Detecting linguistic variation with geographic sampling - lite version

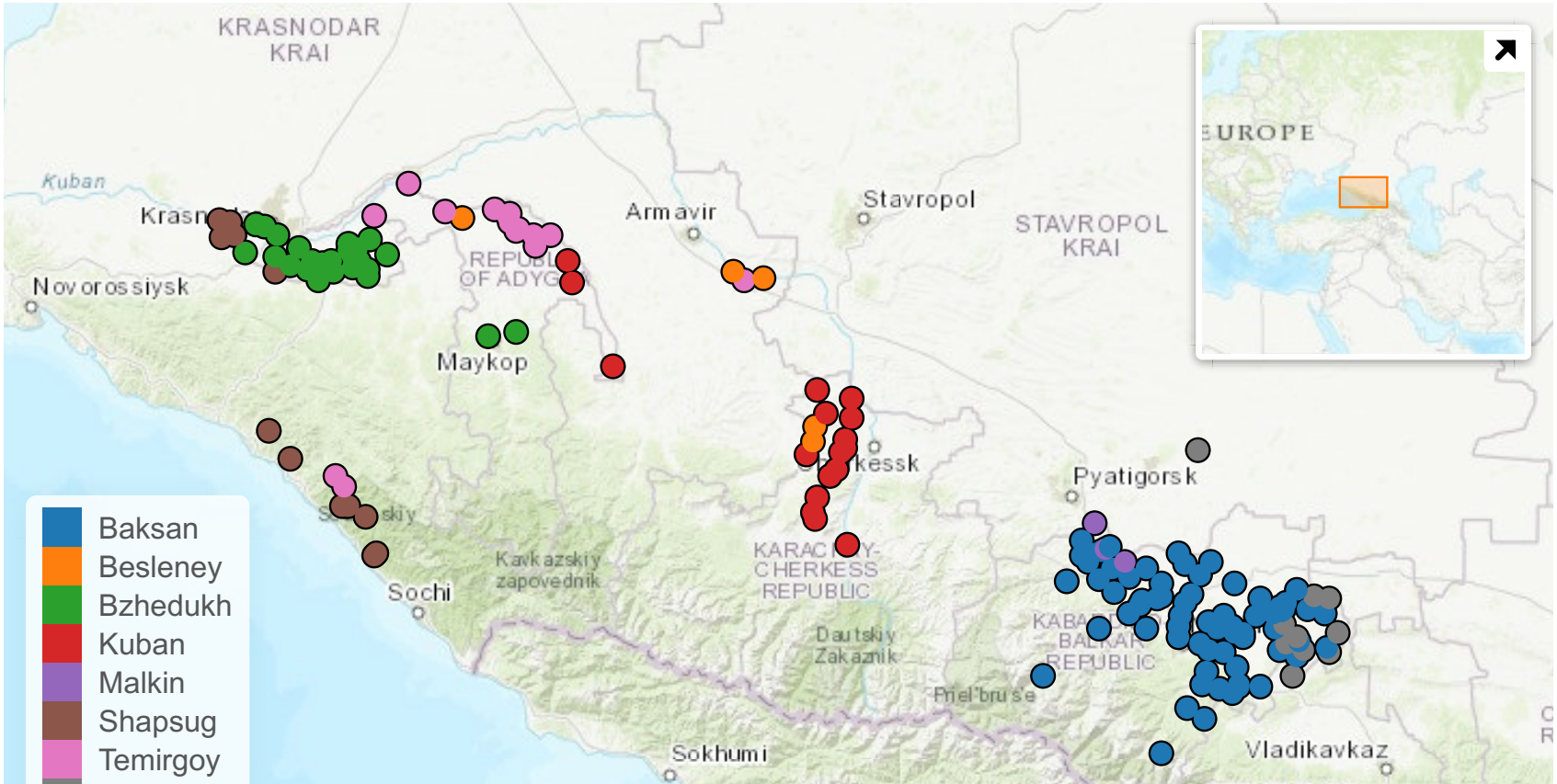
Ezequiel Koile, George Moroz

Linguistic Convergence Laboratory, NRU HSE

26 August 2020, 53rd Annual Meeting of the Societas Linguistica Europaea

The problem

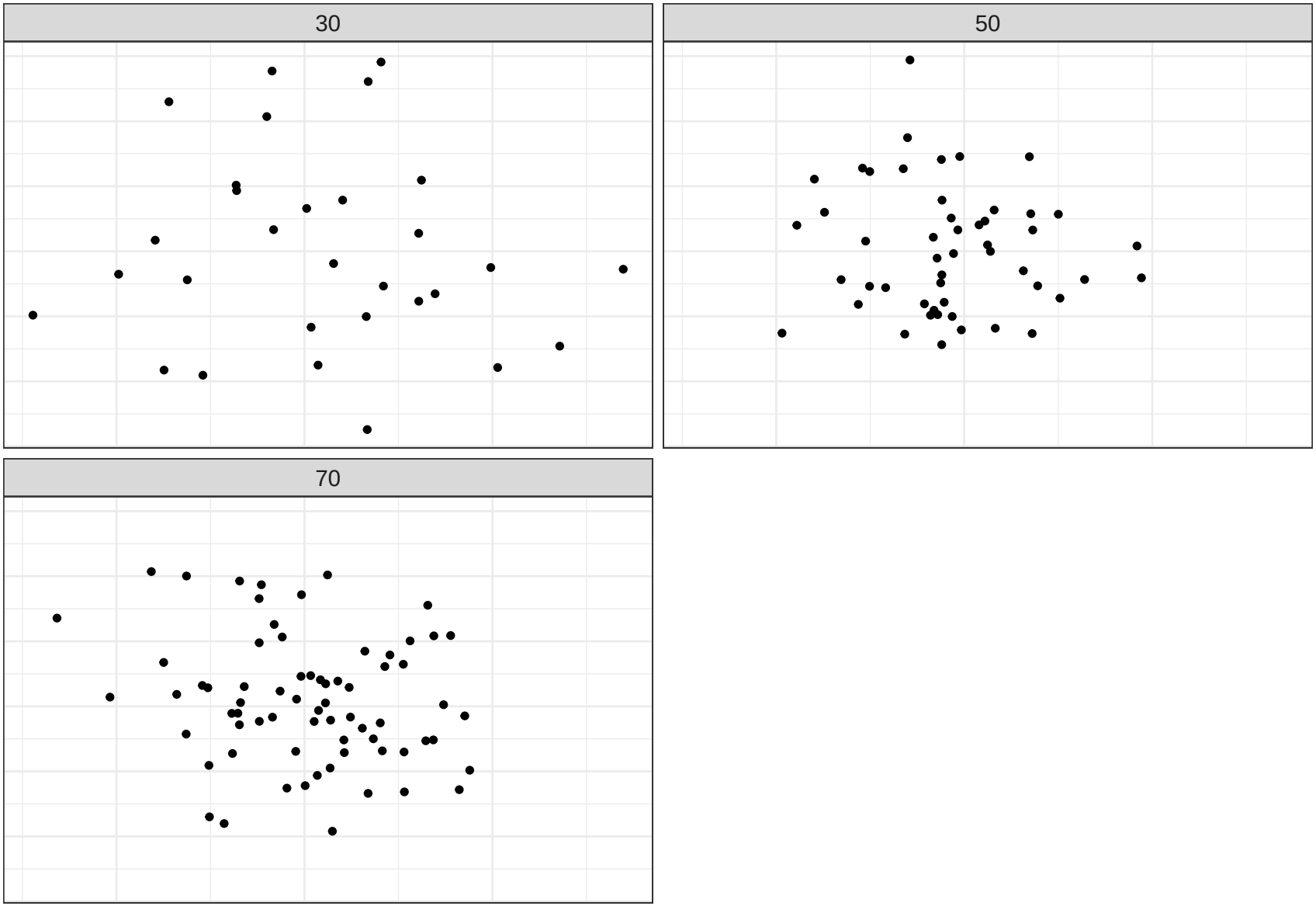
- We are interested in spotting variation of a discrete parameter among the lects spoken in these villages
- It is very impractical to conduct fieldwork in each single village. Therefore, we need to choose a *sample* of locations.
- *Research Question*: How to choose the sample of villages to survey?



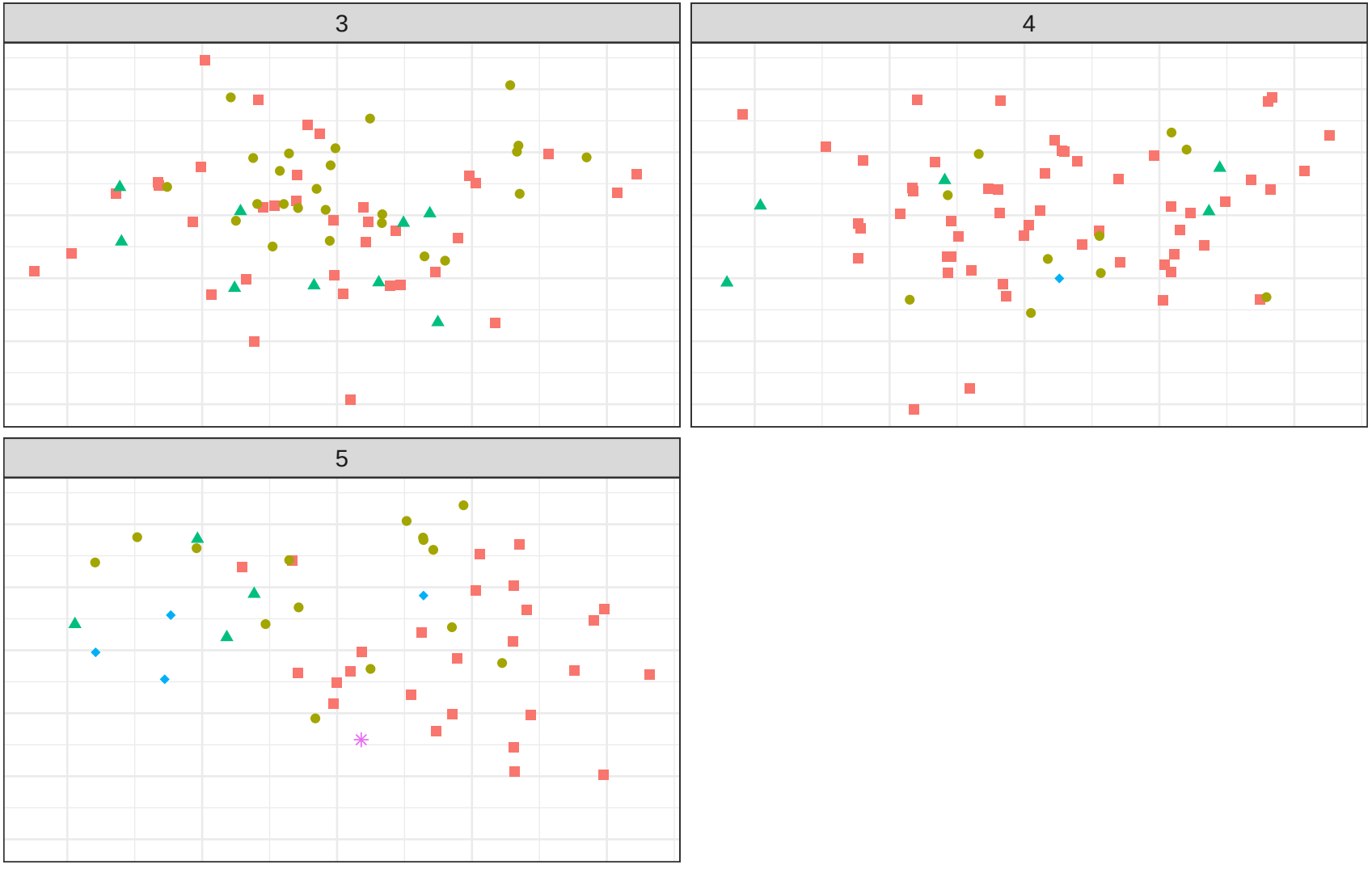
Our approach

- We want to find the amount of variation present for a given feature.
- We use only locations for building our sample.
- We generate clusters with different algorithms (k-means, hierarchical clustering) and pick our sampled locations based on them (package stats, [R Core Team 2020]).
- We compare our results against random geographic sampling for multiple categorical data, in two different scenarios:
  - Simulated data
  - Dialects of Circassian languages

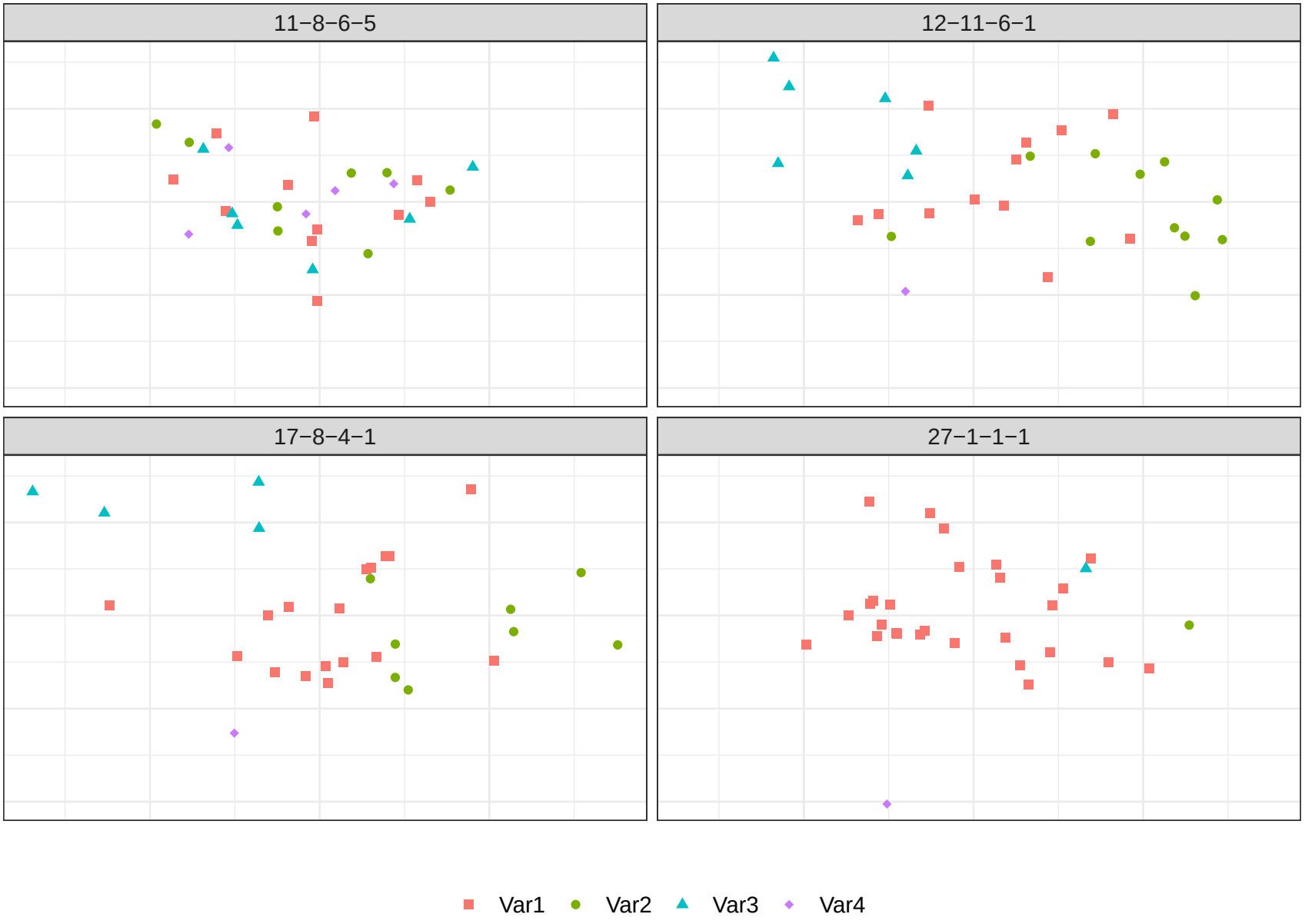
Example of different number of locations (N)



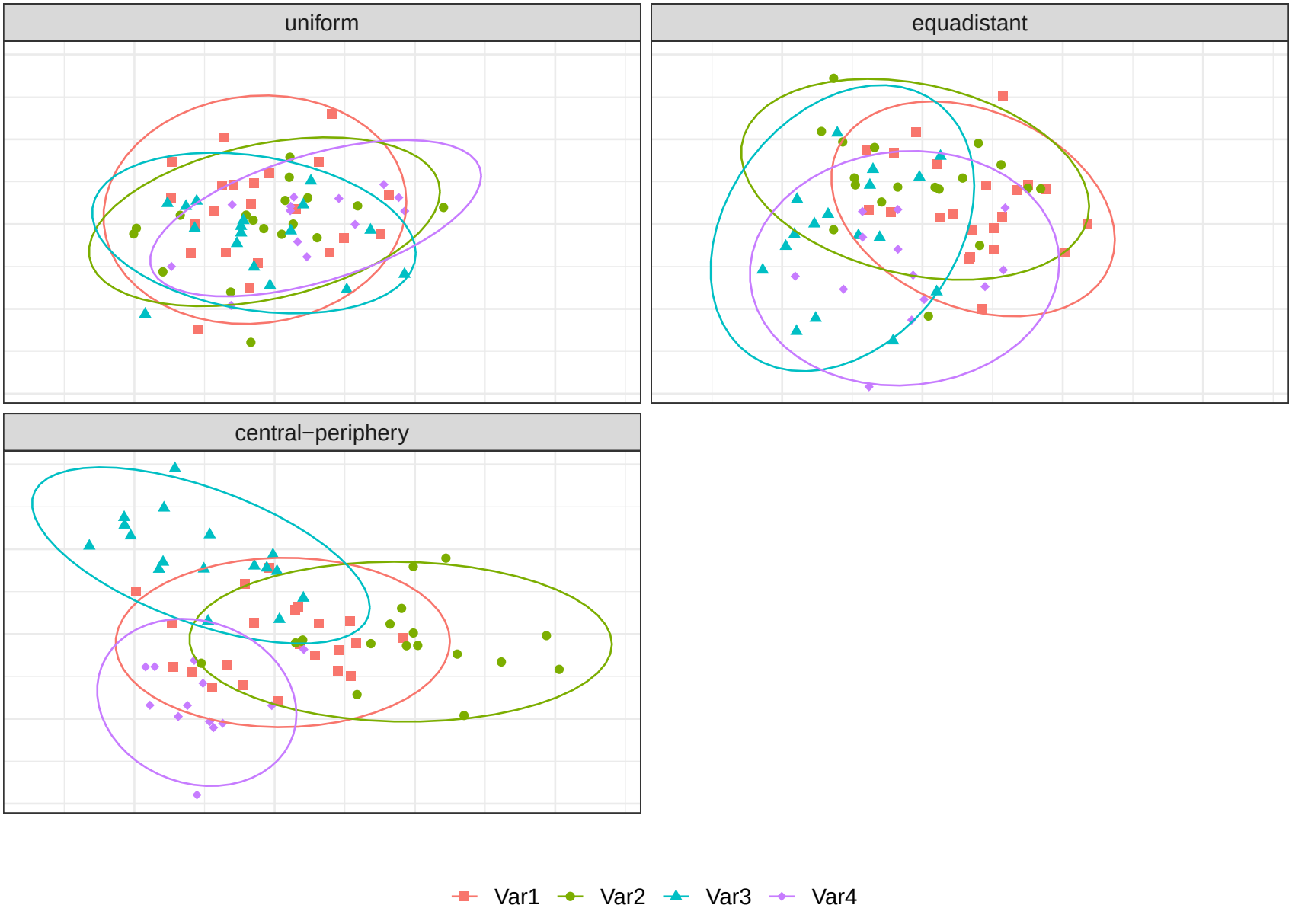
Example of different number of categories (n)



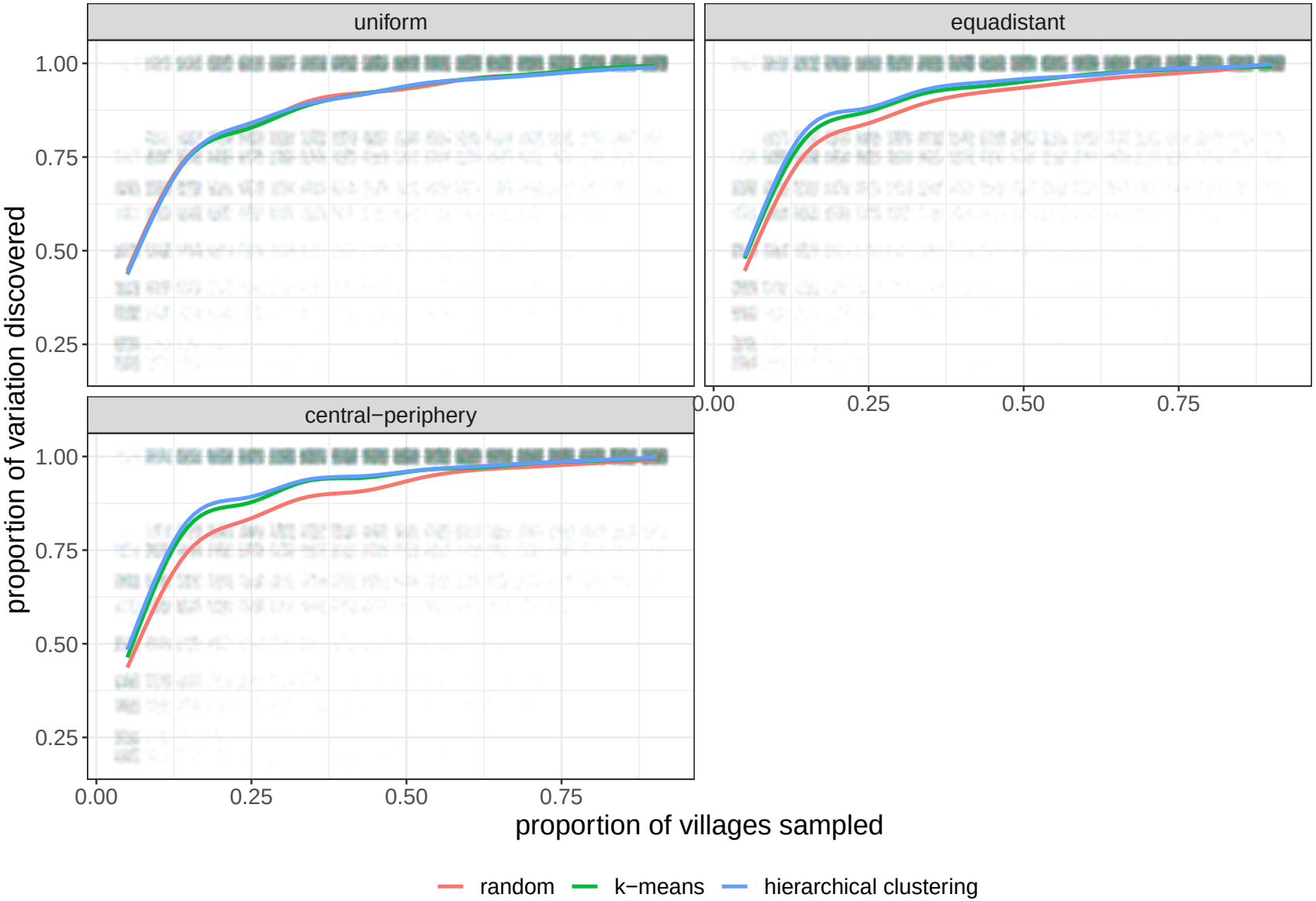
Example of different count configurations (c)



Example of different types of spatial configurations



Results by type of spatial relation



Modelling the variation

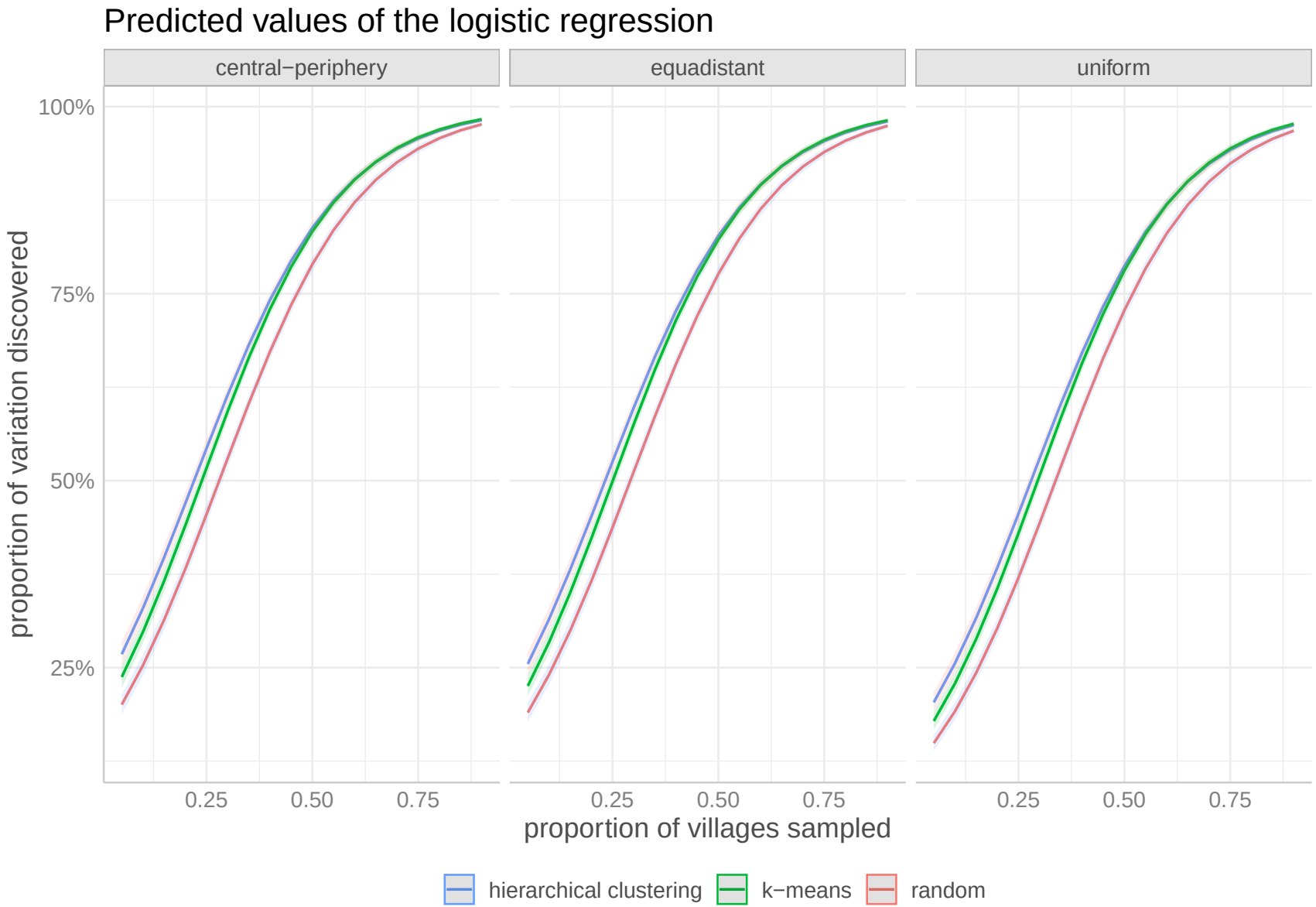
- We run a logistic regression in order to prove those observations by quantifying the relation between:
  - One **binary variable** (outcome):
    - All variation discovered vs. Not all variation discovered
  - Three parameters:
    - Proportion of villages sampled p (numeric: $0.05, \ldots, 0.90$)
    - Type of clustering (hirarchical, k-means, random)
    - Type of geographic distribution (central-periphery, equidistant, uniform)

outcome ~ (spatial configuration + cluster type) * proportion_of_villages

Regression results

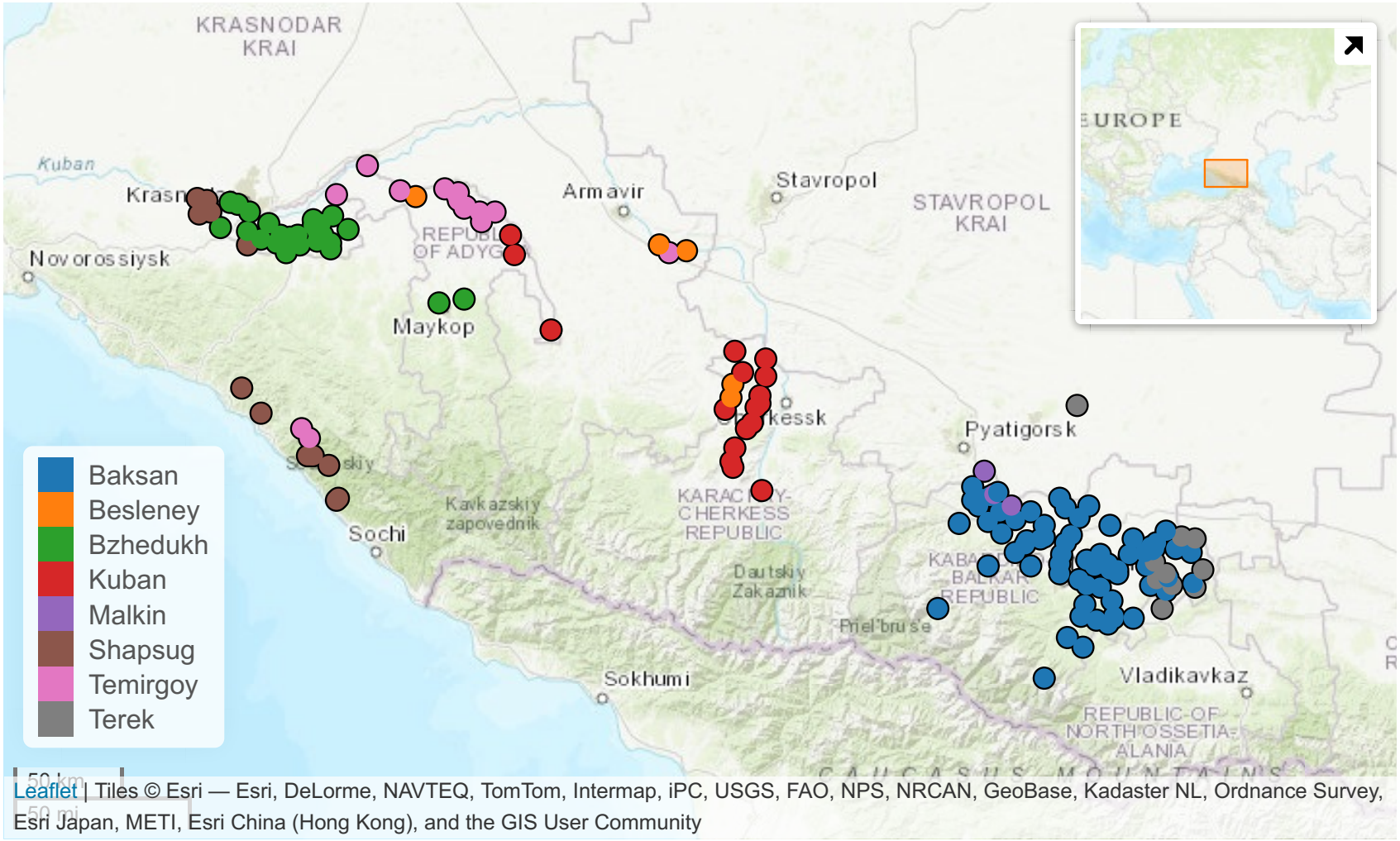
term	estimate	std.error	statistic	p.value
(Intercept)	-2.045	0.048	-42.960	***
equadistant	0.295	0.051	5.736	***
central-periphery	0.362	0.051	7.036	***
k-means	0.208	0.052	4.035	***
h. clustering	0.384	0.051	7.510	***
proportion_of_village	6.057	0.112	54.256	***
equadistant:proportion_of_village	-0.068	0.125	-0.546	0.59
central-periphery:proportion_of_village	-0.050	0.126	-0.396	0.69
k-means:proportion_of_village	0.159	0.126	1.258	0.21
h. clustering:proportion_of_village	-0.121	0.125	-0.969	0.33

Regression results

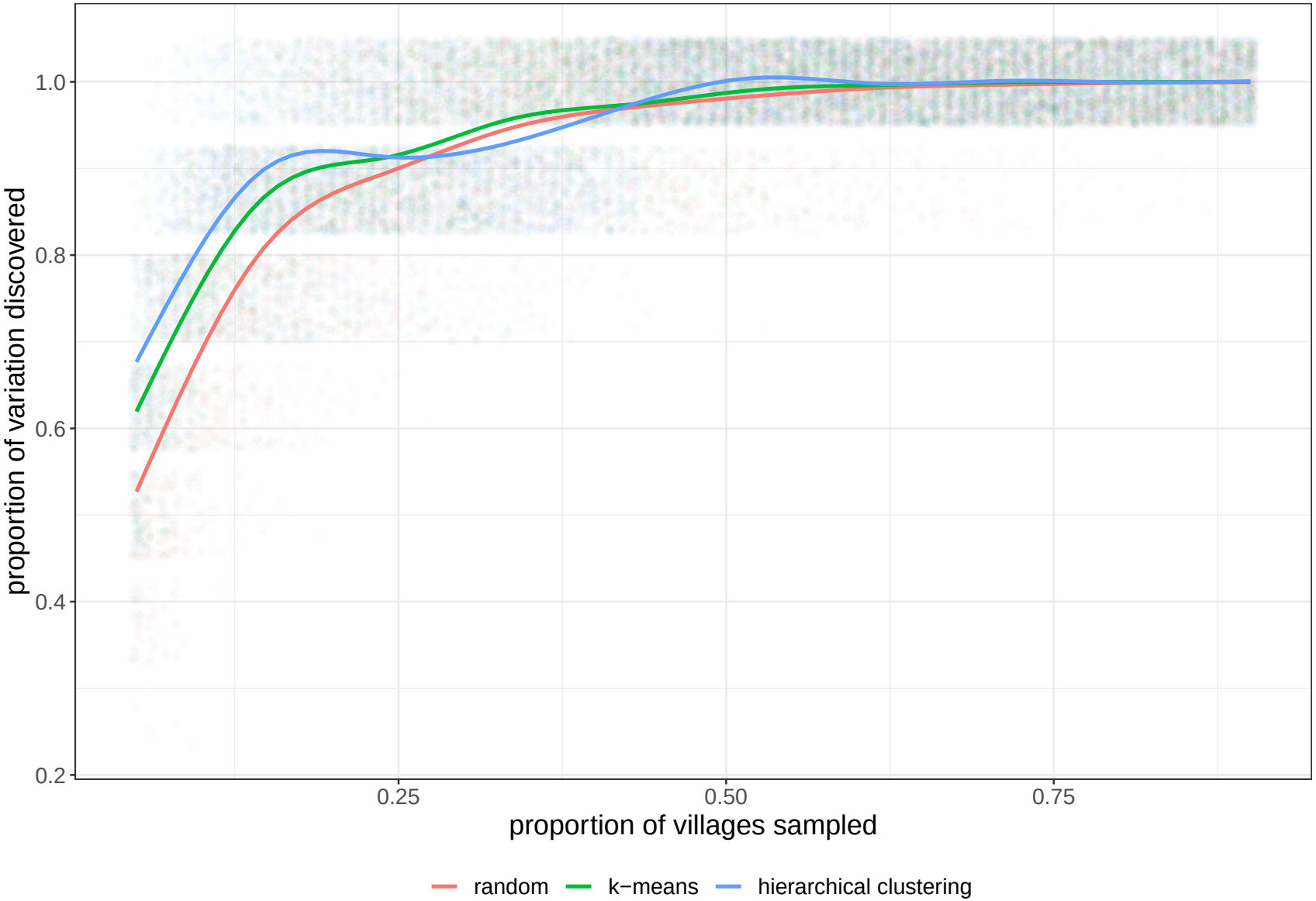


Algorithm evaluation using Circassian data [Moroz 2017]

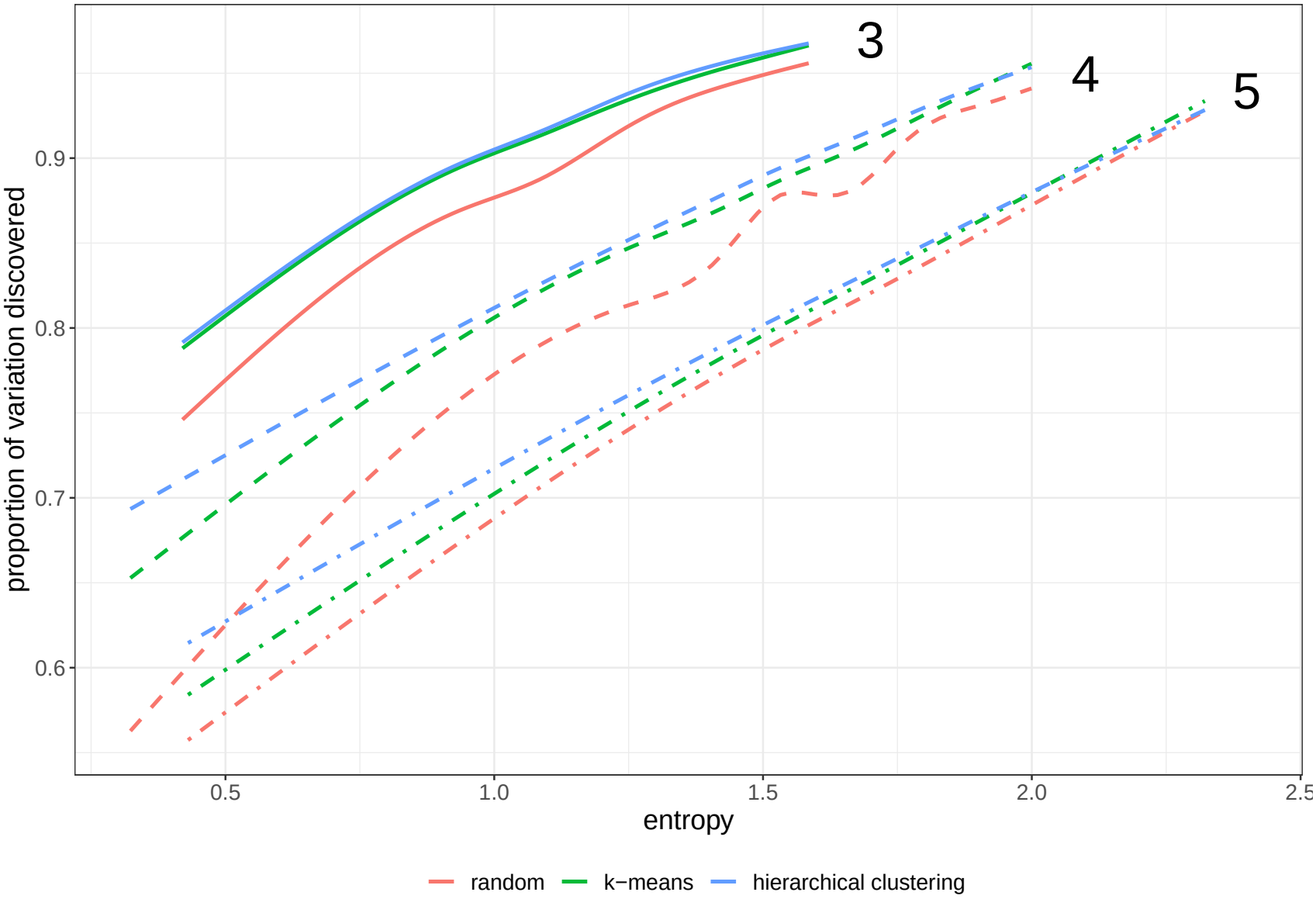
- 158 villages
- proportion of villages sampled: 0.05, 0.06, …, 0.89, 0.9
- true count configuration: 68-27-17-15-13-10-5-3
- 100 runs of each method on the same dataset



Algorithm evaluation using Circassian data [Moroz 2017]



Information entropy: simulated data



Conclusions

- Our algorithm outperforms random sampling on simulated data in the cases where an underlying geographical structure is present, and performs as well as random sampling when variation is uniformly distributed across space
- Our algorithm outperforms random sampling on real Circassian data for small sample proportions, but hirarchical clustering becomes worse than random sampling on larger sample proportions on those specific data
- We found that our algorithm has optimal results when entropy is lower

References

Moroz, G. (2017). *lingtypology: easy mapping for Linguistic Typology*.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.