An Interactive Spatial and Longitudinal Data Dashboard

Faysal Shaikh - CSI 695 - Fall 2021

Presentation overview

1. Background & data sources
2. Database design & creation (DDL)
3. Database manipulation (DML)
4. Final combined database
5. Linking PostgreSQL with Tableau
6. Interactive Tableau dashboard
7. Insights and lessons learned

1. Background & data sources

Timeline and impact of the U.S. opioid epidemic

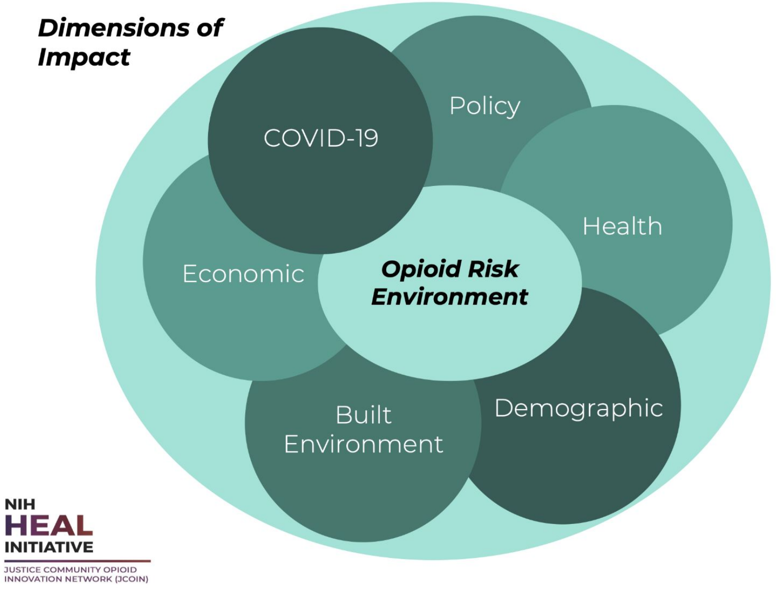
TIMELINE OF THE U.S. OPIOID EPIDEMIC[1]

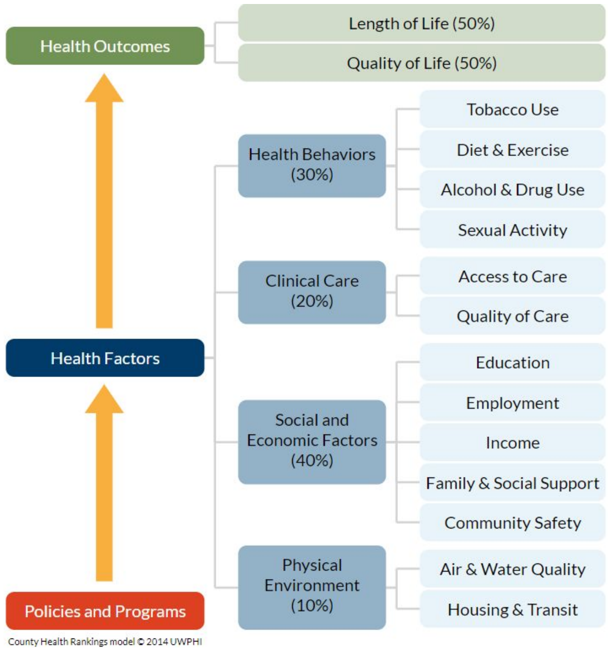
late 1990s	Pharmaceutical companies assure that patients will not become addicted to opioids and rates of opioid prescriptions begin to increase
2016	U.S. opioid overdoses account for over 42,000 deaths, more than any previous year on record
2017	HHS declares the U.S. opioid epidemic a “public health emergency” and announces “5-Point Strategy To Combat the Opioid Crisis”
2019	“Opioid-involved overdoses” account for nearly 50,000 deaths, a new all-time high since 2016

IMPACT OF THE U.S. OPIOID EPIDEMIC[2]

1. *Opioid overdose crisis.* (2021, March 11). National Institute on Drug Abuse. Retrieved November 16, 2021.
2. U.S. Department of Health and Human Services. (2021, October 27). *About the epidemic.* HHS.Gov/Opioids. Retrieved November 16, 2021.

OEPS and CHR opioid epidemic data sources





Opioid environmental policy scan (OEPS)[3]

County Health Rankings & Roadmaps (CHR)[4]

3. Kolak, M., Lin, Q., Paykin, S., Menghaney, M., & Li, A. (2021). *GeoDaCenter/opioid-policy-scan: Opioid environment policy scan data warehouse* (v0.1-beta) [Software]. Zenodo.
4. University of Wisconsin Population Health Institute & Robert Wood Johnson Foundation. County health rankings. County Health Rankings & Roadmaps. Retrieved Nov 16, 2021.

Research question(s)

OEPS: a cross-sectional, multifaceted description of the opioid risk environment.[5]

CHR: a longitudinal, multifaceted description of health factors and outcomes.

Both data sources describe U.S. county-level measures.

Does combining *cross-sectional OEPS data* and *longitudinal CHR data* further detail our picture of the opioid risk environment in U.S. counties?

Does an *interactive data dashboard* visualization of this data *reveal any novel insights* about the opioid risk environment[5] in U.S. counties?

5. Kolak, M. A., Chen, Y. T., Joyce, S., Ellis, K., Defever, K., McLuckie, C., Friedman, S., & Pho, M. T. (2020). Rural risk environments, opioid-related overdose, and infectious diseases: A multidimensional, spatial perspective. *International Journal of Drug Policy, 85.*

Special considerations for longitudinal CHR data

CHR data are available at yearly increments.

To reduce overall dataset size and maximize longitudinal follow-up,
we limit our data to **measures available from all years (2010 - 2021).**

We must utilize schema that can handle **several timepoints per county.**

Special considerations for spatial OEPS data

OEPS data are available at various spatial scales.

To reduce overall dataset size and maximize consistency with CHR data, we limit our data to **measures available at the county level.**

We must utilize a DBMS that can handle **geographic shapefile data**.

2. Database design & creation (DDL)

DBMS choice: PostgreSQL + PostGIS

Why PostgreSQL + PostGIS?

- Excellent SQL compliance
- Integration w/ many tools
- Extensive documentation
- Dedicated PostGIS CrunchyData interactive learning base
- Many additional features found in PostgreSQL over MySQL (esp. for spatial data purposes)

PostgreSQL i.e. POSTGRES

Starting PostgreSQL server & logging in

```
PS project_path> pg_ctl start
waiting for server to start....done
server started

PS project_path> psql -U postgres
Password for user postgres:
psql (13.4)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=#
```

Tablespace & database creation in postgreSQL

```
postgres=# CREATE TABLESPACE csi695
postgres-#   LOCATION 'D:\Codebase\CSI-695-project\combined-db_CHR-OEPS';
CREATE TABLESPACE

postgres=# CREATE DATABASE chr
postgres-#   WITH TABLESPACE = csi695;
CREATE DATABASE

postgres=# CREATE DATABASE oeps
postgres-#   WITH TABLESPACE = csi695;
CREATE DATABASE

postgres=# CREATE DATABASE combined_db_chr_oeps
postgres-#   WITH TABLESPACE = csi695;
CREATE DATABASE
```

Enable PostGIS extensions

```
postgres=# \connect combined_db_chr_oeps
You are now connected to database "combined_db_chr_oeps" as user "postgres".

combined_db_chr_oeps=# CREATE EXTENSION postgis;
CREATE EXTENSION
combined_db_chr_oeps=# CREATE EXTENSION postgis_topology;
CREATE EXTENSION
combined_db_chr_oeps=# CREATE EXTENSION postgis_sfcgal;
CREATE EXTENSION
combined_db_chr_oeps=# CREATE EXTENSION fuzzystrmatch;
CREATE EXTENSION
combined_db_chr_oeps=# CREATE EXTENSION address_standardizer;
CREATE EXTENSION
combined_db_chr_oeps=# CREATE EXTENSION postgis_tiger_geocoder;
CREATE EXTENSION
...
```

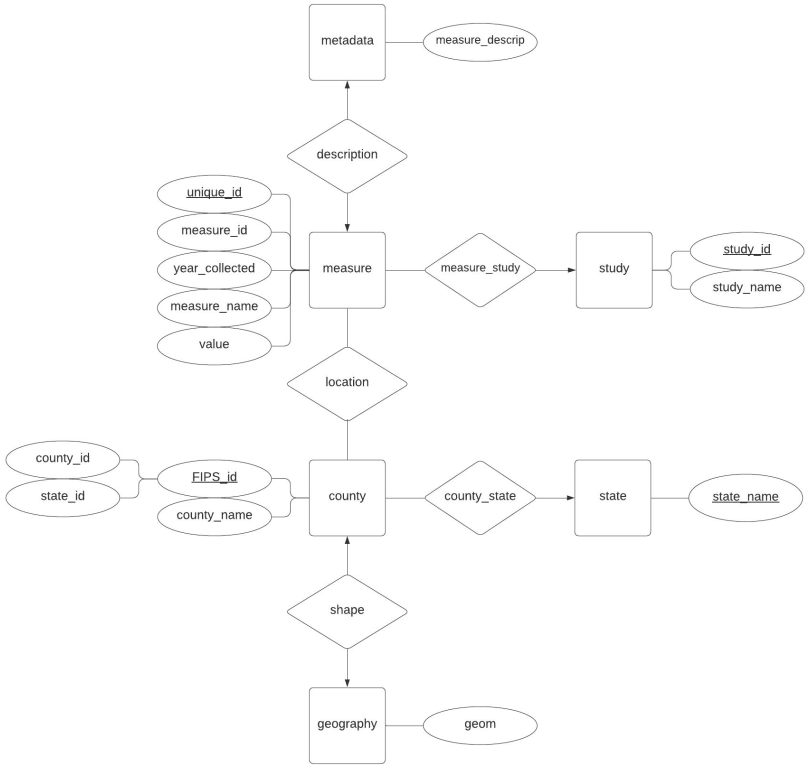
Entity-Relationship (ER) model for OEPS data

Lucidchart

Entity-Relationship (ER) model for CHR data

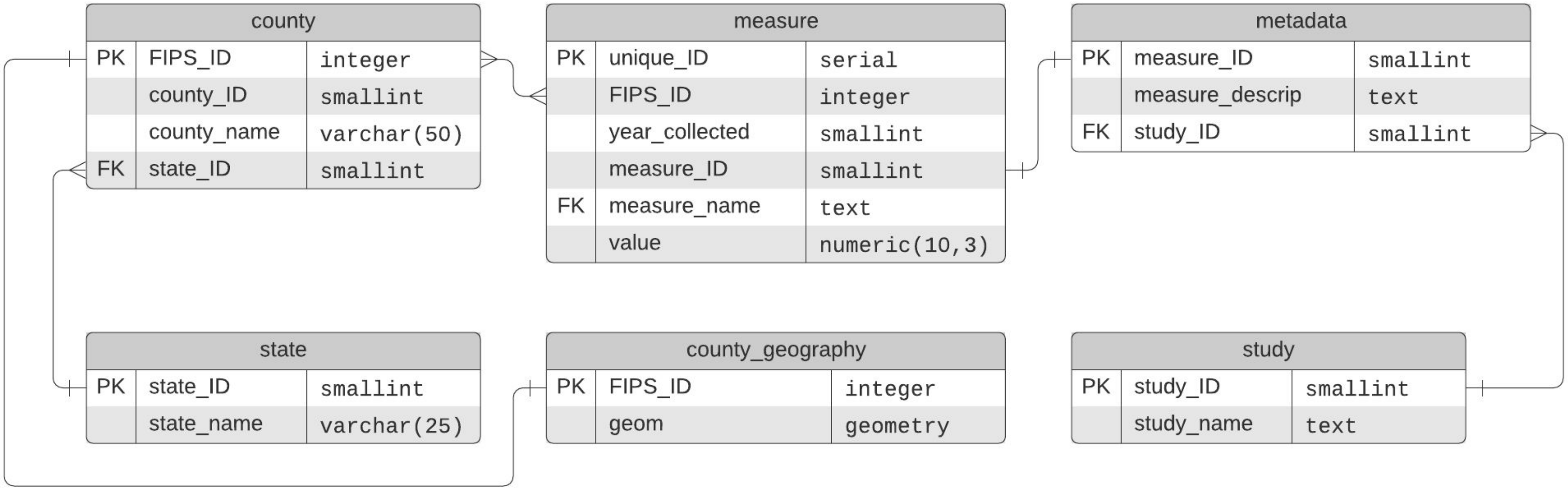
Lucidchart

Entity-Relationship (ER) model for combined data



Lucidchart

Relational model for combined data



Lucidchart

Relation (table) creation in PostgreSQL

```
postgres=# \connect combined_db_chr_oeps
You are now connected to database "combined_db_chr_oeps" as user "postgres".

combined_db_chr_oeps=# CREATE TABLE measure (
combined_db_chr_oeps(#    unique_ID smallserial PRIMARY KEY,
combined_db_chr_oeps(#    FIPS_ID integer,
combined_db_chr_oeps(#    year_collected smallint,
combined_db_chr_oeps(#    measure_ID smallint,
combined_db_chr_oeps(#    measure_name text,
combined_db_chr_oeps(#    value numeric(10,3),
combined_db_chr_oeps(# );
CREATE TABLE
```

Relation (table) creation in PostgreSQL

```
combined_db_chr_oeps=# CREATE TABLE county (
combined_db_chr_oeps(#    FIPS_ID integer PRIMARY KEY,
combined_db_chr_oeps(#    county_ID smallint,
combined_db_chr_oeps(#    county_name varchar(50),
combined_db_chr_oeps(#    state_ID smallint
combined_db_chr_oeps(# );
CREATE TABLE

combined_db_chr_oeps=# CREATE TABLE state (
combined_db_chr_oeps(#   state_ID smallint PRIMARY KEY,
combined_db_chr_oeps(#   state_name varchar(25)
combined_db_chr_oeps(# );
CREATE TABLE
```

Relation (table) creation in PostgreSQL

```
combined_db_chr_oeps=# CREATE TABLE metadata (
combined_db_chr_oeps(#   measure_ID serial PRIMARY KEY,
combined_db_chr_oeps(#   measure_descrip text,
combined_db_chr_oeps(#   study_ID smallint
combined_db_chr_oeps(# );
CREATE TABLE

combined_db_chr_oeps=# CREATE TABLE study (
combined_db_chr_oeps(#   study_ID smallint PRIMARY KEY,
combined_db_chr_oeps(#   study_name text
combined_db_chr_oeps(# );
CREATE TABLE
```

Relation (table) creation in PostgreSQL

```
combined_db_chr_oeps=# ALTER TABLE measure
combined_db_chr_oeps-#   ADD FOREIGN KEY (measure_ID)
combined_db_chr_oeps-#   REFERENCES metadata (measure_ID);
ALTER TABLE

combined_db_chr_oeps=# ALTER TABLE county
combined_db_chr_oeps-#   ADD FOREIGN KEY (state_ID)
combined_db_chr_oeps-#   REFERENCES state (state_ID);
ALTER TABLE

combined_db_chr_oeps=# ALTER TABLE metadata
combined_db_chr_oeps-#   ADD FOREIGN KEY (study_ID)
combined_db_chr_oeps-#   REFERENCES study (study_ID);
ALTER TABLE
```

Relation (table) creation in PostgreSQL

```
combined_db_chr_oeps=# CREATE TABLE county_geography (
combined_db_chr_oeps(#   FIPS_ID integer PRIMARY KEY,
combined_db_chr_oeps(#   geom geometry
combined_db_chr_oeps(# );
CREATE TABLE

combined_db_chr_oeps=# ALTER TABLE county_geography
combined_db_chr_oeps-#   ADD FOREIGN KEY (FIPS_ID)
combined_db_chr_oeps-#   REFERENCES county (FIPS_ID);
ALTER TABLE
```

(I was not able to implement this from the CLI, but this is likely what it **would have** looked like!)

3. Database manipulation (DML)

Preprocessing raw OEPS & CHR data in R

Output preview of `preprocessing_oeps.R`:

```
"COUNTYFP","totPopE","moudMinDis","
bupMinDis","methMinDis","nalMinDis"
,"totUnits","occP","vacantP","mobil
eP","lngTermP","rentalP","unitDens"
,"rcaUrbP","rcaSubrbP","rcaRuralP",
"totPop10","urbPop10","rurlPop10","
cenRuralP","areaSqMi","alcTotal","a
lcDens","alcPerCap","DmySgrg","DmyB
lckBlt","PrcNtvRsrv","dissim.b","in
ter.bw","iso.b","dissim.h","inter.h
w","iso.h","dissim.a","inter.aw",
...
```

`preprocessed_oeps.csv`

Output preview of `preprocessing_chr.R`:

```
"FIPS","State","County",
"% Smokers 2010","Teen Birth Rate
2010","% Uninsured 2010","%
Children in Poverty 2010","Violent
Crime Rate 2010",
"% Smokers 2011","Teen Birth Rate
2011","% Uninsured 2011","%
Children in Poverty 2011","Violent
Crime Rate 2011",
...
```

`preprocessed_chr.csv`

Pseudocode for `preprocessing_oeps.R`

```
# preprocessing_oeps.R
# This program serves to preprocess OEPS data files.

# import libraries: git2r, readr, dplyr, stringr
# specify directory containing all OEPS data files

# first pass: loop all files and create separate data frame objects
# second pass: execute sequential (in order) cumulative pairwise merges

# drop columns we don't want (mostly variations of “county”)
    # in fact, skipped some original data files for “atrocious variations”

# save preprocessed data as CSV file
```

Pseudocode for `preprocessing_chr.R`

```
# preprocessing_chr.R
# This program serves to preprocess CHR data files.

# import libraries: git2r, readxl, dplyr
# specify directory containing all CHR data files

# first pass: loop all files and create separate data frame objects
# second pass: loop data frames to discover columns common between all years
    # sequential (in increasing year) pairwise intersections, successively
    removing columns until only those found in all years are left

# third pass: loop through objects and only keep the common columns
# final pass: merge all data by FIPS, State, County

# save preprocessed data as CSV file
```

Importing preprocessed data into PostgreSQL

I decided to write a “quick” python script to generate an SQL script for data loading (like `LoadData.txt` from class).

We must define `measure_ID` attribute in advanced, or we won’t know the values for the `metadata` table.

The `measure` relation is defined as:

`measure (FIPS_ID, year_collected, measure_ID, measure_name, value)`

Here is an excerpt from the result:

```
INSERT INTO measure VALUES (1001,
NULL, 0, 'totPopE', 55200.0);

INSERT INTO measure VALUES (1001,
NULL, 1, 'moudMinDis',
15.74711697);

INSERT INTO measure VALUES (1001,
NULL, 2, 'bupMinDis', 15.74711697);

...
```

Output preview of `oeps_sql_loader.py`

Importing preprocessed data into PostgreSQL

The previous slide showcased loading of (cross-sectional) OEPS data.

The process is similar for (longitudinal) CHR data, but we must **resume our `measure_ID` from where we left off in the OEPS data.** (This is hardcoded.)

The `measure` relation is defined as:

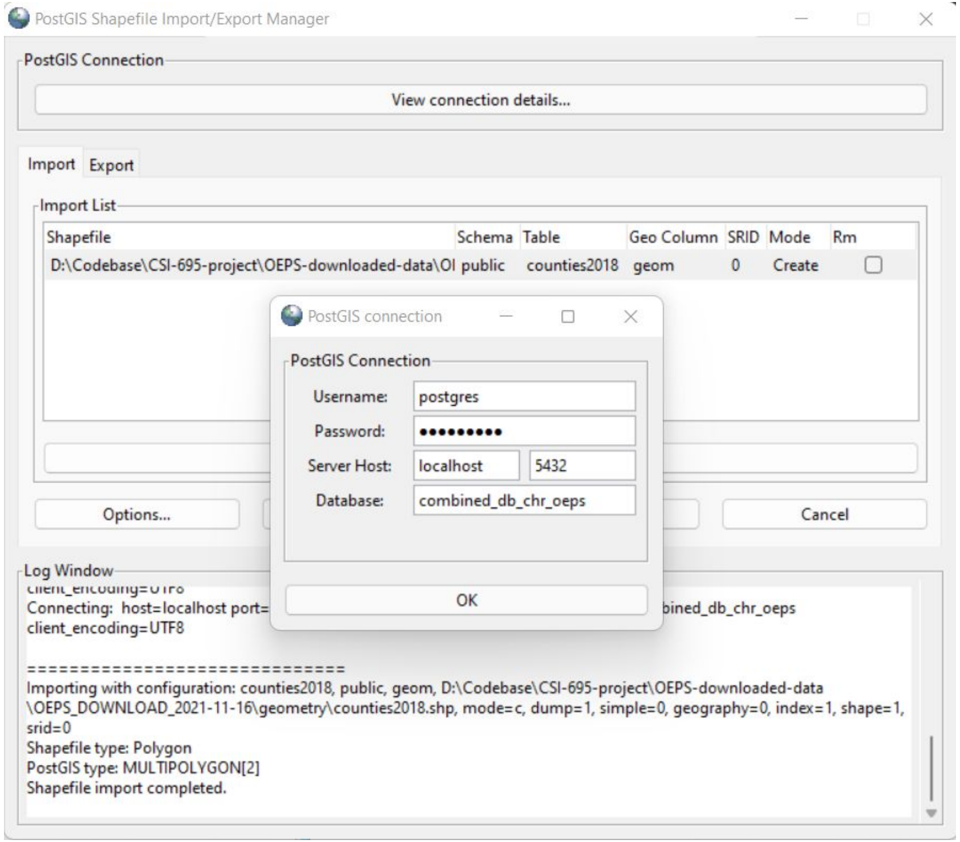
```
measure (FIPS_ID, year_collected,
measure_ID, measure_name, value)
```

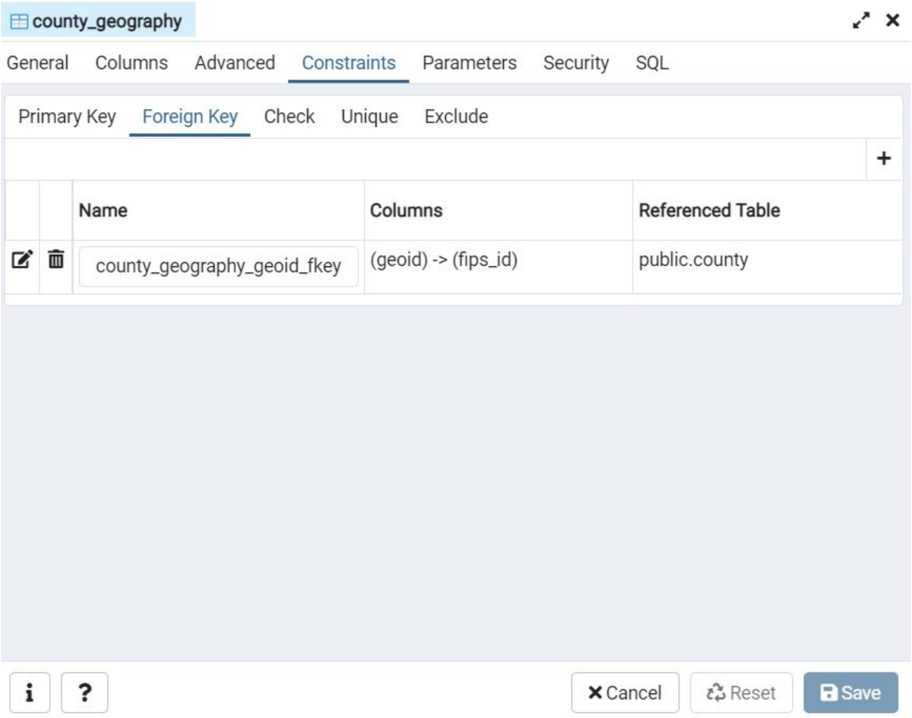
Here is an excerpt from the result:

```
INSERT INTO measure VALUES (1001,
2010, 69, 'Percentage Smokers',
28.14);

INSERT INTO measure VALUES (1001,
2010, 70, 'Teen Birth Rate', 52.6);

INSERT INTO measure VALUES (1001,
2010, 71, 'Percentage Uninsured',
14.0);

...
```

Output preview of `chr_sql_loader.py`

Importing OEPS county shapefiles into PostGIS





4. Final combined database

Row counts from final combined database

```
combined_db_chr_oeps=# SELECT COUNT(*) FROM measure;
394790

combined_db_chr_oeps=# SELECT COUNT(*) FROM county;
2746

combined_db_chr_oeps=# SELECT COUNT(*) FROM county_geography;
3142

combined_db_chr_oeps=# SELECT COUNT(*) FROM state;
51

combined_db_chr_oeps=# SELECT COUNT(*) FROM metadata;
74

combined_db_chr_oeps=# SELECT COUNT(*) FROM study;
2
```

Available data years from final database

```
combined_db_chr_oeps=# SELECT DISTINCT year_collected FROM measure;

 year_collected
----------------
           2013
           2021
           2020

           2015
           2011
           2014
           2010
           2017
           2019
           2016
           2012
           2018
(13 rows)
```

Setup for more complex queries

```
combined_db_chr_oeps=# SELECT * FROM measure LIMIT 0;
 fips_id | year_collected | measure_id | measure_name | value | unique_id
---------+----------------+------------+--------------+-------+----------
(0 rows)


combined_db_chr_oeps=# SELECT * FROM county LIMIT 0;
 fips_id | county_id | county_name | state_id
---------+-----------+-------------+----------
(0 rows)


combined_db_chr_oeps=# SELECT * FROM state LIMIT 0;
 state_id | state_name
----------+------------
(0 rows)
```

Yearly percentage smokers in Fairfax City

```
combined_db_chr_oeps=# SELECT year_collected, measure_name, value FROM measure
combined_db_chr_oeps-#   INNER JOIN county USING(fips_id) INNER JOIN state USING(state_id)
combined_db_chr_oeps-#   WHERE state_name = 'Virginia' AND county_name = 'Fairfax City'
combined_db_chr_oeps-#     AND year_collected IS NOT NULL
combined_db_chr_oeps=#     AND measure_name = 'Percentage Smokers';

 year_collected |    measure_name    | value
----------------+--------------------+--------
           2010 | Percentage Smokers |  2.410
           2011 | Percentage Smokers |  3.400
           2012 | Percentage Smokers |  5.700
           2013 | Percentage Smokers |  6.000
           2016 | Percentage Smokers | 13.200
           2017 | Percentage Smokers | 10.729
           2018 | Percentage Smokers | 11.509
           2019 | Percentage Smokers | 11.509
           2020 | Percentage Smokers | 12.665
           2021 | Percentage Smokers | 11.625
(10 rows)
```

Time-average percentage smokers in Fairfax City

```
combined_db_chr_oeps=# SELECT year_collected, measure_name, AVG(value)
combined_db_chr_oeps-#   OVER (PARTITION BY fips_id, measure_name) FROM measure
combined_db_chr_oeps-#   INNER JOIN county USING(fips_id) INNER JOIN state USING(state_id)
combined_db_chr_oeps-#   WHERE state_name = 'Virginia' AND county_name = 'Fairfax City'
combined_db_chr_oeps-#     AND year_collected IS NOT NULL
combined_db_chr_oeps-#     AND measure_name = 'Percentage Smokers';

 year_collected |    measure_name    |        avg
----------------+--------------------+--------------------
           2010 | Percentage Smokers | 8.8747000000000000
           2011 | Percentage Smokers | 8.8747000000000000
           2012 | Percentage Smokers | 8.8747000000000000
           2013 | Percentage Smokers | 8.8747000000000000
           2016 | Percentage Smokers | 8.8747000000000000
           2017 | Percentage Smokers | 8.8747000000000000
           2018 | Percentage Smokers | 8.8747000000000000
           2019 | Percentage Smokers | 8.8747000000000000
           2020 | Percentage Smokers | 8.8747000000000000
           2021 | Percentage Smokers | 8.8747000000000000
(10 rows)
```

High time-average percentage smoker VA counties

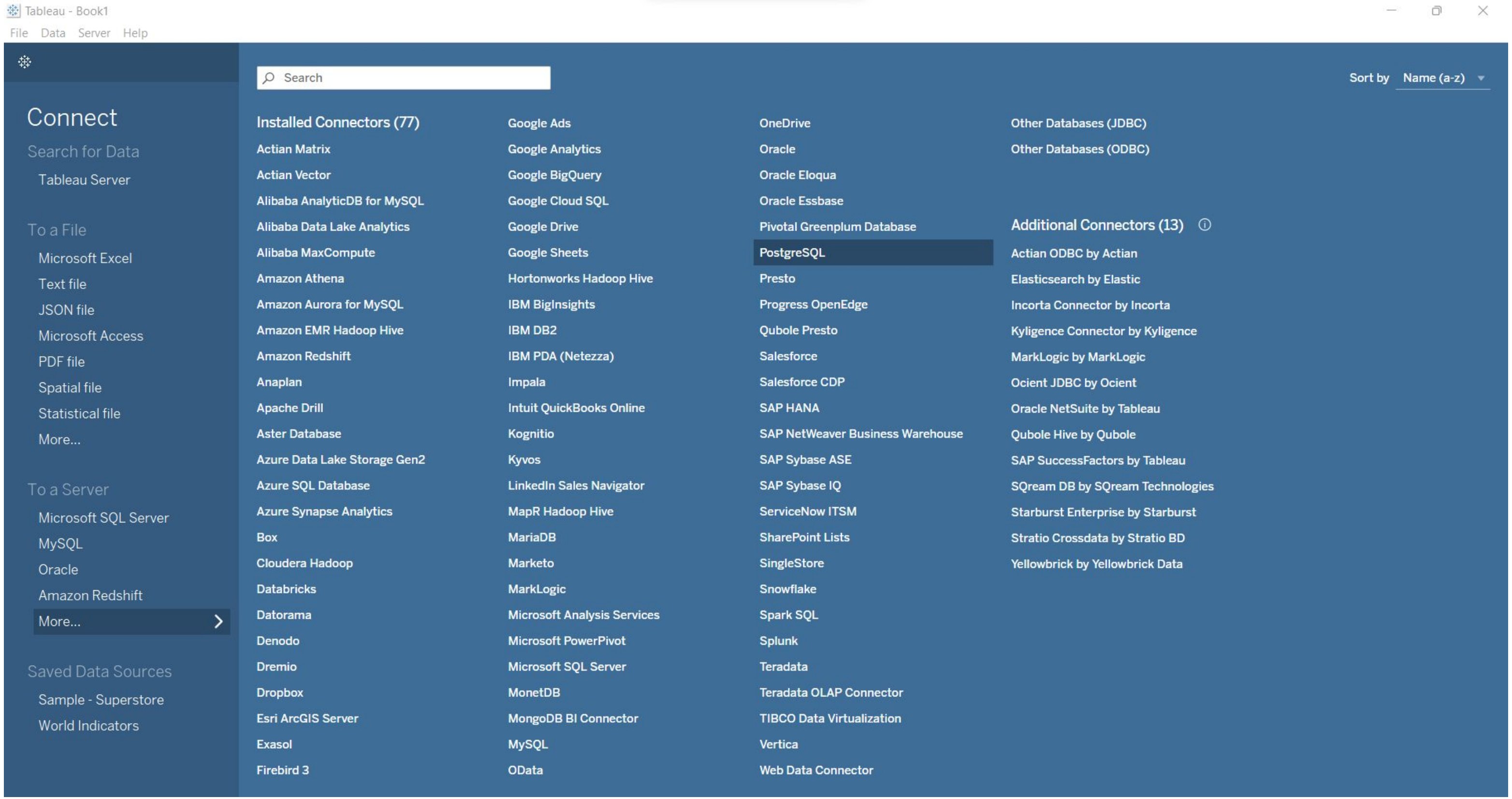
```
combined_db_chr_oeps=# SELECT DISTINCT county_name, state_name, measure_name, AVG(value)
combined_db_chr_oeps-#   OVER (PARTITION BY fips_id, measure_name) FROM measure
combined_db_chr_oeps-#   INNER JOIN county USING(fips_id) INNER JOIN state USING(state_id)
combined_db_chr_oeps-#   WHERE state_name = 'Virginia' AND year_collected IS NOT NULL
combined_db_chr_oeps-#     AND measure_name = 'Percentage Smokers'
combined_db_chr_oeps-#   ORDER BY AVG DESC LIMIT 10;

 county_name | state_name |    measure_name    |        avg
-------------+------------+--------------------+---------------------
 Wise        | Virginia   | Percentage Smokers | 27.4436666666666667
 Dickenson   | Virginia   | Percentage Smokers | 26.9171818181818182
 Carroll     | Virginia   | Percentage Smokers | 26.8813333333333333
 Buchanan    | Virginia   | Percentage Smokers | 25.9701666666666667
 Dinwiddie   | Virginia   | Percentage Smokers | 24.9875833333333333
 Pulaski     | Virginia   | Percentage Smokers | 24.8219166666666667
 Lee         | Virginia   | Percentage Smokers | 24.7314166666666667
 Henry       | Virginia   | Percentage Smokers | 24.4219166666666667
 Scott       | Virginia   | Percentage Smokers | 24.0952500000000000
 Mecklenburg | Virginia   | Percentage Smokers | 23.9395833333333333
(10 rows)
```

Low time-average percentage smoker VA counties

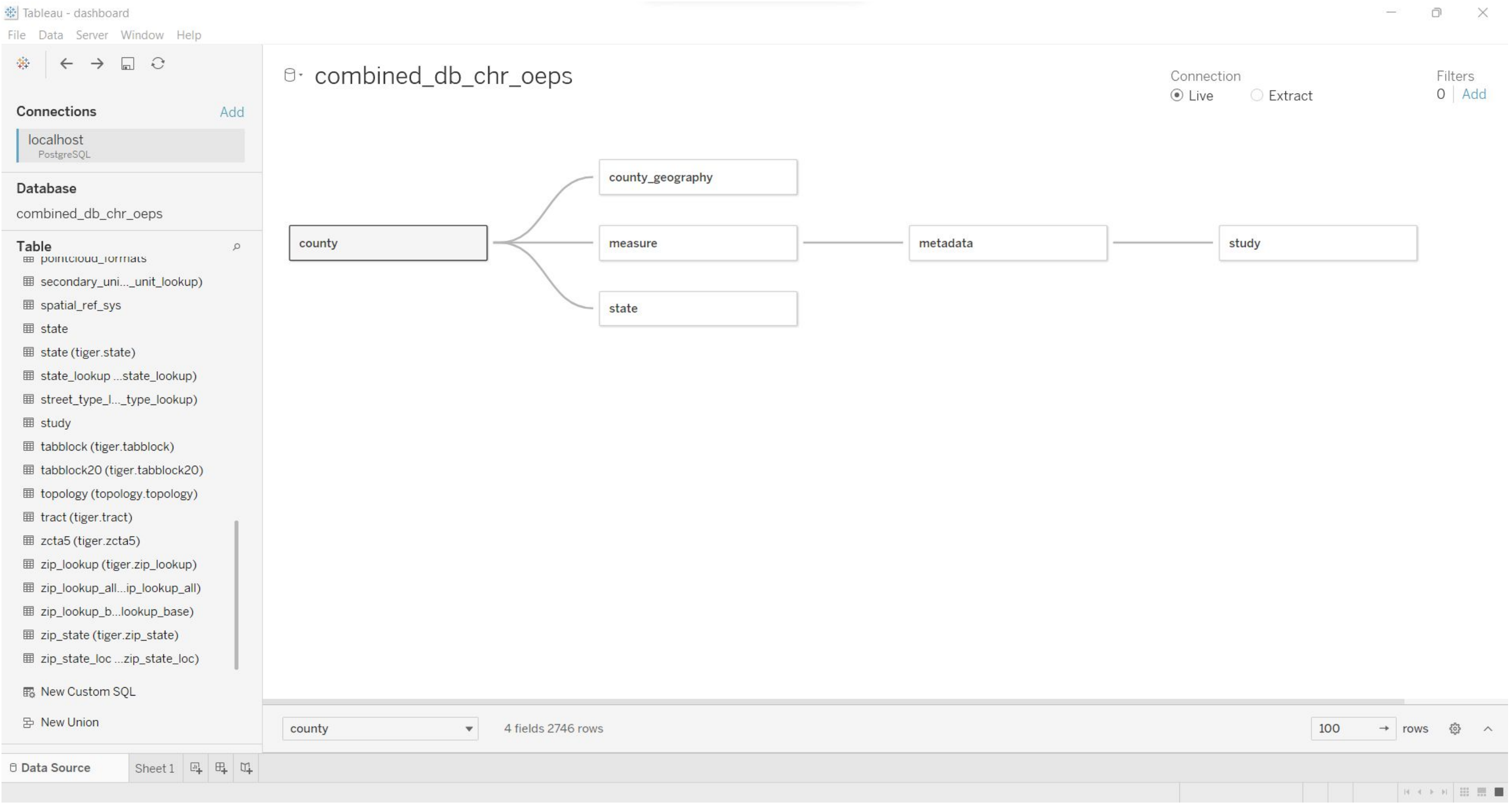
```
combined_db_chr_oeps=# SELECT DISTINCT county_name, state_name, measure_name, AVG(value)
combined_db_chr_oeps-#   OVER (PARTITION BY fips_id, measure_name) FROM measure
combined_db_chr_oeps-#   INNER JOIN county USING(fips_id) INNER JOIN state USING(state_id)
combined_db_chr_oeps-#   WHERE state_name = 'Virginia' AND year_collected IS NOT NULL
combined_db_chr_oeps-#     AND measure_name = 'Percentage Smokers'
combined_db_chr_oeps-#   ORDER BY AVG ASC LIMIT 10;

 county_name  | state_name |    measure_name    |         avg
--------------+------------+--------------------+---------------------
 Fairfax City | Virginia   | Percentage Smokers |  8.8747000000000000
 York         | Virginia   | Percentage Smokers | 11.4976666666666667
 Fairfax      | Virginia   | Percentage Smokers | 11.5245833333333333
 Loudoun      | Virginia   | Percentage Smokers | 11.6141666666666667
 Arlington    | Virginia   | Percentage Smokers | 11.6650000000000000
 James City   | Virginia   | Percentage Smokers | 12.3024166666666667
 Fluvanna     | Virginia   | Percentage Smokers | 12.9787000000000000
 Albemarle    | Virginia   | Percentage Smokers | 13.6415000000000000
 Botetourt    | Virginia   | Percentage Smokers | 14.4337500000000000
 Rockingham   | Virginia   | Percentage Smokers | 14.6154166666666667
(10 rows)
```

5. Linking PostgreSQL with Tableau

Connecting PostgreSQL with Tableau Desktop



Connecting PostgreSQL with Tableau Desktop



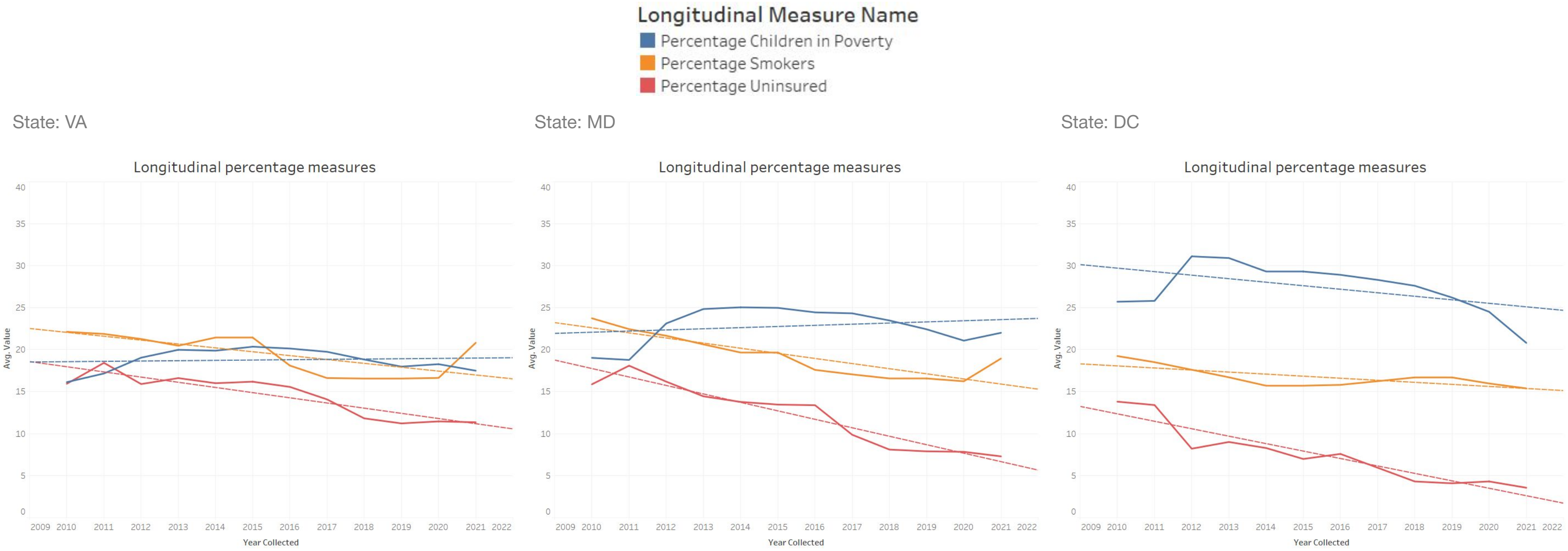
6. Interactive Tableau dashboard

Interactive data dashboard in Tableau Desktop

Interactive data dashboard demo.

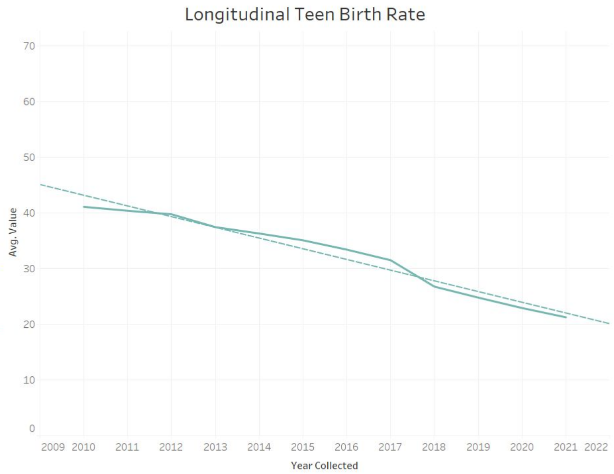
7. Insights and lessons learned

Risk factors (child poverty, smokers, uninsured, teen birth rate) have generally declined in the DMV area from 2010 onward.

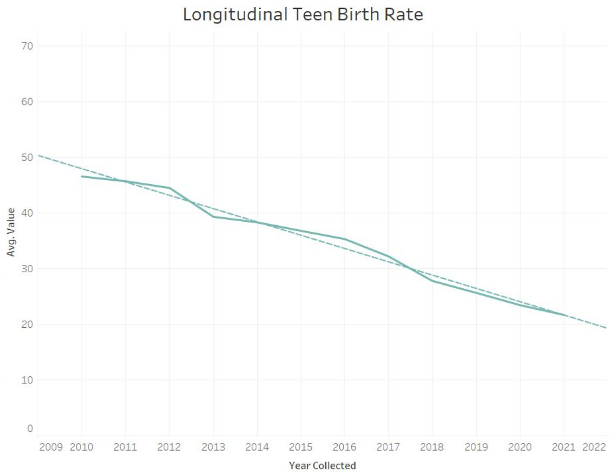


Risk factors (child poverty, smokers, uninsured, teen birth rate) have generally declined in the DMV area from 2010 onward.

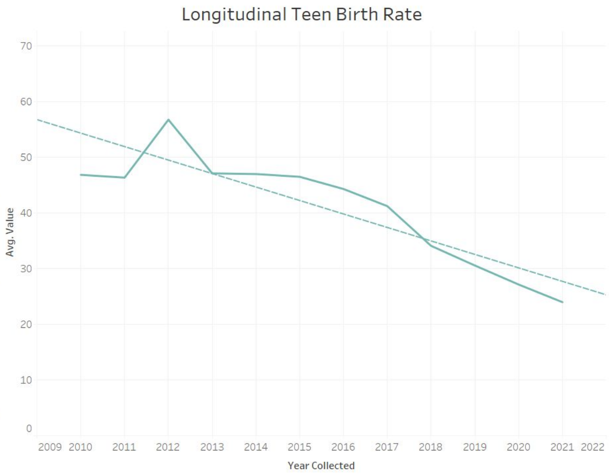
State: VA



State: MD

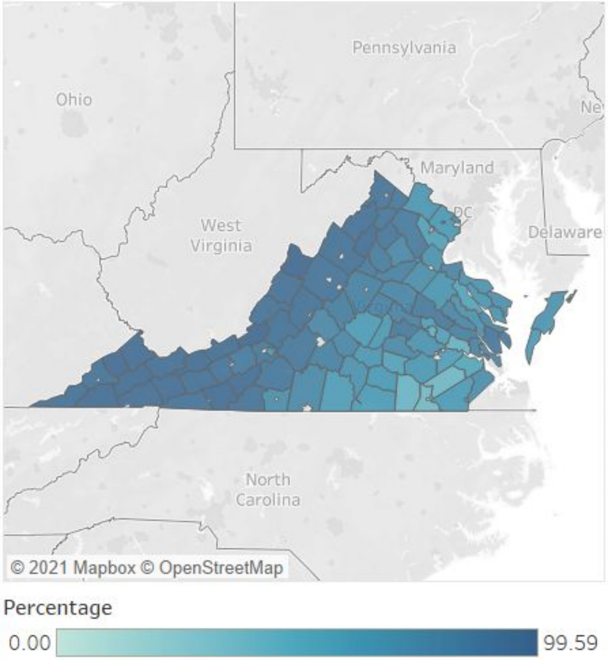


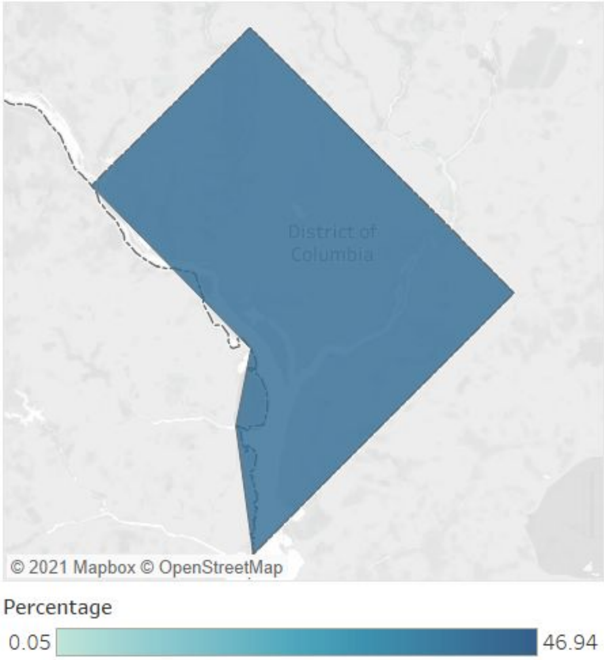
State: DC

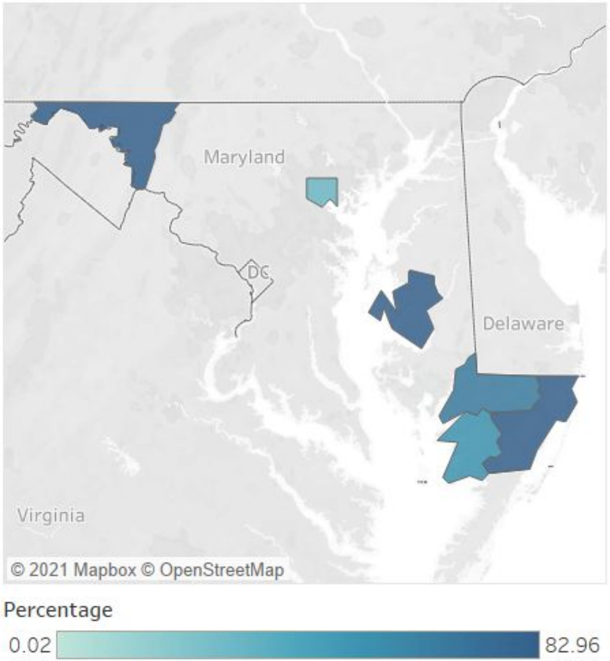


Cross-sectional maps of percentage population by race (e.g., white) seem to correlate spatially with minimum distance to clinics (e.g., naloxone).

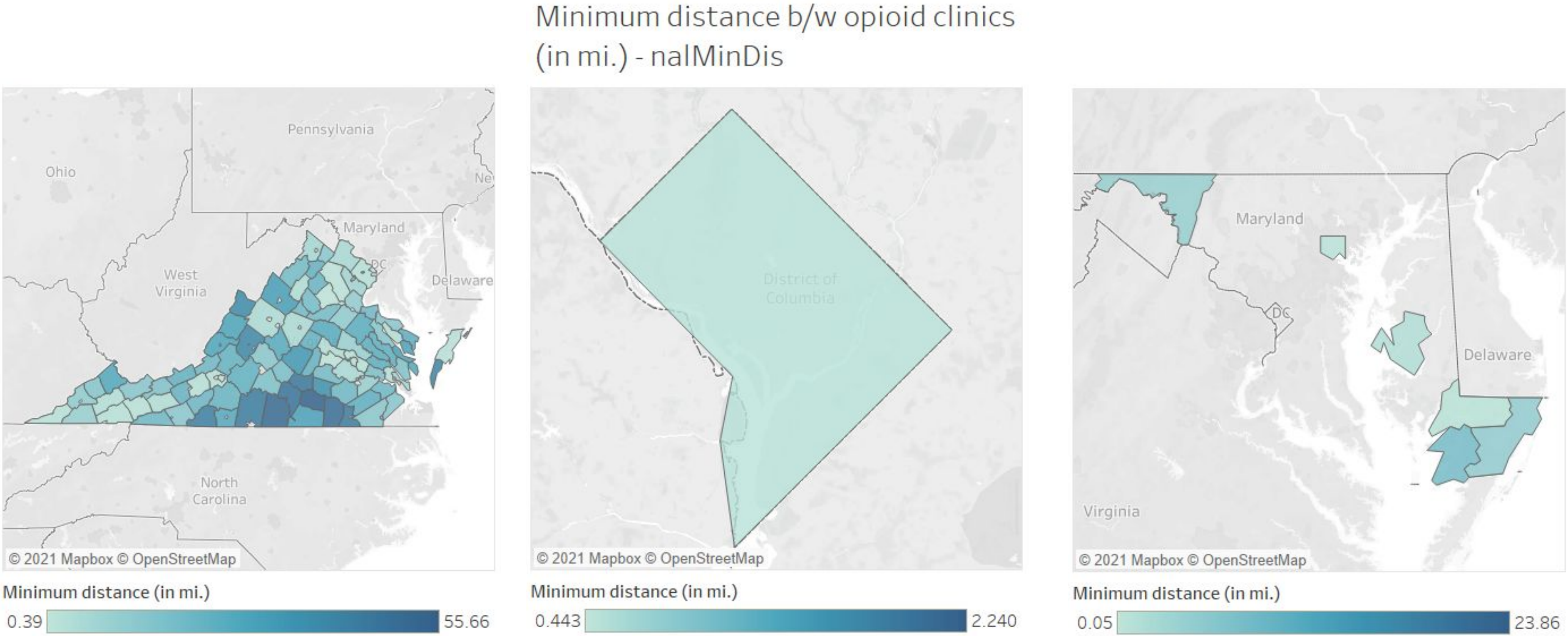
County % composition (by race) - whiteP







Cross-sectional maps of percentage population by race (e.g., white) seem to correlate spatially with minimum distance to clinics (e.g., naloxone).



Lessons learned: computational bottlenecks

Despite **nearly 400,000 rows of data,** PostgreSQL DBMS was able to **execute queries lightning fast (on the order of seconds).**

However, Tableau (and every other GUI-based interface I used) exhibited **major processing speed bottlenecks (especially at “county” granularity).**

Takeaway: it may be useful to **query a smaller subset of the data** prior to sending it to the data visualization dashboard.

Lessons learned: data harmonization

While the prototype dashboard is a start, there are a lot of different variables at **different spatial scales that must be harmonized** in order to plot all data together in an **integrated dashboard.**

With so many different variables at **different measurement scales,** application of some sort of **scaling factor** may be necessary.

Lessons learned: analytics & research questions

Policymakers and stakeholders **may have very different questions than mine;** performing traditional **user-experience experiments** may prove helpful in **identifying the most pertinent questions** for our end users.

Some sort of **composite statistical index** may provide a useful single-value representation of this **extremely high-dimensional data.**

References

Kolak, M. A., Chen, Y. T., Joyce, S., Ellis, K., Defever, K., McLuckie, C., Friedman, S., & Pho, M. T. (2020). Rural risk environments, opioid-related overdose, and infectious diseases: A multidimensional, spatial perspective. *International Journal of Drug Policy*, *85*. https://doi.org/10.1016/j.drugpo.2020.102727

Kolak, M., Lin, Q., Paykin, S., Menghaney, M., & Li, A. (2021). *GeoDaCenter/opioid-policy-scan: Opioid environment policy scan data warehouse* (v0.1-beta) [Software]. Zenodo. https://doi.org/10.5281/zenodo.4747876

Opioid overdose crisis. (2021, March 11). National Institute on Drug Abuse. Retrieved November 16, 2021, from https://www.drugabuse.gov/drug-topics/opioids/opioid-overdose-crisis

University of Wisconsin Population Health Institute & Robert Wood Johnson Foundation. (n.d.). *County health rankings*. County Health Rankings & Roadmaps. Retrieved November 16, 2021, from https://www.countyhealthrankings.org/

U.S. Department of Health and Human Services. (2021, October 27). *About the epidemic*. HHS.Gov/Opioids. Retrieved November 16, 2021, from https://www.hhs.gov/opioids/about-the-epidemic/index.html