

Spatial Synthetic Population Generation Using Simulated Annealing in Go

Hugh, Andi, Ric

Go (Golang)

A Compiled, Concurrent, Systems Programming Language

- **Type:** Statically-typed, compiled language with a C-like syntax but memory safety and garbage collection.
- **Compilation:** Compiles directly to machine code (not to a VM). **Produces a single, fast, dependency-free binary for Linux, Windows, macOS, and more.**
- **Concurrency Model:** Based on **CSP** (Communicating Sequential Processes). Lightweight **goroutines** and **channels** provide a simpler and more efficient alternative to traditional thread-based concurrency.
- **Systems Programming Focus:** **Designed for modern multi-core** and networked machines.
- **Much Much faster than Python and easier than JAVA**

"Do One Thing and Do It Well"

1. **Do One Thing and Do It Well:**
Programs are small, focused tools that excel at a single, specific task.
2. **Write Programs to Work Together:**
The output of one program should be able to serve as the input to another.
This is enabled by...
3. **Everything is a Text Stream:** Use plain text as the universal interface.
(JSON CSV ...)

Creating synthetic area households for Scotland

Constraint data by area

geography_code	s1_hh_urban_rural %urban	s1_hh_urban_rural %rural	s2_hh_size%hhsz _1	s2_hh_size%hhsz _2
S00135307	0	62	22	22
S00135308	0	33	7	17
S00135309	0	71	26	32
S00135310	65	0	21	23

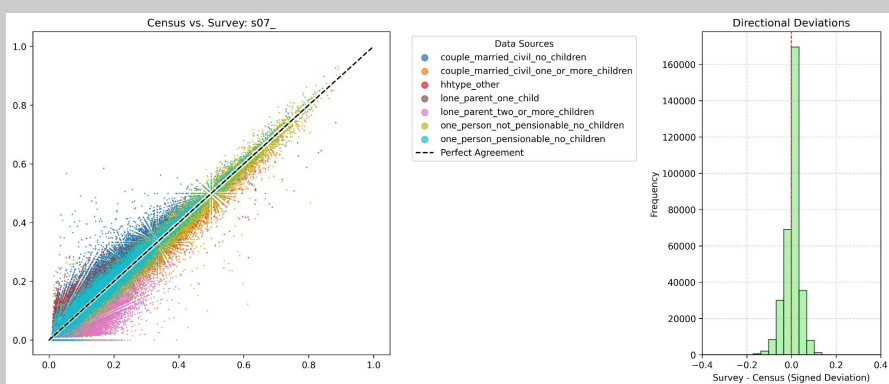
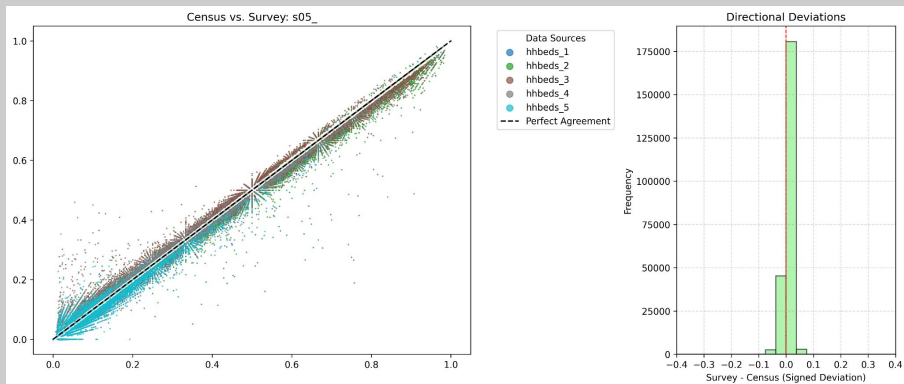
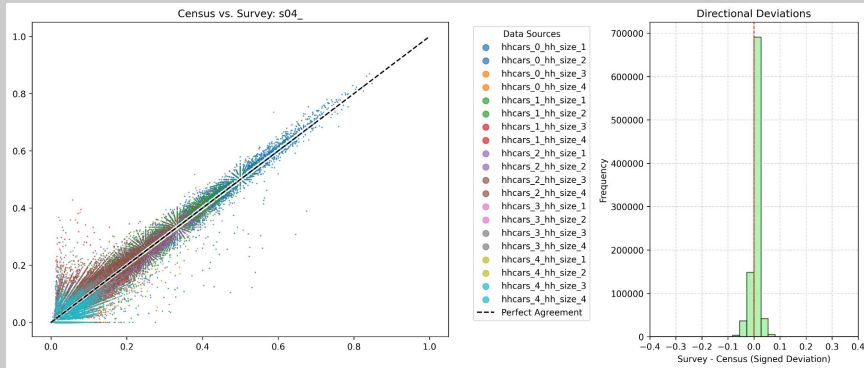
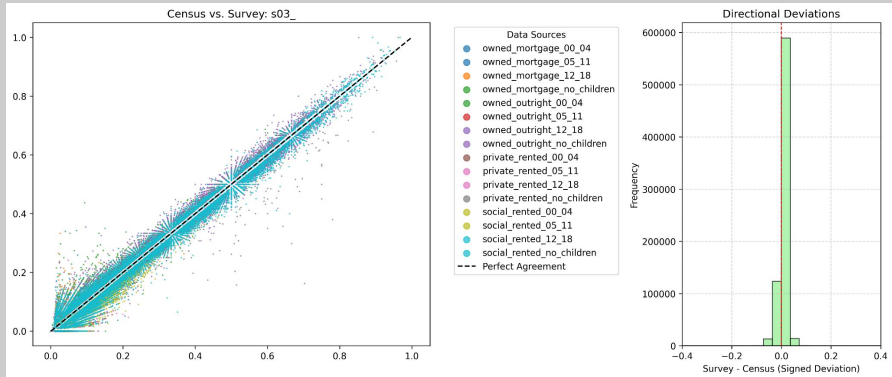
Household data (one hot)

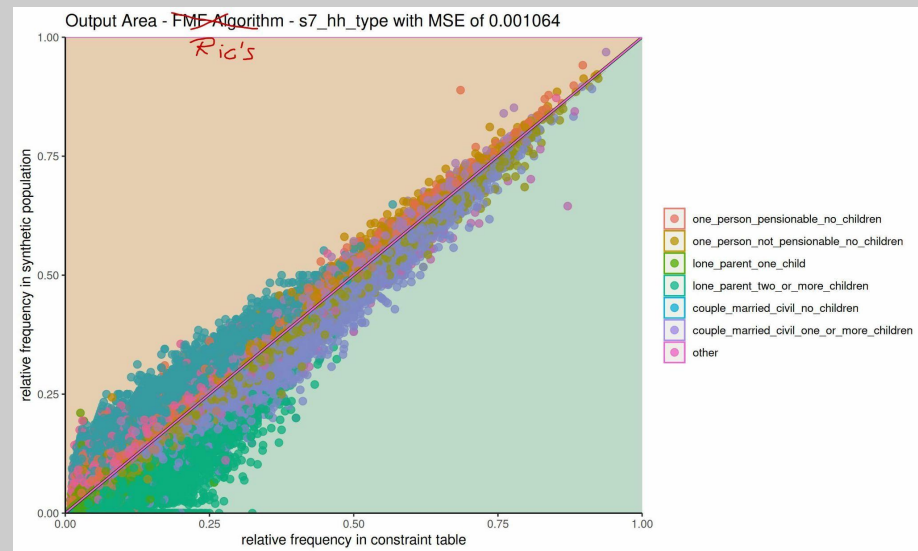
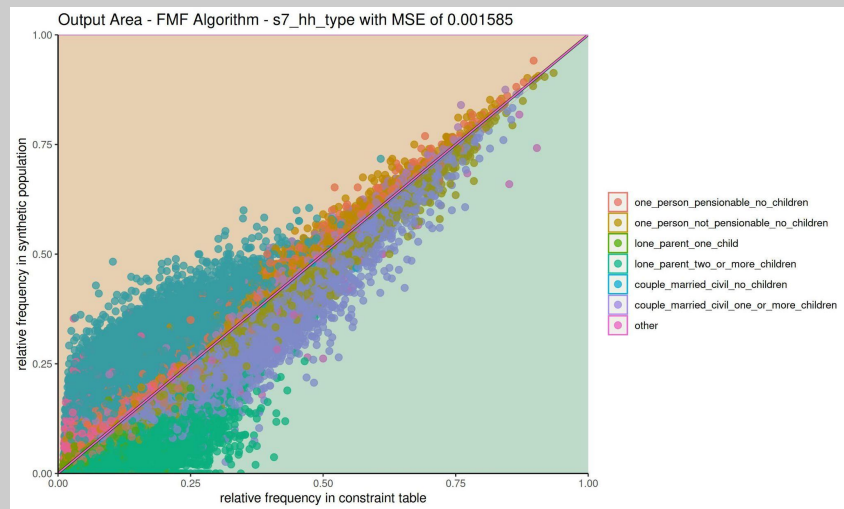
id	s1_hh_urban_rural% urban	s1_hh_urban_rural%r ural	s2_hh_size%hhsz_ 1	s2_hh_size%hhsz_ 2
68006826	1	0	0	1
68013626	1	0	1	0
68020426	1	0	0	0
68027226	1	0	0	1
68047626	1	0	0	0

GoSynthPop population generation setup

```
{
  "constraints": {
    "file": "data/census2022_all_go.csv"
  },
  "microdata": {
    "file": "data/us_hh_export_go.csv"
  },
  "output": {
    "file":
"results/synthetic_population_0608.csv"
  },
  "validate": {
    "file": "results/synthPopSurvey0608.csv"
  }
}
```

```
{
  "initialTemp": 5000.0,
  "minTemp": 0.00001,
  "coolingRate": 0.999,
  "reheatFactor": 0.8,
  "fitnessThreshold": 0.0001,
  "minImprovement": 0.0001,
  "maxIterations": 500000000,
  "windowSize": 1000,
  "change": 100000,
  "distance": "EUCLIDEAN",
  "useRandomSeed": "yes",
  "randomSeed": 42
}
```





Creating synthetic households
of individuals from household
data

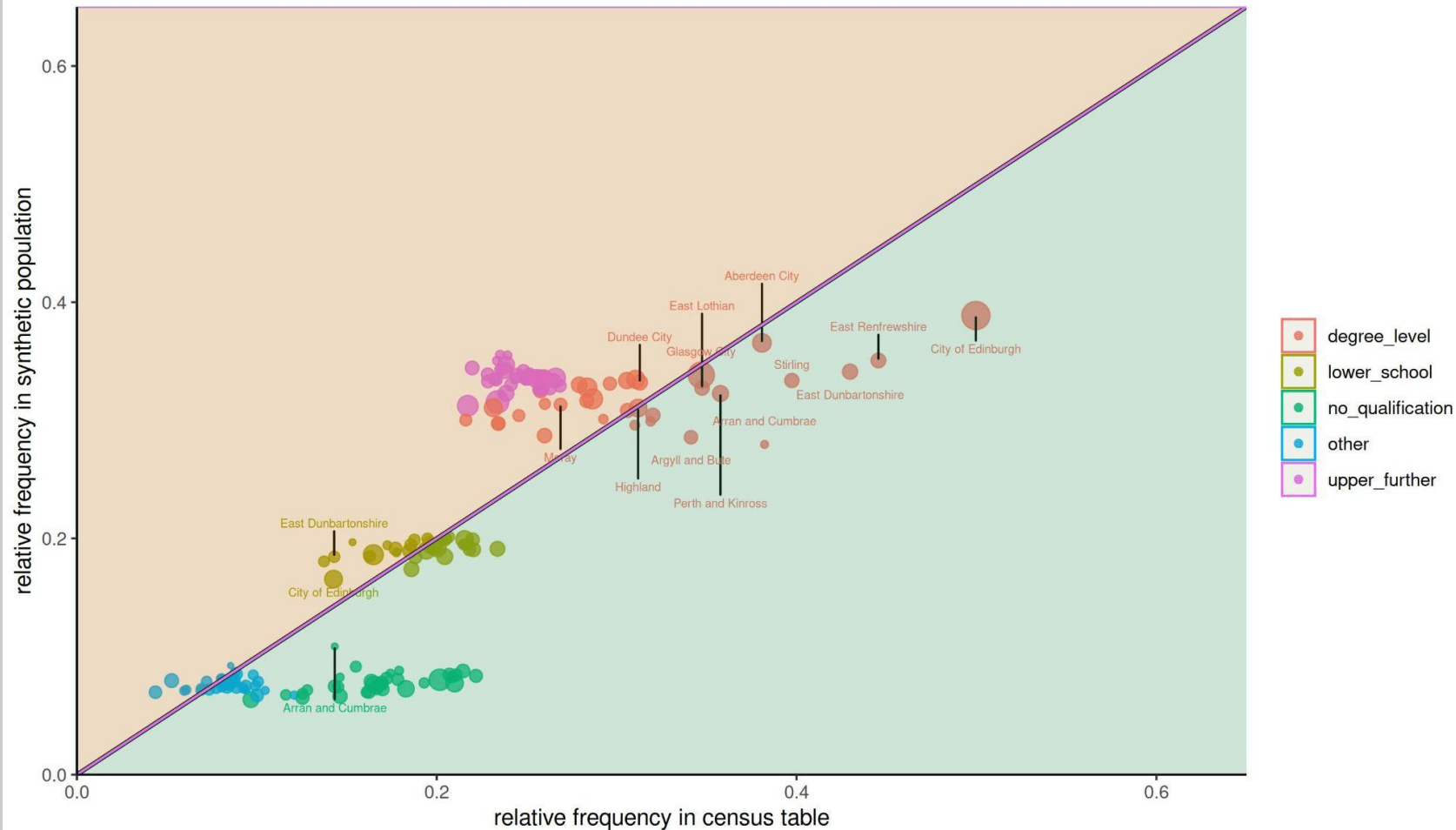
Constraint data by area

s13_unpaid_carer_hh_size%carer_2_hh_size_3	s14_edu_count_no_qualification	s14_edu_count_lower_school	s14_edu_count_upper_further	s14_edu_count_degree_level	s14_edu_count_other
3	13	24	27	44	10
1	7	8	7	24	8
3	27	21	18	64	15
2	11	39	40	31	15
1	12	17	27	52	9

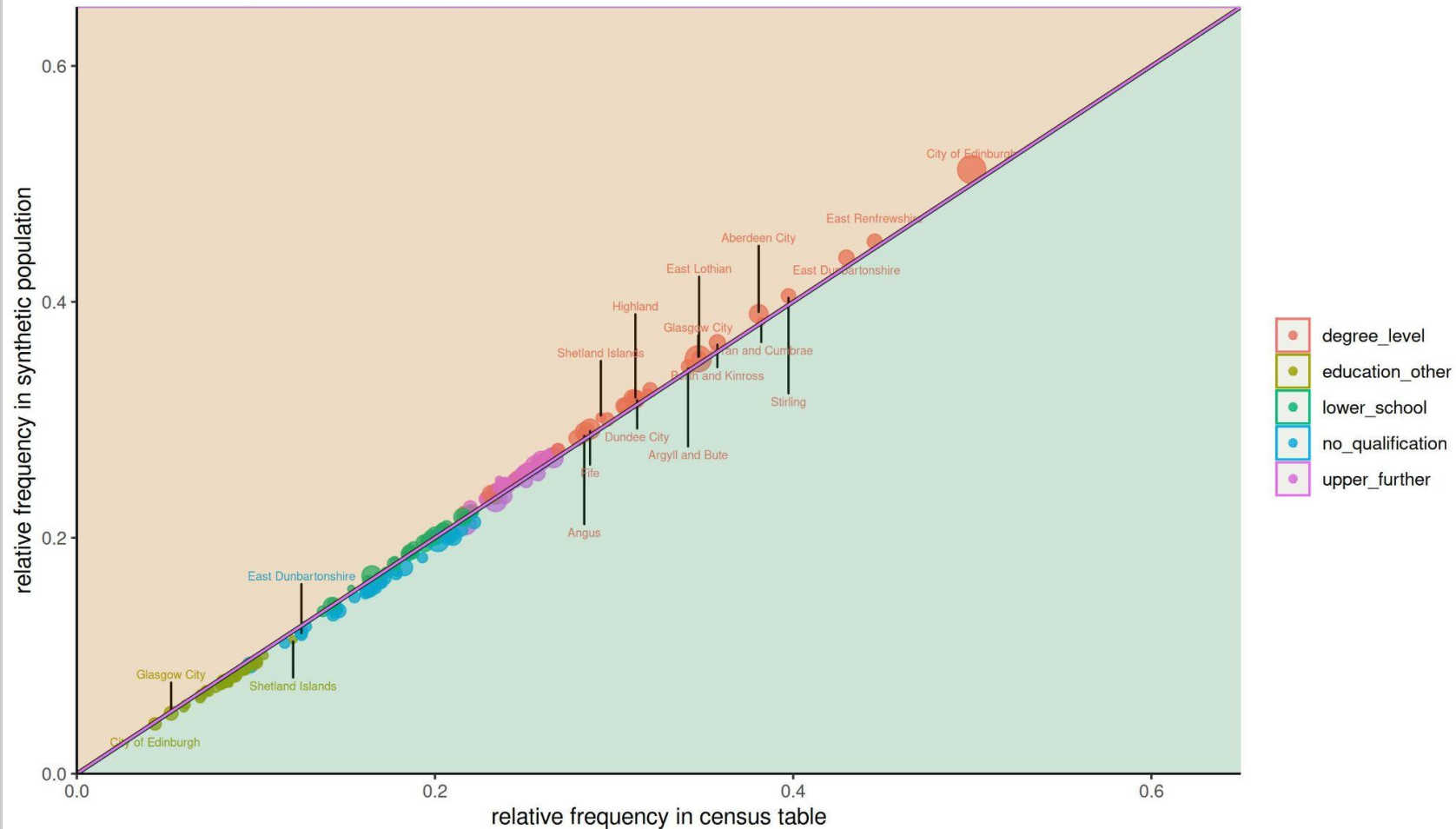
Household data

s13_unpaid_carer_hh_size%carer_2_hh_size_3	s14_edu_count_no_qualification	s14_edu_count_lower_school	s14_edu_count_upper_further	s14_edu_count_degree_level	s14_edu_count_other
0	0	2	0	0	0
0	0	0	0	0	1
0	0	0	1	1	0
0	0	0	1	1	0
1	0	0	0	2	0
0	0	1	2	1	1

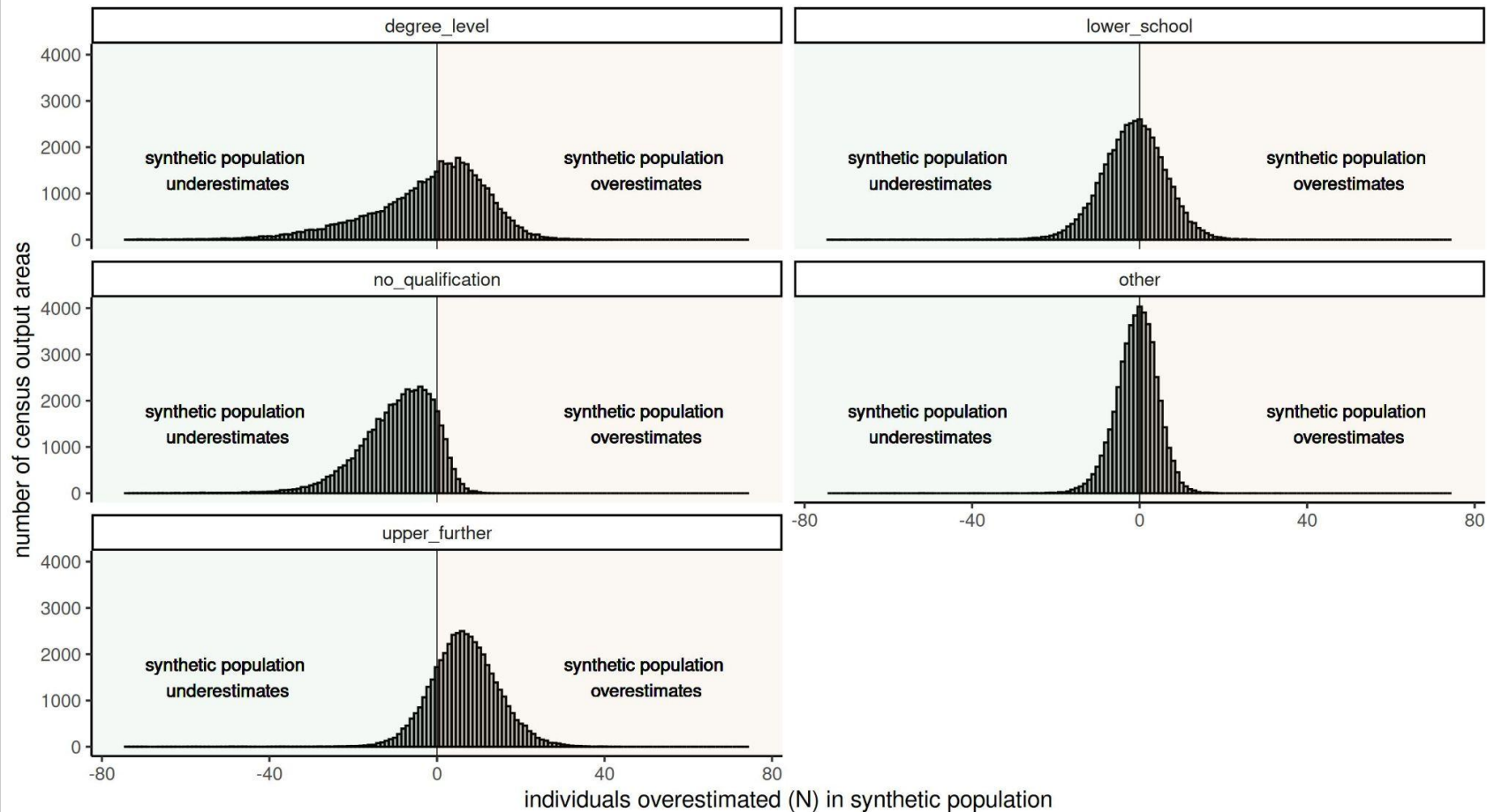
Local Authority - FMF Software - Education at Individual Level



Local Authority - GO Software - Education at Individual Level

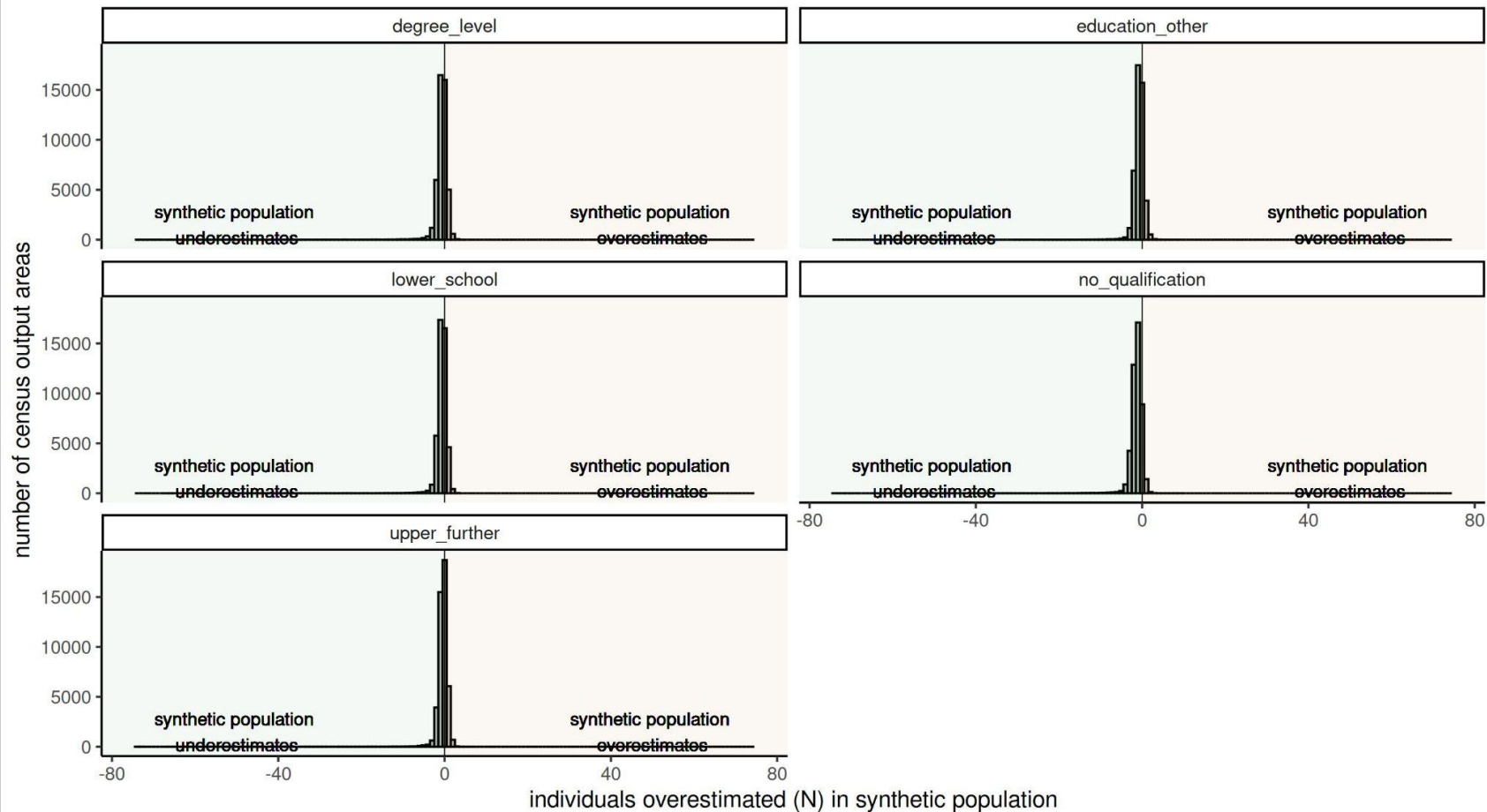


highest education counts; both sexes combined



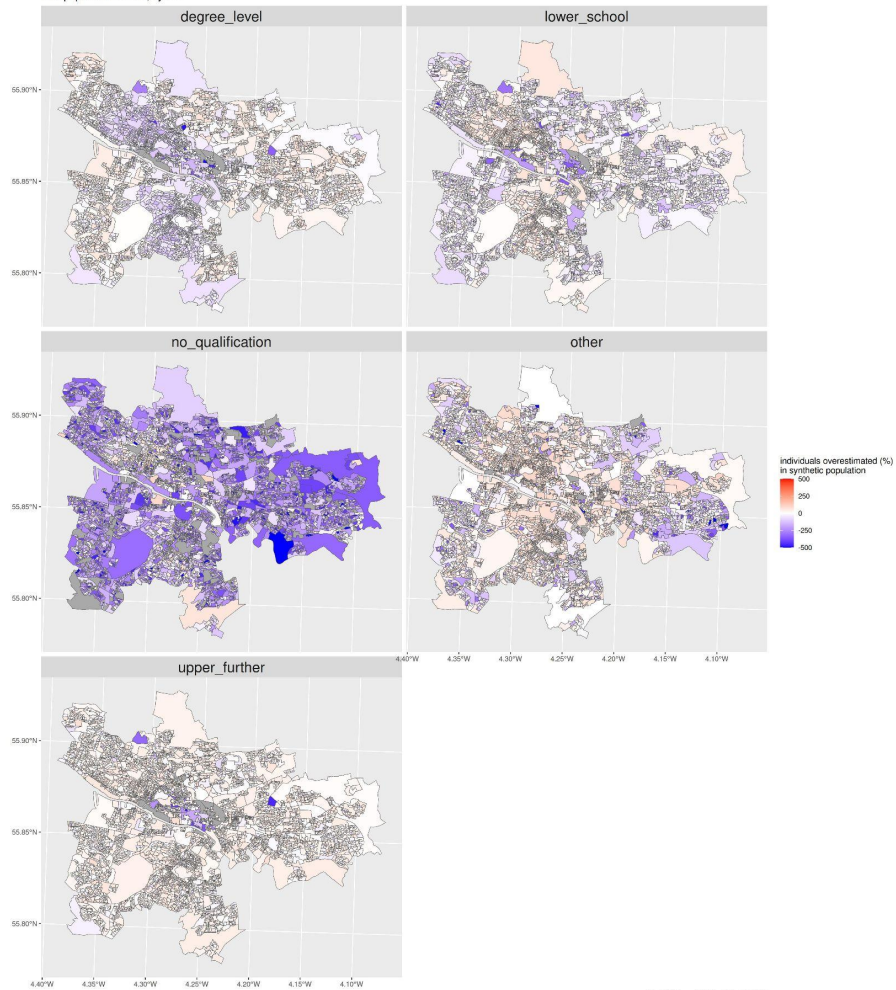
$$N(\text{SynthPop}) - N(\text{Census})$$

highest education counts; both sexes combined

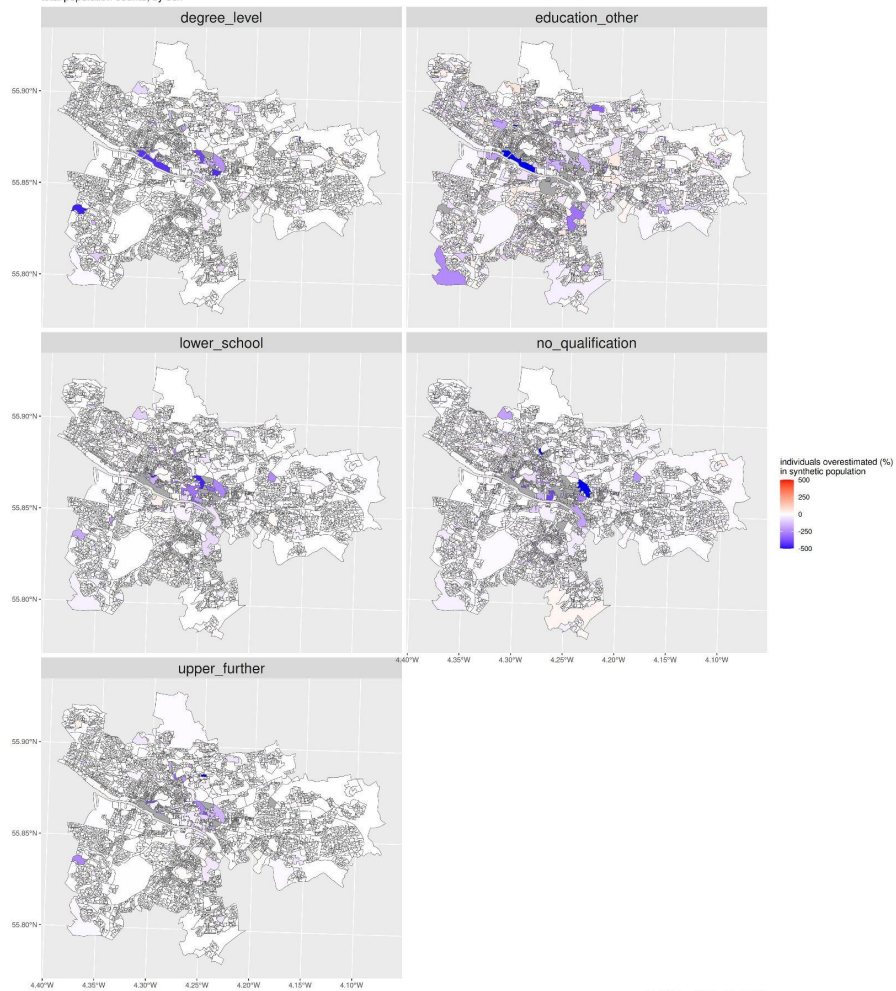


$$N(\text{SynthPop}) - N(\text{Census})$$

total population counts; by sex

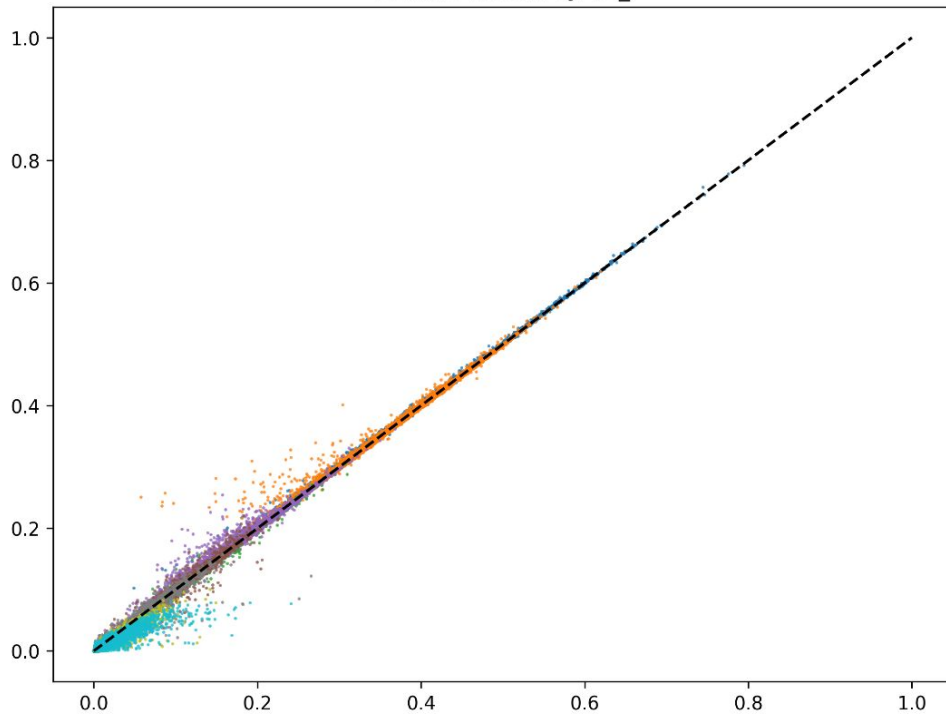


total population counts: by sex

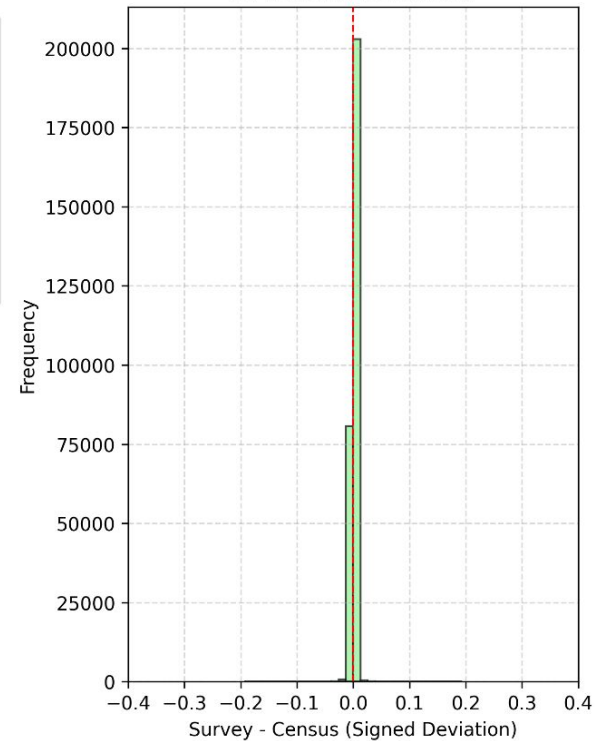


Creating synthetic area households for England and Wales

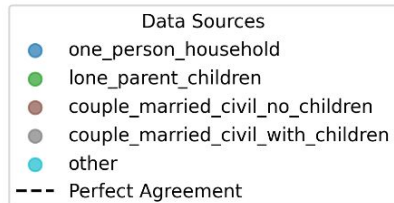
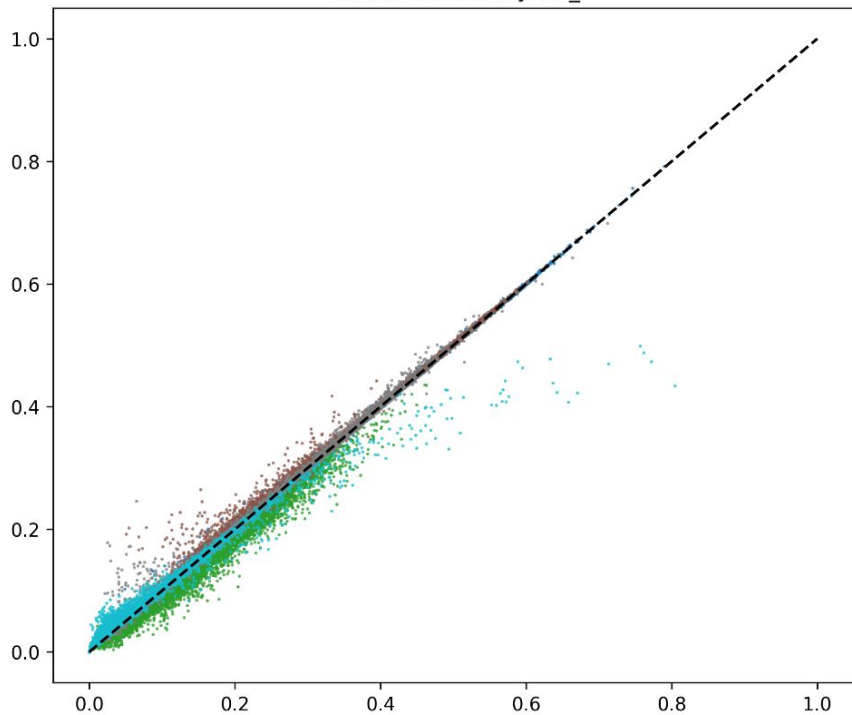
Census vs. Survey: s2_



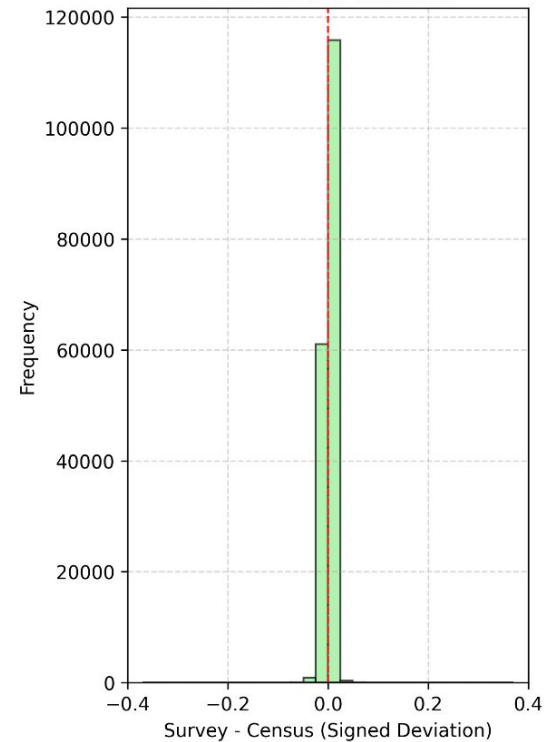
Directional Deviations



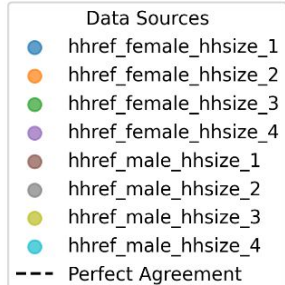
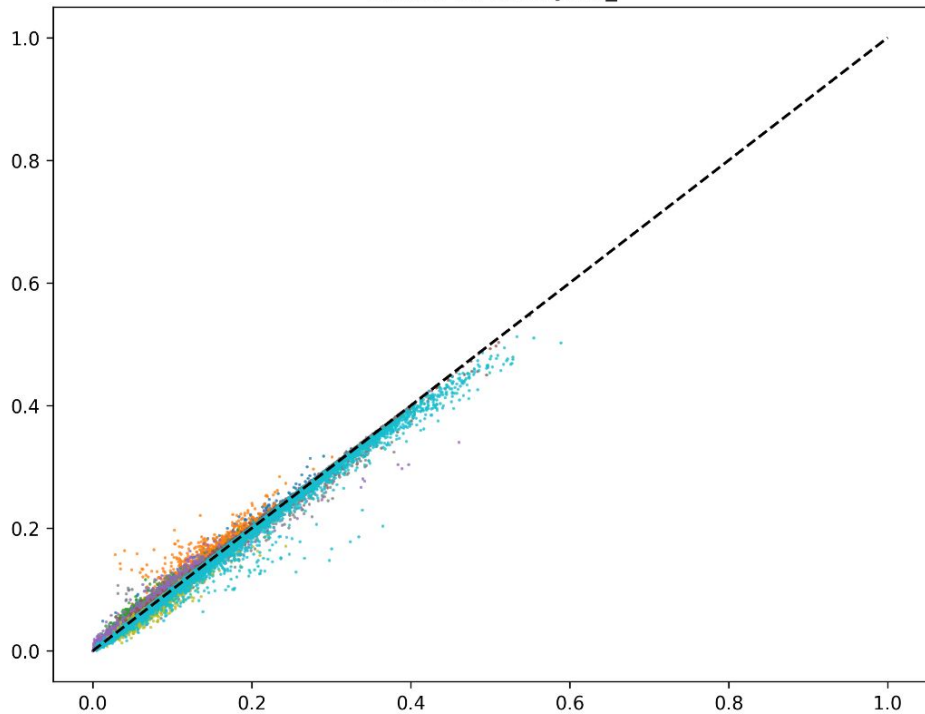
Census vs. Survey: s7_



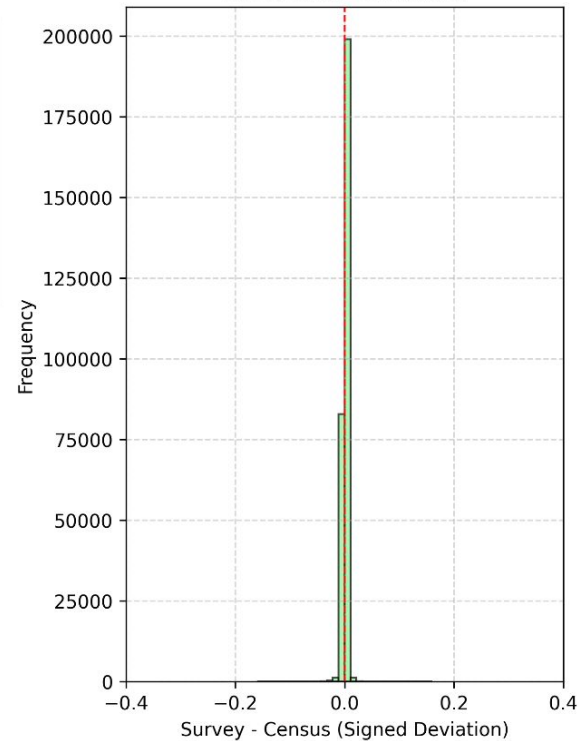
Directional Deviations



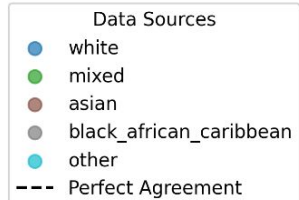
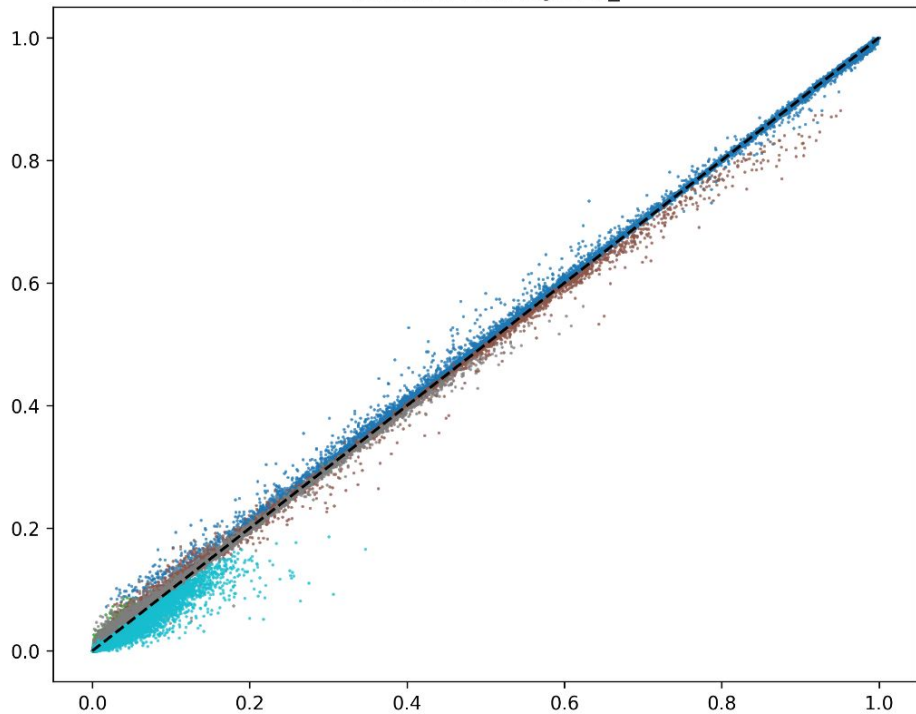
Census vs. Survey: s9_



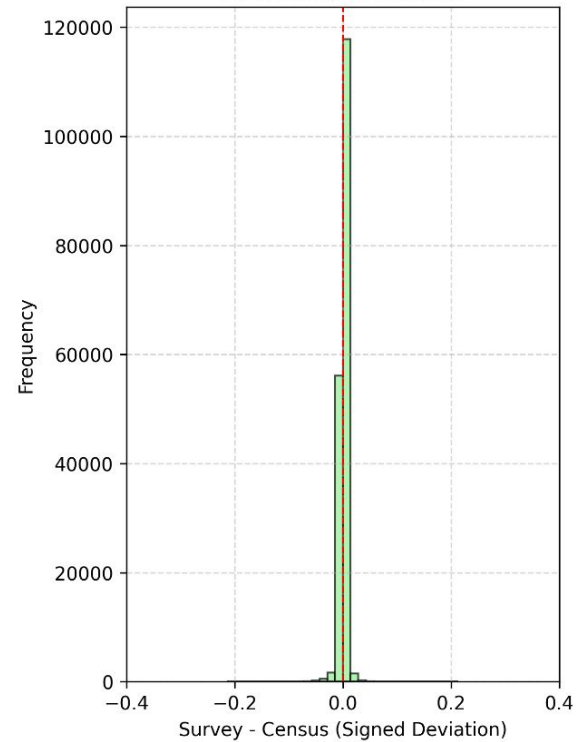
Directional Deviations



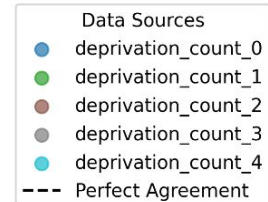
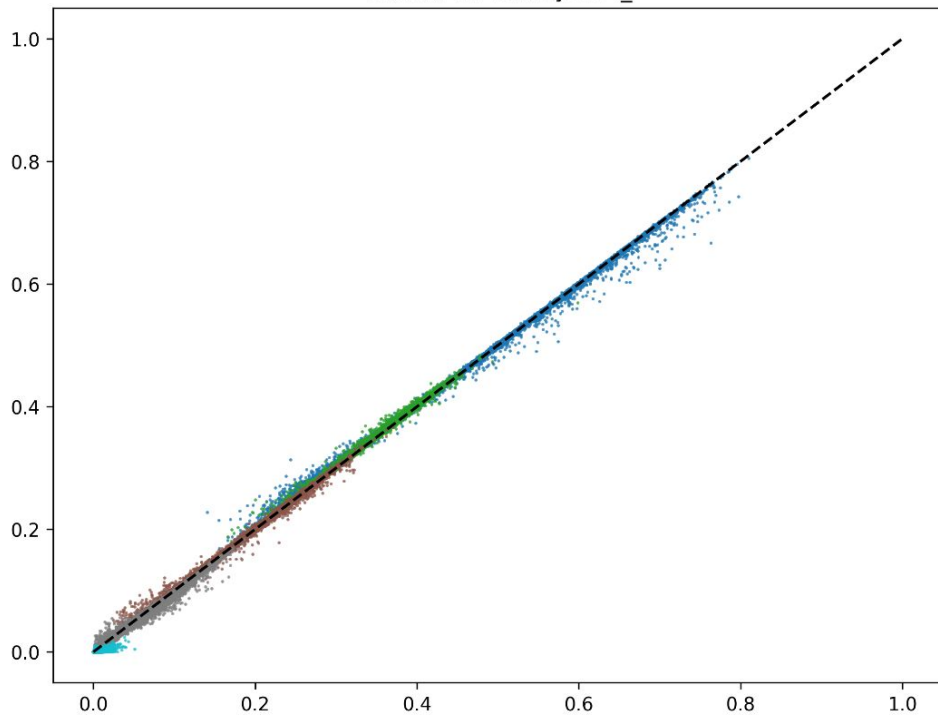
Census vs. Survey: s10_



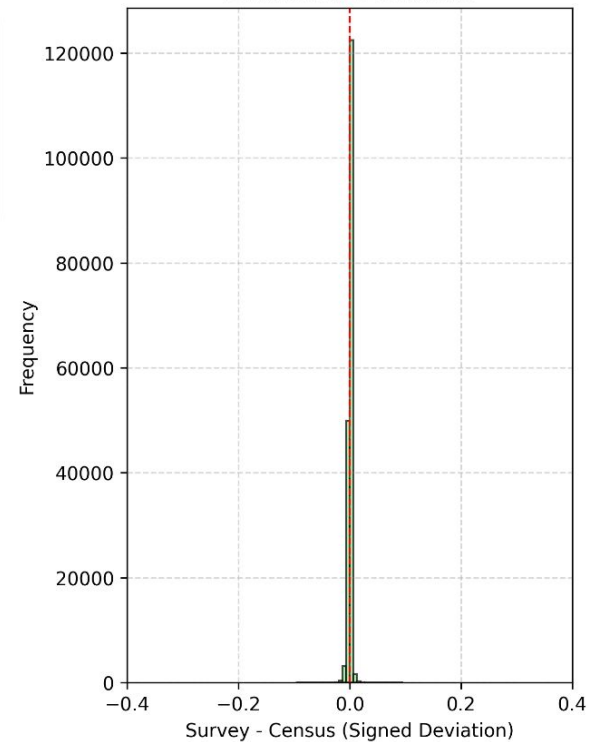
Directional Deviations





Census vs. Survey: s14_



Directional Deviations



#	Objectives	Key Deliverables	Start	End	Status 
1.1	To create synthetic individual population and household datasets	1. An open-source synthetic population of GB individuals nested within households, which captures economic and health distributions in the population			
			1-Apr-24	31-Mar-26	
1.2	To augment the population with relevant health and economic variables using matching techniques	2. Open-source Python code for the (re)creation of the dataset and for the dynamic updating of the population			
			1-Apr-24	31-Mar-26	
1.3	To use the synthetic population to analyse the distribution of economic and health outcomes for sub-groups and spatial zones	3. Equity-focused analyses to support policy design			
			1-Apr-26	31-Mar-28	Not started

#	Objectives	Key Deliverables	Start	End	Status 
1.1	To create synthetic individual population and household datasets	1. An open-source synthetic population of GB individuals nested within households, which captures economic and health distributions in the population			
			1-Apr-24	31-Mar-26	
1.2	To augment the population with relevant health and economic variables using matching techniques	2. Open-source GO code for the (re)creation of the dataset and for the dynamic updating of the population			
			1-Apr-24	31-Mar-26	
1.3	To use the synthetic population to analyse the distribution of economic and health outcomes for sub-groups and spatial zones	3. Equity-focused analyses to support policy design			
			1-Apr-26	31-Mar-28	Not started

Next ?

1. Individual education of England and Wales
2. Northern Ireland
3. Different household and individual constraints
4. Statistical analysis of output
5. I would like to develop GoSynthpop (UK-808) further
 - a. Hugh's IPF (Anonymous populations)
 - b. Other algorithms
 - i. Genetic Algorithm
 - ii. Particle swarm