Synthesizing NNDSS Case Data

CSTE DRPASS Workgroup Discussion

2023-03-07

Chad Heilig, PhD
Data scientist

Presentation overview

Background

NNDSS Synthetic Case Generator

Sharing code and synthetic data

Background

CDC-GTRI* Innovation Collaborative

Can we

make fake records of notifiable diseases

with

the same traits as real records

while

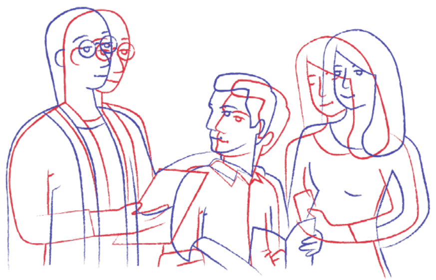
protecting patient privacy

And

make it easy to do again?

Georgia Tech Research Institute, a University-affiliated Research Center (UARC)

Why would synthetic surveillance data be useful?



NNDSS data access and use is restricted

- Data contains private information
- Data use agreements must be signed

A synthetic NNDSS dataset could …

- Preserve statistical patterns like incidence and seasonality while protecting individual patient information
- Facilitate new tools and methods for analysis and visualization
- Explore different data formats and provisioning models (e.g., JSON)
- Test cloud-based architecture for a case-based surveillance pipeline

Source data

Generator uses real data, initially 2015–2021

- NETSS legacy flat file (NETSS, NBS master, HL7 Gen V1 and V2)
- HL7 message with repeating race field (HL7 v.2.5.1 Gen V2)

Only Core data are used, no disease-specific elements

- NETSS extended data was not included
- HL7 data includes most current record of a single case, with minimal suppression

NNDSS Synthetic Case Generator

Overview of method

Main steps for generating synthetic case data

Select a jurisdiction and notifiable condition

Retrieve real NNDSS data (NETSS or HL7 format) and preprocess

- Extract summary information on demographics and dates

Generate synthetic data in 3 steps

1. Time-based incidence patterns (# reports per day over ~6 years)
2. Case-level demographics
3. Case-level event dates

Time-based incidence patterns

Case-level demographics

Case-level event dates

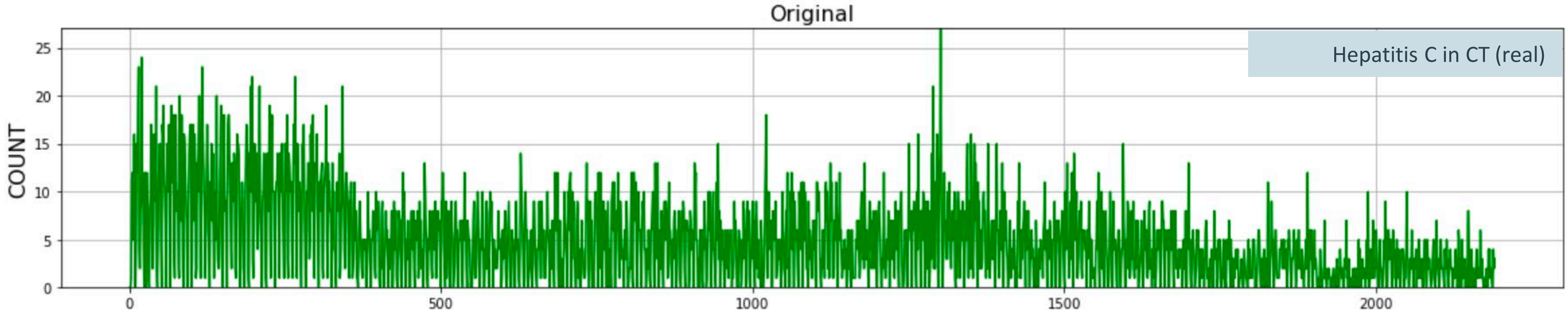
NNDSS Synthetic Case Generator

Real data: time series of record counts

Hepatitis C (10106) in Connecticut

January 2015 – December 2020 (72 months = 2,192 days)

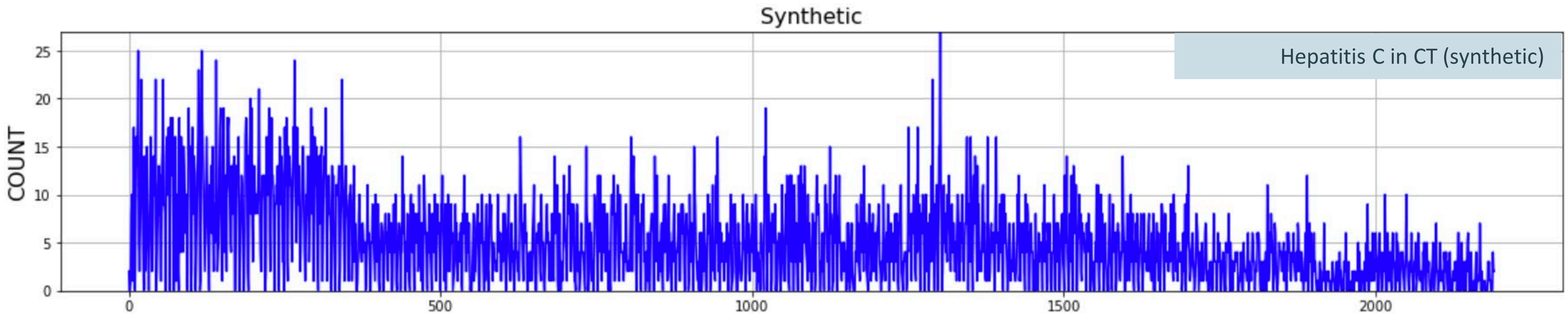
10,747 case records in source data (~4.9/day), roughly decreasing over time

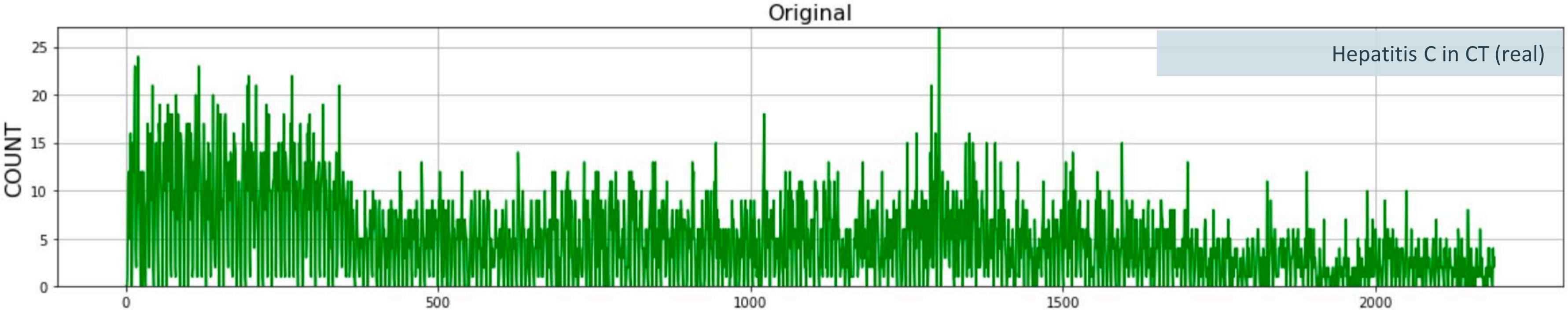


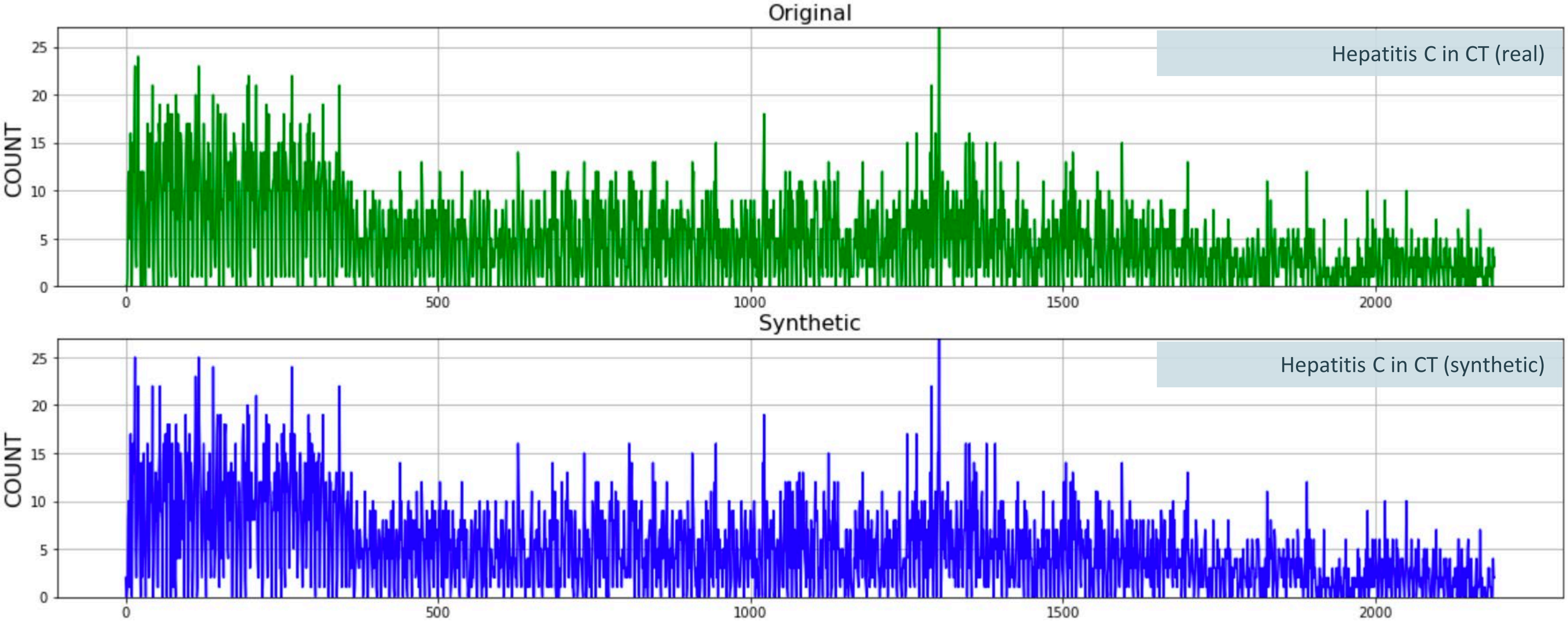
Synthetic data: time series of record counts

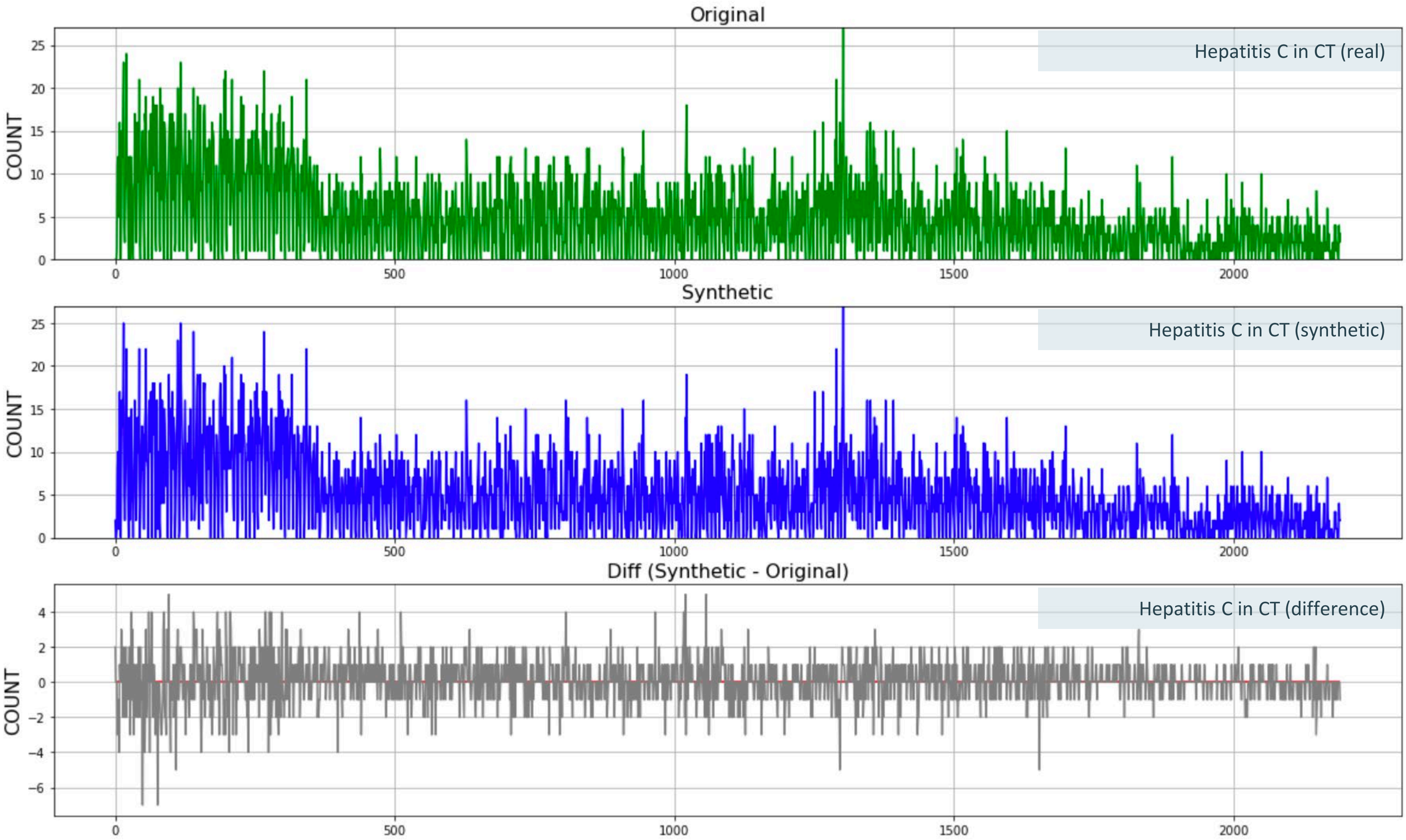
For each day in period, generate a random number of records

- Signal processing using Fourier methods and adaptive noise
- This step yields a (random) total number of records per day and over period
- Time series of synthetic record counts mimics properties of real record counts




Original
Hepatitis C in CT (real)
COUNT
25
20
15
10
5
0
0
500
1000
1500
2000


Original
Hepatitis C in CT (real)
COUNT
Synthetic
Hepatitis C in CT (synthetic)
COUNT


Original
Hepatitis C in CT (real)
COUNT
Synthetic
Hepatitis C in CT (synthetic)
COUNT
Diff (Synthetic - Original)
Hepatitis C in CT (difference)
COUNT

Synthetic data: time series for other conditions

Examples of 4 other conditions (in supplement)

- Lyme Disease (11080) in Connecticut
  - 4-5/day, seasonal with decreasing peaks and episodic spikes
- Campylobacteriosis (11020) in Georgia
  - 3-4/day, mildly seasonal with increasing peaks
- Cryptosporidiosis (11580) in Arizona
  - Outbreak around late 2016, otherwise <1/day
- Vibriosis (11545) in Massachusetts
  - Seasonal with generally low incidence, occasionally 1–2/day

Time-based incidence patterns

Case-level demographics

Case-level event dates

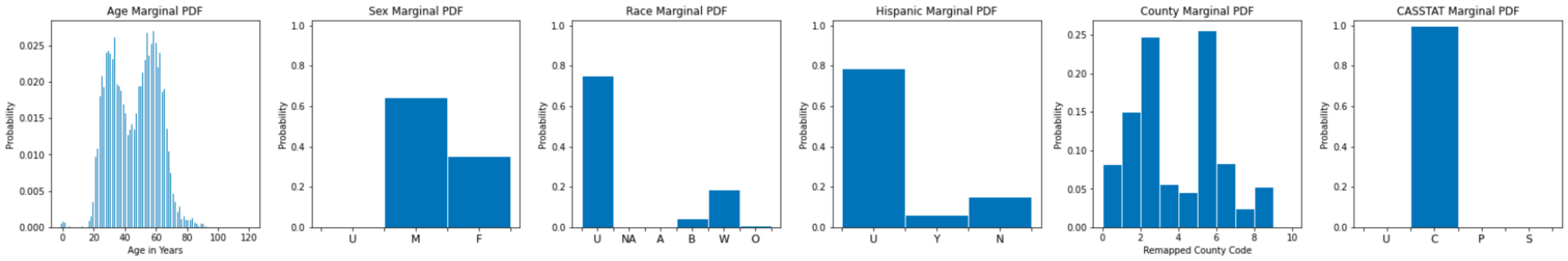
NNDSS Synthetic Case Generator

Real data: demographics (1)

Marginal distributions of age, sex, race, ethnicity, county, case status, and pregnancy status (only HL7 source)

Simulating each individual characteristic is straightforward

What if characteristics, such as age and race, are related?



Real data: demographics (2)

What if demographic characteristics are correlated?

- Suppose that the age distribution of a condition tends to be younger for persons of race A than for persons of race B
- If age and race are not independent in the source data, then we want synthetic data to show similar relationship

We want the synthetic demographics to mimic the marginal means *and covariation* of the real data

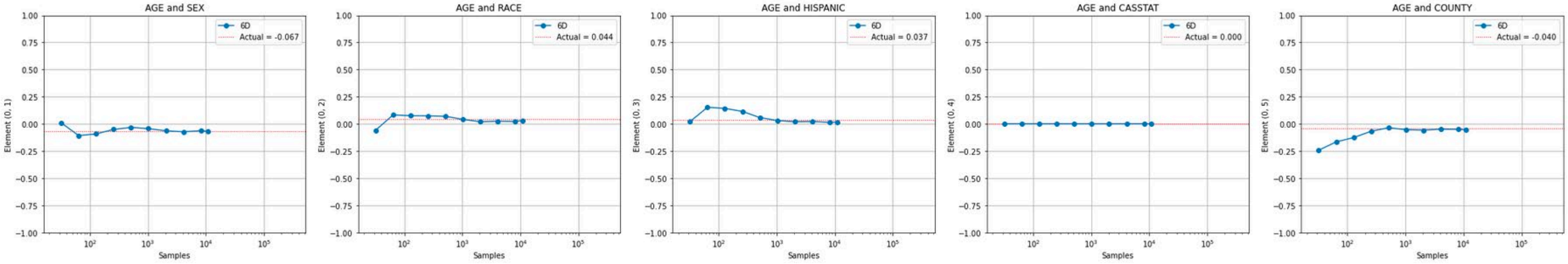
[Details sketched in supplemental slide]

Synthetic data: demographics

Generate with same univariate and bivariate properties

Check that fake data preserve properties of real data

- Look at the correlation of age with 5 other variables
- Focus on the *difference* between real and synthetic correlations; 0 is best
- With ~500 fake records, correlations are nearly identical



Time-based incidence patterns

Case-level demographics

Case-level event dates

NNDSS Synthetic Case Generator

Real data: event dates (1)

So far, the synthetic data broadly preserve 2 sets of properties

- Time-based properties of incidence
- Case-level demographic characteristics

Next, focus on 7 case-level event dates (listed by % available in source)

- First electronic submission (100%)
- Report (97%), derived from 4 types of reporting dates
- Start of investigation (71%)
- Onset of illness (48%)
- Diagnosis (38%)
- Admitted to hospital (5%)
- Death (1%)

Real data: event dates (2)

Case-level event dates would be complicated to model

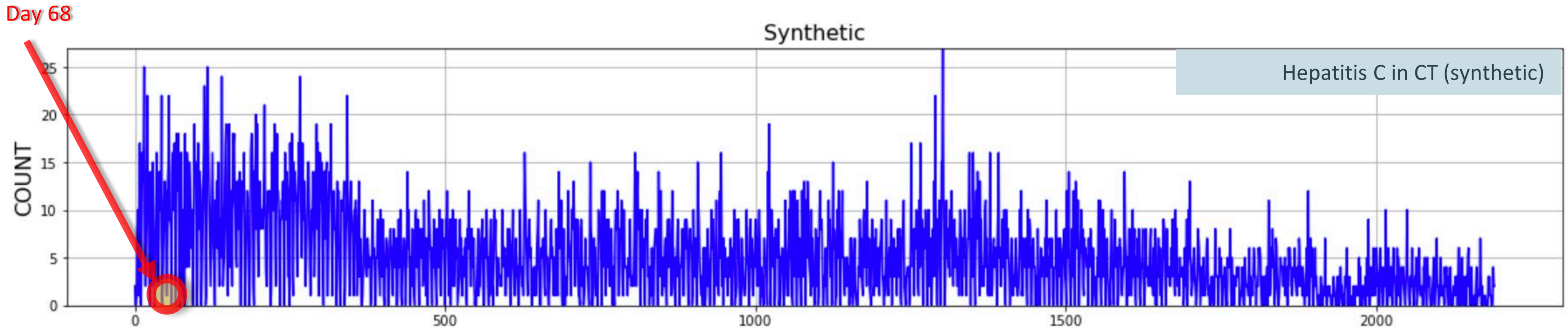
- Source dates are often blank
  - Some blanks represent incomplete data
  - Some events didn't happen, such as hospitalization and death
- A statistical model for dates would be complex and fragile

Let's walk through an example of generating synthetic event dates, then we'll summarize

Synthetic data: event date example (1)

Suppose we go through synthetic cases 1 by 1 and pause at day 68

- About 700 synthetic cases spread over days 0–67
- At day 68 (2021-03-10), by chance the generator places 1 event (see below)
- By chance, the generator assigns sex = female, status = confirmed case
- Using chance, the generator constructs event dates for this case, as follows



Synthetic data: event date example (2)

1. Synthetic: index date 2021-03-10, female (F), confirmed case (C)
2. Randomly sample 1 real record where sex=F and case status=C
3. In real record, compute earliest date and offsets
4. In synthetic record, event dates = index date + real offsets

Set	Sex	Stat	Onset	Diagnosis	Report	Admission	Death
Synth	F	C	TBD	TBD	TBD	TBD	TBD
Real							
Real							
Synth							

Synthetic data: event date example (3)

1. Synthetic: index date 2021-03-10, female (F), confirmed case (C)
2. Randomly sample 1 real record where sex=F and case status=C
3. In real record, compute earliest date and offsets
4. In synthetic record, event dates = index date + real offsets

Set	Sex	Stat	Onset	Diagnosis	Report	Admission	Death
Synth	F	C	TBD	TBD	TBD	TBD	TBD
Real	F	C	2019-08-14	2019-09-27	2019-09-29	NA	NA
Real							
Synth							

Synthetic data: event date example (4)

1. Synthetic: index date 2021-03-10, female (F), confirmed case (C)
2. Randomly sample 1 real record where sex=F and case status=C
3. In real record, compute earliest date and offsets
4. In synthetic record, event dates = index date + real offsets

Set	Sex	Stat	Onset	Diagnosis	Report	Admission	Death
Synth	F	C	TBD	TBD	TBD	TBD	TBD
Real	F	C	2019-08-14	2019-09-27	2019-09-29	NA	NA
Real	F	C	0	+44	+46	NA	NA
Synth							

Synthetic data: event date example (5)

1. Synthetic: index date 2021-03-10, female (F), confirmed case (C)
2. Randomly sample 1 real record where sex=F and case status=C
3. In real record, compute median‡ date and offsets
4. In synthetic record, event dates = index date + real offsets

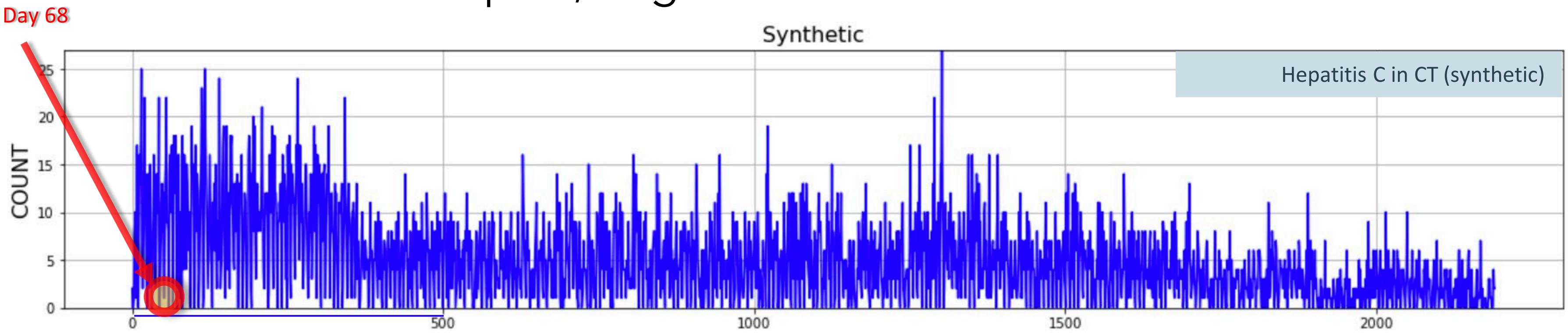
Set	Sex	Stat	Onset	Diagnosis	Report	Admission	Death
Synth	F	C	TBD	TBD	TBD	TBD	TBD
Real	F	C	2019-08-14	2019-09-27	2019-09-29	NA	NA
Real	F	C	0	+44	+46	NA	NA
Synth	F	C	2021-03-10	2021-04-23	2021-04-25	NA	NA

Synthetic data: event date example, summary

In synthetic generator

1. For a given synthetic record, note the case-level sex* and case status*
2. From real data, randomly sample a record with the same sex and case status
3. Index date (from synthetic record) + Date offsets (from sampled real record)

No need for a complex, fragile statistical model!



Main steps for generating synthetic case data

Select a jurisdiction and notifiable condition

Retrieve real NNDSS data, preprocess real data

Generate synthetic data in 3 steps

1. Time-based incidence patterns (# reports per day over ~6 years)
2. Case-level demographics
3. Case-level event dates

Sharing code and synthetic data

Restricted use

Unrestricted public use

Is it safe to share code and synthetic data?

Resemblance of fake to real data, if any, is coincidental

- Generator is trained on real data
- All generated data are fake
- *Is it possible to re-identify any source records with confidence? Especially for rare conditions or demographic combinations?*

3 considerations

- Share Python code
- Share synthetic data within CDC
- Share synthetic data outside CDC

Share generator code?

Generator code has been found to be secure

- Even if an intruder has the code and synthetic data, they cannot reverse engineer source values from the code

Generator code will be posted to a public Git repository

[All code is written in Python]

Share synthetic data?

Within CDC?

- Initial disclosure risk analysis supports sharing synthetic data
- Cautious restrictions, pending further disclosure risk analysis
- Real source data remain tightly restricted

Outside CDC?

- Assess views of CDC programs and jurisdictions
- Especially check infrequent patterns
  - Less common conditions
  - Less common demographics

Discussion

CDC contact information
Voice: 1-800-CDC-INFO (232-4636)
TTY: 1-888-232-6348
www.cdc.gov

Previous examples of synthetic health data

MITRE: Synthea

- Models of disease progression, standards of care → HL7/CDA records

CSELS: synthetic emergency department records

- Recurrent neural network to generate realistic chief complaints

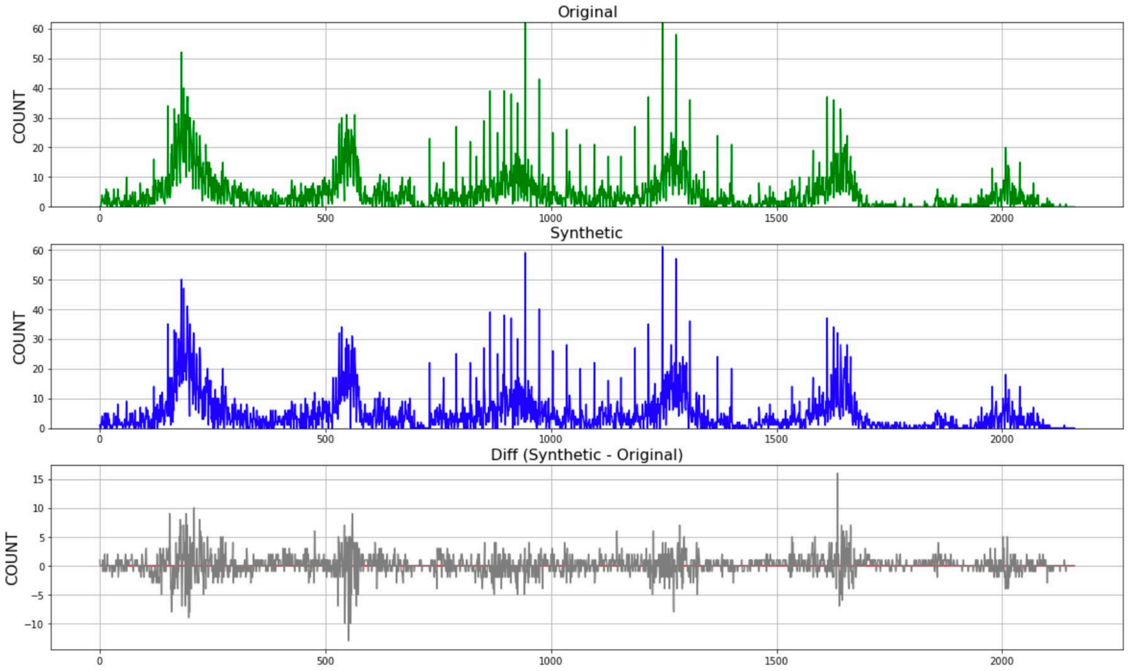
Georgia Tech Research Institute: Synthetic Health Data Generator

- Simulated drug exposures, condition exposures, and procedure occurrences

Georgia Tech: synthetic patient records

- Generative adversarial networks (a deep neural method)

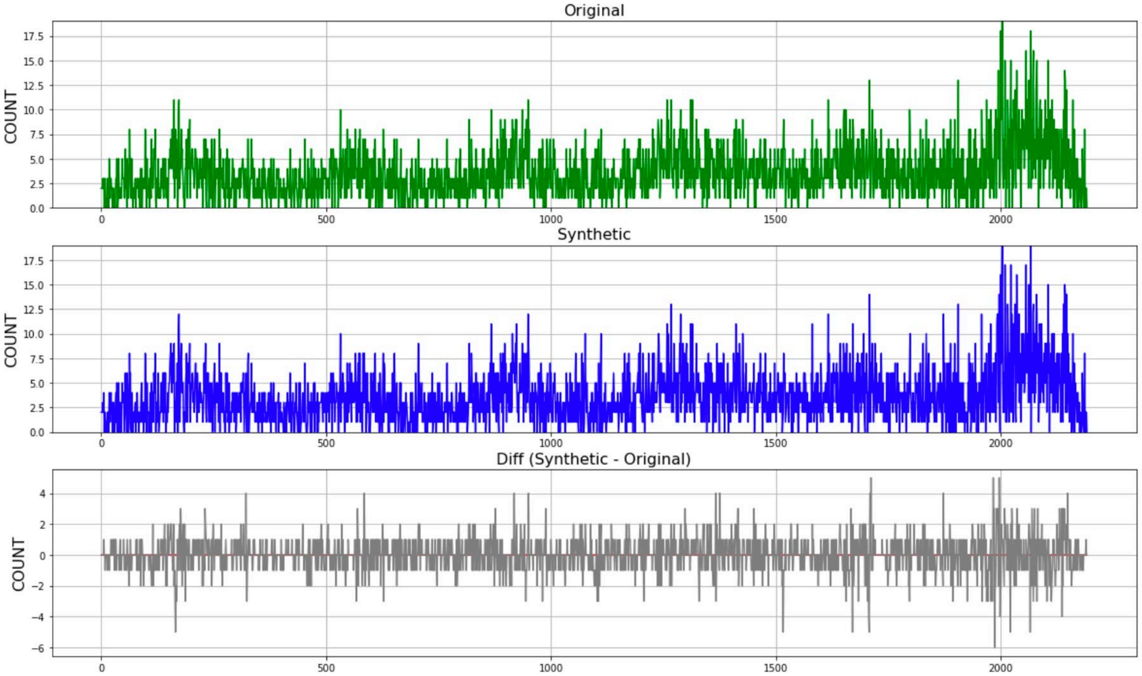
Lyme Disease (11080) in Connecticut
Series: 72 months, ~10,000 case records



Seasonal with decreasing peaks and episodic spikes

Campylobacteriosis (11020) in Georgia

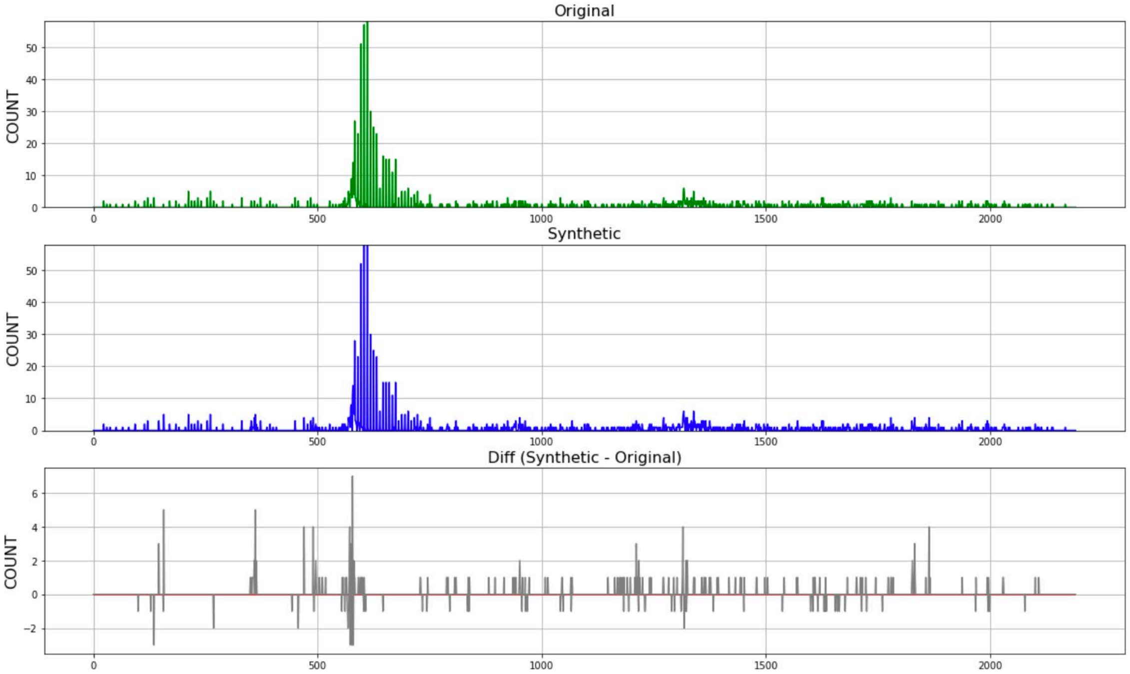
Series: 72 months, ~7,900 case records



Mildly seasonal with increasing peaks

Cryptosporidiosis (11580) in Arizona

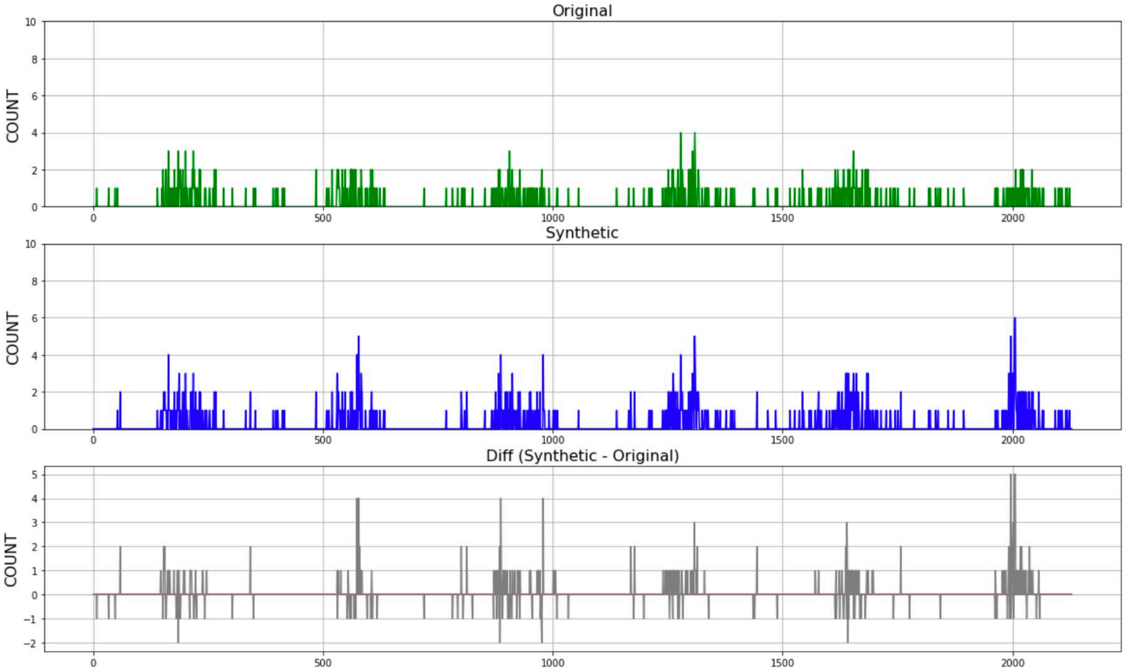
Series: 72 months, ~1,200 case records



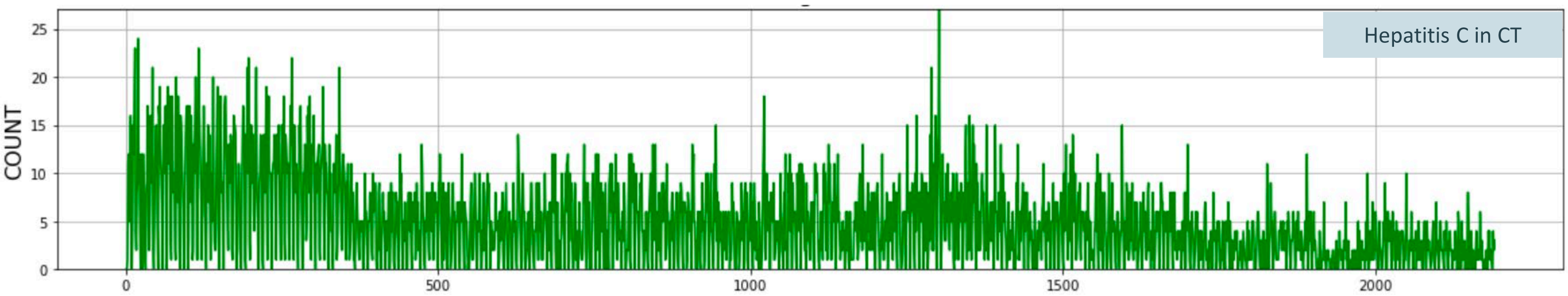
Outbreak around 600–750 days with a few cases most other days

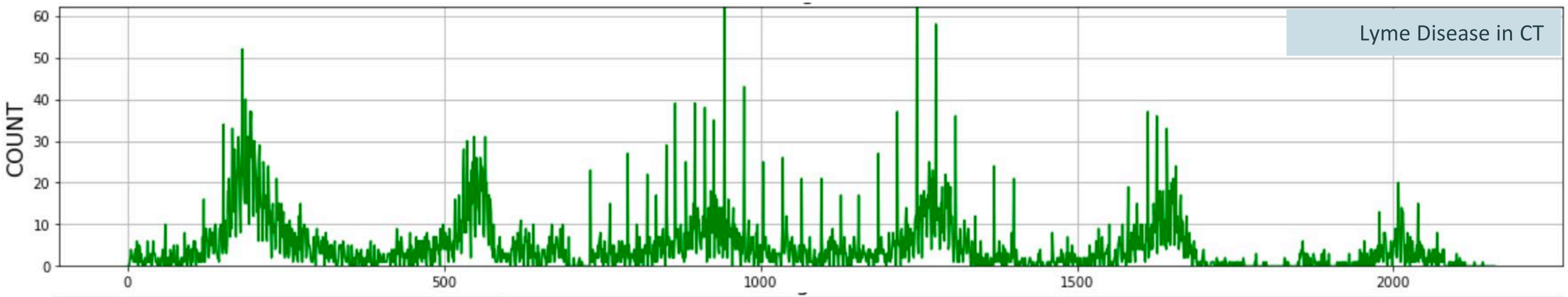
Vibriosis (11545) in Massachusetts

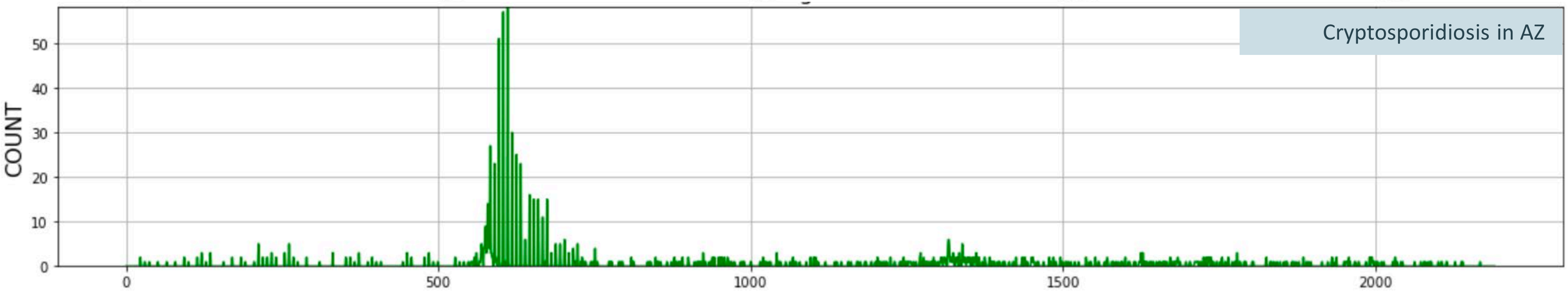
Series: 70 months, ~500 case records



Seasonal with generally low incidence of up to 1–2/day


Hepatitis C in CT
COUNT
0
5
10
15
20
25
0
500
1000
1500
2000


Lyme Disease in CT
COUNT
0
10
20
30
40
50
60
0
500
1000
1500
2000


Cryptosporidiosis in AZ
COUNT
0
10
20
30
40
50
0
500
1000
1500
2000

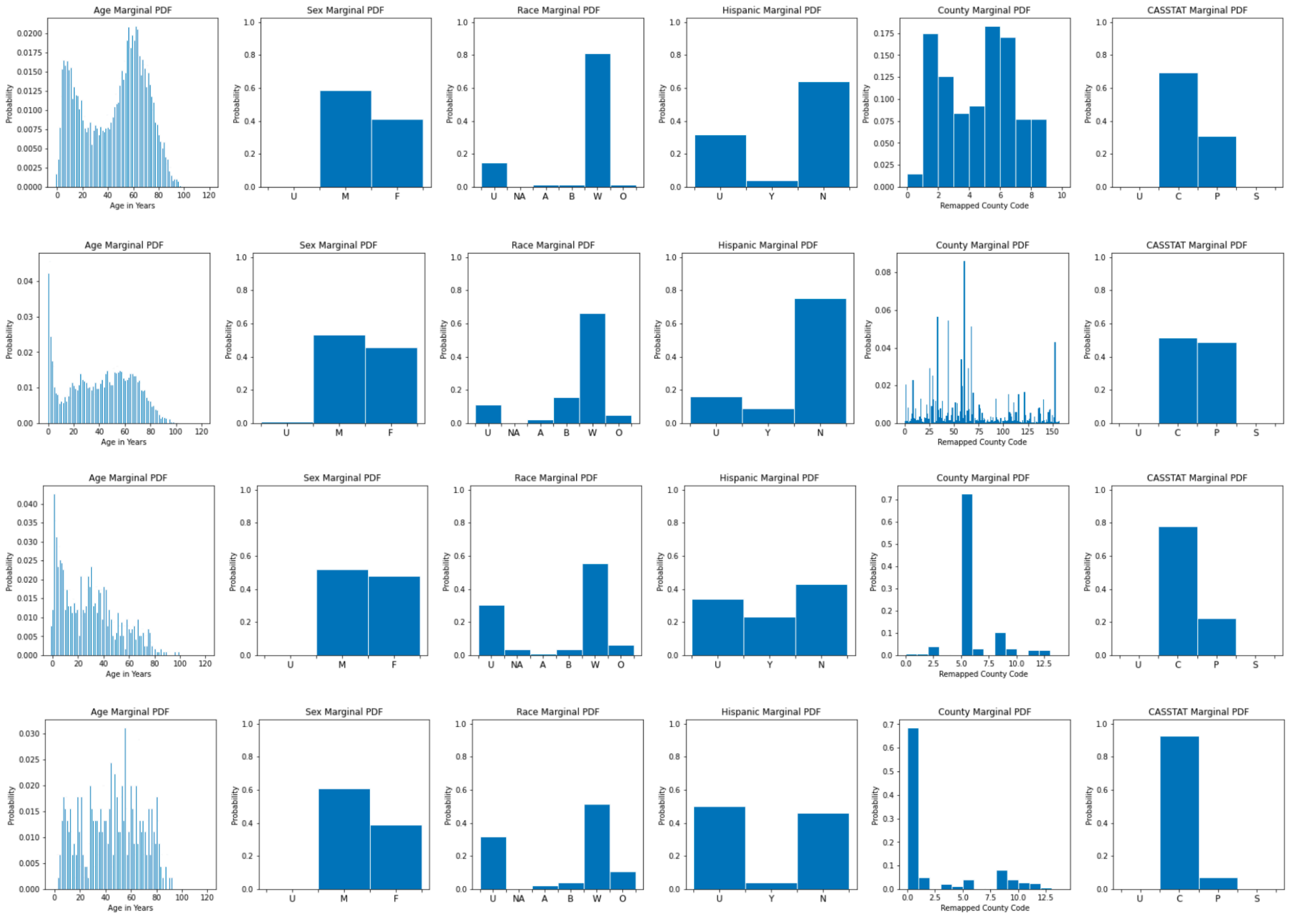
Mimic margins and covariation from real data

1. Start with observed, discrete values for all 6 or 7 characteristics
2. Probability transform: multivariate discrete → multivariate normal
3. Generate pseudorandom normals* with same mean and covariance
4. Inverse transform: multivariate normal* → multivariate discrete*
   (original scale)
5. Check that synthetic data preserve joint properties of real data

* marks synthetic data

More important than each step: This process works well.

Age Sex Race Ethnicity County Case status



Lyme Disease (11080) in CT

Campylobacteriosis (11020) in GA

Cryptosporidiosis (11580) in AZ

Vibriosis (11545) in MA

Comparison of synthetic and real data

Two jurisdictions: MI and AL

Exact matches:

- NETSS: 16% for MI; ~35% for AL
- HL7 Gen V2:
  - MI: 0.07% regardless of using reporting county or not
  - AL: 0.3% if using reporting county; 7% if using subject county

Matches on selected columns (excluding date columns)

- NETSS: 86% for MI; 95% for AL
- HL7 Gen V2:
  - MI: 81% if using reporting county; 30% if using subject county
  - AL: 80% if using reporting county; 72% if using subject county

Different impact on high-, medium-, or low-incidence conditions

Conclusion from the match analysis

Synthetic data occasionally match real source data

- More so for NETSS data

Prevalent conditions

- Should not be a problem

Low-incidence conditions

- More analyses may be needed to assess disclosure risks

Acknowledgements

GTRI:

Richard Boyd, Austin Himschoot, Aaron Potvien

CDC:

Leslyn McNabb, Sara Johnston, Tara Strine, Lesliann Helmus, Mukesh Hamal, Michelle Lin, Aaron Kite-Powell, Erik Knudsen, Lara Bull, Matt Ritchey, Chesley Richards, Aubrey Garner, Jen Adjemian, Paula Yoon, Michael Iademarco