API-231 / GIS-PubPol
Meeting 12 (Changes of Geographic Support)

Yuri M. Zhukov
Visiting Associate Professor of Public Policy
Harvard Kennedy School

March 5, 2024

Motivation: theoretically relevant units $\neq$ spatial units at which data are available

Example: data for different variables are available at different units

Figure 1: Outcome

Figure 2: Treatment

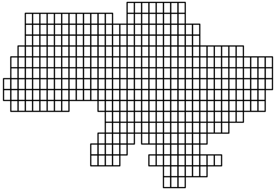


Figure 3: Instrument

Example: borders, number of units change over time

Figure 4: 1937

Figure 5: 1945

Figure 6: 1991

Example: data are measured at different levels of geographic precision

Figure 7: admin 0

Figure 8: admin 1

Figure 9: admin 2

Example: different definitions of same units across data sources

Figure 10: admin 2

Figure 11: "admin 2"

Figure 12: admin 2

The dilemma for analysts

1. Conduct analysis at theoretically inappropriate units
   - this is only possible if all data are available for those same units

or

2. Convert the data to a common set of (more appropriate) units
   - this is an intermediate, messy step
   - it *always* entails some information loss
   - it can lead to measurement error and biased estimation of quantities of interest
   - problem is well-known in geostatistics and social science
   - but no best practices exist for implementation, comparison, evaluation

Changes of support

Definitions

What are change of support problems?

1. *Geographic support*: area, shape, size, and orientation associated with a variable's spatial measurement
2. *Change of support (CoS) problem*: making statistical inferences about a variable at one support by using data from a different support

Related topics:

- ecological inference (EI): deducing micro variation from aggregate data
- modifiable areal unit problem (MAUP): statistical inferences depend on the geographical regions at which data are observed

EI and MAUP are both special cases of CoS problems

The complexity of a CoS depends on

1. Relative scale: aggregation, disaggregation, hybrid
2. Relative nesting: whether one set of units falls completely, neatly inside other

Nesting and scale

Illustration

Let's consider three sets of units (from the U.S. state of Georgia)

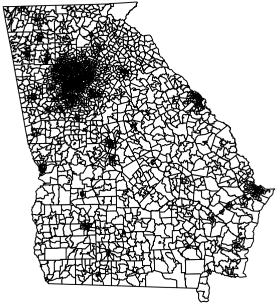


Figure 13: precincts

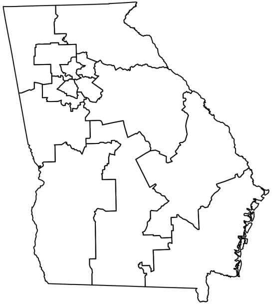


Figure 14: constituencies

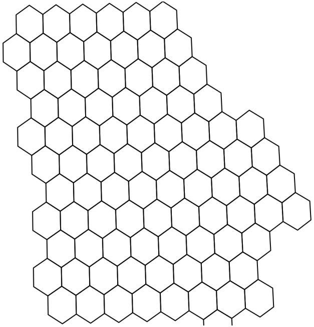


Figure 15: .5° grid

1. Suppose one wants to change the support from *precincts to constituencies*
   - scale: are source units smaller or larger than destination units?
   - nesting: do source units fit completely/neatly into destination units?

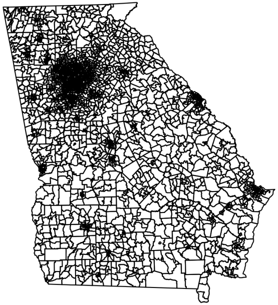


Figure 16: source units

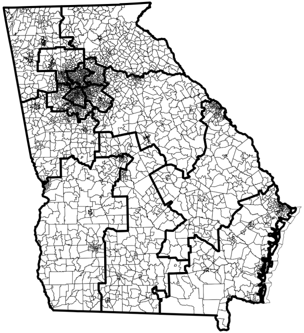


Figure 17: source ∩ destination

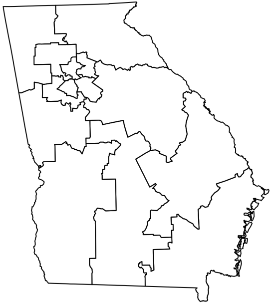


Figure 18: destination units

2. Suppose one wants to change the support from *constituencies to grid cells*
   - scale: are source units smaller or larger than destination units?
   - nesting: do source units fit completely/neatly into destination units?

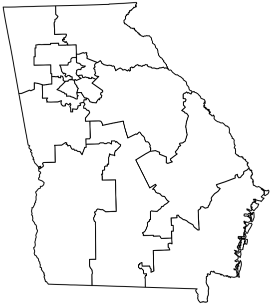


Figure 19: source units

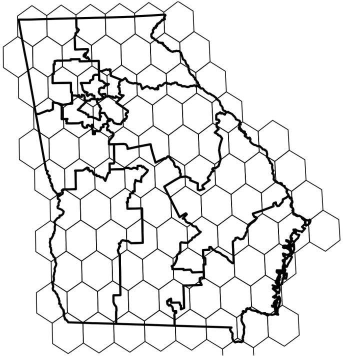


Figure 20: source ∩ destination

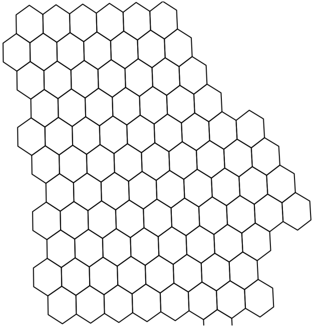


Figure 21: destination units

1. Change of support #1 looks like an aggregation of nested units
2. Change of support #2 looks like (mostly?) disaggregation of non-nested units

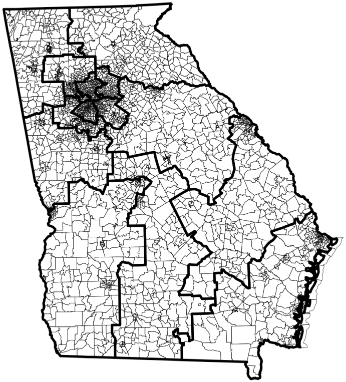


Figure 22: precinct → constituency

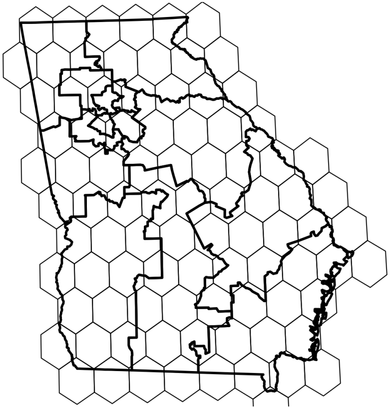


Figure 23: constituency → grid

Some considerations

- many CoS problems require both aggregation and disaggregation
- just because units are politically nested doesn't mean they are geometrically nested (e.g. measurement error, imprecision of boundaries)
- not always easy to "eyeball" these things
- to get a better read on this, we need quantitative measures

Figure 24: Guesstimation ain't easy

Informally

relative scale:

- share of intersections where source units smaller than destination units

relative nesting:

- share of source units that cannot be split across destination units

Formally

- $\mathcal{G}_S$: set of source polygons, indexed $i = 1, \dots, N_S$
- $\mathcal{G}_D$: set of destination polygons, indexed $j = 1, \dots, N_D$
- $\mathcal{G}_{S \cap D}$: intersection of $\mathcal{G}_S$ & $\mathcal{G}_D$, indexed $i \cap j = 1, \dots, N_{S \cap D}$
- a_i: area of source polygon i; a_j: area of destination polygon j
- $a_{i \cap j}$: area of intersection $i \cap j$

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

- values of 1 = aggregation; values of 0 = disaggregation; 0-1 = hybrid

relative nesting: $RN = \frac{1}{N_S} \sum_i^{N_S} \sum_j^{N_D} \left(\frac{a_{i \cap j}}{a_i} \right)^2$

- values of 1 = full nesting; values of 0 = no nesting; 0-1 = partial nesting

Application of relative scale and nesting to Georgia data: any surprises here?

Relative scale

source → destination	(a)	(b)	(c)
(a) precincts	–	1.00	1.00
(b) constituencies	0.00	–	0.12
(c) .5° grid	0.00	0.89	–

Relative nesting

source → destination	(a)	(b)	(c)
(a) precincts	–	0.98	0.92
(b) constituencies	0.01	–	0.29
(c) .5° grid	0.05	0.54	–

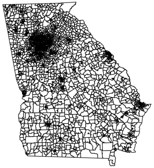


Figure 25: (a)

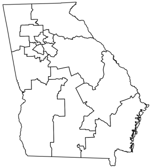


Figure 26: (b)

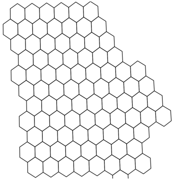


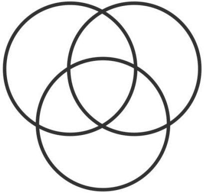
Figure 27: (c)

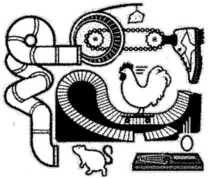
Change of support algorithms

A **CoS algorithm** specifies a transformation between source and destination units

- x: is a variable being transformed from support $\mathcal{G}_S$ to $\mathcal{G}_D$
- $x_{\mathcal{G}D}$: is true value of variable x in destination units $\mathcal{G}_D$
- $\widehat{x_{\mathcal{G}D}}^{(k)} = f_k(x_{\mathcal{G}S})$: estimated value of $x_{\mathcal{G}D}$, calculated w/ CoS algorithm k

these range from simple geometric operations to complex model-based predictions





Types of variables

1. Extensive (depend on area and scale)
   - aggregates are (weighted) sums
   - must satisfy the pycnophylactic (mass-preserving) property:
     - if area is split or combined, its values must be split or combined
     - sum of values in destination units must equal sum in source units
   - examples: population counts, event counts, acreage, mineral deposits
2. Intensive (don't depend on area and scale)
   - aggregates are (weighted) means
   - examples: population density, vote margins, median income
   - intensive variables are often functions of extensive variables (density = mass/vol.)
   - best practice: reconstruct in destination units from transformed components ($\widehat{\text{mass}}_{\mathcal{GD}}/\widehat{\text{volume}}_{\mathcal{GD}} = \widehat{\text{density}}_{\mathcal{GD}}$)

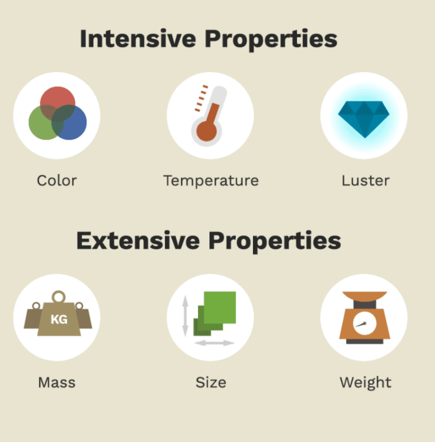


Figure 28: Examples

Areal interpolation

Areal weighting is the default CoS method in many commercial and open-source GIS

1. Advantages
   - easy to implement
   - requires information only on geometry of source and destination units
   - no need for ancillary data
2. Disadvantages
   - assumes that the phenomenon of interest is uniformly distributed in source units
   - this becomes less problematic if source units are relatively small
   - but more problematic as source units increase in size

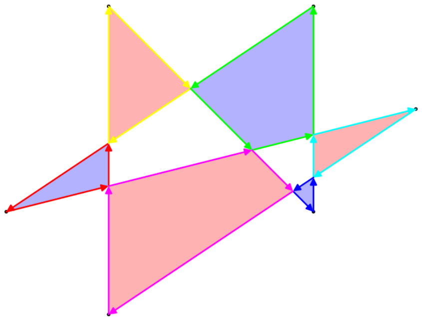
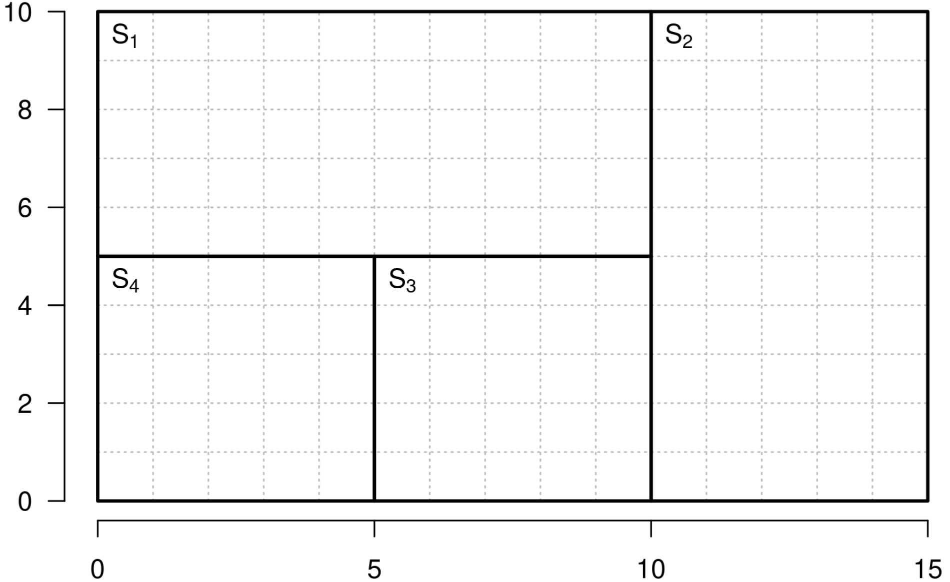
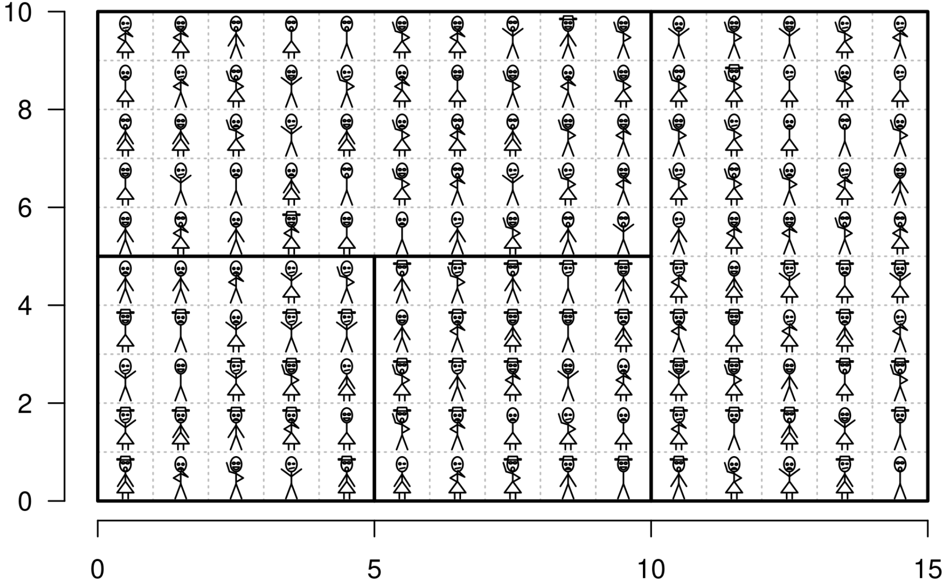


Figure 29: Overlapping areas

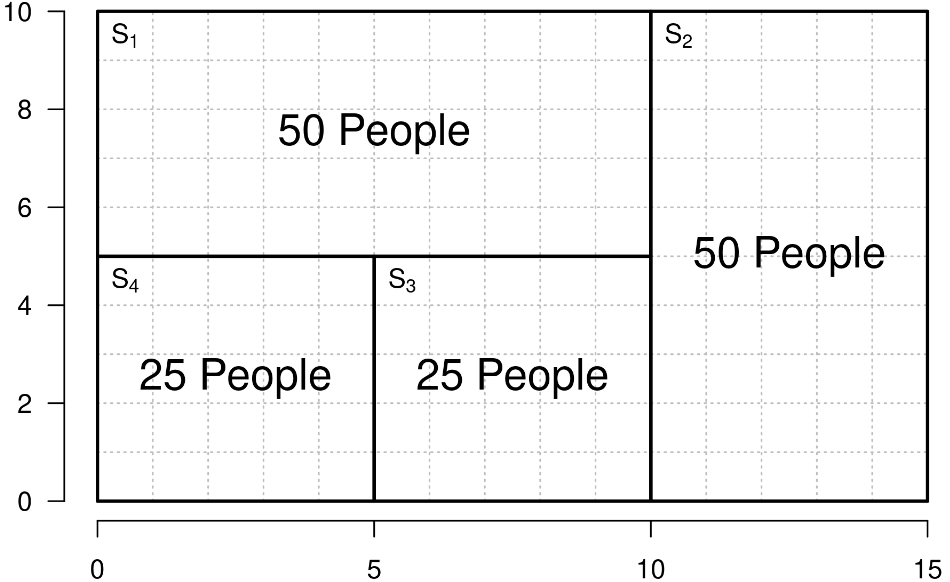
Illustration: suppose a city is divided into 4 sectors: S_1, S_2, S_3, S_4



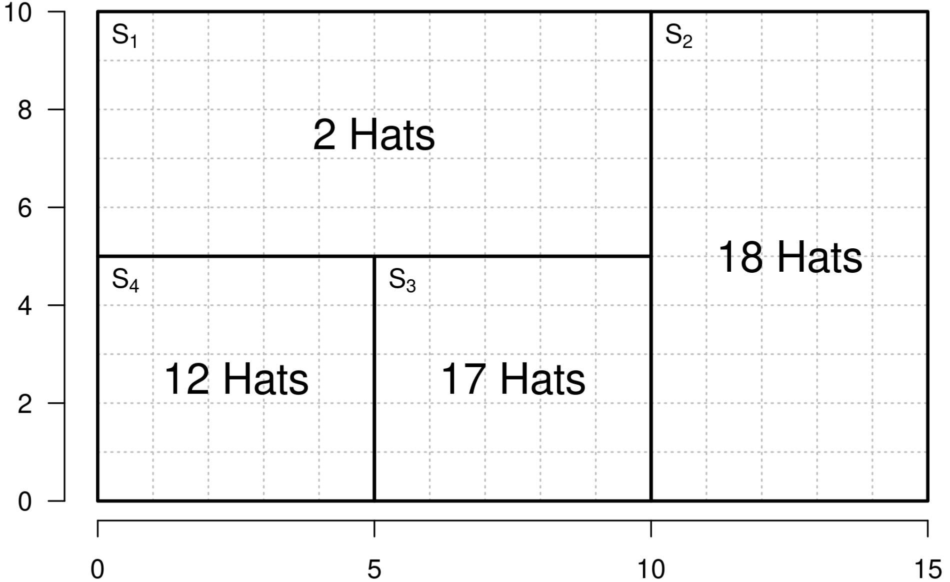
The city's population ($N = 100$) is distributed across the 4 sectors. 49% wear hats.



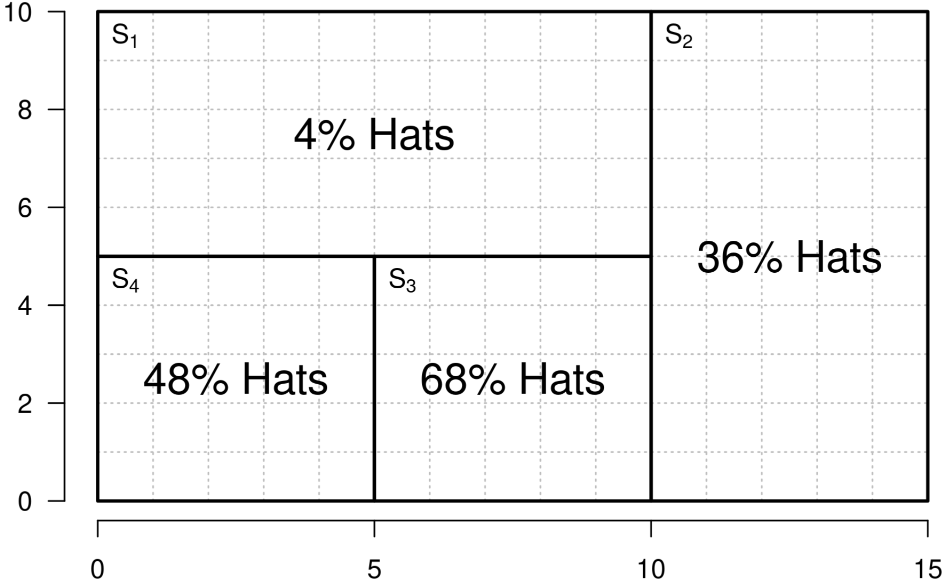
But we don't actually have micro data on where people live, just regional totals.



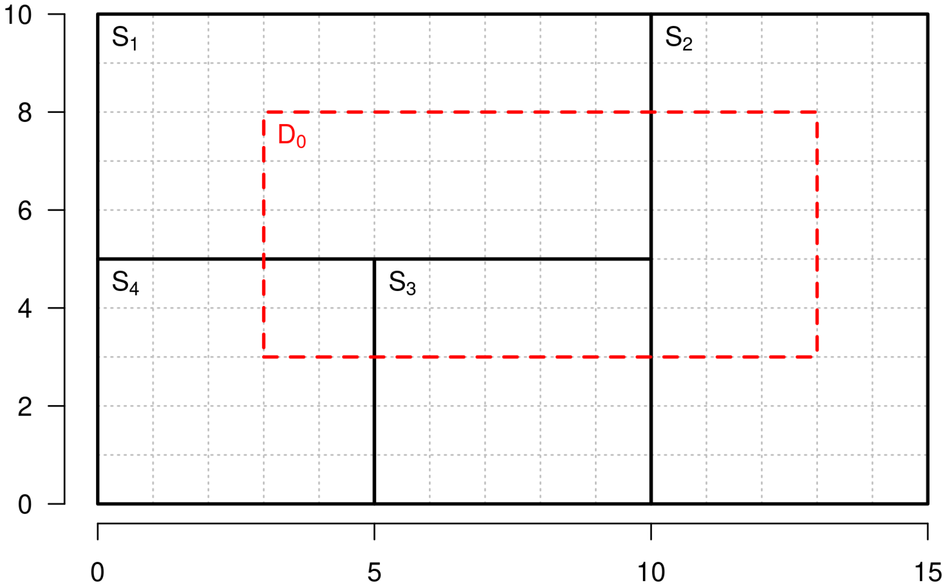
We know how many people live in each sector, and how many of them wear hats.



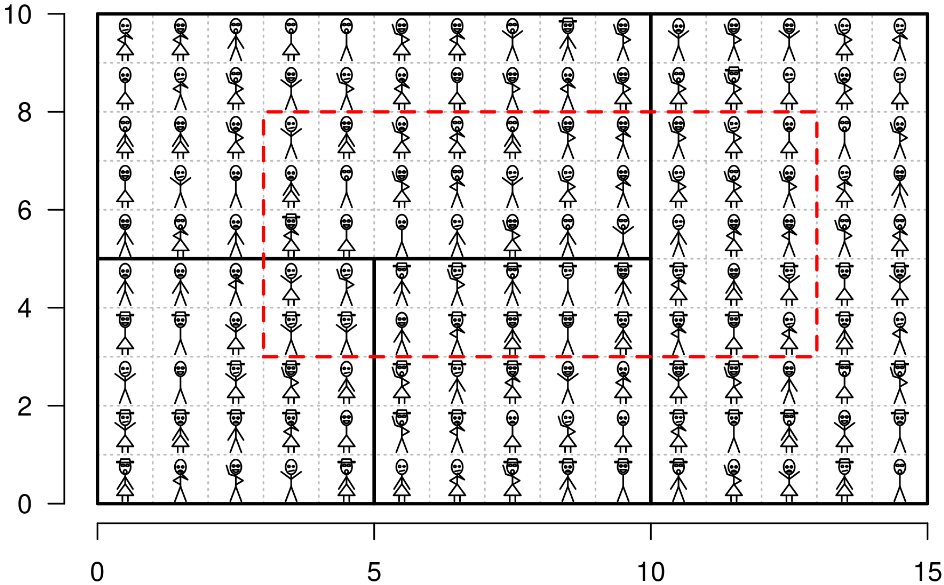
From this, we know that S_1 has a much lower share of hat wearers than S_2, S_3, S_4.



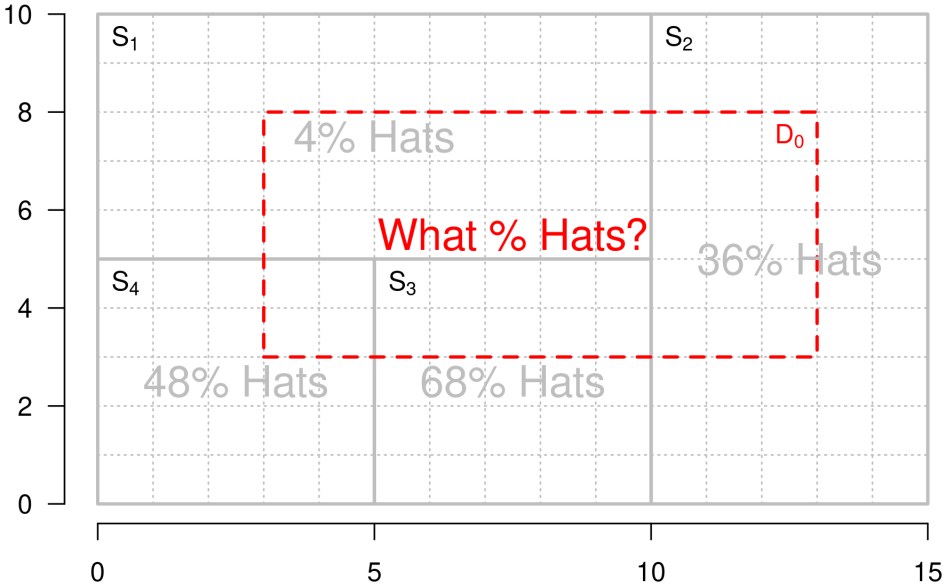
Due to redistricting, a city council member's district has switched from S_1 to D_0.



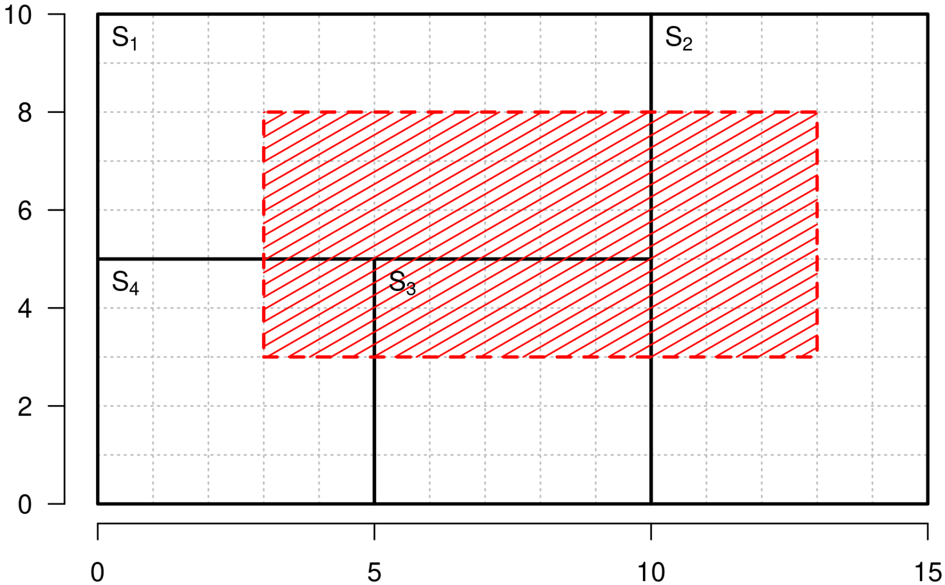
With micro data, you can count how many people are in D_0, and what % wear hats.



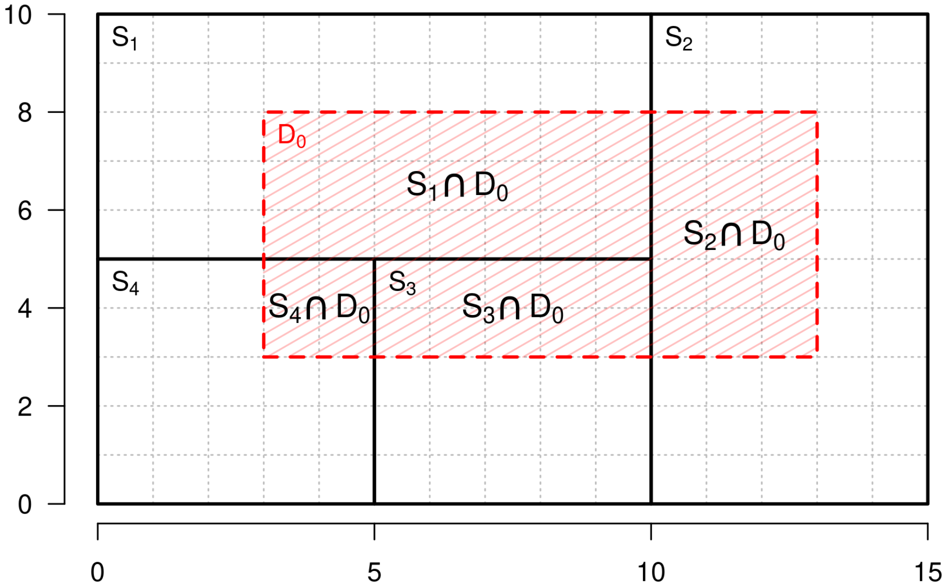
Without micro data, you have to estimate this from aggregate statistics. But how?



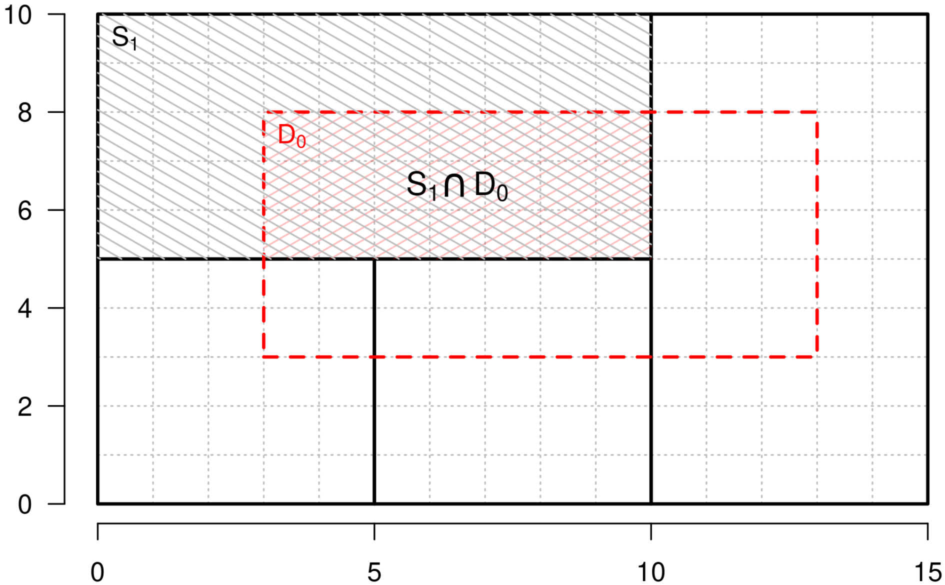
Let's think about what the area of the new region in D_0 actually represents.



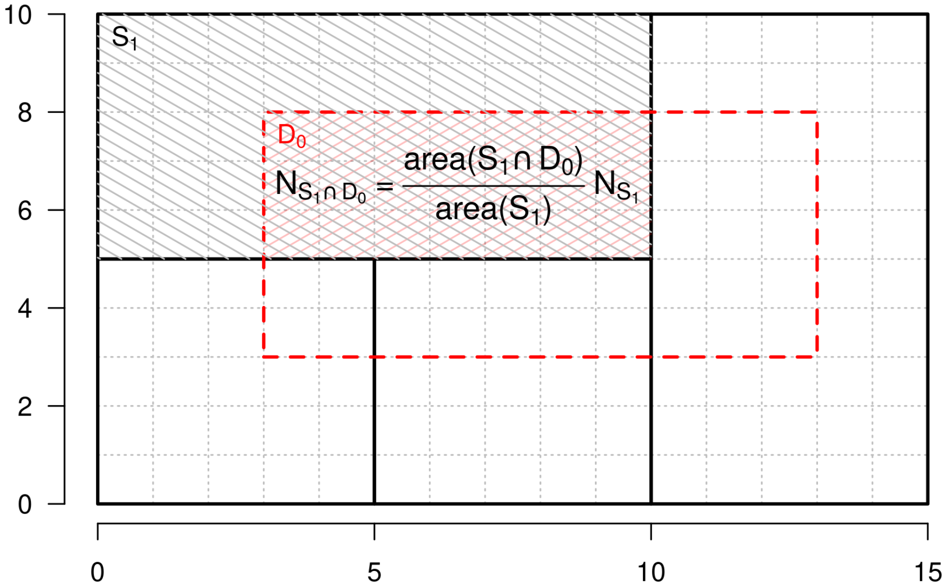
This polygon is a combination of four intersections of $S_1, \ldots, S_4$ with D_0.



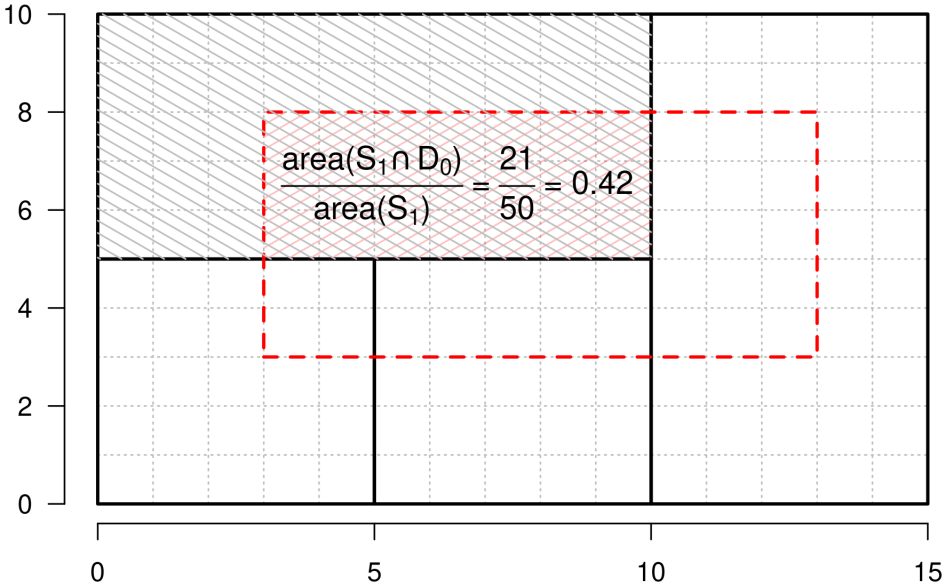
The number of people living in intersection $S_1 \cap D_0$ is a subset of those living in S_1.



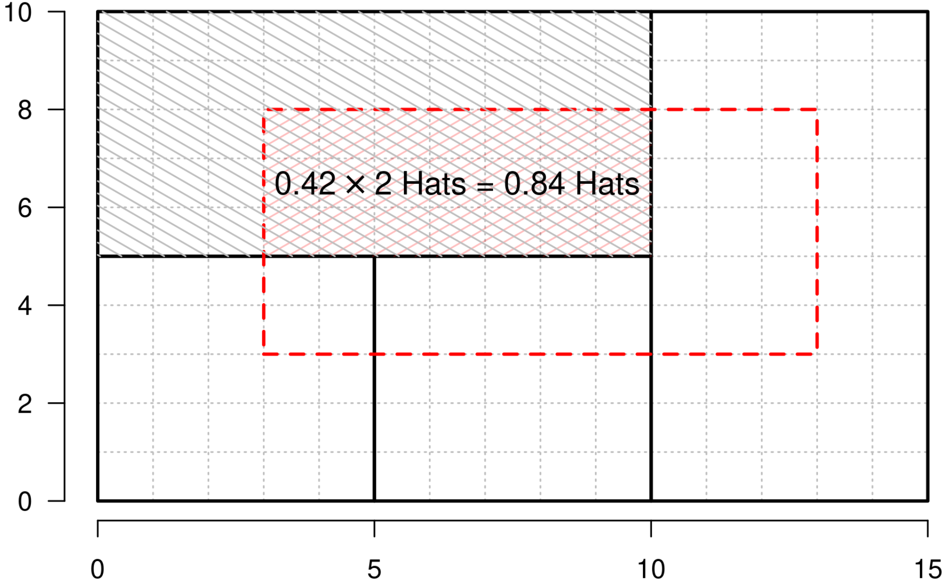
Let's assume that pop size $N_{S_1 \cap D_0}$ is proportional to relative area of $S_1 \cap D_0$ vs S_1.



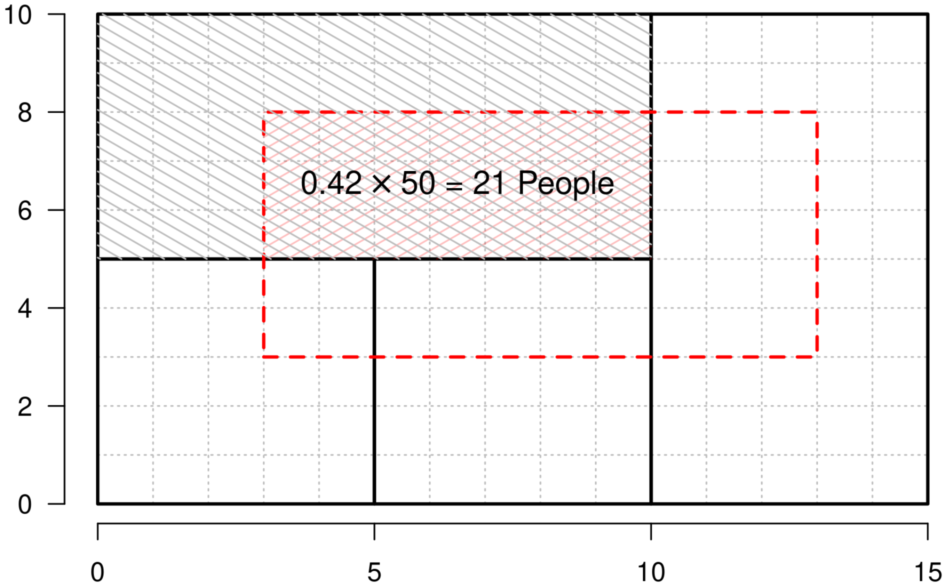
From the map, we see that area$(S_1 \cap D_0) = 3 \times 7 = 21$ and area$(S_1) = 5 \times 10 = 50$.



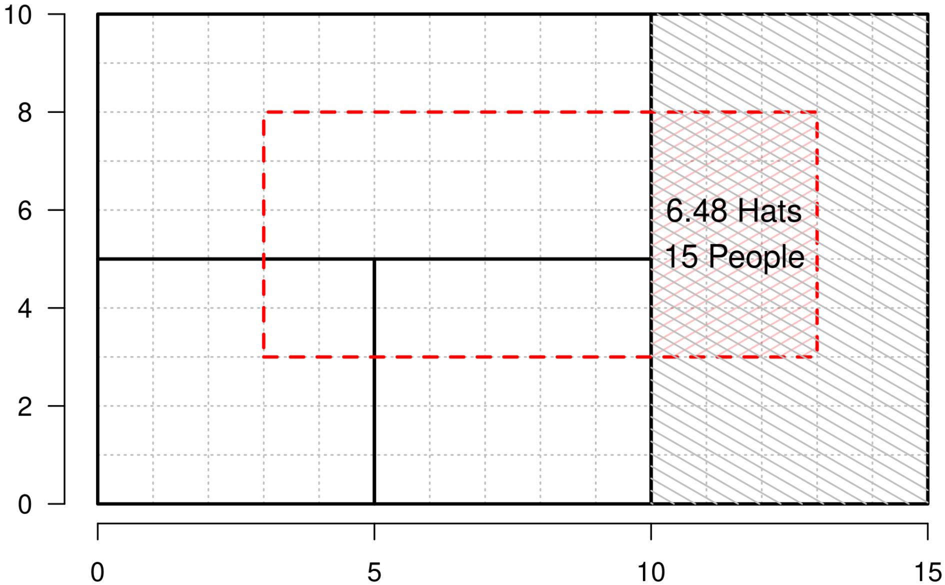
Multiply this "area weight" by the number of hats in S_1 to get subtotal for $S_1 \cap D_0$.



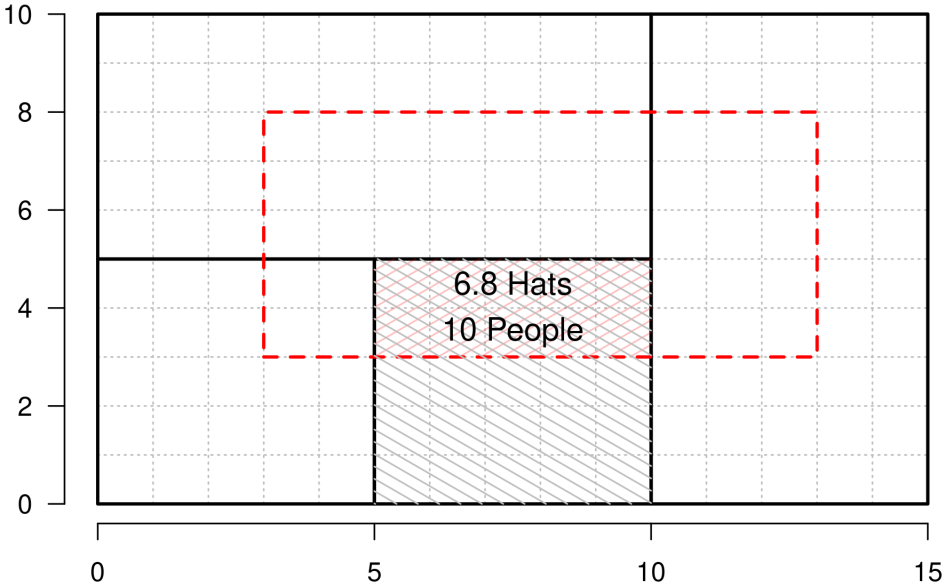
Multiply "area weight" by number of people in S_1 to get sub-population of $S_1 \cap D_0$.



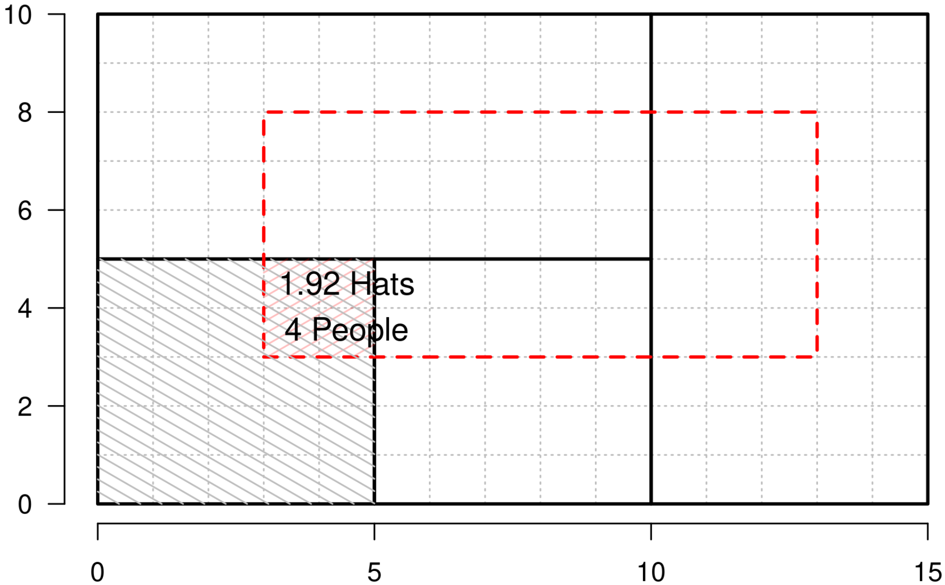
Repeat exercise for $S_2 \cap D_0$: $\frac{15}{50} \times 18$ Hats $= 6.48$ Hats, $\frac{15}{50} \times 50$ People $= 15$ People



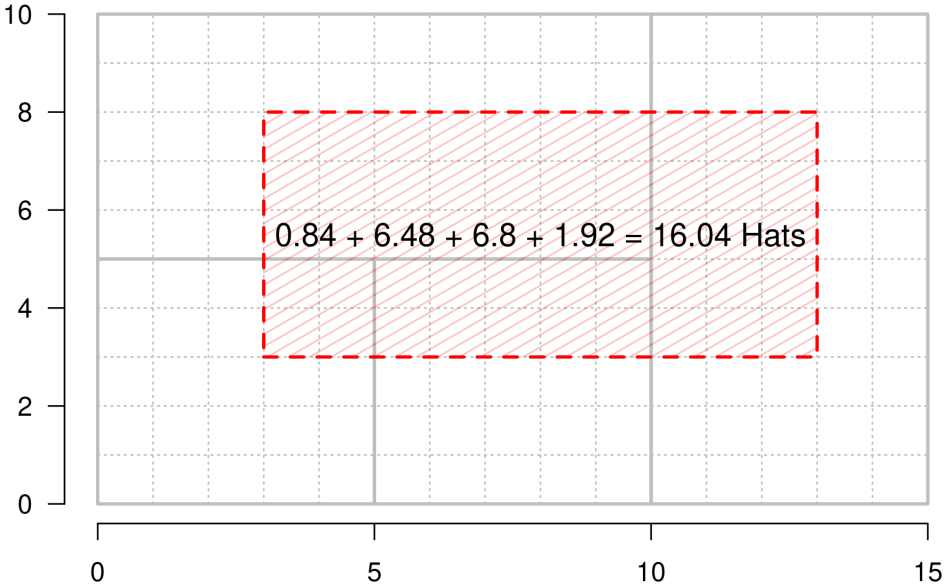
Repeat exercise for $S_3 \cap D_0$: $\frac{10}{25} \times 17$ Hats $= 6.8$ Hats, $\frac{10}{25} \times 25$ People $= 10$ People



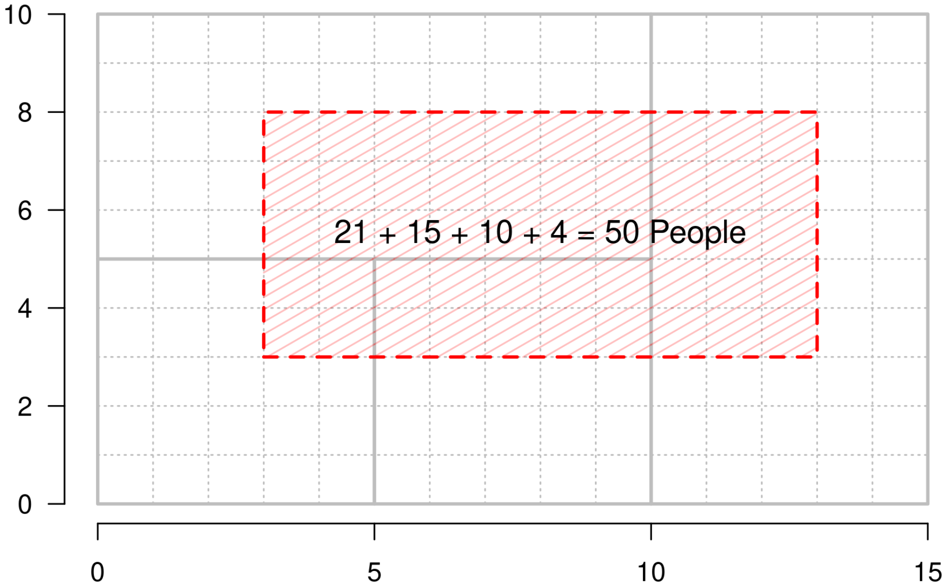
Repeat exercise for $S_4 \cap D_0$: $\frac{4}{25} \times 12$ Hats $= 1.92$ Hats, $\frac{4}{25} \times 25$ People $= 4$ People



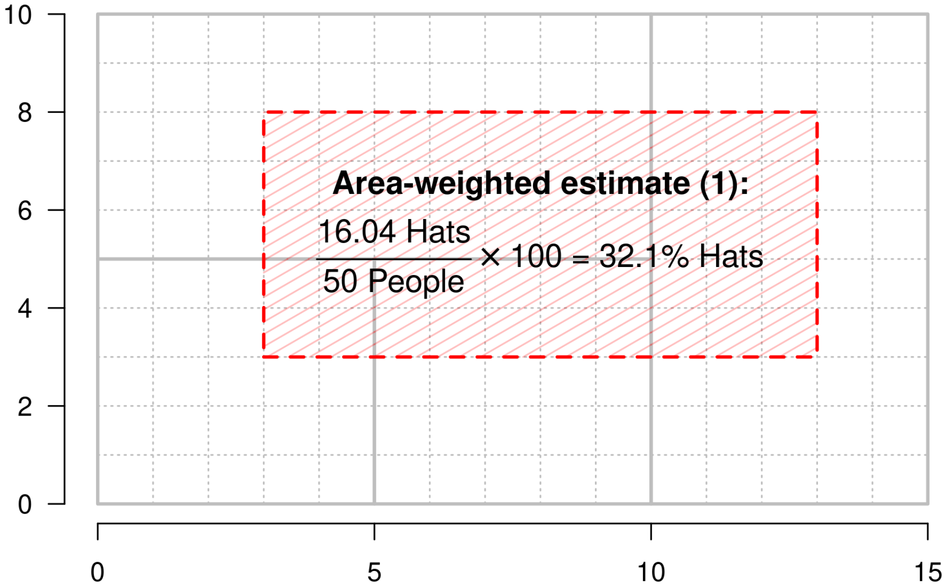
Combine the four subtotals into an area-weighted estimate of hats for all of D_0.



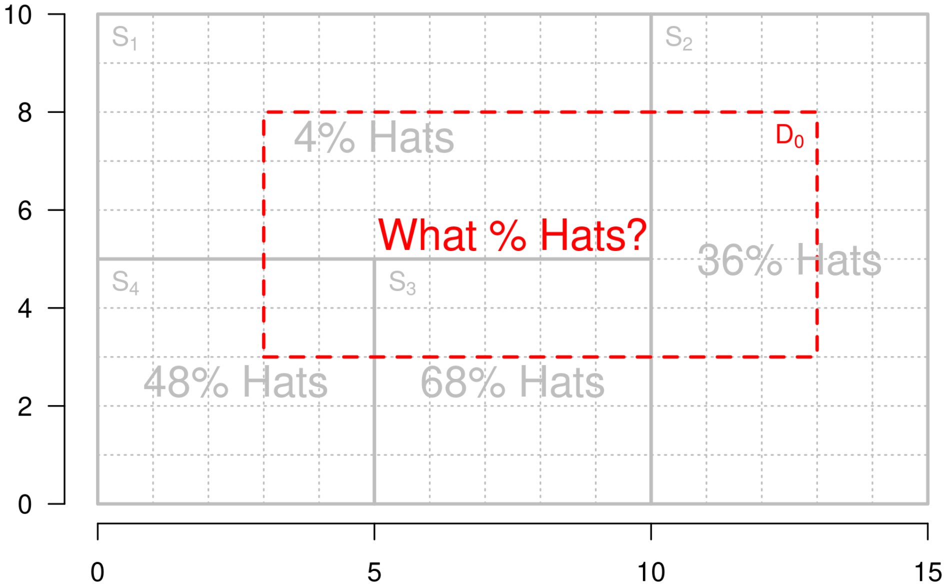
Combine the four subtotals into an area-weighted population estimate for all of D_0.



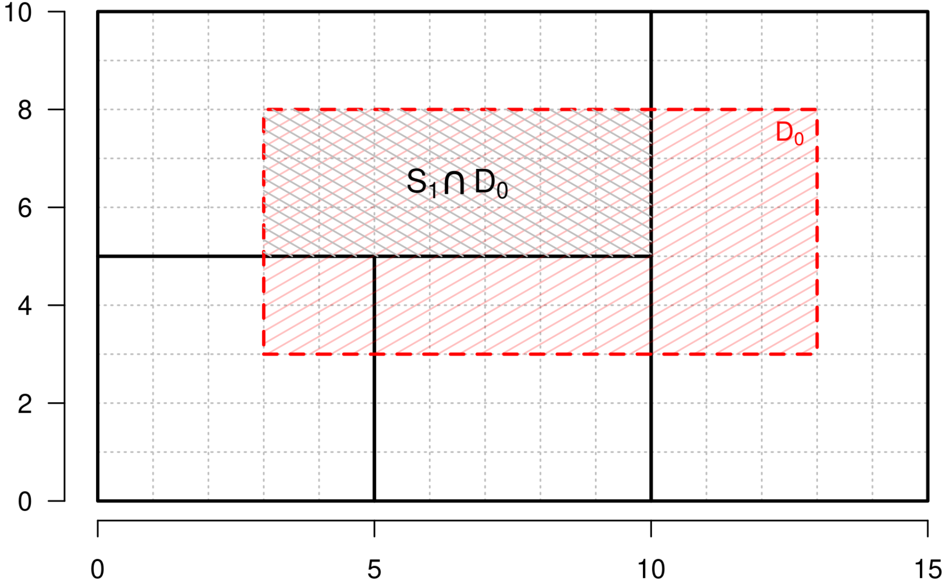
Divide weighted # of hats by weighted population to get "% Hats" estimate for D_0.



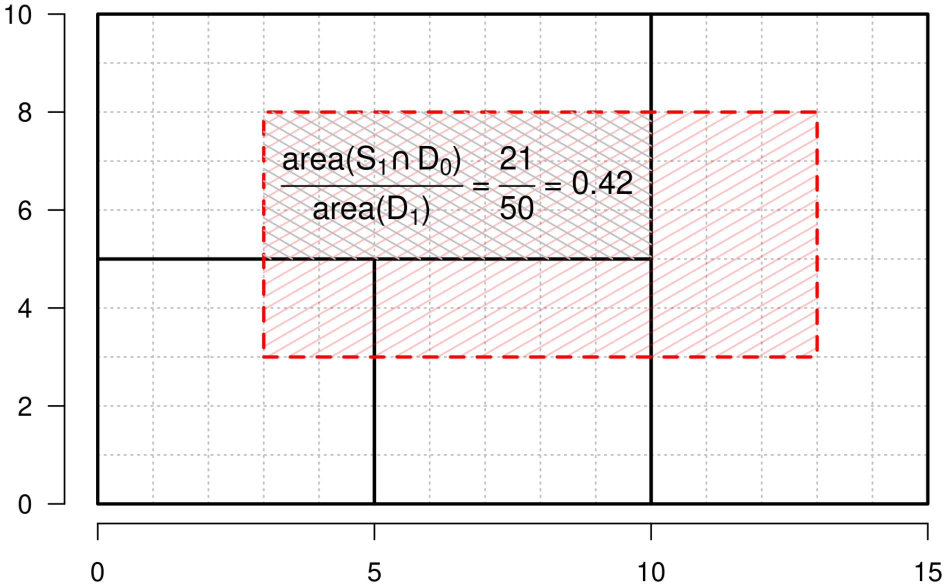
Can't we interpolate %'s directly, instead of nominator and denominator separately?



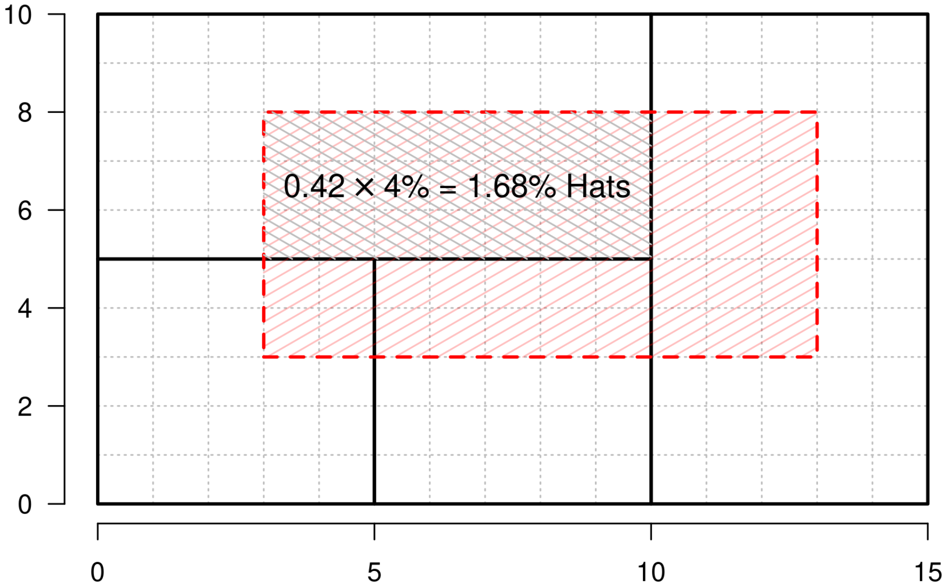
Yes, but the weights would be different: $\frac{\text{area}(S_1 \cap D_0)}{\text{area}(D_0)}$, proportional to destination D_0.



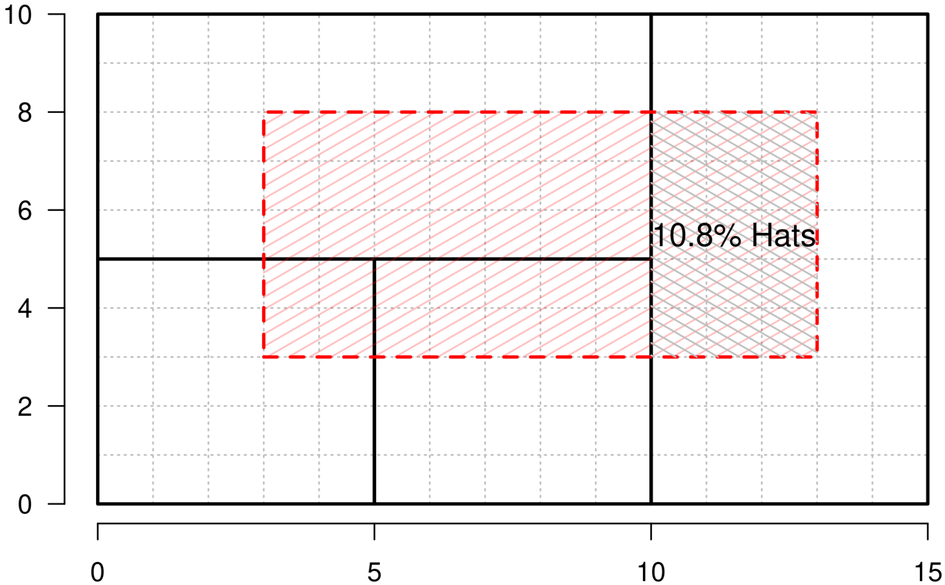
The area of $S_1 \cap D_0$ is $3 \times 7 = 21$, and area$(D_0) = 5 \times 10 = 50$, so $w = 0.42$ again.



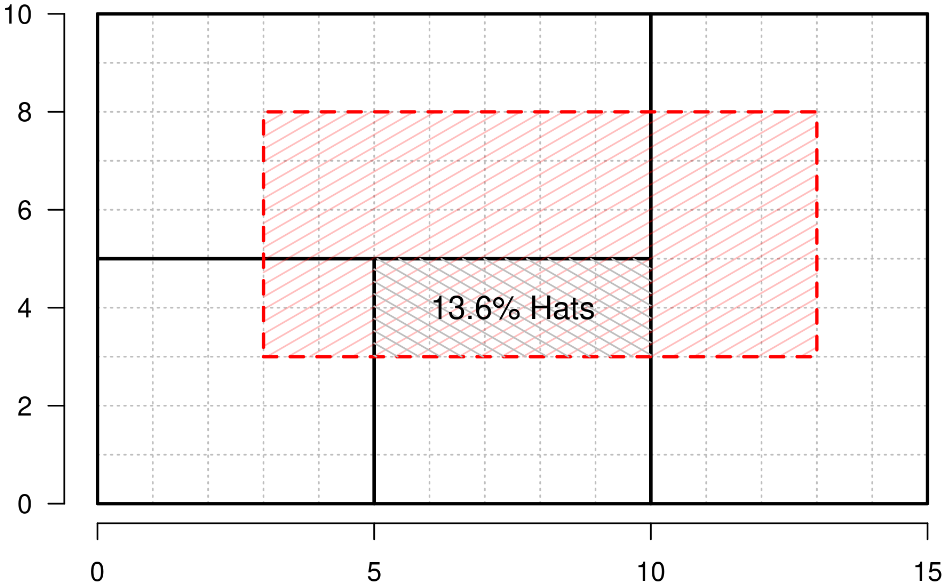
Multiplying the weight by "% Hats" in S_1, we get $\frac{21}{50} \times 4\% = 1.68\%$ Hats.



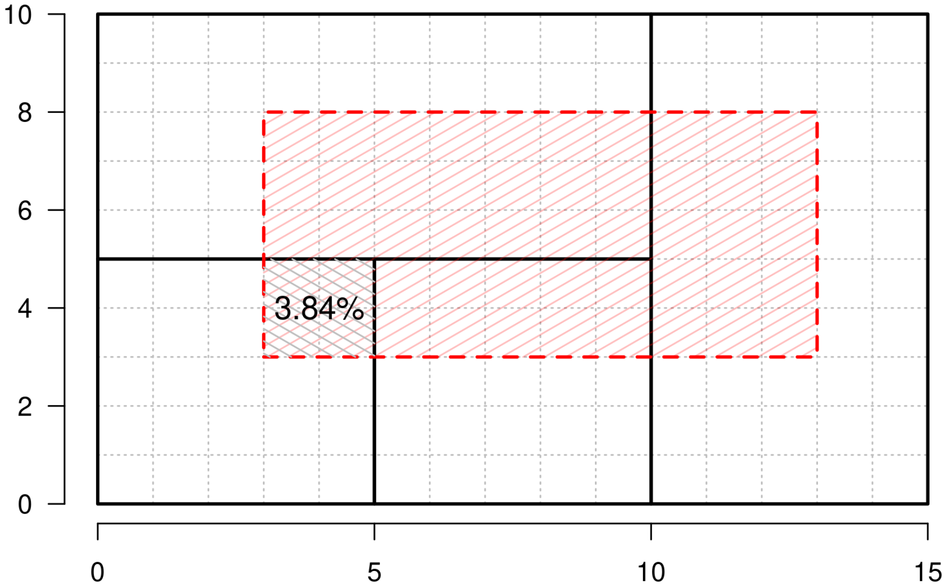
Repeat for $S_2 \cap D_0$: area weight $\frac{15}{50} \times 36\%$ Hats in $S_2 = 10.8\%$ Hats.



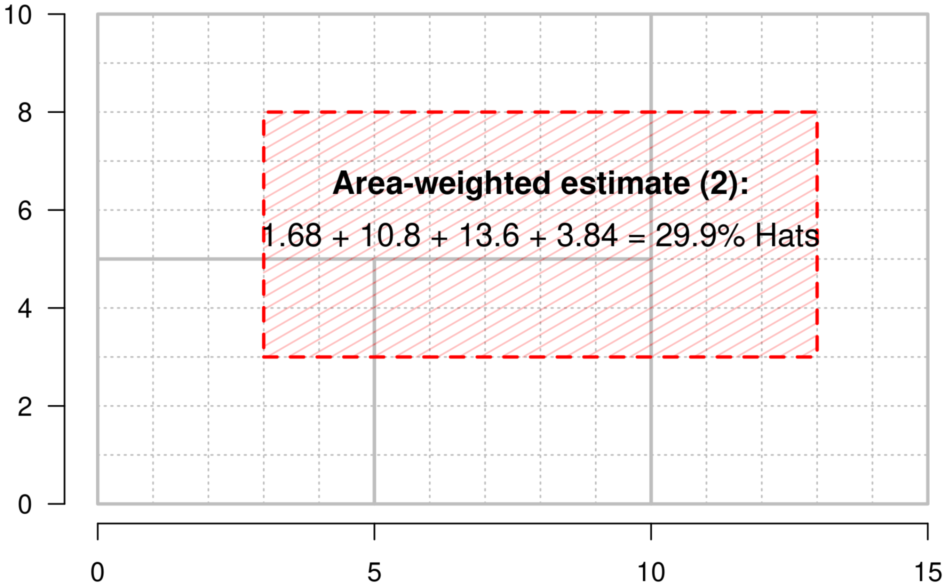
Repeat for $S_3 \cap D_0$: area weight $\frac{10}{50} \times 68\%$ Hats in $S_3 = 13.6\%$ Hats.



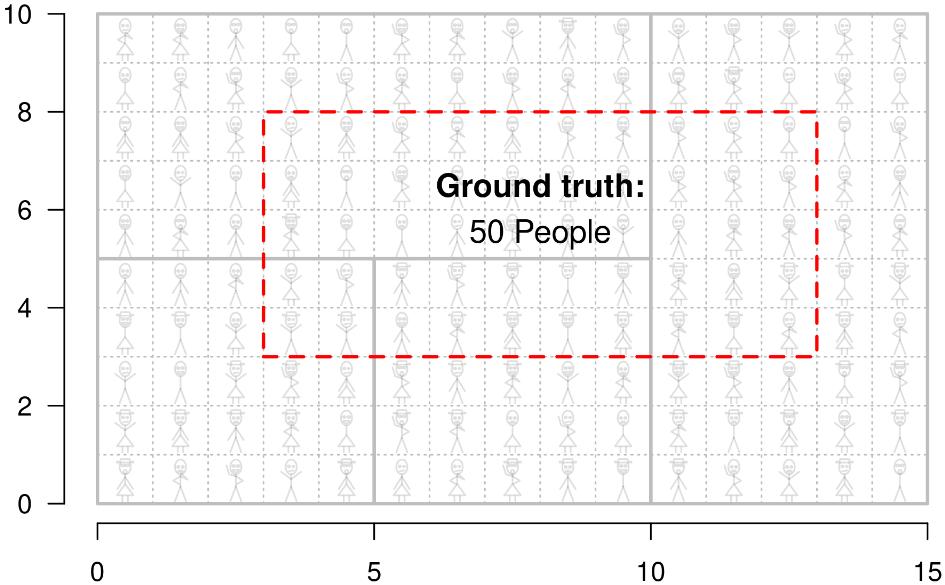
Repeat for $S_4 \cap D_0$: area weight $\frac{4}{50} \times 48\%$ Hats in $S_4 = 3.84\%$ Hats.



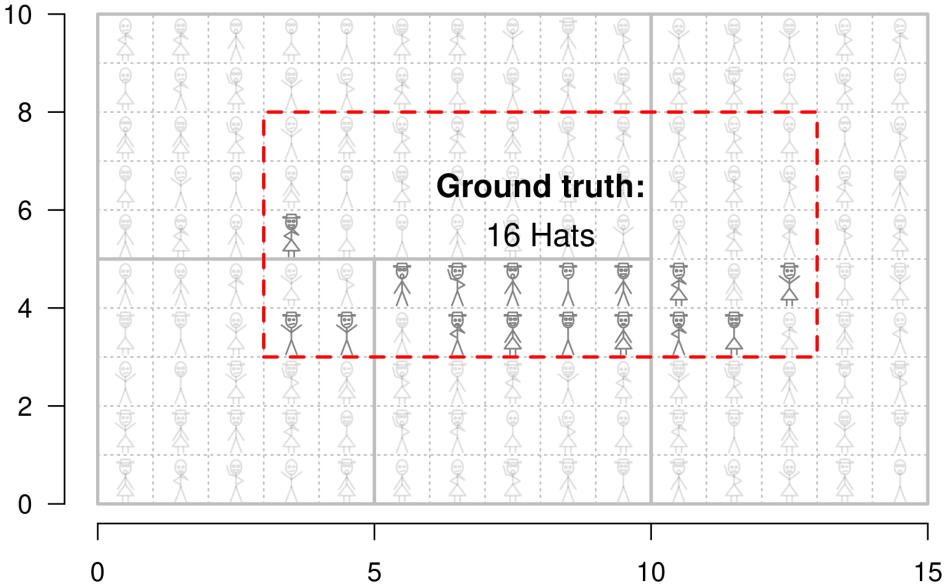
Combine the four subtotals into an area-weighted estimate of "% Hats" for all of D_0.



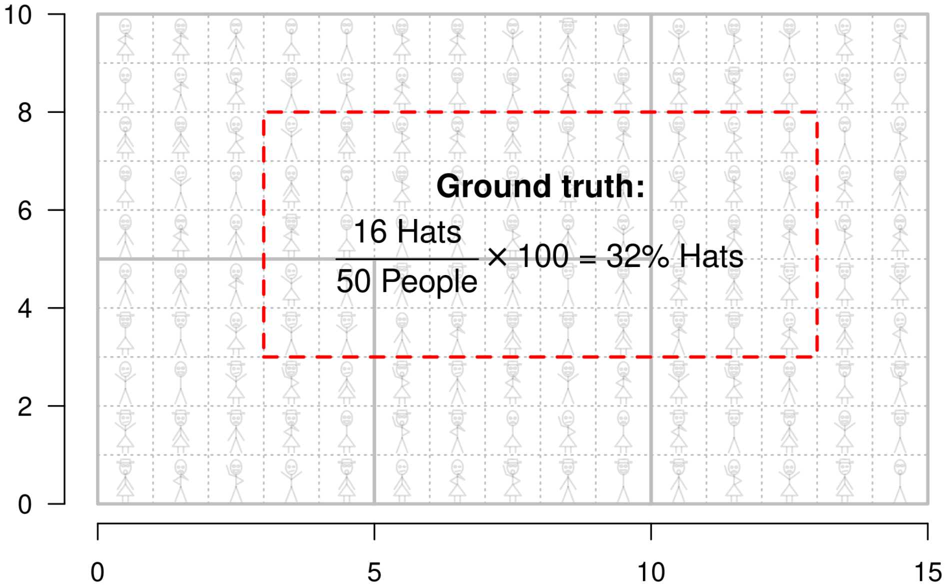
Let's compare these estimates to the ground truth (count how many people in D_0).



Let's compare these estimates to the ground truth (count how many hats in D_0).



Our 1st weighted estimate (32.04%, extensive) is closer than 2nd (29.9%, intensive).



Pseudocode for areal interpolation

1. Intersect $\mathcal{G}_S$ and $\mathcal{G}_D$, creating a third polygon layer $\mathcal{G}_{S \cap D}$,
   - 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.
2. Compute area weights for each intersection $i \cap j$,
   a) for *extensive variables*: $w^{(\text{ext})}_{i \cap j} = \frac{a_{i \cap j}}{a_i}$
   (i.e. share of i's area represented by intersection $i \cap j$)
   b) for *intensive variables*: $w^{(\text{int})}_{i \cap j} = \frac{a_{i \cap j}}{a_j}$
   (i.e. share of j's area contributed by intersection $i \cap j$)
3. Combine weighted statistics for each destination polygon j:
   a) $\hat{x}_j = \sum_{i \cap j}^{N_{\cap j}} w_{i \cap j} x_{i \cap j}$, where $x_{i \cap j}$ is the value of x in intersection $i \cap j$ and $N_{\cap j}$ is the number of intersections in j

Areal interpolation is just one of many potential CoS methods

Examples:

- simple overlay
- population weighted interpolation
- ordinary kriging
- universal kriging
- thin-plate splines and random forests

these differ in their assumptions (e.g. uniformity vs. heterogeneity) and requirements (e.g. ancillary data)

... what's more important is not the choice of CoS algorithm, but the *relative scale and nesting* of source and destination units

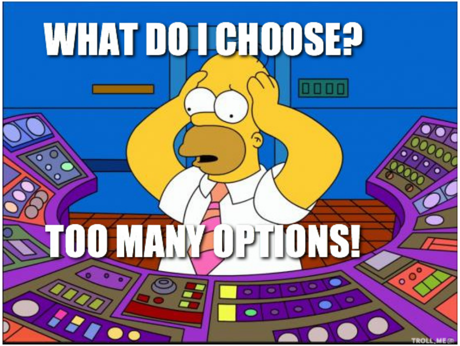
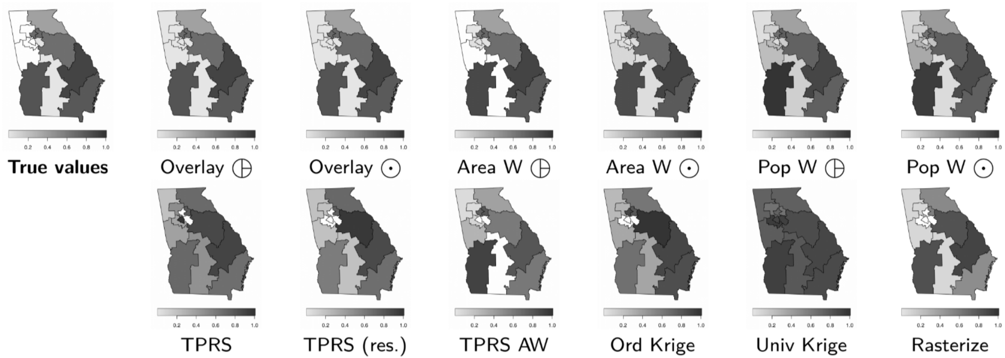


Figure 62: Choice paralysis

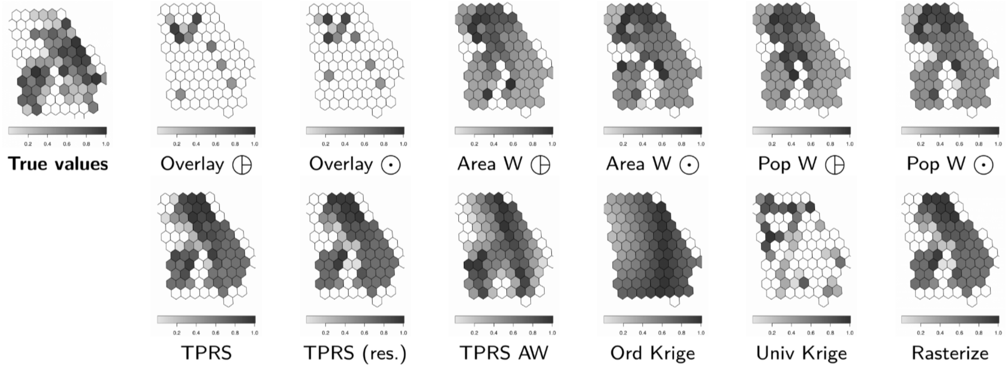
Assessing transformation quality

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



Different CoS algorithms → Different transformed values

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



But how do RS, RN affect the quality of transformations (prediction error, rank correlation, estimation bias), holding CoS algorithm constant?

Higher RS, $RN \rightarrow$ **Lower prediction error** relative to true values

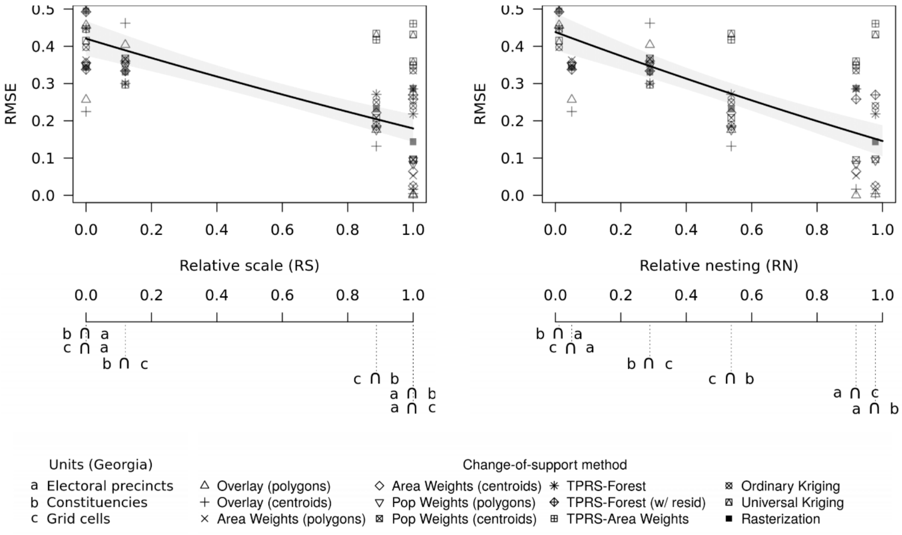


Figure 63: How RN and RS affect root mean squared error

Higher RS, RN → **Higher correlation** b/w transformed values & true values

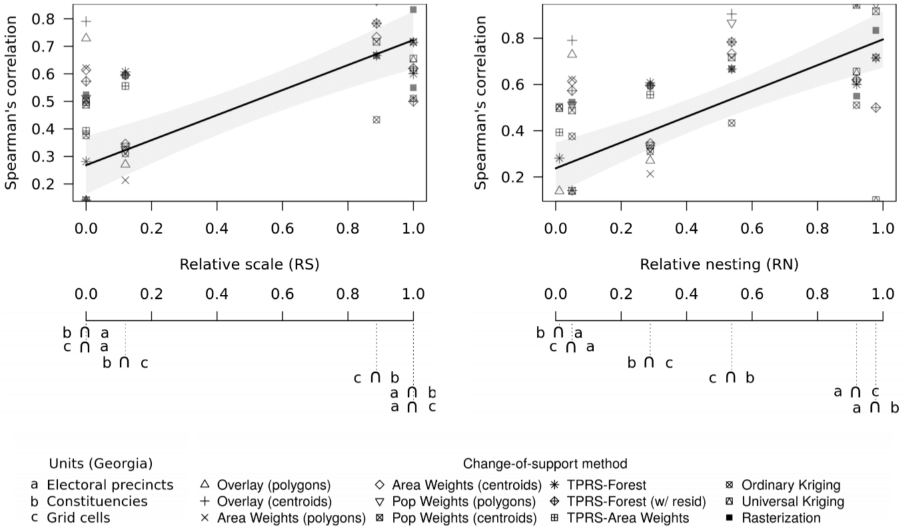


Figure 64: How RN and RS affect correlation

Higher RS, RN $\rightarrow$ **Less bias** in regression coefficients

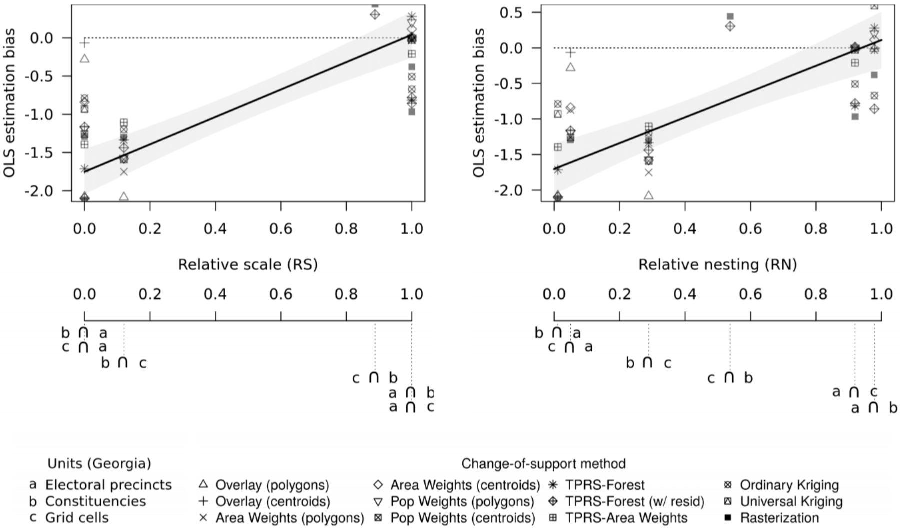


Figure 65: How RN and RS affect OLS estimation bias

What is to be done?

1. General recommendations:
   - consider relative scale and nesting as *ex ante* measures of CoS complexity
   - check face validity of transformed values through visualization
2. If "ground truth" data (micro data, cross-unit IDs) are available:
   - validate transformed values with micro data
   - use micro data as source units
   - match on common ID (if units are well-nested)
3. If "ground truth" data are not available:
   - be transparent about limitations/assumptions
   - partial validation (if micro data available for some regions)
   - report results from alternative CoS algorithms when possible

Bad news: RN and RS can be calculated in R (`SUNGEO::nesting()`), not QGIS
(but you can still do CoS in QGIS, using good judgement and common sense!)