesting**

In Appendix A3, we also consider how divergence between RN and RS affects the relative performance of CoS methods. Not much, we conclude. The absolute proximity of RS and RN to 0 or 1 is far more predictive of transformation quality than their divergence.

Our simulations confirm that the patterns from our analysis of election data — higher RN and RS are better — hold in a more general set of cases. This includes changes of support between units of highly variable sizes and degrees of nesting, and transformations of variables with different properties and distributional assumptions.

3 What Is to Be Done?

Changes of support with medium to high relative nesting and scale tend to produce higher quality transformations, in terms of lower error rates, higher rank correlation, and lower OLS estimation bias. These patterns are consistent across CoS algorithms, in applications involving both extensive and intensive variables. While some CoS methods do perform better than others in specific contexts, no method stands out unconditionally dominating the rest. We were unable to find a "winner" in our comparison of a dozen algorithms, using data from elections and Monte Carlo simulations that mimic different geospatial patterns.

What does this mean for analysts performing CoS operations? At a minimum, we recommend reporting RN and RS coefficients for all CoS operations as *ex ante* measures of transformation complexity. This requires no data beyond the geometries of source and destination polygons, and enables readers to assess the risk of poor inference: the higher the numbers, the more reliable the potential results — and midrange numbers should give one far more confidence than lower numbers. Also, for face validity of transformed values, it is good practice to map the new distribution, and visually inspect it for strange discontinuities, missingness, "unnatural" smoothness or uniformity, and other obvious errors (as in Figure 2). And we urge researchers to implement sensitivity analyses with alternative CoS methods, to show that their results do not rest on the assumptions of particular algorithms.

Beyond this, additional advice depends on the availability of two types of "ground truth" data relevant to changes of support: atomic-level information on the distribution of the variable being transformed (e.g., precinct-level vote tallies, locations of individual events), and information on the proper assignment of units (e.g., cross-unit IDs).

If researchers have access to irreducibly lowest-level (ILL) data on variables of interest — in addition to their aggregate values in source units — we recommend using ILL data to validate spatial transformations directly. Examples of ILL data include precinct-level

votes, point coordinates of events, ultimate sampling units, individual-level (micro) data, and other information that cannot be disaggregated further. With such data, one can implement CoS methods as in the above analyses, and select the algorithm that yields the smallest errors and highest correlation with aggregates of true values in destination units. Alternatively, researchers may simply use the ILL data as their source features, since they are likely to have high RN and RS scores with destination units.

If IDs for destination polygons are available for source units *and* RN and RS are sufficiently high, then spatial transformations via CoS algorithms may be unnecessary. For each destination polygon, one needs only to identify the source features that share the common identifier (e.g., county name, state abbreviation), and compute group statistics for those features. To increase feasibility, however, the ID variable must exist, and provide a one-to-one or many-to-one mapping. A source feature cannot be assigned to more than one destination unit without additional assumptions about how its values are (re-)distributed.

What if there are no lower-level data for validation, and no common IDs for unit assignment? In this situation, our general advice — reporting RN and RS, visually inspecting the results, and performing sensitivity analyses — still holds. Yet the third of these steps has pitfalls. Since we cannot know which set of estimates is closest to true values, rerunning one's analysis with alternative CoS algorithms can create temptations either to be biased and choose the numbers one likes best, or to give equal weight to all algorithms, including some that could be wildly off the mark.

While cross-validation without ground truth data is a difficult topic that is beyond the scope of this article, we briefly illustrate one potential path in Appendix A9. Specifically, one can report the results of multiple CoS methods, along with a measure of how divergent each set of results is from the others, using outlier detection tests. As an analogy, this is like using multiple, imperfect instruments to detect the amount of oil underground. We may never know the true amount. But learning (for instance) that only one of the instruments has detected the presence of hydrocarbons is useful, both in the search for oil — that instrument might get it right — but also in evaluating the outlier instrument for future efforts. We caution that no set of results should be included or excluded from analysis solely on the basis of an outlier detection test. It is possible for an outlier to be more accurate than the average, and alternatively, an algorithm giving results close to the

average may be quite inaccurate. Yet if a CoS method frequently gives output that differs systematically from other CoS methods, further investigation may be warranted into the deviant algorithm. At the very least, this would help contextualize one's results.

To provide researchers with routines, documentation, and source code to implement these and other procedures with their own data, we developed an open-source software package (`SUNGEO`), available through the Comprehensive R Archive Network and GitHub. It includes functions to calculate RN, RS and related metrics, as well as functions to execute — and compare — most of the CoS methods discussed in this article. These tools should enable researchers to explore options, elucidate the consequences of choices, and design CoS strategies to meet their needs.

4 Conclusion

When integrating data across spatial units, seemingly benign measurement decisions can produce non-trivial consequences. The accuracy of spatial transformations depends on the relative nesting and scale of source and destination units. We introduced two simple, non-parametric measures that assess the extent to which the units are nested and the range from aggregation to disaggregation. We have shown that the two measures are predictive of the quality of spatial transformations, with higher values of RN and RS associated with lower error rates, higher correlation between estimated and true values, and less severe OLS estimation bias. These measures can serve as *ex ante* indicators of spatial transformation complexity and error-proneness, even in the absence of "ground truth" data for validation. We also provide open-source software to help researchers implement these procedures.

Because changes of support entail information loss, the consequences of these problems will depend in part on whether one uses spatially-transformed estimates for description or inference. Researchers have leeway when using transformed measures for mapping and visualization, so long as these transformed estimates correlate with the (unobserved) ground truth. The situations become more precarious when using interpolated measures for inference. Both Type II and Type I errors are possible. In the case of extensive variables, transformations with lower RN and RS scores generally result in the under-estimation of OLS coefficients, increasing the chances of false negatives. Yet there are also cases where

estimation bias is in the opposite direction (e.g., intensive variables with low RN, RS), increasing the chances of false positives. More research is needed on these situations.

We reiterate that CoS problems are ignored at the peril of accurate inference, and there is no silver bullet. To be as transparent as possible, researchers should document their measurement choices. This includes reporting relative nesting and scale, checking the face validity of the output, and avoiding reliance on a single CoS algorithm. We encourage future research to explore new methods for integrating spatially misaligned data.

Funding

Funding for this research was provided by an NSF RIDIR grant under award number SES-1925693.

Acknowledgements

We are grateful to the following for providing helpful comments: Wendy Tam Cho, Karsten Donnay, Rob Franzese, Jeff Gill, Max Goplerud, Andrew Linke, Walter Mebane, Kevin Quinn, Jonathan Rodden, Melissa Rogers and Yuki Shiraito.

Data availability statement

Replication data and code [will be] available at doi.org/XXXXXX. The R package is available at cran.r-project.org/package=SUNGEO and github.com/zhukovyuri/SUNGEO.

References

Anselin, Luc and Wendy K Tam Cho. 2002. "Spatial effects and ecological inference." *Political analysis* 10(3):276–297.

Benmelech, Efraim, Claude Berrebi and Esteban F. Klor. 2015. "Counter-Suicide-Terrorism: Evidence from House Demolitions." *Journal of Politics* 77(1):27–43.

Branton, Regina, Valerie Martinez-Ebers, Tony E Carey and Tetsuya Matsubayashi. 2015. "Social protest and policy attitudes: The case of the 2006 immigrant rallies." *American Journal of Political Science* 59(2):390–402.

Bullock III, Charles S. 2018. "The History of Redistricting in Georgia." *Georgia Law Review* 52(4):1057–1104.

Chen, Jowei. 2010. "The effect of electoral geography on pork barreling in bicameral legislatures." *American Journal of Political Science* 54(2):301–322.

Cho, Wendy K Tam and Charles F Manski. 2008. "Cross level/ecological inference." *Oxford handbook of political methodology* pp. 547–569.

Cho, Wendy K Tam and Yan Y Liu. 2016. "Toward a talismanic redistricting tool: A computational method for identifying extreme redistricting plans." *Election Law Journal* 15(4):351–366.

Comber, Alexis and Wen Zeng. 2019. "Spatial interpolation using areal features: A review of methods and opportunities using new forms of data with coded illustrations." *Geography Compass* 13(10):e12465.

Cook, Scott J and Nils B Weidmann. 2022. "Race to the Bottom: Spatial Aggregation and Event Data." *International Interactions* pp. 1–21.

Cressie, Noel. 1996. "Change of support and the modifiable areal unit problem." *Geographical Systems* 3(2-3):159–180.

Darmofal, David and Christopher Eddy. 2020. *Spatial data*. Vol. 2 SAGE Publications Ltd. pp. 424–435. DOI: 10.4135/9781526486387.

Davidson, Marty. 2022*a*. "TPRS-Areal Weighting for Administrative Boundaries." Working paper.

Davidson, Marty. 2022*b*. "TPRS-Forest Estimation for Spatial Data." Working paper.

Donnay, Karsten and Andrew Linke. 2016. Geomerge: Spatial Data Integration. Technical report Working paper.

Eckert, Fabian, Andrés Gvirtz, Jack Liang and Michael Peters. 2020. A Method to Construct Geographical Crosswalks with an Application to US Counties since 1790. Technical report National Bureau of Economic Research.

Gibbs, Jack P and Walter T Martin. 1962. "Urbanization, technology, and the division of labor: International patterns." *American sociological review* pp. 667–677.

Goovaerts, Pierre. 1997. *Geostatistics for natural resources evaluation*. Oxford University Press on Demand.

Goplerud, Max. 2016. "Crossing the Boundaries: An Implementation of Two Methods for Projecting Data across Boundary Changes." *Political Analysis* 24(1):121–129.

Gotway, Carol A and Linda J Young. 2002. "Combining incompatible spatial data." *Journal of the American Statistical Association* 97(458):632–648.

Gotway, Carol A and Linda J Young. 2007. "A geostatistical approach to linking geographically aggregated data from different sources." *Journal of Computational and Graphical Statistics* 16(1):115–135.

Hirschman, Albert O. 1945. *National power and the structure of foreign trade*. Vol. 105 Univ of California Press.

King, Gary. 1997. *A solution to the ecological inference problem: Reconstructing individual behavior from aggregate data*. Princeton University Press.

Kollman, Ken, Allen Hicken, Daniele Caramani, David Backer and David Lublin. 2022. *Constituency-Level Elections Archive*. Ann Arbor, MI: Center for Political Studies, University of Michigan.

Lee, Dong Wook and Melissa Rogers. 2019. "Measuring geographic distribution for political research." *Political Analysis* 27(3):263–280.

Matheron, Georges. 1989. The internal consistency of models in geostatistics. In *Geostatistics*. Springer pp. 21–38.

Morgenstern, Hal. 1982. "Uses of ecologic analysis in epidemiologic research." *American journal of public health* 72(12):1336–1344.

Openshaw, Stan and Peter J. Taylor. 1979. "A million or so correlation coefficients, three experiments on the modifiable areal unit problem." *Statistical applications in the spatial science* (21):127–144.

Robinson, William S. 1950. "Ecological correlations and the behavior of individuals." *American Sociological Review* 15(3):351–357.

Rozenas, Arturas and Yuri Zhukov. 2019. "Mass Repression and Political Loyalty: Evidence from Stalin's 'Terror by Hunger'." *American Political Science Review* 113(2):569–583.

Zhu, Li, Lance A Waller and Juan Ma. 2013. "Spatial-temporal disease mapping of illicit drug abuse or dependence in the presence of misaligned ZIP codes." *GeoJournal* 78(3):463–474.

Online Appendix

Contents

A1. Review of empirical political science literature

As we mention in the main text, we conducted a review of subnational empirical research published in top political science journals since 2010. We utilized a keyword search of the *Web of Science* search engine to find a total of 100 peer-reviewed articles that utilized some form of subnational/geospatial empirical research. We make no claims that this subset of articles contains the entire population of subnational/geospatial empirical articles within political science, but we are confident that this provides a representative sample of the extant literature within the field. Below we enumerate the articles in our sample, and note which ones include steps taken by researchers to change the support of key variables.

Author & Year	DOI	Journal	CoS by Authors
Gehring (2021)	10.1017/S0003055418000709	APSR	
Harris & Posner (2019)	10.1017/S0003055418000709	APSR	✓
Hankinson (2018)	10.1017/S0003055418000035	APSR	
Tajima et al. (2018)	10.1017/S0003055418000138	APSR	
Braun (2016)	10.1017/S0003055415000544	APSR	
Cederman et al. (2011)	10.1017/S0003055411000207	APSR	✓
Bohmelt et al. (2020)	10.1111/ajps.12494	AJPS	
Nall et al. (2018)	10.1111/ajps.12305	AJPS	
Knutsen et al. (2017)	10.1111/ajps.12268	AJPS	
Monogan et al. (2017)	10.1111/ajps.12278	AJPS	
Stokes (2016)	10.1111/ajps.12220	AJPS	✓
Nyhan & Montgomery (2015)	10.1111/ajps.12143	AJPS	

Williams & Whitten (2015)	10.1111/ajps.12124	AJPS	
Branton et al. (2015)	10.1111/ajps.12159	AJPS	✓
Wallace et al. (2014)	10.1111/ajps.12060	AJPS	
Bhavani et al. (2014)	10.1111/ajps.12045	AJPS	✓
Mukherjee & Singer (2010)	10.1111/j.1540-5907.2009.00417.x	AJPS	
Cho & Gimpel (2010)	10.1111/j.1540-5907.2009.00419.x	AJPS	
Wimpy et al. (2021)	10.1086/710089	JOP	
Montgomery & Nyhan (2017)	10.1086/690301	JOP	
Bove & Bohmelt (2016)	10.1086/684679	JOP	
Nall (2015)	10.1086/679597	JOP	✓
Benmelech et al. (2015)	10.1086/678765	JOP	✓
Boehmke et al. (2012)	10.1017/S0022381612000321	JOP	✓
Bell et al. (2012)	10.1017/S0022381611001642	JOP	
Weidmann (2011)	10.1017/S0022381611000831	JOP	✓
Clemens et al. (2015)	10.1111/lsq.12067	LSQ	
Cortina (2020)	10.1177/1065912919854135	PRQ	
Briggs (2019)	10.1177/1065912918798489	PRQ	
Croicu & Kreutz (2017)	10.1177/1065912916670272	PRQ	
Minkoff & Lyonos (2019)	10.1177/1532673X17733799	APR	✓
Smith & Weinberg (2016)	10.1177/1532673X15602755	APR	
Gill (2021)	10.1177/1532440020930197	SPPQ	
Darmofal et al. (2019)	10.1177/1532440019851806	SPPQ	
Pacheco (2017)	10.1177/1532440017705150	SPPQ	
Parinandi (2013)	10.1177/1532440013484477	SPPQ	
Boehmke & Skinner (2012)	10.1177/1532440012438890	SPPQ	
Carson et al. (2012)	10.1177/1532440012438892	SPPQ	
Gilardi & Wasserfallen (2014)	10.1017/S0007123414000246	BJPS	
Bell et al. (2013)	10.1017/S0007123413000100	BJPS	
Gibler & Braithwaite (2013)	10.1017/S000712341200052X	BJPS	
Gatesman & Unwin (2021)	10.1017/pan.2020.22	PA	
Juhl (2021)	10.1017/pan.2020.23	PA	
Betz et al. (2021)	10.1017/pan.2020.26	PA	
Saxon (2020)	10.1017/pan.2019.45	PA	✓
Vande Kamp (2020)	10.1017/pan.2019.35	PA	
Juhl (2020)	10.1017/pan.2019.12	PA	
Betz et al. (2018)	10.1017/pan.2018.10	PA	

Harbers & Ingram (2017)	10.1017/pan.2017.4	PA	
Goplerud (2016)	10.1093/pan/mpv029	PA	✓
Franzese et al. (2012)	10.1093/pan/mpr049	PA	
Steinwand (2011)	10.1093/pan/mpr026	PA	
Abramson & Carter (2021)	10.1017/S0020818320000545	IO	
Christensen (2019)	10.1017/S0020818318000413	IO	
Sommerer & Tallberg (2019)	10.1017/S0020818318000450	IO	
Cunningham & Sawyer (2017)	10.1017/S0020818317000200	IO	
Branch (2016)	10.1017/S0020818316000199	IO	
Steinwand (2015)	10.1017/S0020818314000381	IO	
Chaudoin et al. (2015)	10.1017/S0020818314000356	IO	
Neumayer et al. (2014)	10.1017/S0020818313000362	IO	
Neumayer & Pluemper (2010)	10.1017/S0020818309990191	IO	
Brazys & Kotsdam (2020)	10.1093/isq/sqaa072	ISQ	
Jones & Zeitz (2019)	10.1093/isq/sqz068	ISQ	
Reeder (2018)	10.1093/isq/sqy016	ISQ	✓
Bohmelt et al. (2017)	10.1093/isq/sqx067	ISQ	
Zhukov & Stewart (2013)	10.1111/isqu.12008	ISQ	
Barthel & Neumayer (2012)	10.1111/j.1468-2478.2012.00757.x	ISQ	
Cao (2010)	10.1111/j.1468-2478.2010.00611.x	ISQ	
Kosec & Mogues (2020)	10.1017/S0043887120000027	WP	
Wilfahrt (2018)	10.1017/S0043887117000363	WP	✓
Baccini et al. (2014)	10.1017/S0043887114000124	WP	
Obinger & Schmitt (2011)	10.1017/S0043887111000025	WP	
Cammett & Issar (2010)	10.1017/S0043887110000080	WP	✓
Schvitz et al. (2021)	10.1177/00220027211013563	JCR	
Echevarria-Coco et al. (2021)	10.1177/0022002720958470	JCR	
Polo (2020)	10.1177/0022002720930811	JCR	
Ito & Elliot (2020)	10.1177/0022002719885428	JCR	
Koren (2019)	10.1177/0022002719833160	JCR	✓
Moro & Sberna (2018)	10.1177/0022002717693049	JCR	
Bohnet et al. (2018)	10.1177/0022002716665209	JCR	
Miller et al. (2018)	10.1177/0022002716649232	JCR	
Minhas & Radford (2017)	10.1177/0022002716639100	JCR	
Schultz (2017)	10.1177/0022002715620470	JCR	
Osorio (2015)	10.1177/0022002715587048	JCR	

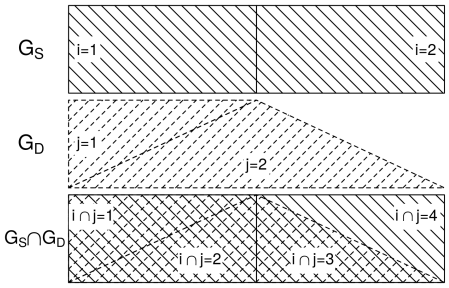
Schutte (2015)	10.1177/0022002713520534	JCR	
Baccini & Duer (2015)	10.1177/0022002713516844	JCR	
Fjelde & Hultman (2014)	10.1177/0022002713492648	JCR	✓
McDoom (2014)	10.1177/0022002713484282	JCR	
Althaus et al. (2012)	10.1177/0022002711422340	JCR	
Buhaug et al. (2011)	10.1177/0022002711408011	JCR	✓
Weidmann & Ward (2010)	10.1177/0022002710371669	JCR	
Chang & Manion (2021)	10.1177/0010414021989762	CPS	
Spater & Tranvik (2019)	10.1177/0010414019830721	CPS	
Lechler & McNamee (2018)	10.1177/0010414018758760	CPS	
Ejdemyr et al. (2018)	10.1177/0010414017730079	CPS	
De Juan (2017)	10.1177/0010414016688006	CPS	✓
Maehler & Pierskalla (2015)	10.1177/0010414014545012	CPS	✓
Ward & Cao (2012)	10.1177/0010414011434007	CPS	
Neumayer & Pluemper (2012)	10.1177/0010414011429066	CPS	
Elkink (2011)	10.1177/0010414011407474	CPS	

A2. Scale and nesting metrics

This section provide details on the nesting and scale metrics used in the main text, as well as several alternative metrics included in the `SUNGEO` R package (`nesting()` function).

Let $\mathcal{G}_S$ be a set of source polygons, indexed $i = 1, \ldots, N_S$, and $\mathcal{G}_D$ be a set of destination polygons, indexed $j = 1, \ldots, N_D$. Let $\mathcal{G}_{S \cap D}$ be the intersection of polygons 1 and 2, indexed $i \cap j = 1, \ldots, N_{S \cap D} : N_{S \cap D} \geqslant \max(N_S, N_D)$. Let a_i be the area of source polygon i, and a_j be the area of destination polygon j. Let $a_{i \cap j}$ be the area of $i \cap j : a_{i \cap j} \leqslant \min(a_i, a_j)$. Let $1(\cdot)$ be a Boolean operator, equal to 1 if statement "$\cdot$" is true, and 0 otherwise.

Each intersection $i \cap j$ can be mapped to its parent polygons i and j, using a $N_{S \cap D} \times 3$ matrix of indices $M_{S \cap D}$. For illustrative purposes, consider the stylized example below:



$$M_{S \cap D} = \begin{array}{|c|c|c|} \hline i \cap j & i & j \\ \hline 1 & 1 & 1 \\ 2 & 1 & 2 \\ 3 & 2 & 2 \\ 4 & 2 & \varnothing \\ \hline \end{array} \quad \text{where } N_{S \cap D} = 4$$

$M_{i \cap D}$ is the subset of $M_{S \cap D}$ indexing the $N_{i \cap D}$ intersections of polygon i. For $i = 1$:

$$M_{1 \cap D} = \begin{array}{|c|c|c|} \hline i \cap j & i & j \\ \hline 1 & 1 & 1 \\ 2 & 1 & 2 \\ \hline \end{array} \quad \text{where } N_{1 \cap D} = 2$$

Similarly, $M_{S \cap j}$ is the subset corresponding to destination polygon j. For $j = 2$:

$$M_{S \cap 2} = \begin{array}{|c|c|c|} \hline i \cap j & i & j \\ \hline 2 & 1 & 2 \\ 3 & 2 & 2 \\ \hline \end{array} \quad \text{where } N_{S \cap 2} = 2$$

If a polygon (or a part of a polygon) from $\mathcal{G}_S$ does not overlap with any features from $\mathcal{G}_D$, the corresponding row in $M_{S \cap D}$ will have an empty value for j (and vice versa):

$$M_{2 \cap D} = \begin{array}{|c|c|c|} \hline i \cap j & i & j \\ \hline 3 & 2 & 2 \\ 4 & 2 & \varnothing \\ \hline \end{array}$$

where $i \cap j = 4$ corresponds to a part of source polygon $i = 2$ that does not overlap with any polygons from $\mathcal{G}_D$. Note that this intersection is not empty; it just has a "single parent" and cannot be mapped to any destination features j.

We can now define our nesting and scale metrics.

- *Relative nesting* (RN). Captures how closely source and destination boundaries align:

$$RN = \frac{1}{N_S} \sum_{i}^{N_S} \sum_{i \cap j}^{N_{i \cap D}} \left(\frac{a_{i \cap j}}{a_i} \right)^2 \tag{A2.1}$$

  which is the share of source units that cannot be split across multiple destination units. Values of 0 indicate no nesting (every source unit can be split across multiple destination units) and values of 1 indicate full nesting (no source unit can be split across multiple destination units).

- *Relative scale* (RS). Captures whether a task is one of aggregation or disaggregation:

$$RS = \frac{1}{N_{S \cap D}} \sum_{i \cap j}^{N_{S \cap D}} 1(a_i < a_j) \tag{A2.2}$$

which is the share of source units that are smaller than destination units. Its range is from 0 to 1, where values of 1 indicate pure aggregation (all source units are smaller than destination units) and values of 0 indicate no aggregation (all source units are at least as large as destination units). Values between 0 and 1 indicate a hybrid (i.e. some source units are smaller, others are larger than target units).

- *Relative nesting, symmetric* (RN-sym). Alternative measure of RN, ranges from -1 to 1:

$$RN\text{-}sym = \frac{1}{N_S}\sum_{i}^{N_S}\sum_{i\cap j}^{N_{i\cap D}}\left(\frac{a_{i\cap j}}{a_i}\right)^2 - \frac{1}{N_D}\sum_{j}^{N_D}\sum_{i\cap j}^{N_{S\cap j}}\left(\frac{a_{i\cap j}}{a_j}\right)^2 \tag{A2.3}$$

  which is the difference between the nesting of source units within destination units, $1/N_S\sum_i^{N_S}\sum_{i\cap j}^{N_{i\cap D}}\left(a_{i\cap j}/a_i\right)^2$ (i.e. RN from standpoint of $\mathcal{G}_S$), and the nesting of destination units within source units, $1/N_D\sum_j^{N_D}\sum_{i\cap j}^{N_{S\cap j}}\left(a_{i\cap j}/a_j\right)^2$ (RN from standpoint of $\mathcal{G}_D$). Values of 1 indicate that source units are perfectly nested within destination units; -1 indicates that destination units are perfectly nested within source units.

- *Relative scale, symmetric* (RS-sym). Alternative measure of RS, ranges from -1 to 1:

$$RS\text{-}sym = \frac{1}{N_{S\cap D}}\sum_{i\cap j}^{N_{S\cap D}} 1(a_i < a_j) - 1(a_i > a_j) \tag{A2.4}$$

  which is a difference between two proportions: $1/N_{S\cap D}\sum 1\left(a_i < a_j\right)$, or the share of source units that is smaller than destination units (i.e. RS from standpoint of $\mathcal{G}_S$), and $1/N_{S\cap D}\sum 1\left(a_i > a_j\right)$, the share that is larger (i.e. RS from standpoint of $\mathcal{G}_D$). Its range is from -1 (pure disaggregation, all source units are larger than target units) to 1 (pure aggregation, all source units are smaller than target units). Values of 0 indicate that all source units are the same size as target units.

- *Relative nesting, conditional* (RN-nn). RN for source units that are not fully nested:

$$RN^{(\text{nn})} = \frac{1}{N_{S^\star}}\sum_{i}^{N_{S^\star}}\sum_{i\cap j}^{N_{i\cap D}}\left(\frac{a_{i\cap j}}{a_i}\right)^2 \tag{A2.5}$$

  where $S^\star$ denotes the set of source units with $\frac{1}{N_{i\cap D}}\sum_{i\cap j}^{N_{i\cap D}}\frac{a_{i\cap j}}{a_i} < 1$.

- *Relative scale, conditional* (RS-nn). RS for source units that are not fully nested:

$$RS^{(\text{nn})} = \frac{1}{N_{S^\star \cap D}} \sum_{i \cap j}^{N_{S^\star \cap D}} 1(a_i < a_j) \tag{A2.6}$$

where $S^\star$ denotes the set of source units with $\frac{1}{N_{i \cap D}} \sum_{i \cap j}^{N_{i \cap D}} \frac{a_{i \cap j}}{a_i} < 1$.

- *Proportion intact* (PI). A nesting metric that requires no area calculations at all:

$$PI = \frac{1}{N_S} \sum_{i}^{N_S} 1 \left(N_{i \cap D} - \sum_{i \cap j}^{N_{i \cap D}} 1 \left(\mathcal{G}_{i \cap j} = \varnothing \right) = 1 \right) \tag{A2.7}$$

This measure ranges from 0 to 1, where 1 indicates full nesting (i.e. every source unit is intact/no splits), and 0 indicates no nesting (i.e. no source unit is intact/all are split).

- *Proportion fully nested* (PFN). A stricter version of PI, which also requires that source units are fully contained within destination units (in PI, source units outside the boundaries of the destination layer are considered "intact"; in PFN, they are not).

$$PFN = \frac{1}{N_S} \sum_{i}^{N_S} 1 \left(\frac{1}{N_{i \cap D}} \sum_{i \cap j}^{N_{i \cap D}} \frac{a_{i \cap j}}{a_i} = 1 \right) \tag{A2.8}$$

This measure ranges from 0 to 1, where 1 indicates full nesting (i.e. every source unit is intact AND is fully contained within a single destination unit), and 0 indicates no nesting (i.e. no source unit is intact OR none are contained within destination units).

- *Relative overlap* (RO). Assesses extent of spatial overlap between source and destination polygons. Let α_S be the combined area of all source polygons. Let α_D be the combined area of all destination polygons. Let $\alpha_{S(-D)}$ be the combined area of all source polygons, excluding the area covered by destination polygons. Let $\alpha_{D(-S)}$ be the combined area of all destination polygons, excluding the area covered by source polygons.

$$RO = \frac{\alpha_{S(-D)}}{\alpha_S} - \frac{\alpha_{D(-S)}}{\alpha_D} \tag{A2.9}$$

this measure is scaled between -1 and 1. Values of 0 indicate perfect overlap (there is no part of source units that fall outside of destination units, and vice versa). Values between 0 and 1 indicate a source "underlap" (some parts of source polygons fall outside of destination polygons; more precisely, a larger part of source polygon area falls outside destination polygons than the other way around). Values between -1 and 0

indicate a destination "underlap" (some parts of destination polygons fall outside of source polygons; a larger part of destination polygon area falls outside source polygons than the other way around). Values of -1 and 1 indicate no overlap (all source units fall outside destination units, and vice versa). This is a theoretical limit only; in the R package, the function returns an error if there is no overlap.

Table A2.2 reports the pairwise correlations between these metrics, for the Monte Carlo simulations described in the main text. As the table suggests, the correlations are generally strongly positive, but not always perfect — especially in the case of PI and RO, which are capturing conceptually different properties of changes of support.

Metric	RN	RS	RN-sym	RS-sym	RN-nn	RS-nn	PI	PFN	RO
RN	1.00	0.97	1.00	0.97	1.00	0.97	0.81	0.73	0.40
RS	0.97	1.00	0.97	1.00	0.97	0.99	0.75	0.66	0.34
RN-sym	1.00	0.97	1.00	0.97	1.00	0.97	0.82	0.73	0.40
RS-sym	0.97	1.00	0.97	1.00	0.97	0.99	0.75	0.66	0.34
RN-nn	1.00	0.97	1.00	0.97	1.00	0.97	0.80	0.72	0.40
RS-nn	0.97	0.99	0.97	0.99	0.97	1.00	0.69	0.61	0.34
PI	0.81	0.75	0.82	0.75	0.80	0.69	1.00	0.93	0.37
PFN	0.73	0.66	0.73	0.66	0.72	0.61	0.93	1.00	0.35
RO	0.40	0.34	0.40	0.34	0.40	0.34	0.37	0.35	1.00

Table A2.2: Correlation between alternative nesting metrics.

A3. The relationship between RN and RS

The strong correlations reported in Table A2.2 raise several important questions about how these metrics relate to each other. First, are some of these metrics more strongly predictive of transformation quality than others? Second, are these metrics redundant? After we condition on RN, for example, does RS add any explanatory value in accounting for variation in transformation quality? Third, how frequently do these metrics diverge, and what are the implications of such divergence for analysis?

Our Monte Carlo simulations confirm that some nesting metrics — particularly RN and its variants — have particularly strong explanatory power as predictors of transformation quality. Table A3.3 reports a "horse race" evaluation of the nesting metrics' ability to explain transformation quality, as measured by RMSE, Spearman's correlation and OLS estimation bias. Specifically, we replicated the semi-parametric regressions in equation (5), each time with a different nesting metric on the right-hand side, and compared goodness-of-fit diagnostics across these specifications. Across all three fit diagnostics —

Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC) and the sum of squared residuals (Deviance) — RN consistently outperforms RS. Moreover, some variants of RN (e.g. RN-sym, RN-nn) offer marginal performance gains over the original.

Diagnostic	Metric	AIC	BIC	Deviance
log(RMSE)	RN-sym	1054353.12	1054526.11	380590.73
	RN-nn	1054655.82	1054828.82	380905.01
	RN	1054664.63	1054837.63	380914.17
	RS	1055875.27	1056048.27	382173.74
	RS-sym	1055875.27	1056048.27	382173.74
	PI	1056026.38	1056199.38	382331.25
	RS-nn	1056188.93	1056361.93	382500.76
	PFN	1058636.84	1058799.02	385064.64
	RO	1066254.95	1066427.95	393145.37
Spearman's correlation	RN-sym	-919652.25	-919479.26	1748.54
	RN-nn	-918183.71	-918010.71	1755.55
	RN	-918126.76	-917953.77	1755.83
	RS-nn	-901759.84	-901586.84	1835.97
	RS	-896572.42	-896399.42	1862.12
	RS-sym	-896572.42	-896399.42	1862.12
	PI	-871872.61	-871699.61	1991.87
	RO	-852239.56	-852066.56	2101.41
	PFN	-843389.86	-843227.67	2152.75
OLS estimation bias	RN-nn	1949057.83	1949230.82	4365847.48
	RN	1949062.99	1949235.98	4365908.90
	RN-sym	1949114.91	1949287.91	4366527.12
	RO	1949711.79	1949884.79	4373640.09
	RS-nn	1949718.72	1949891.72	4373722.76
	RS	1949852.81	1950025.81	4375322.37
	RS-sym	1949852.81	1950025.81	4375322.37
	PI	1950136.94	1950309.94	4378713.70
	PFN	1950556.52	1950718.71	4383750.44

Table A3.3: Relative performance of nesting metrics in explaining transformation quality.

If RN generally "outperforms" RS as a predictor of transformation quality, then why should we bother with RS at all? Is there any added value in calculating and reporting RS scores, once we account for RN? We compared the performance of several nested models, including:

1. RN: a baseline specification with just RN in the spline function, as in equation (5).

2. $RN + RS$: an expanded, additive specification with separate splines for RN and RS.

3. $RN \times RS$: an expanded, interactive specification with separate splines for RN and RS, and a multiplicative interaction between the two splines.

To assess whether including RS in these specifications improves model fit, we performed a series of Likelihood Ratio Tests, reported in Tables A3.4-A3.5. The null hypothesis in all cases is that the more parsimonious model (e.g. RN only) fits the data just as well as the expanded model (e.g. $RN + RS$). The alternative hypothesis is that the expanded model fits the data significantly better than the restricted model. We were able to reject the null hypothesis for all of the three diagnostic measures (RMSE, correlation, OLS bias), in simulations with both intensive and extensive variables. In every instance, the ratio of the likelihoods is significantly different from 1; adding RS to the baseline specification results in lower residual deviance, and (generally) lower BIC scores.

Outcome	Model	BIC	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
log(RMSE)	RN	157769.99	91665.00	29943.43			
	RN+RS	157605.63	91662.00	29878.62	3	64.81	<0.001***
	RN×RS	157570.22	91653.00	29833.60	9	45.02	<0.001***
Spearman's correlation	RN	-229356.87	91663.00	438.96			
	RN+RS	-229678.68	91660.00	437.26	3	1.7	<0.001***
	RN×RS	-229844.84	91651.00	435.97	9	1.28	<0.001***
OLS estimation bias	RN	487440.56	91663.00	1091477.23			
	RN+RS	487391.25	91660.00	1090482.45	3	994.77	<0.001***
	RN×RS	487363.51	91651.00	1088930.41	9	1552.05	<0.001***

Table A3.4: Likelihood ratio tests (intensive variable).

Outcome	Model	BIC	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
log(NRMSE)	RN	108282.65	90993.00	17476.86			
	RN+RS	108168.13	90990.00	17448.31	3	28.55	<0.001***
	RN×RS	108182.85	90981.00	17431.44	9	16.87	<0.001***
Spearman's correlation	RN	-133522.36	90993.00	1226.19			
	RN+RS	-133707.06	90990.00	1223.25	3	2.95	<0.001***
	RN×RS	-133962.02	90981.00	1218.45	9	4.8	<0.001***
OLS estimation bias	RN	91874.20	90993.00	14593.57			
	RN+RS	91659.69	90990.00	14553.73	3	39.84	<0.001***
	RN×RS	91526.41	90981.00	14516.03	9	37.7	<0.001***

Table A3.5: Likelihood ratio tests (extensive variable).

One of the reasons why RS does not appear redundant in the Likelihood Ratio Tests may be that RS and RN are capturing conceptually different geometric properties — (dis)aggregation vs. nesting — and the two scores occasionally numerically diverge. As we have seen in Table 1 of the main text, it is possible to obtain a (near-)perfect RS score in the absence of perfect nesting. Such cases, judging by our simulations and real-world examples, are not uncommon in practice. They can arise due to both measurement error (e.g. small misalignments due to an imprecise representation of border features) and structural differences between source and destination units (e.g. as in the grid-to-constituency example in Table 1).

Figure A3.1 shows histograms of the distributions of RS and RN values across our Monte Carlo simulations, along with a scatterplot of RS as a function of RN. The two distributions have very different shapes. RS has a bimodal distribution, with peaks around $RS = 0$ and $RS = 1$. RN appears more normally distributed, with a single mode around $RN = 0.5$ and almost no values at the extremes of $RN = 0$ or $RN = 1$. The relationship between the two measures resembles a logistic curve, in which $RS < RN$ for values of $RN < 0.5$ and $RS > RN$ for $RN > 0.5$. We can parameterize this relationship as follows,

$$\widehat{RS} = \left(1 + e^{6.2 - 12.75 \cdot RN}\right)^{-1}$$

where -6.2 and 12.75 are intercept and slope estimates from a logit regression of RS on RN. This fitted curve appears as a solid black line in the rightmost pane of Figure A3.1.

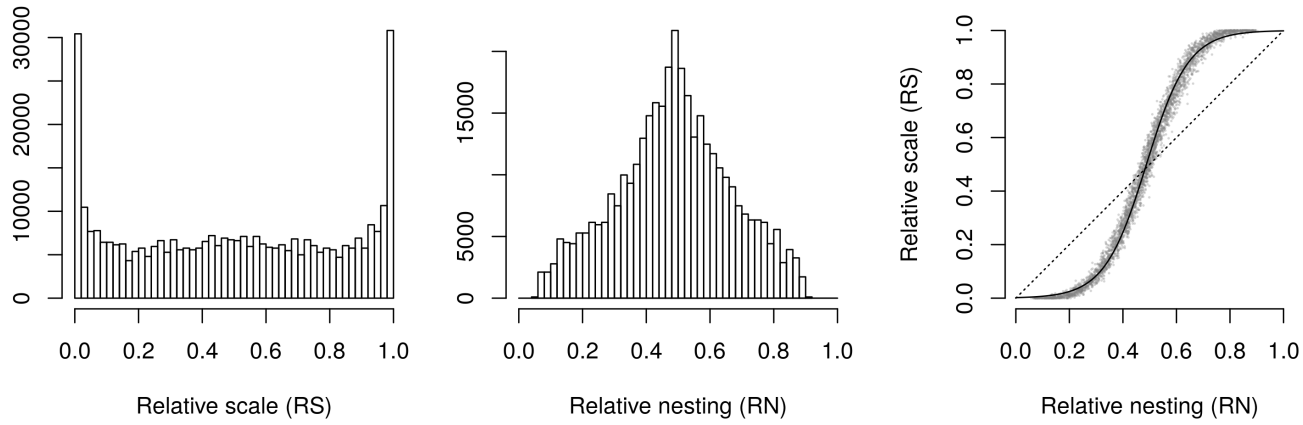


Figure A3.1: How the distributions of RS and RN differ from each other.

This analysis suggests that divergence between RN and RS, while systematic, tends to be limited to a relatively narrow range. The largest absolute differences $|RS - RN|$ are below 0.3 in our simulations (0.35 in our analysis of election data), and there are no cases in which $RS > 0.5$ and $RN < 0.5$ (or vice versa).

Figure A3.2 takes stock of how such divergences might impact analysis — and whether some CoS methods outperform others in such instances. The values in the cells represent average (N)RMSE, correlation and OLS bias in three subsets of simulations:

1. $RS < RN$, where $RS = 0$ *and* $RN \geqslant 0.16$ (90th percentile) (i.e. bottom-left of the scatterplot in Figure A3.1, where the curve begins to turn upward).

2. $RS = RN$, where $0.45 \leqslant RS \leqslant 0.55$ *and* $0.45 \leqslant RN \leqslant 0.55$.

3. $RS > RN$, where $RS = 1$ *and* $RN \leqslant 0.79$ (10th percentile) (i.e. top-right of the scatterplot in Figure A3.1, where the curve begins to flatten).

Figure A3.2 suggests that no CoS method has a clear comparative advantage in cases where RN and RS diverge. As we have already established, transformation quality generally improves as RN and RS increase, so the statistics where $RS = 1 > RN$ unsurprisingly look more favorable than those where $RS = 0 < RN$. The relative performance of the CoS methods, moreover, does not radically change across these subsets of simulations. With some minor exceptions (e.g. simple overlays with centroids when $RS < RN$), no CoS method emerges as a local winner. Methods that fare relatively poorly overall, like population-weighted interpolation, also perform quite poorly in these more specific instances of divergence as well.

RMSE

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	0.69	0.97	0.7	0.42
Overlay (centroids)	2.37	1.03	1.39	6.24
Area Weights (polygons)	0.65	0.98	0.63	0.36
Area Weights (centroids)	0.64	0.98	0.63	0.34
Pop Weights (polygons)	9.35	3.72	5.27	2.97
Pop Weights (centroids)	13.3	11.69	8.57	3.49
TPRS-Forest	0.63	0.97	0.62	0.34
TPRS-Forest (w/ resid)	0.66	0.98	0.65	0.37
TPRS-Area Weights	0.73	1.01	0.73	0.42
Ordinary Kriging	0.65	1.01	0.63	0.34
Universal Kriging	0.65	1.01	0.63	0.34
Rasterization	0.66	0.98	0.65	0.37
Median	0.66	0.99	0.65	0.37

Spearman's correlation

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	0.92	0.83	0.93	0.96
Overlay (centroids)	0.8	0.86	0.81	0.76
Area Weights (polygons)	0.94	0.85	0.96	0.98
Area Weights (centroids)	0.94	0.85	0.96	0.98
Pop Weights (polygons)	0.91	0.82	0.93	0.96
Pop Weights (centroids)	0.91	0.83	0.92	0.95
TPRS-Forest	0.94	0.86	0.95	0.98
TPRS-Forest (w/ resid)	0.94	0.84	0.95	0.97
TPRS-Area Weights	0.83	0.79	0.8	0.9
Ordinary Kriging	0.94	0.79	0.96	0.98
Universal Kriging	0.94	0.81	0.96	0.98
Rasterization	0.94	0.84	0.95	0.97
Median	0.94	0.84	0.95	0.97

OLS estimation bias

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	-0.09	-0.11	-0.1	-0.12
Overlay (centroids)	-1.4	0.05	-1.56	-2.29
Area Weights (polygons)	0.05	0.2	0.06	-0.06
Area Weights (centroids)	0.06	0.2	0.07	-0.03
Pop Weights (polygons)	-0.59	-0.41	-0.8	-0.22
Pop Weights (centroids)	-0.65	-0.53	-0.79	-0.43
TPRS-Forest	0.1	0.29	0.09	-0.02
TPRS-Forest (w/ resid)	0.06	0.18	0.04	-0.03
TPRS-Area Weights	0.1	0.3	0.07	-0.003
Ordinary Kriging	0.48	0.46	0.1	-0.02
Universal Kriging	0.15	0.42	0.11	-0.02
Rasterization	0.06	0.18	0.04	-0.03
Median	0.06	0.19	0.05	-0.03

(a) Intensive variable (Gaussian)

NRMSE

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	0.5	1.87	0.21	0.09
Overlay (centroids)	0.44	1.58	0.21	0.09
Area Weights (polygons)	0.11	0.16	0.1	0.07
Area Weights (centroids)	0.06	0.08	0.07	0.04
Pop Weights (polygons)	0.11	0.16	0.1	0.07
Pop Weights (centroids)	0.06	0.08	0.07	0.04
TPRS-Forest	0.14	0.14	0.14	0.15
TPRS-Forest (w/ resid)	0.14	0.14	0.14	0.15
TPRS-Area Weights	0.19	0.19	0.18	0.22
Ordinary Kriging	0.13	0.13	0.13	0.15
Universal Kriging	0.21	0.18	0.2	0.24
Rasterization	0.21	0.21	0.2	0.24
Median	0.14	0.16	0.14	0.12

Spearman's correlation

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	0.76	0.63	0.73	0.95
Overlay (centroids)	0.68	0.46	0.68	0.95
Area Weights (polygons)	0.92	0.85	0.91	0.98
Area Weights (centroids)	0.95	0.93	0.95	0.99
Pop Weights (polygons)	0.91	0.86	0.91	0.98
Pop Weights (centroids)	0.95	0.93	0.95	0.99
TPRS-Forest	0.84	0.81	0.83	0.91
TPRS-Forest (w/ resid)	0.84	0.81	0.82	0.91
TPRS-Area Weights	0.5	0.43	0.46	0.67
Ordinary Kriging	0.84	0.82	0.84	0.91
Universal Kriging	0.92	0.92	0.92	0.95
Rasterization	0.45	0.38	0.41	0.6
Median	0.84	0.82	0.83	0.95

OLS estimation bias

	Overall	RS<RN	RS=RN	RS>RN
Overlay (polygons)	-1.15	-2.19	-1.14	-0.22
Overlay (centroids)	-1.21	-2.31	-1.19	-0.2
Area Weights (polygons)	-0.64	-0.97	-0.68	-0.27
Area Weights (centroids)	-0.09	-0.16	-0.11	-0.03
Pop Weights (polygons)	-0.65	-0.99	-0.69	-0.28
Pop Weights (centroids)	-0.1	-0.17	-0.1	-0.04
TPRS-Forest	-0.65	-0.74	-0.71	-0.51
TPRS-Forest (w/ resid)	-0.7	-0.79	-0.79	-0.53
TPRS-Area Weights	-0.65	-1.1	-0.58	-0.31
Ordinary Kriging	-0.64	-0.69	-0.71	-0.54
Universal Kriging	-1.24	-1.18	-1.25	-1.09
Rasterization	-1.33	-1.5	-1.44	-0.84
Median	-0.65	-0.98	-0.71	-0.29

(b) Extensive variable (Poisson)

Figure A3.2: **Transformation quality when RN and RS diverge**

A4. Overview of change-of-support methods

Simple overlays

We consider two types of overlay methods: polygon-to-polygon and point-to-polygon. For point coordinates in the latter transformation, we used polygon centroids.

- *Simple overlay (polygons).* The source layer $\mathcal{G}_S$ is a set of polygons, representing administrative units, constituencies or other discrete areas of interest. The destination layer $\mathcal{G}_D$ is a second, different set of polygons. For each destination polygon $j \in \{1, \ldots, N_S\}$, the algorithm identifies the source polygon $i \in \{1, \ldots, N_S\}$ with which it overlaps. If i overlaps with multiple destination polygons, we assign it to the destination polygon with which it shares the largest areal overlap. For each destination unit j, the algorithm then computes statistics (e.g. sum, mean) for the source polygon $i \cap j$ assigned to it.

- *Simple overlay (centroids).* The source layer $\mathcal{G}_S$ is a set of points, representing the centroids of polygons or other fixed address (e.g. event location). The destination layer $\mathcal{G}_D$ is a set of polygons. For each polygon j, the algorithm identifies the set of points that fall within it, and calculates statistics for these overlapping points $i \cap j$.

Simple overlays are the industry standard for the aggregation of event data, which are typically stored as point locations. Its primary advantages are its speed and ease of implementation, which requires no re-weighting or geostatistical modeling. Its primary disadvantage is its one-to-one or many-to-one mapping of source-to-destination units, which can generate missing values in $\mathcal{G}_D$, particularly when $\mathcal{G}_S$ are points, when $N_S \ll N_D$, or when destination units are smaller in area relative to source units. This is less of a problem when missing values can be treated as "true zeroes" (e.g. event counts). It is more of a problem for most other social science applications (e.g. votes, surveys).

Area-weighted interpolation

We consider two variants of areal interpolation: polygon-to-polygon and point-to-polygon. For point coordinates in the latter transformation, we used polygon centroids.

- *Area weights (polygons).* The source layer $\mathcal{G}_S$ and destination layer $\mathcal{G}_D$ are sets of (different) polygons, representing administrative units, constituencies or other discrete areas of interest. The algorithm intersects the two polygon layers, creating a third polygon layer $\mathcal{G}_{S \cap D}$, where each feature $i \cap j \in \{1, \ldots, N_{S \cap D}\}$ is a part of source polygon i that falls inside destination polygon j. The algorithm then computes area weights, proportional to the share of j's area contributed by each source polygon. Each intersection $i \cap j$ receives weight $w_{i \cap j}^{(\text{area})} = \frac{a_{i \cap j}}{a_j}$, where $a_{i \cap j}$ is the area of $i \cap j$ and a_j is the area of j. For each polygon j, the algorithm calculates weighted statistics for overlapping source features. For intensive variables, these statistics are typically weighted averages of values in intersections, $x_j = \sum_{i \cap j} w_{i \cap j}^{(\text{area})} x_{i \cap j}$, where $x_{i \cap j}$ is the value of some variable x in intersection $i \cap j$. For extensive variables, these statistics are typically sums of values in

all constituent intersections, $x_j = \sum_{i \cap j} x_{i \cap j}$, adjusted so as to satisfy the pycnophylactic (mass-preserving) property.

- *Area weights (centroids).* The source layer $\mathcal{G}_S$ is a set of points, representing the centroids of polygons or other fixed address (e.g. event location). The destination layer $\mathcal{G}_D$ is a set of polygons. This method includes an additional, intermediate step to convert the point features into polygons, by creating a Voronoi tessellation of the study area. During the tessellation stage, the algorithm creates N_S polygons, such that for any polygon l_i corresponding to point i, all points inside l_i are closer to i than to any other point $-i$. This is followed by a polygon-to-polygon interpolation stage, as described in the previous paragraph.

Areal weighting is the default CoS method built in to many commercial and open-source Geographic Information Systems. In contrast to simple overlays, interpolation by design leaves no gaps or missing regions. It is also easy to implement and requires information only on the geometries of source and destination units, with no need for ancillary data. Its point-to-polygon variant is particularly attractive if the boundaries of source units are unknown. However, this method rests on several important assumptions, which are fully satisfied in only very rare cases. Most notably, it assumes that the phenomenon of interest is uniformly distributed in source polygons. The point-to-polygon variant is also sensitive to assumptions about boundary placement made in the creation of the synthetic tessellated polygons.

Population-weighted interpolation

We consider two variants of population-weighted interpolation: polygon-to-polygon and point-to-polygon. As above, we used polygon centroids as point coordinates.

- *Population weights (polygons).* This method requires three spatial data layers. The source layer $\mathcal{G}_S$ and destination layer $\mathcal{G}_D$ are sets of (different) polygons, representing administrative units, constituencies or other discrete areas of interest. The third, ancillary layer is a raster of population levels Π, which fully overlaps with the area of $\mathcal{G}_S$ and $\mathcal{G}_D$. The algorithm intersects the two polygon layers, creating a third polygon layer $\mathcal{G}_{S \cap D}$, where each feature $i \cap j \in \{1, \ldots, N_{S \cap D}\}$ is a part of source polygon i that falls inside destination polygon j. The algorithm then computes population weights, proportional to the share of j's population contributed by each source polygon. Each intersection $i \cap j$ receives weight $w_{i \cap j}^{(\text{pop})} = \frac{p_{i \cap j}}{p_j}$, where $p_{i \cap j}$ is the population count of intersection $i \cap j$ and p_j is the population of j. For each polygon j, the algorithm calculates weighted statistics for overlapping source features, as described above.

- *Population weights (centroids).* The source layer $\mathcal{G}_S$ is a set of points, representing the centroids of polygons or other fixed address (e.g. event location). The destination layer $\mathcal{G}_D$ is a set of polygons. The third, ancillary layer is the raster of population levels Π. As with the second, point-based area-weighting method, this method includes an intermediate step to convert the point features into tessellated polygons, and calculates population levels for each intersection between these polygons and $\mathcal{G}_D$. This is followed by a population-weighted polygon-to-polygon interpolation stage, as described in the previous paragraph.

This is an extension of the area weighting method, which dispenses with the uniformity assumption and seeks to account for variation in the underlying distribution of x. Although we use the term "population" here, this method is extensible to any set of ancillary data that researchers consider to be predictive of this distribution. The primary disadvantages are data scarcity (e.g. contemporaneous population data are not always available), the assumption that this ancillary layer is indeed predictive of x, as well as assumptions made during the tessellation stage in the point-based version.

Thin-plate regression spline methods

We consider two methods that employ thin-plate regression splines: TPRS-Forest estimation and TPRS-Areal Weighting. These methods allow researchers to use external variables when interpolating between source and designation polygons. In addition, these methods provide a way to construct uncertainty measures for the designation unit by exploiting geographic variation in the XY-coordinates of the source polygon units. Finally, these methods provide a way to interpolate spatial data with non-Gaussian error distributions, such as binary, count, or categorical data. These two methods differ, however, in how they handle information from the source polygon. TPRS-Forest estimation excludes random noise fluctuations, or non-systematic variation, when mapping from the source to designation polygon. TPRS-Areal Weighting, on the other hand, interpolates all information, including random noise fluctuations, from the source to designation polygon.

Thin-plate regression splines (TPRS) (Duchon, 1977; Wood, 2003) estimate a nonparametric smooth function $f(\cdot)$ — in our case, $f(\text{Long}, \text{Lat})$ — by minimizing

$$||\mathbf{y} - \mathbf{f}|| + \lambda J_{md}(f)$$

where $\mathbf{y}$ is a vector of y_i's, $\mathbf{f} = |f(x_1), \ldots, f(x_n)|'$, $\mathbf{x}$ is an $N_{S \cap D} \times d$ matrix of predictors (in this case, longitude and latitude), $||\cdot||$ is the Euclidean norm, and λ is a smoothing parameter governing the model degrees of freedom, which can be selected through generalized

cross-validation or the Akaike Information Criterion. J_{md} is a "wiggliness penalty" for f:

$$J_{md} = \int \dots \int_{\mathcal{R}_d} \sum_{v_S! \dots v_d! = m} \frac{m!}{v_S! \dots v_d!} \left(\frac{\delta^m f}{\delta x_1^{v_S} \dots \delta x_d^{v_d}} \right) dx_1 \dots dx_d$$

where m is the order of differentiation, satisfying $2m > d$. In our two-predictor case, the wiggliness penalty becomes

$$J_{22} = \int \int \left(\frac{\delta^2 f}{\delta \text{Long}^2} \right)^2 + \left(\frac{\delta^2 f}{\delta \text{Lat}^2} \right)^2 + 2 \left(\frac{\delta^2 f}{\delta \text{Long}^2 \text{Lat}^2} \right)^2 d\text{Long} d\text{Lat}$$

The advantage of thin-plate regression splines is that they avoid the knot placement problems of conventional regression spline modeling, reducing the subjectivity of the model fitting process. They also nest smooths of lower rank within smooths of higher rank.

- *TPRS-Forest Estimation.* This process begins by fitting a thin-plate regression spline to the source polygon and predicting conditional mean and standard error estimates to the designation polygon units. Next, we fit a random forest model to the source polygon and predict conditional mean estimates to the designation polygon units.

- *TPRS-Areal Weighting.* This process begins by intersecting the source and designation polygon units. Next, we fit a thin-plate regression spline to the source polygon and predict conditional mean and standard error estimates to the designation polygon units. Third, we conduct simple areal weighting using the TPRS residuals and the shape boundaries of the source and designation polygon units. Finally, we construct error bounds by bootstrapping the estimated values: conditional mean, areal weighted residuals, and standard error.

Kriging methods

We consider two block kriging methods in the main text: ordinary and universal. The primary difference is that the second requires ancillary data, while the first does not.

- *Ordinary Kriging.* The source layer $\mathcal{G}_S$ is a set of points, representing (in our case) the centroids of polygons. However, the points can represent any other fixed address (e.g. sampling or event location). The destination layer $\mathcal{G}_D$ is a set of polygons. At the point level, ordinary kriging interpolates a value $Z(x_0)$ of random field $Z(x)$ at unobserved

location x_0, using data from observed location x_i. The kriging estimator is:

$$\hat{Z}(x_0) = \sum_{i=1}^{n} w_i(x_0) Z(x_i)$$

where $w_i(x_0),\ i = 1, \ldots, n$ is a spatial weight. These weights are based on a variogram model, which describes the degree to which nearby locations have similar values:

$$\hat{\gamma}(d) = \frac{1}{2n(d)} \sum_{d_{iq}=d} (Z(x_i) - Z(x_q))^2$$

where $\hat{\gamma(d)}$ is estimated semivariance, $n(d)$ is number of point pairs (x_i, x_q) separated by distance d, and $Z(x_i)$ is value of a variable at location x_i. As locations become farther apart, they should become more dissimilar and have higher semivariance $\gamma(d)$. We select a variogram model appropriate to the data by minimizing the sum of squared residuals from the sample. To interpolate at point x_0 based on points $x_1, \ldots, x_{N_S}$, the weights $w_1, \ldots w_{N_S}$ must be found, by solving the system of linear equations:

$$\begin{bmatrix} \gamma(d_{11}) & \gamma(d_{12}) & \cdots & \gamma(d_{1N_S}) & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \gamma(d_{N_S1}) & \gamma(d_{N_S2}) & \cdots & \gamma(d_{N_SN_S}) & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix} = \begin{bmatrix} w_1 \\ \vdots \\ w_{N_S} \\ \lambda \end{bmatrix} \begin{bmatrix} \gamma(d_{10}) \\ \vdots \\ \gamma(d_{N_S0}) \\ 1 \end{bmatrix}$$

where $\gamma(d_{ij})$ is the semivariance for the distance between points x_i and x_j, and λ is the trend parameter. Ordinary kriging assumes an unknown constant trend: $\lambda(x) = \lambda$. Point-level interpolation by ordinary kriging is given by:

$$\hat{Z}(x_0) = \begin{pmatrix} w_1 \\ \vdots \\ w_{N_S} \end{pmatrix}' \begin{pmatrix} Z(x_1) \\ \vdots \\ Z(x_{N_S}) \end{pmatrix}$$

Ordinary kriging error is:

$$var\left(\hat{Z}(x_0) - Z(x_0)\right) = \begin{pmatrix} w_1 \\ \vdots \\ w_{N_S} \\ \lambda \end{pmatrix}' \begin{pmatrix} \gamma(d_{10}) \\ \vdots \\ \gamma(d_{N_S0}) \\ 1 \end{pmatrix}$$

This approach can be extended to yield predictions for areal units, via *block kriging* (Cressie, 1993; Chiles and Delfiner, 2009). Let B be an area ("block") that forms the spatial support of $Z(B)$. These blocks can be regularly-shaped grid cells, or irregular polygons. In our case, we can specify a separate block B_j for each destination polygon in $\mathcal{G}_D$. The block kriging predictor is a weighted average of point-level measurements

$$\hat{Z}(B_j) = \sum_{i=1}^{n} w_i(B_j) Z(x_i)$$

This approach is equivalent to predicting multiple points in region B_j, and averaging those values over B_j (Young et al., 2009). Using the predicted values of this random field, the algorithm computes statistics (e.g. means, sums) for each destination polygon j, preserving the pycnophylactic property for extensive variables as appropriate.

- *Universal Kriging.* This method requires three or more spatial data layers. As above, the source layer $\mathcal{G}_S$ is a set of points (e.g. centroids) and destination layer $\mathcal{G}_D$ is a set of polygons. The third, ancillary layer is a raster of population levels Π, which fully overlaps with the bounding box of $\mathcal{G}_S$ and $\mathcal{G}_D$. The algorithm interpolates a value $Z(x_0)$ of random field $Z(x)$ at unobserved location x_0, using data from observed location x_i *and* population values observed at x_0 and x_i. We then extend this approach to estimate block averages for each polygon in $\mathcal{G}_D$.

Kriging is widely used in natural and environmental sciences as a solution to the change-of-support problem (Gotway and Young, 2007). Unlike the interpolation and overlay methods, this is a model-based approach, which can ascertain the uncertainty of estimates. However, kriging is highly sensitive to variogram model selection, and some of its assumptions (particularly regarding the smoothness of interpolated values over space) can be problematic for social science.

A5. Analysis of Swedish electoral data

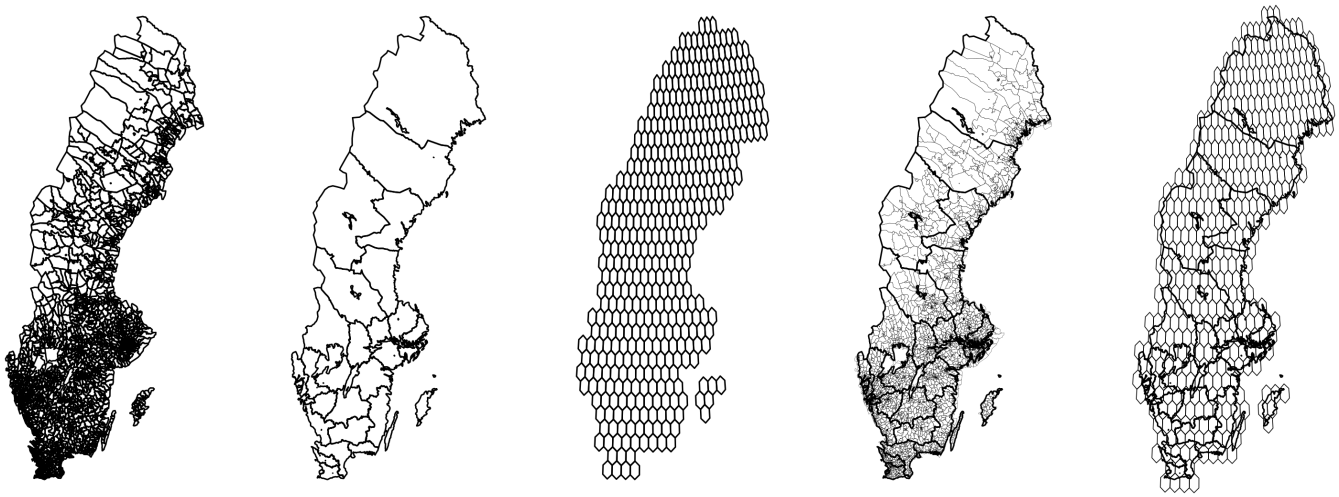
The current section replicates the main text's CoS analysis of electoral data from the U.S. state of Georgia, with analogous data from Sweden's 2010 Riksdag (unicameral legislature) elections. Figure A5.3 shows the spatial data layers used in this analysis, which correspond to those in Figure 1 (precincts, constituencies, 0.5° hexagonal grid cells).[1]

[1]The precincts boundaries are from data.val.se/val/val2010/statistik. The constituency boundaries are from Kollman et al. (2017).

Table A5.6 reports relative scale and nesting coefficients for these polygons (counterpart to Table 1). Notably, as Table A5.6b shows, while precincts should be fully nested within constituencies "in real life," this is not technically the case in the geospatial data ($RN = 0.90$). The nesting coefficient for precincts-constituencies is about the same as it is for precincts-grid, although RS is larger for the former pair. This surprisingly low RN is likely due to measurement error, the differential precision of the two geospatial boundary datasets, and other discrepancies (e.g. coastal features, bodies of water).

Figure A5.4 illustrates several examples of transformed values of Top-2 Competitiveness alongside true values, for (a) precinct-to-constituency and (b) constituency-to-grid CoS. Figure A5.5 reports fit diagnostics for CoS transformations of Swedish election results. The results here are consistent with those for Georgia (Figure 3).

Figure A5.3: **Spatial data layers** (Sweden)



(a) Precincts (b) Constituencies (c) .5° grid (d) a + b (e) c + b

Table A5.6: **Relative scale and nesting of polygons in Figure A5.3.**

(a) Relative nesting (RN)

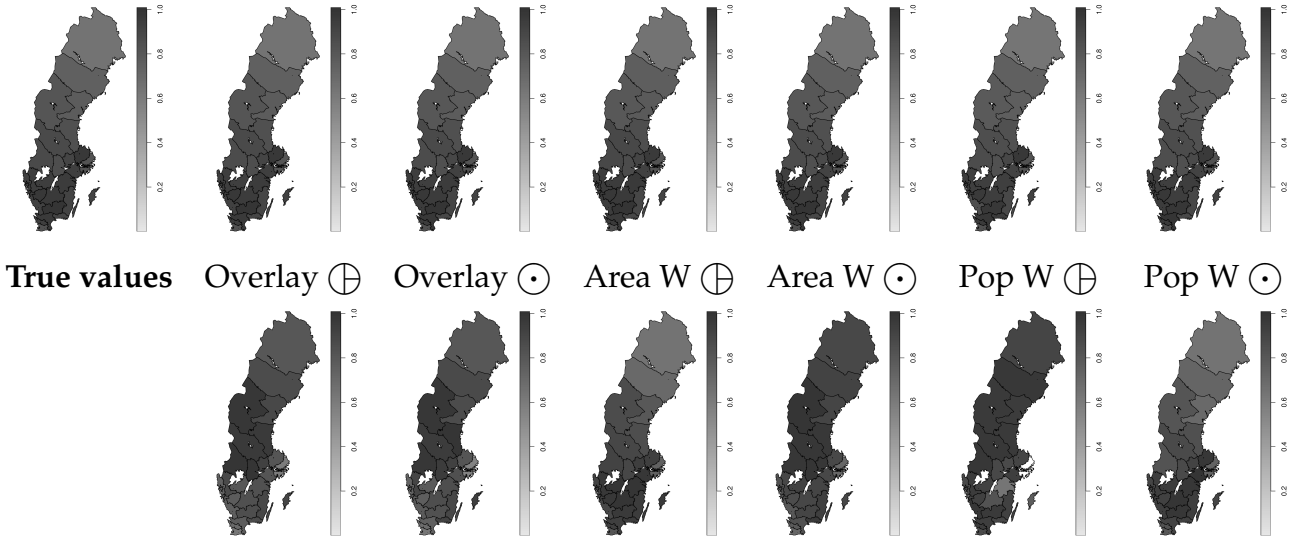
Source	Destination		
	(a)	(b)	(c)
(a) Precincts	–	0.90	0.90
(b) Constituencies	0.02	–	0.19
(c) .5° grid	0.27	0.66	–

(b) Relative scale (RS)

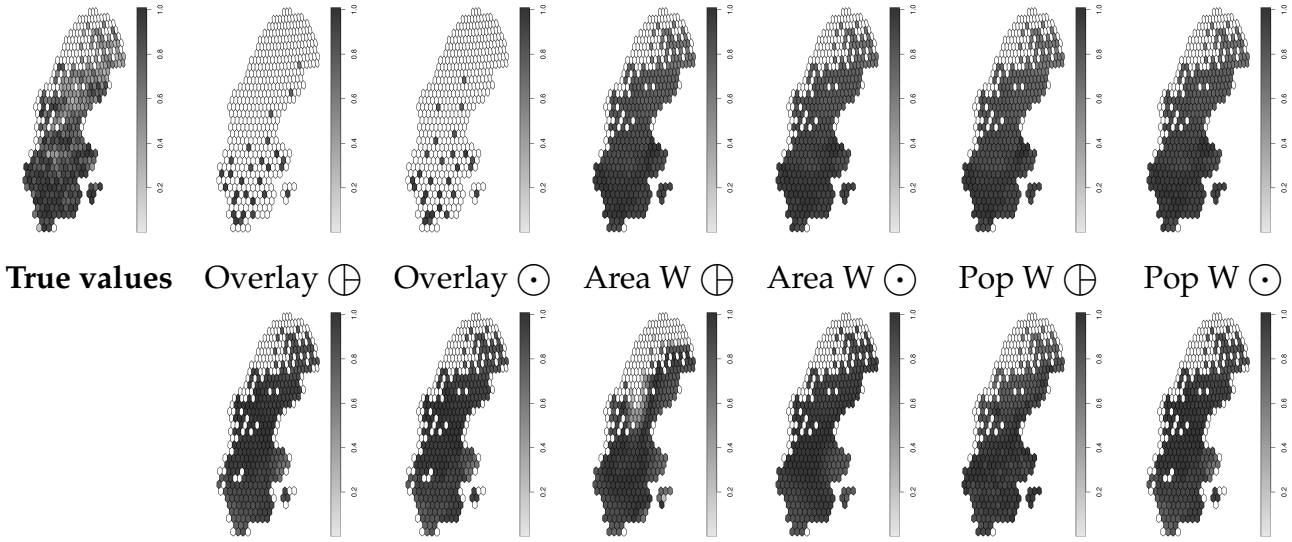
Source	Destination		
	(a)	(b)	(c)
(a) Precincts	–	1.00	0.94
(b) Constituencies	0.00	–	0.01
(c) .5° grid	0.09	0.99	–

Figure A5.4: **Output from change-of-support operations** (Sweden). ⨁: source features are polygons. ⊙: source features are polygon centroids.

(a) Precinct-to-constituency



(b) Constituency-to-grid



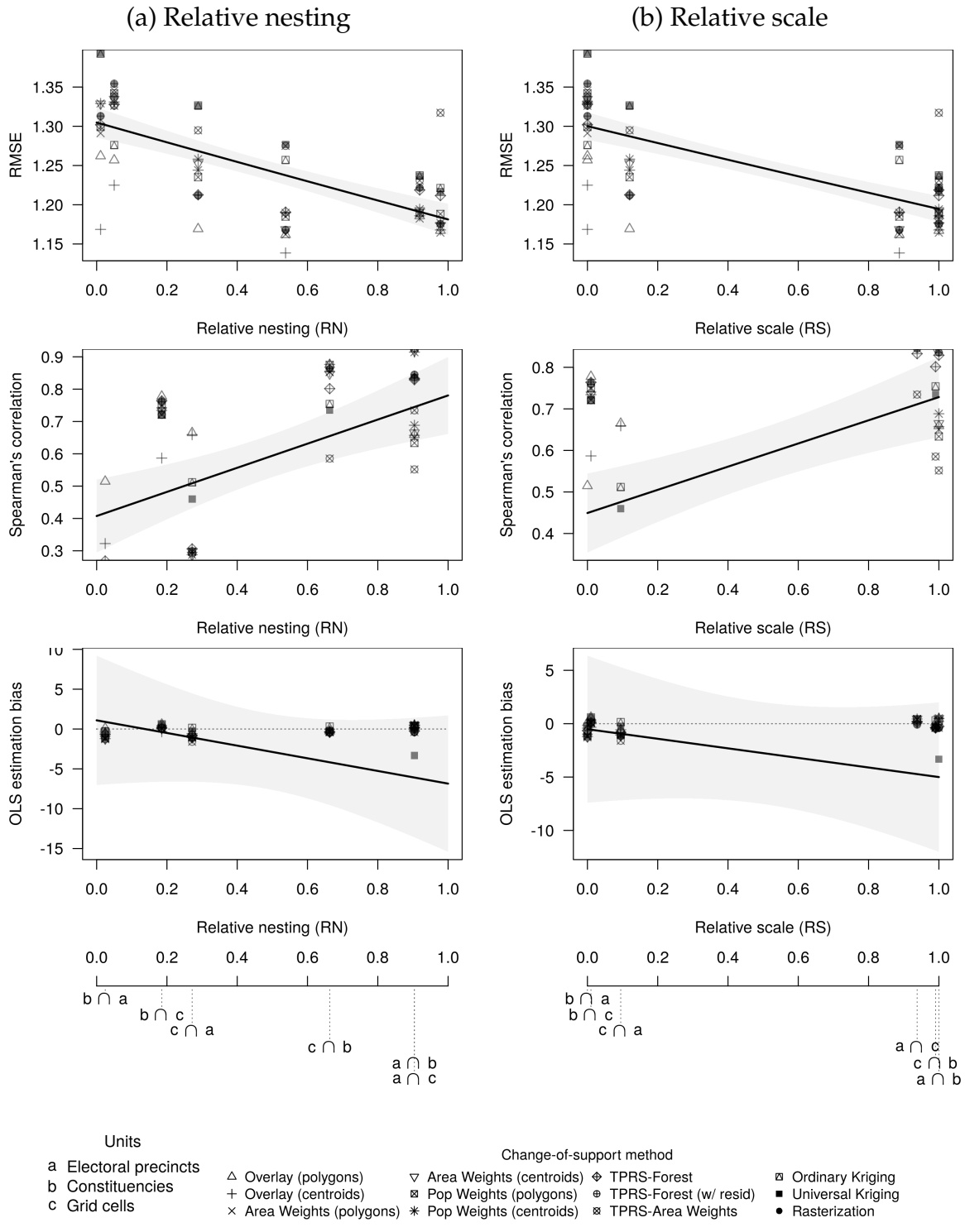


Figure A5.5: **Relative nesting, scale, and transformations of election data** (Sweden)

A6. Direct vs. indirect transformations of intensive variables

The conceptual distinction between intensive and extensive variables can obscure the fact that intensive variables are often composites of multiple extensive variables. While some intensive variables, like air temperature, can be directly observed "in the wild," others are functions of extensive variables divided or normalized by other extensive variables, like volume, area or population count. For example, in order to establish a party's vote share in an election (an intensive variable), one must first observe the total number of valid votes cast (an extensive variable) along with the number of valid votes the party received (another extensive variable), and divide the latter by the former. To take another example, Top-2 Competitiveness — the variable we used in our electoral data analysis — can be calculated from several combinations of components:

$$\begin{aligned}\text{Top-2 Competitiveness} &= 1 - \text{winning party vote share margin} \\ &= \frac{\text{valid votes} - \text{winning party vote count margin}}{\text{valid votes}} \\ &= \frac{\text{valid votes} - (\text{votes for winner} - \text{votes for runner-up})}{\text{valid votes}}\end{aligned}$$

of which only "winning party vote share margin" is an intensive variable, albeit one which is itself a function of multiple constitutive extensive variables. Indeed, some extensive variables are also composites: "vote count margin" is the difference between the vote counts of the winner and runner-up.

What is the appropriate way to handle CoS operations for composite intensive variables? The choice boils down to the following:

1. **Direct transformation**: perform CoS operations directly on the intensive variable, rather than its extensive components. In our example, this means treating competitiveness as a single, self-contained variable, and attempting to calculate its average values in destination units.

2. **Indirect transformation**: perform CoS operations on a variable's extensive components, and use the transformed values of these individual components to reconstruct the intensive variable within destination units. The transformed values for all extensive variables must satisfy the pycnophylactic (mass-preserving) property. In our case, this means calculating sums of valid votes in each destination unit, identifying winners and runners-up in destination units by ranking the parties by aggregate number of votes they each received, and then using these components to calculate competitiveness scores within destination units.

The choice between these two approaches depends in part on data availability. Direct transformation is suitable for a "data-poor" scenario, where the researcher only has access to a composite measure and not the underlying variables used to construct it. Indirect transformation, which we adopted in the main text, is better-suited for a "data-rich" scenario, where the researcher has access to the full complement of component variables. Yet data availability is not the only consideration that is relevant here.

As Table A6.7 shows, the comparative advantages of direct vs. indirect transformations depend on the relative nesting and scale of source and destination units. When RS and RN are high, as in the case of precinct-to-constituency transformations in Georgia (a to b), the indirect approach outperforms the direct one. Errors are generally smaller, and correlations slightly higher, when using multiple CoS operations on individual extensive components rather than a single CoS operation on the composite measure. When RS and RN are lower, as in virtually all other cases in Table A6.7, transforming the single composite measure yielded more reliable results than transforming individual components.

Diagnostic	Source	Destination	RS	RN	Direct	Indirect
RMSE	a. Precincts	b. Constituencies	1.00	0.98	1.20	1.17
	a. Precincts	c. Grid cells	1.00	0.92	1.21	1.18
	b. Constituencies	a. Precincts	0.00	0.01	1.32	1.49
	b. Constituencies	c. Grid cells	0.12	0.29	1.25	1.50
	c. Grid cells	a. Precincts	0.00	0.05	1.32	4.98
	c. Grid cells	b. Constituencies	0.89	0.54	1.20	1.24
Spearman's correlation	a. Precincts	b. Constituencies	1.00	0.98	0.73	0.70
	a. Precincts	c. Grid cells	1.00	0.92	0.85	0.77
	b. Constituencies	a. Precincts	0.00	0.01	0.45	0.07
	b. Constituencies	c. Grid cells	0.12	0.29	0.53	0.34
	c. Grid cells	a. Precincts	0.00	0.05	0.67	0.45
	c. Grid cells	b. Constituencies	0.89	0.54	0.72	0.61
OLS estimation bias	a. Precincts	b. Constituencies	1.00	0.98	0.94	-1.03
	a. Precincts	c. Grid cells	1.00	0.92	0.35	-0.31
	b. Constituencies	a. Precincts	0.00	0.01	0.32	-2.07
	b. Constituencies	c. Grid cells	0.12	0.29	2.62	-1.59
	c. Grid cells	a. Precincts	0.00	0.05	-0.60	-1.40
	c. Grid cells	b. Constituencies	0.89	0.54	1.35	-2.05

Table A6.7: Transformation quality when interpolating component intensive variables directly ("Direct") vs. reconstructing them from extensive components ("Indirect").

This analysis suggests that an indirect strategy of reconstructing secondary statistics

from transformed constituent variables pays the most dividends in cases of aggregation across nested units. When units are less nested, the indirect approach can actually backfire — the reconstructed composite becomes less accurate because, at lower values of RN and RS, the transformations of each component themselves become less accurate. In such cases, the researcher may be better off transforming the composite measure directly.

A7. Monte Carlo study design

At each iteration, our simulations executed the following routine:

1. Draw a set of (random) source ($\mathcal{G}_S$) and destination ($\mathcal{G}_D$) polygons. Let N_S be the number of source polygons, and N_D be the number of destination polygons. Create a bounding box $\mathcal{B}$ defined by coordinate set $\{x_{\min}, x_{\max}, y_{\min}, y_{\max}\}$. Within $\mathcal{B}$, sample a random set of N_S points. Create N_S tessellated polygons such that for any polygon $j \in \{1, \ldots, N_S\}$ corresponding to point $i \in \{1, \ldots, N_S\}$, all points inside j are closer to i than to any other point $-i$. Repeat procedure for the N_D destination polygons.

2. Randomly allocate X values (e.g. votes, margins) across $\mathcal{G}_S$ and $\mathcal{G}_D$.

   - For **intensive variables** (scale-independent, like population density or vote shares), we simulated values from a Gaussian Random Field (GRF). For each unit in the intersection $\mathcal{G}_S \cap \mathcal{G}_D$, draw a value of x from a mean zero GRF $\{X_n\}_{n \in G}$, simulated with the sequential simulation algorithm (Goovaerts, 1997). We model spatial autocorrelation in this field with a variogram, which describes the degree to which nearby locations have similar values. The semivariance $\gamma(d)$ is formally defined as the squared difference in values between locations:

$$\hat{\gamma}(d) = \frac{1}{2n(d)} \sum_{d_{ij}=d} \left(X\left(c_i\right) - X\left(c_j\right)\right)^2 \tag{A7.10}$$

     where $n(d)$ is the number of point pairs separated by distance d, and $X(c_i)$ is the value of variable $x(\mathbf{c})$ at location c_i. The variogram can be used for spatial prediction by fitting a parametric model to the variogram, specifying the model type (e.g. exponential, spherical, Matern), partial sill (magnitude of variation), range (maximum distance d), and nugget variance (micro-variability, measurement error). The range parameter here governs the degree of spatial correlation (because $n(d)$ is increasing in d, more observations influence each other as range increases).

     We fit two variogram models:

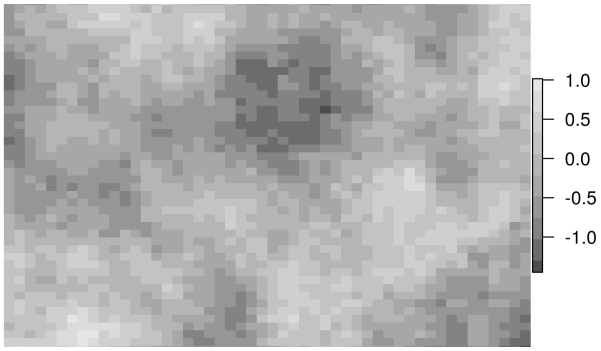
(a) *Spatially autocorrelated data*. Matern covariance semi-variogram model, with range of 2000, partial sill of 1, nugget variance of 0.

(b) *Spatially random data*. Matern covariance semi-variogram model, with range of 1, partial sill of 1, nugget variance of 0.

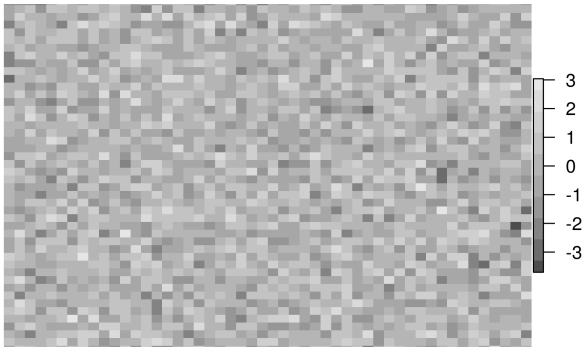
The sequential simulation algorithm follows a random path through locations $c_1, \ldots, c_n$. At each location c_i (e.g. centroid of each intersection $\mathcal{G}_S \cap \mathcal{G}_D$), it computes the conditional distribution of $X(c_i)$ given the data and previously simulated values. It draws a value from this distribution, and assigns it to c_i. It then proceeds to the next unvisited location, until all n locations have assigned values.

Figure A7.6 illustrates two realizations of the GRF, simulated using the (a) autocorrelated and (b) spatially random variogram models.

Figure A7.6: Two examples of Gaussian Random Fields used in simulations



(a) Spatially autocorrelated



(b) Spatially random

- For **extensive variables** (scale-dependent, like population counts or event counts), we simulated values from a Poisson point process (PPP). For each unit in the intersection $\mathcal{G}_S \cap \mathcal{G}_D$, count the number of events x from a Poisson point process (PPP) model. Let $\mathcal{C}$ denote a bounded spatial region, let $A(\mathcal{C})$ represent the area of $\mathcal{C}$, let $X(\mathcal{C})$ be the number of events realized in $\mathcal{C}$, and let λ be the intensity parameter. The probability that exactly k events occur in region $\mathcal{C}$ is

$$P(X(S) = k) = \frac{(\lambda A(\mathcal{C}))^k e^{-\lambda A(\mathcal{C})}}{k!} \quad \forall A(\mathcal{C}) > 0,\ k = 0, 1, 2, \ldots \tag{A7.11}$$

We fit two PPP models:

(a) *Spatially autocorrelated data*. Inhomogeneous Poisson process, where $\lambda(\text{long}, \text{lat})$ is a function of spatial coordinates (assumes intensity is variable over $\mathcal{C}$). We

used a spherical functional form, where intensity is highest in the center, and lower on the periphery:

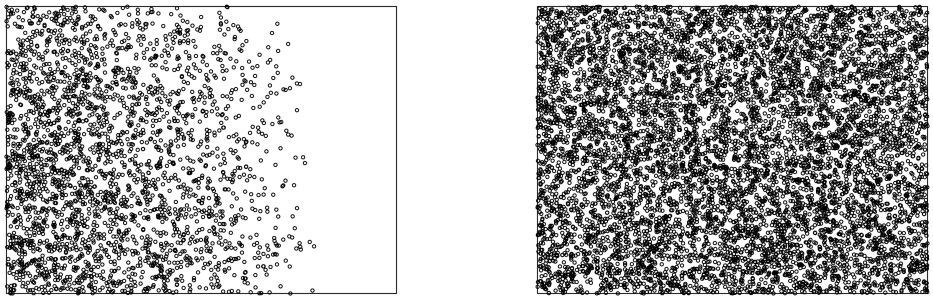
$$\lambda(\text{long}, \text{lat}) = \lambda_{\text{max}} - \lambda_0 \sqrt{(\text{long} - \text{long}_0)^2 + (\text{lat} - \text{lat}_0)^2} \tag{A7.12}$$

where $\lambda_{\text{max}} = 100$ and $\lambda_0 = 10$, and $(\text{long}_0, \text{lat}_0)$ is a central coordinate pair, whose exact location in $\mathcal{C}$ varies randomly across simulations.

(b) *Spatially random data*. Homogeneous Poisson process, where λ is a positive constant (assumes constant intensity over $\mathcal{C}$). We set $\lambda = \lambda_{\text{max}} = 100$.

Figure A7.7 illustrates two realization of the PPP, simulated using the (a) autocorrelated/inhomogeneous and (b) spatially random/homogeneous PPP models.

Figure A7.7: Two examples of Poisson Point Processes used in simulations

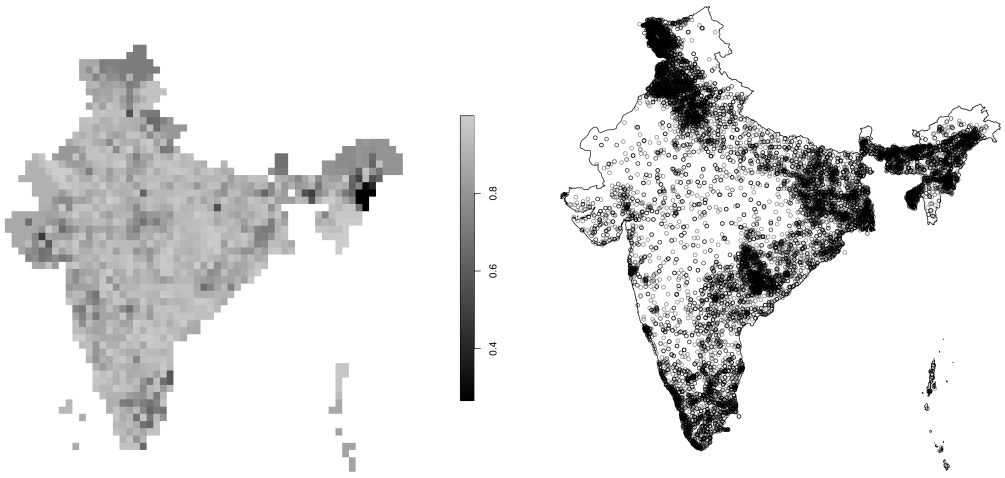


(a) Inhomogeneous PPP (b) Homogeneous PPP

By way of comparison, Figure A7.8 shows the geographic distribution of electoral competitiveness (A7.8a) and political violence (A7.8b) in India.[2] While real-world data do not neatly align with a particular known distribution (e.g., Gaussian, Poisson), the types of clustering and heterogeneities seen here bear some resemblance to those in the spatially-autocorrelated GRF (Figure A7.6a) and inhomogeneous PPP (Figure A7.7a). For example, the heterogeneous pattern in Figure A7.7a is similar to the differential point densities in the north of India in Figure A7.8b, around Jammu and Kashmir.

[2]For this illustration, we used CLEA data from the 1996 Indian general election (Kollman et al., 2022) and xSub multi-source event data on political violence (Zhukov, Davenport and Kostyuk, 2019).

Figure A7.8: Geographic distribution of electoral and violence data in India



(a) Electoral competitiveness (b) Political violence

After x_i values are drawn for each of the sub-units in intersection $\mathcal{G}_S \cap \mathcal{G}_D$, calculate synthetic variable $y_i = \alpha + \beta x_i + \epsilon_i$, with parameters $\alpha = 1$, $\beta = 2.5$ and $\epsilon \sim N(0, 1)$.

Calculate aggregated x and y values for each set of polygons:

- Aggregate over $\mathcal{G}_D$ to get "true values" of x, y in $\mathcal{G}_S$: $(x_{\mathcal{G}S}, y_{\mathcal{G}S})$
- Aggregate over $\mathcal{G}_S$ to get "true values" of x, y in $\mathcal{G}_D$: $(x_{\mathcal{G}D}, y_{\mathcal{G}D})$
- Repeat for two types of aggregated variables: intensive (aggregates are means), and extensive (aggregates are sums). Aggregated values for extensive variables must satisfy the pycnophylactic (mass-preserving) property.

3. Change the geographic support of X from $\mathcal{G}_S$ to $\mathcal{G}_D$, compare true values of x in $\mathcal{G}_D$ to these spatially-transformed values of x. Let K be a set of CoS algorithms (e.g. areal interpolation, kriging, etc.). Each $k \in \{1, \ldots, K\}$ specifies a mapping/transformation between geometries $\mathcal{G}_S$ and $\mathcal{G}_D$ Let $x_{\mathcal{G}D}$ be the "true" value of x in destination polygons $\mathcal{G}_D$ Let $\widehat{x_{\mathcal{G}D}}^{(k)} = f_k(x_{\mathcal{G}S})$ be the estimated value of $x_{\mathcal{G}D}$, calculated using CoS algorithm k For each k, calculate

   - Root mean squared error (RMSE): $\sqrt{\sum_j \frac{1}{N_D}(x_{j\mathcal{G}D} - \widehat{x_{j\mathcal{G}D}})^2}$, for intensive variables.
   - Normalized RMSE (NRMSE): $\frac{\sqrt{\sum_j \frac{1}{N_{\mathcal{G}D}}(x_{j\mathcal{G}D} - \widehat{x_{j\mathcal{G}D}})^2}}{\max(x_{\mathcal{G}D}) - \min(x_{\mathcal{G}D})}$, for (scale-dependent) extensive variables.

- Spearman's rank correlation coefficient: $\frac{\sum_i^{N_D}(R_i(x_{\mathcal{G}D})-\bar{R}(x_{\mathcal{G}D}))(R_i(\widehat{x_{\mathcal{G}D}})-\bar{R}(\widehat{x_{\mathcal{G}D}}))}{\sqrt{\sum_i^{N_D}(R_i(x_{\mathcal{G}D})-\bar{R}(x_{\mathcal{G}D}))^2}\sqrt{\sum_i^{N_D}(R_i(\widehat{x_{\mathcal{G}D}})-\bar{R}(\widehat{x_{\mathcal{G}D}}))^2}}$, where $R_i(\cdot)$ is the rank of observation i, and $\bar{R}(\cdot)$ is the sample mean rank.
- OLS estimation bias: difference between "true" value $\beta = 2.5$ and estimate of $\hat{\beta}$ from regression of y on transformed values of x: $y_{j\mathcal{G}D} = \alpha + \beta\widehat{x_{j\mathcal{G}D}} + \epsilon$.

We ran this simulation for all $N_S \in [10, \ldots, 200]$ and $N_D \in [10, \ldots, 200]$, totalling $191^2 = 36481$ potential CoS operations, from aggregation ($N_S = 200, N_D = 10$) to disaggregation ($N_S = 10, N_D = 200$). We repeated this process 100 times, with different random seeds.

A8. Additional Monte Carlo results

The main text reports Monte Carlo results only for the RN coefficient. Figures A8.9 and A8.10 show an analogous set of results for the RS coefficient.

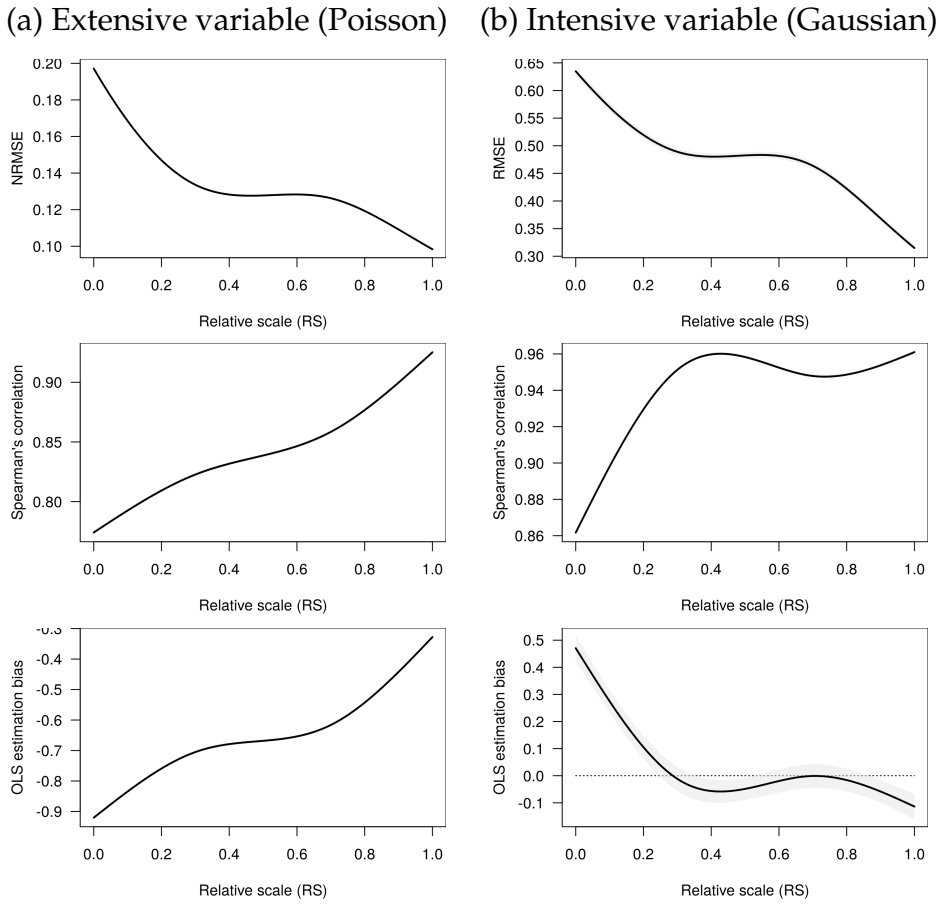
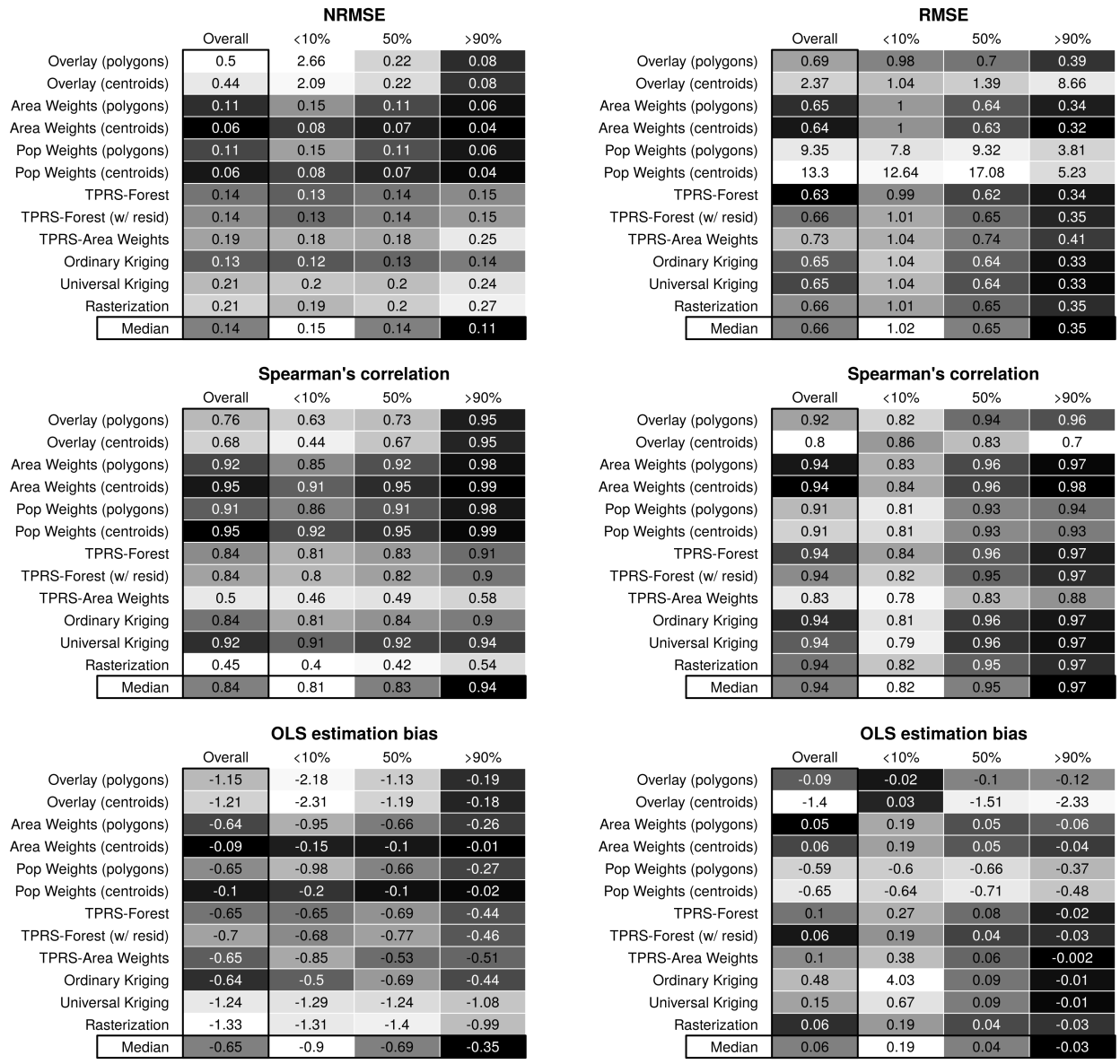


Figure A8.9: **Relative scale and transformations of synthetic data**



(a) Extensive variable (Poisson) (b) Intensive variable (Gaussian)

Figure A8.10: **Transformation quality at different percentiles of relative scale**

A9. Sensitivity analyses with multiple CoS methods

One of our central recommendations calls for researchers to perform sensitivity analyses using alternative CoS methods. While using multiple CoS methods avoids reliance on a single potentially idiosyncratic algorithm, this pluralism is not without pitfalls of its own. Different CoS methods may produce divergent results, and adjudicating between these results is not always straightforward. Direct validation is impossible without ground

truth data, as we cannot know which set of estimates is closest to true values. The existence of divergent estimates, with no clear hierarchy among them, creates temptations to "cherry pick." Yet the opposite approach — treating all results, including deviant ones, as equally valid — can be just as misleading.

We propose a middle path, in which researchers report the results of multiple CoS methods, along with a measure of how divergent each set of results is from the others. Where traditional cross-validation is not possible, we recommend using outlier detection tests to establish which CoS methods produce similar results, and which ones are deviant. The deviance in this case reflects not how distant a result is from the "truth," but how distant it is from results obtained through other methods. In the example below, we use Rosner (1983)'s generalized extreme Studentized deviate test to identify outliers; yet the same logic can be extended to other outlier tests (e.g. χ^2, Dixon, Grubbs).

Let κ denote the number of CoS methods under consideration, indexed by $k = \{1, \ldots, \kappa\}$. Each of these methods yields an estimate, $\hat{x}_k$, which can represent the sample mean of a spatially-transformed variable, a regression coefficient estimate from transformed data, or any other quantity of interest. We will assume that at least $\kappa - \lfloor \kappa/2 \rfloor$ of these estimates come from the same Gaussian distribution, while up to $\lfloor \kappa/2 \rfloor$ estimates may come from a different distribution (where $\lfloor \kappa/2 \rfloor$ is the largest integer less than or equal to $\kappa/2$).

Let $\hat{x}^{(j)}$ be the j-th most extreme value of $\hat{x}$, such that $\hat{x}^{(1)}$ is the value with the farthest distance from the sample mean $\bar{x}$, $\hat{x}^{(2)}$ is the second-farthest from the mean, and so on. Let $\hat{\mathbf{x}}^{(j)}$ be the set of estimates at least as extreme as $\hat{x}^{(j)}$. Let $\bar{x}^{(j)}$ and $s^{(j)}$ be the mean and standard deviation, respectively, of the $\kappa - j$ estimates that remain after removing the j most extreme values. For each $j = 0, \ldots, \lfloor \kappa/2 \rfloor - 1$, we will compare the test statistic

$$R_{j+1} = \frac{|\hat{x}^{(j)} - \bar{x}^{(j)}|}{s^{(j)}}$$

against its corresponding critical value

$$\lambda_{j+1} = \frac{t_{p,\kappa-j-2}(\kappa - j - 1)}{\sqrt{(\kappa - j - 2 + t_{p,\kappa-j-2})(\kappa - j)}}$$

where $t_{p,\kappa-j-2}$ is the p-th quantile of Student's t-distribution with $\kappa - j - 2$ degrees of freedom, $p = 1 - \frac{\alpha/2}{\kappa - j}$, and Type I error level $\alpha = .05$. If $R_j > \lambda_j$ then the j most extreme values are outliers. Measure k is an outlier if it is among these j values, $\hat{x}_k \in \hat{\mathbf{x}}^{(j)}$.

Let ω_k denote the proportion of tests in which measure k is flagged as an outlier:

$$\omega_k = \frac{1}{\lfloor K/2 \rfloor} \sum_{j}^{\lfloor K/2 \rfloor} 1\left(R_j > \lambda_j\right) \cdot 1\left(\hat{x}_k \in \hat{\mathbf{x}}^{(j)}\right)$$

Table A9.8 reports ω_k values for the CoS methods used in our Monte Carlo simulations. The quantity of interest here is the sample mean of the transformed value of variable X, as generated by each algorithm. The table suggests that simple overlay methods produce the most divergent results of all methods considered, with one algorithm (overlay-centroids) being flagged as an outlier in 86 percent of all tests. Whether this method should therefore be excluded from analysis is at the discretion of the researcher, although the high value of ω_k certainly suggests that relying exclusively on simple overlays could be problematic.

Method	ω_k
Rasterization	0.001
TPRS-Forest (w/ resid)	0.001
TPRS-Forest	0.003
Area Weights (polygons)	0.01
Ordinary Kriging	0.01
Universal Kriging	0.02
Area Weights (centroids)	0.02
TPRS-Area Weights	0.16
Pop Weights (polygons)	0.27
Pop Weights (centroids)	0.32
Overlay (polygons)	0.38
Overlay (centroids)	0.86

Table A9.8: Rosner's outlier tests ($\hat{X}$ estimates in Monte Carlo simulations).

Several caveats are in order. Most emphatically, no set of results should be excluded from analysis solely on the basis of an outlier detection test. In the absence of ground truth data, for all we know, the outlier result may be the only "correct" one, while all the others are truly "wrong." By the same token, just because a result is not an outlier does not mean it is necessarily "correct." For instance, rasterization has the lowest ω_k in Table A9.8, but its performance across most cross-validation exercises was middling at best.

An outlier detection test is not a substitute for cross-validation. Outlier tests can tell us how close a series of results are to each other, but not how close they are to the (unobserved) truth. Our recommendation, therefore, is that researchers use ω_k values in the spirit of transparency and discovery, rather than as a discrete numerical threshold or

screening device. At a minimum, we recommend that researchers report ω_k values alongside their main results, to place their findings in context. If the output of a CoS method is frequently flagged as an outlier, further investigation may be warranted into why these results are so deviant. A researcher can then look "under the hood" of the offending algorithm and see whether a well-motivated justification exists for keeping or removing the method from the ensemble. Similarly, researchers whose analysis relies on one primary CoS method can use ω_k to reassure readers that their results are not anomalous.

A10. R package code examples

The `SUNGEO` R package provides tools to calculate nesting metrics and implement some of the CoS methods described here. The package can be installed and loaded as follows:

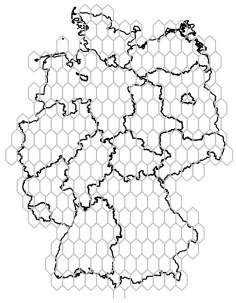
```
# Install package
> install.packages("SUNGEO", dependencies = TRUE)

# Load package:
> library(SUNGEO)
```

Let's begin by loading some spatial data for illustrative purposes: polygons representing German legislative districts and hexagonal grid cells:

```
# Load data
> data(clea_deu2009)
> data(hex_05_deu)

# Preview
> plot(clea_deu2009["geometry"])
> plot(hex_05_deu["geometry"],add=TRUE,border="grey")
```



The grid cells appear generally smaller than legislative districts, and not nested. We can use the `nesting()` function to calculate scale and nesting metrics for the two sets of polygons. Let's calculate nesting metrics for a change of support from district to grid:

```
# Calculate all nesting metrics for a district-to-grid CoS
> nest_1 <- SUNGEO::nesting(
+               poly_from = clea_deu2009,
+               poly_to = hex_05_deu
+               )
> str(nest_1)
List of 12
 $ rs        : num 0.0252
 $ rn        : num 0.16
 $ rs_sym    : num -0.95
 $ rn_sym    : num -0.511
 $ rs_nn     : num 0.0252
 $ rn_nn     : num 0.16
 $ p_intact  : num 0
 $ full_nest: num 0
 $ ro        : num -0.175
 $ gmi       : num 0.84
```

In this scenario, $RS = 0.025, RN = 0.16$, indicating disaggregation across non-nested units. Now let's check the opposite direction:

```
# Calculate all nesting metrics for a grid-to-district CoS
> nest_2 <- SUNGEO::nesting(
+               poly_from = hex_05_deu,
+               poly_to = clea_deu2009
+               )
> str(nest_2)
List of 12
 $ rs        : num 0.976
 $ rn        : num 0.67
 $ rs_sym    : num 0.953
 $ rn_sym    : num 0.511
 $ rs_nn     : num 0.97
 $ rn_nn     : num 0.528
```

```
 $ p_intact : num 0.588
 $ full_nest: num 0.302
 $ ro       : num 0.175
 $ gmi      : num 0.33
```

Here, $RS = 0.98, RN = 0.67$, indicating aggregation and more (but not perfect) nesting. To save computational time, we can modify the `metrix` option to extract specific metrics (e.g. just RN) rather the full battery.

```
# Calculate just RN
> nest_3 <- SUNGEO::nesting(
+               poly_from = hex_05_deu,
+               poly_to = clea_deu2009,
+               metrix = "rn"
+               )
> nest_3
$rn
[1] 0.6702956
```

To identify which source units remain intact and which are split (among other quantities), we can use the option `by_unit=TRUE` to obtain the unit-level components.

```
# Disaggregate nesting metrics by unit (where feasible)
> nest_4 <- SUNGEO::nesting(
+               poly_from = hex_05_deu,
+               poly_to = clea_deu2009,
+               by_unit = TRUE
+               )
nest_4$by_unit
> nest_4$by_unit
     index rs       rn    rs_alt     rn_alt rs_nn    rn_nn p_intact
  1:     1  1 3.19e-06 0.9740827 0.00178652     1 3.19e-06        1
  2:     2  1 4.91e-02 0.9493340 0.22154270     1 4.91e-02        1
  3:     3  1 8.18e-02 0.9493340 0.28602730     1 8.18e-02        1
  4:     4  1 4.88e-03 0.9493340 0.06983717     1 4.87e-03        1
  5:     5  1 4.15e-03 0.9493340 0.06443348     1 4.15e-03        1
 ---
```

```
253:   253  1 1.93e-01 0.9322391 0.43939827     1 1.93e-01         1
254:   254  1 3.43e-01 0.9322391 0.58569486     1 3.43e-01         1
255:   255  1 2.63e-01 0.9019284 0.51319413     1 2.63e-01         1
256:   256  1 2.43e-01 0.9019284 0.49270627     1 2.43e-01         1
257:   257  1 1.01e-01 0.9019284 0.31718422     1 1.01e-01         1
     full_nest       gmi
  1:         0 0.9999968
  2:         0 0.9509188
  3:         0 0.9181884
  4:         0 0.9951228
  5:         0 0.9958483
 ---
253:         0 0.8069292
254:         0 0.6569615
255:         0 0.7366318
256:         0 0.7572405
257:         0 0.8993942

# Visualize rn_i on a map
> hex_05_deu$rn_i <- nest_4$by_unit[,rn]
> plot(hex_05_deu["rn_i"])
```

The SUNGEO package also has routines for batch geocoding of addresses (geocode_osm(), geocode_osm_batch()), overlays (point2poly_simp()), interpolation (poly2poly_ap()), and other CoS methods. Please see the package help files for additional information.

References

Chiles, Jean-Paul and Pierre Delfiner. 2009. *Geostatistics: modeling spatial uncertainty*. Vol. 497 2nd ed. John Wiley & Sons.

Cressie, Noel. 1993. *Statistics for spatial data*. John Wiley & Sons.

Duchon, J. 1977. *Construction Theory of Functions of Several Variables*. Berlin: Springer chapter Splines minimizing rotation-invariant semi-norms in Solobev spaces.

Goovaerts, Pierre. 1997. *Geostatistics for natural resources evaluation*. Oxford University Press on Demand.

Gotway, Carol A and Linda J Young. 2007. "A geostatistical approach to linking geographically aggregated data from different sources." *Journal of Computational and Graphical Statistics* 16(1):115–135.

Kollman, Ken, Allen Hicken, Daniele Caramani, David Backer and David Lublin. 2022. *Constituency-Level Elections Archive*. Ann Arbor, MI: Center for Political Studies, University of Michigan.

Kollman, Ken, Allen Hicken, Daniele Caramani, David Backer, David Lublin, Joel Selway and Fabricio Vasselai. 2017. *GeoReferenced Electoral Districts Datasets (Beta)*. Ann Arbor, MI: Center for Political Studies, University of Michigan.

Rosner, Bernard. 1983. "Percentage points for a generalized ESD many-outlier procedure." *Technometrics* 25(2):165–172.

Wood, Simon N. 2003. "Thin plate regression splines." *Journal of the Royal Statistical Society, Series B* 65(1):95–114.

Young, Linda J, Carol A Gotway, Greg Kearney and Chris DuClos. 2009. "Assessing uncertainty in support-adjusted spatial misalignment problems." *Communications in StatisticsTheory and Methods* 38(16-17):3249–3264.

Zhukov, Yuri M, Christian Davenport and Nadiya Kostyuk. 2019. "Introducing xSub: A New Portal for Cross-National Data on Sub-National Violence." *Journal of Peace Research* 56(4):604–614.