Working Title on Simulation of Synthetic Ecosystems

Shannon K. Gallagher, Lee F. Richardson, Samuel L. Ventura, and William F. Eddy
Carnegie Mellon University Department of Statistics

Abstract

This is not yet abstract but will become so with the passage of time.

1 Introduction

With the increased ability in computing in the past two decades, agent based modeling (ABM) has gained significance in civic engineering ADD CITATIONS [2], finance ADD CITATIONS, and especially epidemiology ADD citations (FRED, Porco's latest, find others). In particular, agent based models allow epidemiologists to model the spread of disease and also simulate disease prevention strategies as in FRED Paper citation.

ABM as input relies on pre-specified *agents* or microdata which represent individual objects or individuals with a given set of characteristics. Generally, the agents represent diverse populations. As ABM necesitates that agents interact with one another (and possibly their environment), agents with a richer set of qualities are preferred. We call the agents together with their environment a *synthetic ecosystem*.

For instance we have a table which represents a family in the United States. Finish example. they interact at work, school, home, have race/age/etc.

Ultimately, ABM modelers desire to create useful models that are reflect reality, and have value guiding decision making. As such, we need to create accurate microdata as input to these models. Expanding the work of Wheaton et al. in [11], we intended to focus on syntheic ecosystems within the United States. However, when the Ebola epidemic broke out during the summer of 2014, we extended our model to affected countries in Western Africa, such as Sierra Leone, Liberia, Mali, and more. Challenges quickly arose that did not occur in dealing with the United States due to the quality and type of data available.

In response to these challenges, we develop a flexible, modular proram, Synthetic Populations and Ecosystems of the World (SPEW), geared toward generating specific ecosystems for users. At its core, SPEW creates ecosystem's by consolidating three data sources

1. Population Counts
2. Locations
3. Public Use Microdata Sample (PUMS)

along with additional sources of data such as workplaces and schools. As compared to previous instations of synthetic ecosystems, SPEW is flexible enough to incorporate any human poplation given the availability of data. We currently have generated the United States, Canada, and 80 other countries with standard data avaiable, which amounts to more than (impressively large number) of synthetic people.

Although we can generate the world on a supercomputer over a few days for ourselves, agent based models are a heavy computational burden, and we provide code for individuals who can easily create moderate sized populations (∼10 million individuals) for their use cases.

The rest of the paper is organized as follows. In Section **??**, we discuss the evolution of synthetic populations for ABMs. In Section 3, we describe the data sources used. In Secton 4, we describe in detail how we created our ecosystems. In Section 5, we discuss the synthetic ecosystems we released, and how accurate they are. Finally, in Section 6, we summarize what we have done so far and what we plan to do in future releases of datasets.

2 Prior Work

The first working ABMs can be traced back to the the late 1960s and 70s with Conway's Game of Life [1], along with Schelling's segregation simulation [10]. The first model being an agent based model with deterministic decision rules and the latter probalistic. In both cases, the actual agents are very simple representing agents with one or two qualitites.

As technology progressed, so has the work with ABMs, which can be found in epidemiology ([5] and [8]), logistics [7], civil science [2], [9] and more. Most of these applications focused more on the outputs of the ABMs rather than the inputs or agents.

For our purposes, the biggest development came in 1996. Beckman et. al [2] were particularly interested in creating accurate agents for modeling traffic simulation in Chicago, and they incorporated Deming and Stephan's Iterative Proportional Fitting Procedure (IPFP) [4] as a way of matching population demographics which tables representing their marginal distributions. They utilized the TRANSIMS look up acronym software which still exists today. The IPFP is a way to find the Maximum Likelihood Estimator (MLE) for cells of a contingency table given the marginal totals for certain variables. Using this technique to first create a contingency table from existing marginal totals and sample microdata, Beckman devised sampling weights in which to create full and accurate synthetic ecosystems.

Wheaton et al. [11] extended Beckman's program to generate synthetic ecosystems of the entire United States matching on the variables: number of children, household income ($), household size, household population, and vehicles available, disseminating the data at a county level and using marginal totals at a block group level (see Figure 3.1). Their synthetic ecosystem population totals are based off the 2010 Decennial US Census. In addition to the four variables that were matched on, Wheaton incorporated schools and workplaces for which the individuals of the synthetic ecosystem would attend. These synthetic ecosystems were designed specifically for ABMs and both [5] and [8] incorporate them in their models. Limiting capabilities of the Wheaton population include which agent qualities to match on and adherence to the 2010 Decennial Census numbers. In addition to the household and individual populations, Wheaton produced a separate group quarters population including assisted living facilities, prisons, dorms, etc.

While our specific purpose is to create synthetic populations for ABMs, it should be noted that there is lot's of research done creating synthetic populations for privacy purposes. A Bayesian approach to population generation is implemented by Hu, Reiter, and Wang [6], which creates completely synthetic data, rather than sampling multiple copies from microdata as in in the IPF or naive sampling. However, it should be noted that Hu et. al's population is generated with the aim of privacy and not necessarily for the purpose of input to use in ABMs. Hu's populations are designed for communities with the order of magnitude of about 10^4 individuals and it is currently unclear how household populations can be combined with individual populations.

3 Data

A major challenge in generating our synthetic ecosystems is the collection and integration of data across a variety of sources. As mentioned in section 1, the three required data sources for our ecosystems are population counts, geographies, and a Public Use Microdata Sample (PUMS). By population counts, we mean a table which lists each region, and the number of people and/or households within it. By geographies, we generally mean a ESRI shapefile (`.shp` extension) which contains the location, shape, and other attributes for each region. Finally, by PUMS we mean a sample from the population of interest of individuals and their various characteristics, such as age, sec, income, etc...

There are two key steps to assembling the necessary data for generating our synthetic ecosystems: Collection and Integration. By collection, we mean the identification and download of the raw data sources. By integration, we refer to the process of making sure that these raw data-sources are aligned with one another. For example, integrating our population counts with our geographies means that we are making sure the region names/identifiers contained within our shapefiles match with the names of our population counts. As one can imagine, the ease with which we could collect and integrate our data-sources together varied by country. In order to make our efforts reproducible, we have made all of the code we used to download, and integrate the data available online, at the following addresss:

`https://github.com/leerichardson/spew_olympus`

We have included this for two main purposes. The first, is that it makes things easier for use when we need to track down issues that arise when generating our synthetic ecosystems. The second reason is that, if desired, users of our synthetic ecosystems will be able to understand down every decision we made, and the journey from turning our raw data-sources into our synthetic populations.

The synthetic ecosystems we have made available generally fall into three categories, roughly divided by the availability of PUMS data. First, we used United States Census data to generate synthetic ecosystems, following the work done by [11]. The Census provides a tremendous amount of integrated, detailed information, which allows us to build onthe work of other to create our most extensive synthetic ecosystems. The second group of ecosystems we used centered around IPUMS [3] data, which contained both PUMS and geographies for 83 countries throughout the world. The IPUMS datasets were the key reason we were able to generate populations for so many countries throughout the world. The final category of synthetic populations we created can be viewed as a custom category. This is where the majority of development will come from in our ecosystems, and we detail Canada as an example of a custom synthetic ecosystem we were able to quickly develop using our infrastructure.

In the remainder of the section, we will detail the sources we used for these three categories.

3.1 United States

Nationwide data is available from the US Census for all three of necessary data sources, and since they all originate from the same organization, the data already highly integrated. We have a detailed description fo the data in Appendix A.

For population totals, we have both household and individual counts available from the American Community Survey (ACS) Summary Files (SF). These counts are available at the block group level, a census unit consisting of about 100 double check households. However, we work at the tract level which is the union of of census block groups and consists of about 4,000 people per tract. The advantage of using tracts over block groups is they are less variable

with the passage of time than block groups and some conditional tables of block groups are suppressed by the Census for privacy reasons.

In addition to providing marginal counts, the Census provides PUMS data de-identified individuals from 5%? of the population. This data also comes from the ACS, and in our current iteration we use the one year, 2013 PUMS surveys. Due to privacy reasons, the locations of the individuals in the PUMS are only available at the Public Use Micro Area (PUMA) level.

As illustrated in Figure 3.1, there is no direct relationship between PUMAs and counties, and counties are usually the desired input for ABM. This discrepancy between the data highlights the challenge of synthesizing data, even in a highly harmonized place like the United States.

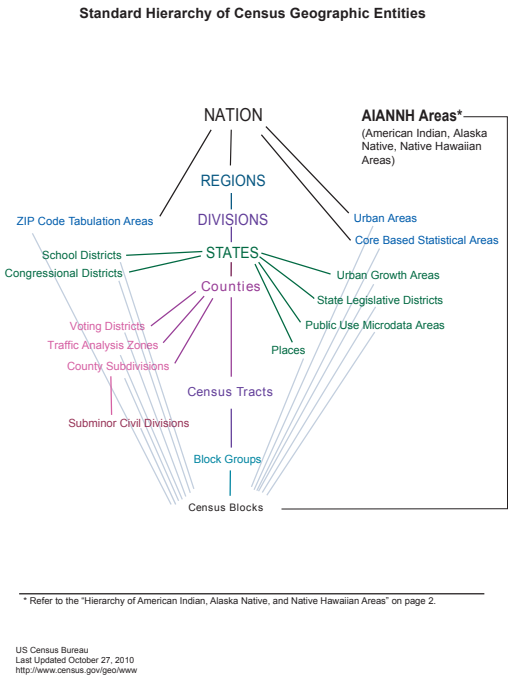

Figure 1: From `census.gov`. Geographical hierarchy of US regions. Of note, we see that PUMAs and counties do not have a nested relationship, an issue which we handle by using the largest common geography of these two: the census tract.

Along with the counts and microdata, we also have to include locations for our synthetic agents by incorporating regional geographies. Borders are dynamic, especially as we move down the geographical hierarchy, which adds a final challenge to consolidating our data sources for use in SPEW. For the United States, we use the Census Topologically Integrated Geographic Encoding and Referencing (TIGER) products for the different borders which allow us to assign locations our synthetic agents.

3.2 IPUMS

In contrast to the USA, it was more difficult to find harmonized sources of data for other countries. However, we did track down our three required data sources for 83 different countries, largely stemming from the availability of IPUMS data for these countries of interest.

The IPUMS data we have available IPUMS-I [3] are simply PUMS data for many countries in the world. In our case, we were able to download PUMS files for 83 countries from the IPUMS website. In these data-sets, the main results such as

For international population totals, we use `geohive.com`. Geohive has the equivalent of level 2 geography, which are the equivalent of states for nearly every country in the world. The levels represent the granularity of the regions with a larger level being more granular than the previous. We have an example of different levels in Table **??**. For some countries, we have Level 3 geography available, which would be the equivalent of counties in the US.

The counts, in comparison to the US, represent population totals only. This presents a challenge for us because we sample from households PUMS, which in turn generate the people. There are many solutions to this issue, and one we employ entails finding the household average for each country and using that to find the number of households per region. In general, there is a tradeoff in balancing the correct populations of people and households, but this tradeoff can be mitigated using more advanced sampling techniques such as mean matching, or the Iterative Proportional Fitting (IPF) algorithm. Again, this just emphasizes the importance of the user's objectives. We can design a population to accurately reflect the variables the user needs for her research.

3.3 Custom Data: Canada as an Example

As a final example here, we detail the data we used to create a Canadian Synthetic population. In our view, the Canadian synthetic population represents the way in which we will generate the majority of our ecosystems going forward: Finding data from a location of interest, integrating the sources together, and using SPEW to output a synthetic ecosystem.

In this particular example, we downloaded each of our three data sources from the Statistics Canada website. Once we had the data downloaded, we were able to write a few computer scripts which converted the data into the format suitable for analysis. In particular, we ran the data-set through a series of checks which come with the spew package. Once the data-set made it through these checks, we knew that it was ready for use in SPEW.

The key point in the generation of the Canadian synthetic ecosystem was how quickly we were able to put it together. Specifically, all we needed was links to the three particular data-sources, and after a few hours of munging, we had the synthetic ecosystem. The rapid nature of creating the Canadian Synthetic ecosystem comes largely from the utility of the spew R package we have developed. In particular, we knew exactly how the data needed to look, and once we had it in place we were confident that spew could generate the required ecosystem. While dealing with heterogeneous data-sources always requires some leg-work, we believe our infrastructure has greatly enhanced both the speed and quality with which we can generate synthetic ecosystems.

4 Methods: Description of SPEW

We've seen in the previous section the data we've collected and integrated in order to generate synthetic ecosystems. In particular, we have noticed that the three necessary elements for generating a synthetic ecosystem are Population counts, geographies, and sample microdata.

In this section, we will describe how we go from these integrated data sources towards an entire synthetic ecosystem.

Because we were generating synthetic ecosystems for various locations, it became clear that we needed a generalized engine which can take integrated data, and output synthetic ecosystems. From this, we developed an R Package, `spew`, which stands for Synthetic Populations and Ecosystems of the World. Our goal for spew is that if the user could provide integrated data from the three required sources, spew would output a synthetic ecosystem. Note that having an engine such as spew allowed us to separate the two primary tasks we encountered in generating sythetic ecosystems: Collection/integration of the data, and turning this data into a synthetic ecosystem.

Once spew was developed, we used it to generate all of our synthetic populations (USA, Canada, Argentina, etc...). We've made all of the code for spew available online at the following URL:

`https://github.com/leerichardson/spew`

where interested users can download it and use it to generate their own synthetic ecosystems.

4.1 How the Engine works

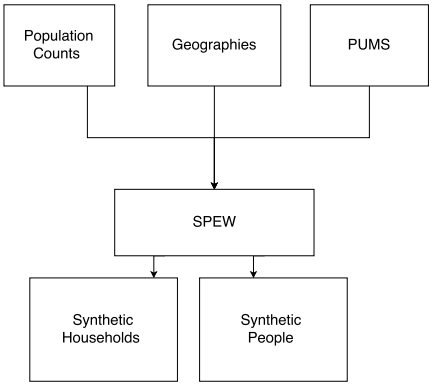

Figure 2: This diagram shows what spew does as a high level: Takes formatted population counts, geographices, and microdata and outputs synthetic ecosystems

At a high level, spew performs the function of taking our three integrated data sources, and outputs a synthetic ecosystem, see **??** for a demonstration. More specifically, spew works by splitting a location into mutually exclusive regions, the union of which adds up to the entire location. From this, we generate a synthetic population for each one of these regions.

It's important to point out that our PUMS data usually contains a variable which corresponds to a specific region within the synthetic population. This variable is usually a superset of many smaller regions, and we refer to it as the `puma_id`. Thus, for each region we typically subset the PUMS data to contain only data from the corresponding `puma_id`. This leads to synthetic ecosystems which are more representative of the marginal distributions of each tract.

For example, in the United States we generate a unique synthetic population for each tract. In this case, we can think of each tract as one of our mutually exclusive regions. Note that each tract is contained within a Public Use Microdata Area (PUMA), and the United States

PUMS data has a variable indicating which PUMA each record is located within. Thus, for each tract we subset the PUMS data to contain all samples from the particular PUMA the tract is located in. Once we have the correct PUMS data, we sample the appropriate number of households for the particular tract. Next, we sample the location of each household using the Tiger shapefile, which serves the role of our geographies. Note that before we can generate the populations, we verify that all of the regions in the geography files match up with all of the regions in the population counts file. Finally, we organize these tracts into subdirectories organized by PUMA. Thus, the default United States synthetic population has a subdirectory for each state, each PUMA within the state, and a synthetic population

input : Population counts, geographies, microdata, other ...
1. Verify each data source has necessary components ;
2. Verify data sources align with one another ;
for *Every Region* **do**
    1. Sample Households ;
    2. Sample Locations ;
    3. Attach People to Households ;
    4. Add other data as desired (eg: schools, workplaces, etc...) ;
end
output: Synthetic Households, People, etc...

Algorithm 1: Pseudocode for generating Synthetic Ecosystems with SPEW

It's important to point out that the pseudocode in the above algorithm is fairly general. In particular, one could use any method they wanted to sample households, locations, and even people within households. Also note that while the three required data-sources needed to generate the synthetic households and people, there is in principle no type of data, be it schools, workplaces, hospitals, mosquitoes, etc.., that we could not include into this framework.

This generality of SPEW is by design, and we think that it is one of our best features. Because we are dealing with many heterogenous data-sources, and can not predict the future types of ecosystems which will be requested by Agent Based Modelers, we strove to create an engine in which it would be easy to implement new features and requests as desired. For instance, we imagine an interested party could simply give us an algorithm and data to assign agents to schools, and we could then easily incorporate and run this through spew.

Right now, spew uses very rudimentary methods for these different steps. In particular, instead of using IPF as in [2], we employ simple random sampling. This means that we are simply re-sampling the pre-existing records until we have enough for a synthetic ecosystem. Along these sample lines, to sample locations we are simply uniformly sampling over each particular region, without focus on the locational features within the ecosystem. While these basic techniques allowed us to set up the initial framework and get the first round of populations released, there is much room to advance the sophistication of the methods we use. One could imagine using density estimation to sample the households, and attribute based sampling to location them. While our current populations lack sophisticated methods, we believe this gives us lots of room for growth in future iterations of the program.

4.2 Output of the engine

After the integrated data-sources are ran through spew, the result is the synthetic ecosystem. Currently, this means we have both synthetic households and sythetic people. A synthetic ecosystem is structured the same way as the PUMS, except in this case we have records corresponding to 100 % of the population, whereas the PUMS data usually is a sample of 1-5 %

of the population. See figure ADD Example Microdata figure for a visual look at what our synthetic ecosystem looks like.

In terms of what to expect as output, recall in the previous section we described what is known as the `puma_id`. For every synthetic population with a `puma_id`, we will have

4.3 Computing

By design, generating synthetic populations is a very computationally intensive task. Even in just generating one ecosystem, when there are millions of people with many characteristics to be generated, the input/output load is quite expansive. Fortunately, we had access to the Olympus Cluster, hosted by the Pittsburgh Supercomputing Center.

The Olympus cluster is made up of 24 nodes, with one serving as the head node and the other 23 nodes serving as compute nodes. Each node has four cite olympus documentation multi core processors, each of which contains 16 compute cores, so each node has 64 compute cores. This means that, in principle, we can run 1536 processes at one time on Olympus.

One other thing that jumps out when inspection reference algorithm, is that spew is what is referred to as an embarrasingly parallel application. By this, we mean that once we have our integrated data, it easy easy to see that by parallelizing each region, we have a straighforward way of generating our synthetic ecosystems in parallel.

Add in a timing which shows the VALUE of using Olympus

Add in a timing which shows the VALUE of using the ml io section

5 Results and Vetting

Add a section on Automated checks and diagnostics

Add an example output summary from Shannon's diagnostic functions

6 Conclusions and Future Work

A Data List

1. 2006-2010 5-year ACS PUMS
   - Available at: `http://factfinder.census.gov/faces/nav/jsf/pages/searchresults.xhtml?refresh=t`
   - Corresponds to 2000 defined Census geography
   - Household and People populations
   - For detailed information see: `http://www.census.gov/acs/www/data_documentation/documentation_main/`

   (a) `pums_h.csv`
   - The variables correspond to different household attrributes, about 80 of which are weights.

   (b) `pums_p.csv`
   - People population subset of the PUMS
   - The variables correspond to different pepole attributes, around 90 of which are weights.

2. US Census TIGER Shapefiles– 2010

- Available at `https://www.census.gov/geo/maps-data/data/tiger.html`
- Geographical boundaries of different census regions. Currently have block group level, which is the most fine unit diseminated by the Census.

3. National Center for Education Statistics School Data
   - Available at: `http://nces.ed.gov/ccd/elsi/tableGenerator.aspx`
   - Can find school data for given year and region.
   - Variables include enrollment information, latitude and longitude coordinates, and other useful variables.
   - Both public and private school data available
4. ESRI workplace data

References

[1] Andrew Adamatzky. *Game of Life Cellular Automata*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[2] R.J. Beckman, K.A. Baggerly, and M.D. McKay. Creating synthetic baseline populations. *Transportation Research Part A*, 30(6):415–429, 1996.

[3] Minnesota Population Center. Integrated public use microdata series, international: Version 6.3, 2014. [Machine-readable database].

[4] W. Edwards Deming and Frederick F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *The Annals of Mathematical Statistics*, 11(4):pp. 427–444, 1940.

[5] Grefenstette JJ, Brown ST, Rosenfeld R, Depasse J, Stone NT, Cooley PC, Wheaton WD, Fyshe A, Galloway DD, Sriram A, Guclu H, Abraham T, and Burke DS. Fred (a framework for reconstructing epidemic diseases): An open-source software system for modeling infectious diseases and control strategies using census-based populations., October 2013. PubMed PMID: 24103508.

[6] Jingchen Hu, JeromeP. Reiter, and Quanli Wang. Disclosure risk evaluation for fully synthetic categorical data. In Josep Domingo-Ferrer, editor, *Privacy in Statistical Databases*, volume 8744 of *Lecture Notes in Computer Science*, pages 185–199. Springer International Publishing, 2014.

[7] Fu-ren Lin and Shyh-ming Lin. Enhancing the supply chain performance by integrating simulated and physical agents into organizational information systems. *Journal of Artificial Societies and Social Simulation*, 9(4):1, 2006.

[8] Fengchen Liu, WayneTA Enanoria, Jennifer Zipprich, Seth Blumberg, Kathleen Harriman, SarahF Ackley, WilliamD Wheaton, JustineL Allpress, and TravisC Porco. The role of vaccination coverage, individual behaviors, and the public health response in the control of measles epidemics: an agent-based simulation for california. *BMC Public Health*, 15(1), 2015.

[9] David L. Sallach and Charles M. Machal. Introduction: The simulation of social agents. *Social Science Computer Review*, 19:245–248, Fall 2001.

[10] Thomas Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1, 1971.

[11] William D. Wheaton, James C. Cajka, Bernadette M. Chasteen, Diane K. Wagener, Philip C. Cooley, Laxminarayana Ganapathi, Douglas J. Roberts, Justine L. Allpress, and James C. Cajka. Rti press synthesized population databases: A us geospatial database for agent-based models, 2009.