**D211 Task 1: Data Analysis**

**Joseph Duszynski**

**ID #011237236**

**A.1. Datasets and dashboard file**

See the attached file “U.S.\_Chronic\_Disease\_Indicators\_2020.csv” for the external dataset [2]. The Tableau workbook (dashboard) is attached as “WGU\_and\_CDI.twbx”. The WGU medical dataset [1] must be accessed from within the virtual environment, as it’s different from datasets used in D210 and previous courses.

**A.2. Dashboard installation**

After loading the virtual environment, place the “U.S.\_Chronic\_Disease\_Indicators\_2020.csv” file in “C:\Users\Public\Downloads”. Then open pgAdmin 4 and load the “medical\_data” database found in “Servers > PostgreSQL 13 > Databases” and use the query tool (Alt+Shift+Q). Copy and paste the SQL code found in section A.4. (or found in the attached file “D211\_SQL.txt”), execute it, then open the Tableau workbook file “WGU\_and\_CDI.twbx” (extracted to the desktop or wherever you’d like) and login with:

* Server: localhost
* Port: 5432
* Database: medical\_data
* Username: postgres
* Password: Passw0rd!

This was tested successfully in a new virtual environment, but if it doesn’t work as expected, the alternative instructions below would also create the dashboard.

In the “Data Source” page, add a connection under “To a Server > PostgreSQL”, using the login information above. Drag the tables “cdi\_2020”, “patient\_state\_services”, and “asthma\_joined” from the left panel to the central data window. The relationships should form automatically on each table’s “state” column.

For the sheet “Asthma Difference from WGU to CDI rates”, use the “asthma\_joined” table and place “State” as a detail, “AVG(Avg Asthma Cdi)” as a label, and “AVG(Asthma Diff)” as a color (Orange-Blue Diverging, Reversed) in the “Marks” shelf. Select map mode if needed.

For the “Condition Selector” map, using the patient\_state\_services table, add “State” as a “Label” to “Marks” (it may need to be changed to “Geographic Role > State/Province” in its context menu if not automatically detected as being geographical), select “Map”, and add the calculated field “Selected Condition Percentage”:

CASE [Condition Selector]

WHEN 'Allergic Rhinitis' THEN 100 \* SUM([Allergic Rhinitis]) / COUNT([Patient Id])

WHEN 'Anxiety' THEN 100 \* SUM([Anxiety]) / COUNT([Patient Id])

WHEN 'Arthritis' THEN 100 \* SUM([Arthritis]) / COUNT([Patient Id])

WHEN 'Asthma' THEN 100 \* SUM([Asthma]) / COUNT([Patient Id])

WHEN 'Back Pain' THEN 100 \* SUM([Backpain]) / COUNT([Patient Id])

WHEN 'Diabetes' THEN 100 \* SUM([Diabetes]) / COUNT([Patient Id])

WHEN 'Hyperlipidemia' THEN 100 \* SUM([Hyperlipidemia]) / COUNT([Patient Id])

WHEN 'Overweight' THEN 100 \* SUM([Overweight]) / COUNT([Patient Id])

WHEN 'Readmission' THEN 100 \* SUM([Readmis]) / COUNT([Patient Id])

WHEN 'Reflux Esophagitis' THEN 100 \* SUM([Reflux Esophagitis]) / COUNT([Patient Id])

WHEN 'Stroke' THEN 100 \* SUM([Stroke]) / COUNT([Patient Id])

END

as a color (Red-Blue Diverging, Reversed). Create a parameter called “Condition Selector” that takes “String” as its “Data type” and has “Allowable values”:

* Allergic Rhinitis
* Anxiety
* Arthritis
* Asthma
* Back Pain
* Diabetes
* Hyperlipidemia
* Overweight
* Readmission
* Reflux Esophagitis
* Stroke

Right click on it and select “Show Parameter” in its context menu.

Create the calculated field “Patients in state matching filters”:

COUNT([Patient Id])

Then create the calculated field “Patients in state matching filters with selected condition”:

CASE [Condition Selector]

WHEN 'Allergic Rhinitis' THEN SUM([Allergic Rhinitis])

WHEN 'Anxiety' THEN SUM([Anxiety])

WHEN 'Arthritis' THEN SUM([Arthritis])

WHEN 'Asthma' THEN SUM([Asthma])

WHEN 'Back Pain' THEN SUM([Backpain])

WHEN 'Diabetes' THEN SUM([Diabetes])

WHEN 'Hyperlipidemia' THEN SUM([Hyperlipidemia])

WHEN 'Overweight' THEN SUM([Overweight])

WHEN 'Readmission' THEN SUM([Readmis])

WHEN 'Reflux Esophagitis' THEN SUM([Reflux Esophagitis])

WHEN 'Stroke' THEN SUM([Stroke])

END

Add these two fields to the “Marks” shelf as details. Next, create the field “Complication risk”:

Case [Compl Id]

WHEN 1 THEN 'Unknown'

WHEN 2 THEN 'High'

WHEN 3 THEN 'Medium'

WHEN 4 THEN 'Low'

END

And the field “Initial admin”:

Case [Admis Id]

WHEN 1 THEN 'Emergency Admission'

WHEN 2 THEN 'Elective Admission'

WHEN 3 THEN 'Observation Admission'

END

Under “Filters”, add “Initial Days”, “Age”, “Gender”, “Initial admin”, “Services”, and “Complication risk”. For all six of these, again select “Show Filter”. On the right edge of the screen,

* “Condition Selector”: use “Compact List”
* “Initial days”: “Range of Values” and “All Values in Database”
* “Age”: “Range of Values” and “All Values in Database”
* “Gender”: “Single Value (slider)” and “All Values in Database”
* “Initial admin”: “Multiple Values (list)” and “All Values in Database”
* “Services”: “Multiple Values (list)” and “All Values in Database”
* “Complication risk”: “Multiple Values (list)” and “All Values in Database”

For the sheet “CDI Questions with Readmission”, use the table cdi\_2020, then add “state” to the “Marks” shelf and select map (“state” may need to be altered to a geographic state if not automatically detected). Add “Datavaluetype” and “Datavalueunit” as attributes and “AVG(Datavalue)” as a color (Red-Blue Diverging, Reversed) to “Marks”. Create the calculated field “Readmission percentage”:

100 \* SUM([Readmis]) / COUNT([Patient Id])

and add it as a label to “Marks”. Now add “Question”, “Datavalueunit”, and “Datavaluetype” to “Filters”, allowing all options. Right click each of them for “Show Filter”.

Use “Multiple Values (list)” and “Only Relevant Values” for the “Datavalueunit” and “Datavaluetype” widgets. For “Question”, use “Single Value (dropdown)” and “All Values in Database”.

The dashboards are created by dragging each of the sheets into their respective dashboard instance. “Condition Selector” may need its widgets to be manually rearranged for the same order as what was created in the sheet.

**A.3. Dashboard navigation**

The “Asthma Difference Dashboard” is a map with panning and zoom controls depicting the differences in asthma rates by state between the WGU data and 2020 CDI data. The overlaid numerical labels are equivalent to the 2020 CDI rates for the crude rate of asthma in adults by state. The measured difference in rates as depicted by the color gradient is the WGU dataset’s rate of the respective condition minus its rate in the CDI data. As an example, California has 31.46% occurrence of asthma in the WGU data while it’s 9.300% in the 2020 CDI data, giving a difference of 22.16% (shown in the tooltip).

The ”Condition Selector Dashboard” is a map with panning and zoom controls (mouse control or upper left hand corner of map) for WGU data in the U.S. and some overseas territories. “Condition Selector” is a drop down menu for readmission, asthma, diabetes, and other conditions in the dataset that show the rates of those conditions by state according to a color gradient (see the legend underneath the selector). Additionally, the condition rates can be filtered by subsets of length of initial stay, age, gender, initial admission reason, initial service sought, and assessed complication risk. Select the desired boxes or buttons to adjust the filters as needed, initially set to allow all patients (no filtering).

The “CDI Questions with Readmission Dashboard” is a map of the 2020 U.S. CDI data again with zoom and panning capabilities. The “Question” widget on the right hand side will render a color graded map by state for the measured data value of that question. The color legend below gives an approximate visual measure, although mousing over a state will show a tooltip with all desired information. The “Datavalueunit” and “Datavaluetype” widgets will ideally be restricted to only one value each. Depending on the question, the dataset has different values recorded, such as a crude rate as well as an age-adjusted rate. If overlapping values are selected it will report an average. Asterisks for “Data Value Unit/Type” in a tooltip indicate a conflict. The overlaid numerical text on each state is the WGU readmission percentage.

**A.4. SQL code**

SQL code to be used with pgAdmin 4 query tool:

ALTER TABLE public.servicesaddon

ALTER COLUMN overweight

TYPE int USING CASE

WHEN overweight='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN arthritis

TYPE int USING CASE

WHEN arthritis='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN diabetes

TYPE int USING CASE

WHEN diabetes='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN hyperlipidemia

TYPE int USING CASE

WHEN hyperlipidemia='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN backpain

TYPE int USING CASE

WHEN backpain='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN anxiety

TYPE int USING CASE

WHEN anxiety='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN allergic\_rhinitis

TYPE int USING CASE

WHEN allergic\_rhinitis='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN reflux\_esophagitis

TYPE int USING CASE

WHEN reflux\_esophagitis='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN asthma

TYPE int USING CASE

WHEN asthma='Yes' THEN 1

ELSE 0

END;

ALTER TABLE public.patient

RENAME COLUMN hignblood TO highblood;

ALTER TABLE public.patient

ALTER COLUMN stroke

TYPE int USING CASE

WHEN stroke='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN highblood

TYPE int USING CASE

WHEN highblood='Yes' THEN 1

ELSE 0

END,

ALTER COLUMN readmis

TYPE int USING CASE

WHEN readmis='Yes' THEN 1

ELSE 0

END;

ALTER TABLE public.servicesaddon

ADD FOREIGN KEY (patient\_id) REFERENCES public.patient(patient\_id);

ALTER TABLE public.survey\_responses\_addon

ADD FOREIGN KEY (patient\_id) REFERENCES public.patient(patient\_id);

DROP TABLE IF EXISTS CDI\_2020;

CREATE TABLE CDI\_2020 (

ID numeric,

State text,

Question text,

DataValueUnit text,

DataValueType text,

DataValue text,

LowConfidenceLimit numeric,

HighConfidenceLimit numeric,

CONSTRAINT entry\_id PRIMARY KEY (ID)

);

COPY CDI\_2020 FROM 'C:/Users/Public/Downloads/U.S.\_Chronic\_Disease\_Indicators\_2020.csv' DELIMITER ',' CSV HEADER;

ALTER TABLE public.cdi\_2020

ALTER COLUMN datavalue

TYPE text USING CASE

WHEN datavalue='Yes' THEN '1'

WHEN datavalue='No' THEN '0'

ELSE datavalue

END;

ALTER TABLE public.cdi\_2020

ALTER COLUMN datavalue

TYPE numeric USING CASE

WHEN datavalue ~ '^[0-9]' THEN datavalue::numeric

ELSE NULL

END;

DROP TABLE IF EXISTS patient\_state\_services;

CREATE TABLE patient\_state\_services AS

SELECT patient.\*, location.state, servicesaddon.services, servicesaddon.overweight, servicesaddon.arthritis, servicesaddon.diabetes, servicesaddon.hyperlipidemia, servicesaddon.backpain, servicesaddon.anxiety, servicesaddon.allergic\_rhinitis, servicesaddon.reflux\_esophagitis, servicesaddon.asthma

FROM patient

INNER JOIN location

ON patient.location\_id = location.location\_id

INNER JOIN servicesaddon

ON patient.patient\_id = servicesaddon.patient\_id;

DROP TABLE IF EXISTS cdi\_asthma;

CREATE TABLE cdi\_asthma AS (

SELECT state, datavalue AS avg\_asthma\_cdi

FROM cdi\_2020

WHERE question='Current asthma prevalence among adults aged >= 18 years'

AND datavaluetype='Crude Prevalence'

);

DROP TABLE IF EXISTS wgu\_asthma;

CREATE TABLE wgu\_asthma AS (

SELECT state, ROUND(100 \* AVG(asthma), 2) AS avg\_asthma\_wgu

FROM patient\_state\_services

GROUP BY state

);

DROP TABLE IF EXISTS asthma\_joined;

CREATE TABLE asthma\_joined AS(

SELECT cdi\_asthma.state, avg\_asthma\_cdi, (avg\_asthma\_wgu - avg\_asthma\_cdi) AS asthma\_diff

FROM cdi\_asthma

INNER JOIN wgu\_asthma

ON cdi\_asthma.state = wgu\_asthma.state

);

DROP TABLE IF EXISTS cdi\_asthma;

DROP TABLE IF EXISTS wgu\_asthma;

COMMIT;

Tableau data SQL:

SELECT "cdi\_2020"."datavalue" AS "datavalue",

  CAST("cdi\_2020"."datavaluetype" AS TEXT) AS "datavaluetype",

  CAST("cdi\_2020"."datavalueunit" AS TEXT) AS "datavalueunit",

  "cdi\_2020"."highconfidencelimit" AS "highconfidencelimit",

  "cdi\_2020"."id" AS "id",

  "cdi\_2020"."lowconfidencelimit" AS "lowconfidencelimit",

  CAST("cdi\_2020"."question" AS TEXT) AS "question",

  CAST("cdi\_2020"."state" AS TEXT) AS "state"

FROM "public"."cdi\_2020" "cdi\_2020"

SELECT "patient\_state\_services"."additional\_charges" AS "additional\_charges",

  "patient\_state\_services"."admis\_id" AS "admis\_id",

  "patient\_state\_services"."age" AS "age",

  "patient\_state\_services"."allergic\_rhinitis" AS "allergic\_rhinitis",

  "patient\_state\_services"."anxiety" AS "anxiety",

  "patient\_state\_services"."arthritis" AS "arthritis",

  "patient\_state\_services"."asthma" AS "asthma",

  "patient\_state\_services"."backpain" AS "backpain",

  "patient\_state\_services"."children" AS "children",

  "patient\_state\_services"."compl\_id" AS "compl\_id",

  "patient\_state\_services"."diabetes" AS "diabetes",

  "patient\_state\_services"."doc\_visits" AS "doc\_visits",

  "patient\_state\_services"."full\_meals" AS "full\_meals",

  CAST("patient\_state\_services"."gender" AS TEXT) AS "gender",

  "patient\_state\_services"."highblood" AS "highblood",

  "patient\_state\_services"."hyperlipidemia" AS "hyperlipidemia",

  "patient\_state\_services"."income" AS "income",

  "patient\_state\_services"."initial\_days" AS "initial\_days",

  "patient\_state\_services"."job\_id" AS "job\_id",

  "patient\_state\_services"."lat" AS "lat",

  "patient\_state\_services"."lng" AS "lng",

  "patient\_state\_services"."location\_id" AS "location\_id",

  CAST("patient\_state\_services"."marital" AS TEXT) AS "marital",

  "patient\_state\_services"."overweight" AS "overweight",

  CAST("patient\_state\_services"."patient\_id" AS TEXT) AS "patient\_id",

  "patient\_state\_services"."population" AS "population",

  "patient\_state\_services"."readmis" AS "readmis",

  "patient\_state\_services"."reflux\_esophagitis" AS "reflux\_esophagitis",

  CAST("patient\_state\_services"."services" AS TEXT) AS "services",

  CAST("patient\_state\_services"."soft\_drink" AS TEXT) AS "soft\_drink",

  CAST("patient\_state\_services"."state" AS TEXT) AS "state (patient\_state\_services)",

  "patient\_state\_services"."stroke" AS "stroke",

  "patient\_state\_services"."totalcharge" AS "totalcharge",

  "patient\_state\_services"."vitd\_levels" AS "vitd\_levels",

  "patient\_state\_services"."vitd\_supp" AS "vitd\_supp"

FROM "public"."patient\_state\_services" "patient\_state\_services"

SELECT "asthma\_joined"."asthma\_diff" AS "asthma\_diff",

  "asthma\_joined"."avg\_asthma\_cdi" AS "avg\_asthma\_cdi",

  CAST("asthma\_joined"."state" AS TEXT) AS "state (asthma\_joined)"

FROM "public"."asthma\_joined" "asthma\_joined"

Tableau calculated fields:

Readmission percentage:

100 \* SUM([Readmis]) / COUNT([Patient Id])

Patients in state matching filters:

COUNT([Patient Id])

Selected Condition Percentage:

CASE [Condition Selector]

WHEN 'Allergic Rhinitis' THEN 100 \* SUM([Allergic Rhinitis]) / COUNT([Patient Id])

WHEN 'Anxiety' THEN 100 \* SUM([Anxiety]) / COUNT([Patient Id])

WHEN 'Arthritis' THEN 100 \* SUM([Arthritis]) / COUNT([Patient Id])

WHEN 'Asthma' THEN 100 \* SUM([Asthma]) / COUNT([Patient Id])

WHEN 'Back Pain' THEN 100 \* SUM([Backpain]) / COUNT([Patient Id])

WHEN 'Diabetes' THEN 100 \* SUM([Diabetes]) / COUNT([Patient Id])

WHEN 'Hyperlipidemia' THEN 100 \* SUM([Hyperlipidemia]) / COUNT([Patient Id])

WHEN 'Overweight' THEN 100 \* SUM([Overweight]) / COUNT([Patient Id])

WHEN 'Readmission' THEN 100 \* SUM([Readmis]) / COUNT([Patient Id])

WHEN 'Reflux Esophagitis' THEN 100 \* SUM([Reflux Esophagitis]) / COUNT([Patient Id])

WHEN 'Stroke' THEN 100 \* SUM([Stroke]) / COUNT([Patient Id])

END

Patients in state matching filters with selected condition:

CASE [Condition Selector]

WHEN 'Allergic Rhinitis' THEN SUM([Allergic Rhinitis])

WHEN 'Anxiety' THEN SUM([Anxiety])

WHEN 'Arthritis' THEN SUM([Arthritis])

WHEN 'Asthma' THEN SUM([Asthma])

WHEN 'Back Pain' THEN SUM([Backpain])

WHEN 'Diabetes' THEN SUM([Diabetes])

WHEN 'Hyperlipidemia' THEN SUM([Hyperlipidemia])

WHEN 'Overweight' THEN SUM([Overweight])

WHEN 'Readmission' THEN SUM([Readmis])

WHEN 'Reflux Esophagitis' THEN SUM([Reflux Esophagitis])

WHEN 'Stroke' THEN SUM([Stroke])

END

Complication risk:

Case [Compl Id]

WHEN 1 THEN 'Unknown'

WHEN 2 THEN 'High'

WHEN 3 THEN 'Medium'

WHEN 4 THEN 'Low'

END

Initial admin:

Case [Admis Id]

WHEN 1 THEN 'Emergency Admission'

WHEN 2 THEN 'Elective Admission'

WHEN 3 THEN 'Observation Admission'

END

**B. Panopto presentation**

See the attached link: https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=0cffdb8e-e44c-43cd-921e-b175016e10df

**C.1. Dashboard alignment**

As discussed in the task scenario, hospital administrators and analysts would like to understand dynamics driving readmission rates within their healthcare system (in the WGU dataset). The variables collected in the dataset are somewhat limited and are confined to 10000 patients, which could be improved by comparing to a U.S. national survey of many different health metrics and indicators. The WGU data may have abnormally high (or low) instances of certain health conditions, which would require further investigation. By comparing the WGU readmission rates by state to state-level data contained in the 2020 CDI, patterns that wouldn’t otherwise be noticed may emerge. High premature mortality and lack of access to healthcare coverage are two outside metrics that might correlate to WGU readmission rates.

The differences in asthma rates display the discrepancies by state, prompting further investigation into states with significant differences.

The “Condition Selector” interactive map is a convenient way to view state-level percentages of the conditions recorded in the WGU data (e.g. asthma, hyperlipidemia) as well as the readmission rate. Filter options can confine questions and analysis to specific subgroups and show regions, demographics, and health conditions that are worthy of additional investigation or targeted changes.

The ”2020 U.S. CDI Map by Survey Question” map integrates the WGU readmission rates as a label over each state with questions of interest from the CDI dataset. For example, premature mortality may have a visually detectable relationship with readmission rates or identify anomalies worthy of further analysis.

The included dashboards allow administrators to review multiple different variables in one panel for insights into readmission rates and the overall health status of each state’s population.

**C.2. Business intelligence tool**

This project used Tableau Desktop, a data visualization program that can work with data from SQL databases, .csv files, and other common formats. It allows for simple calculations and transformations of data variables to be displayed in many different visual formats such as scatter plots, heatmaps, boxplots, histograms, and geographical maps with a color gradient. Tooltips and customization options make it user friendly and straightforward to create visualizations. Data analysts can use Tableau to present easily understood conclusions from their work for an audience of any technical background.

**C.3. Data cleaning**

The “medical\_data” database data and the provided subset of 2020 U.S. CDI data [2] were largely previously processed. Only a few adjustments were necessary:

1. In the “servicesaddon” table, one-hot encoding (‘Yes’ converted to 1 and ‘No’ converted to 0) was done on the columns “overweight”, “arthritis”, “diabetes”, “hyperlipidemia”, “backpain”, “anxiety”, “allergic\_rhinitis”, “reflux\_esophagitis”, and “asthma”.
2. In the “patient” table, the misspelled column “hignblood” was corrected to “highblood”. The columns “stroke”, “highblood”, and “readmis” were all one-hot encoded as well.
3. Foreign key constraints were introduced for servicesaddon.patient\_id to refer to patient.patient\_id and survey\_responses\_addon.patient\_id referring to patient.patient\_id.
4. The 2020 U.S. CDI data was loaded into the created table cdi\_2020. Its “datavalue” column had some instances of string values (such as ‘Yes’ and ‘No’) mixed with what was mostly numerical data, so it initially had to be loaded as text data. ‘Yes’ values were converted to ‘1’ and ‘No’ values were converted to ‘0’, then every entry was cast as numeric or changed to null if that wasn’t possible.
5. A larger “patient\_state\_services” table was formed by performing inner joins between “patient”, “location”, and “servicesaddon” on their respective shared IDs.
6. Temporary tables “cdi\_asthma” and “wgu\_asthma” were created to record the asthma rates by state. They were joined on their state column to create a persistent table “asthma\_joined” containing the state, asthma rate from the CDI data, and the difference between the WGU rate and CDI rate.

**C.4. Dashboard creation**

The secondary, more detailed installation instructions in section A.2. discuss how the dashboard was created.

**C.5. Data analysis results**

One of the most relevant conclusions from these visualizations is shown in the “Asthma Difference Dashboard”, highlighting the extraordinary differences by state. This depiction allows administrators to quickly identify outlier states with the greatest differences. In particular, nearly every state has asthma rates at least 15% higher than averages from the CDI data. The maps make it very clear that asthma rates are abnormally high and need further investigation.

The ”Condition Selector” map provides essential information by state with multiple filtering options. Administrators can focus on readmission rates or a health condition of interest, identify states that perform well or poorly, and further refine the scope with factors such as the patient’s initial stay, age, complication risk, and others. Restricting the initial stay to ~48 days or less shows there are no readmissions across the entire healthcare system. Removing that restriction clearly shows readmission rates tend to be highest in the central parts of the country, as well as a few states in the southeast and northeast. Adjusting the filters quickly demonstrates readmission rates for patients initially seeking blood work are primarily highest in the South and New England, prompting further inquiry into local community health and hospital practices in those regions.

**C.6. Analysis limitations**

The WGU data has a limited set of health questions responded to in a binary (“yes”/’no”) manner and only 10000 patients. Although patient readmissions can be predicted reasonably well, the underlying causes and lengths of subsequent stays is not present in the current dataset. The binary presence of health conditions such as high blood pressure (as opposed to something numerical) can obscure or oversimplify patterns and relationships. The visualizations in this project cannot distinguish one state with 70% of the population being overweight but not obese with a state where 40% of the population is overweight (but not obese) and 30% is obese. Additional granularity could improve the analysis and its conclusions.

The most recent U.S. CDI data is unfortunately from 2020, while the WGU patient data provides no known timeframe. Should the WGU data be drawn from widely different time periods the conclusions of any analysis would be unreliable as macroscopic health patterns can change even within a single year.

While the increased asthma rates are alarming, it can’t be ruled out that patients with preexisting conditions may be more likely than the general population to require hospital admission. The information for patient admission doesn’t make it clear if asthma played a role, so additional data would be ideal. Furthermore, comparing data with neighboring hospitals would give greater insight into the asthma anomalies.

Some subpopulation sizes (e.g. seven patients in Nevada were admitted for CT scans) are too small to draw confident conclusions (in readmission or other metrics), so comparisons with neighboring hospitals in each county or state would be beneficial.

The CDI data unfortunately didn’t have comprehensive cancer statistics from a recent year, which would have been helpful in investigating any potential relationship between high cancer rates and high readmission rates. Some health questions within the CDI data had null values for some states, making comparisons between states fragmented if not inconsequential for those questions.

Readmission is almost certainly caused by multiple, interdependent factors. Untangling causes without having additional data on why patients are readmitted can be very difficult. The CDI data shows binge drinking tends to be more common in certain states of the Midwest and northern plains while obesity and generally high disease mortality is concentrated in the South. The visualizations can promote areas of further investigation, but they can’t be used to establish causality.

Although state-level data is very helpful for understanding broad regional trends, problematic issues in a particular county or even hospital can go unnoticed or distort conclusions about the entire state. Any data analysis would be improved with data from individual counties or even hospitals.

**D. Sources for third-party code**

**1.** WGU. 2024. D211 Advanced Data Acquisition “Labs on Demand”. “medical\_data” database accessed from pgAdmin 4 within the “Labs on Demand” virtual environment. Retrieved May 19, 2024, from https://tasks.wgu.edu/student/011237236/course/34320018/task/4303/overview.

**2.** CDC. 2024. U.S. Chronic Disease Indicators (CDI), 2023 Release. Retrieved May 19, 2024, from <https://data.cdc.gov/Chronic-Disease-Indicators/U-S-Chronic-Disease-Indicators-CDI-2023-Release/g4ie-h725/data_preview>.

**E. Sources**

No other sources were used.