Western Governors University

D211 – Advanced Data Acquisition

By Krista Moik

Table of Contents

Part 1: Data Dashboards

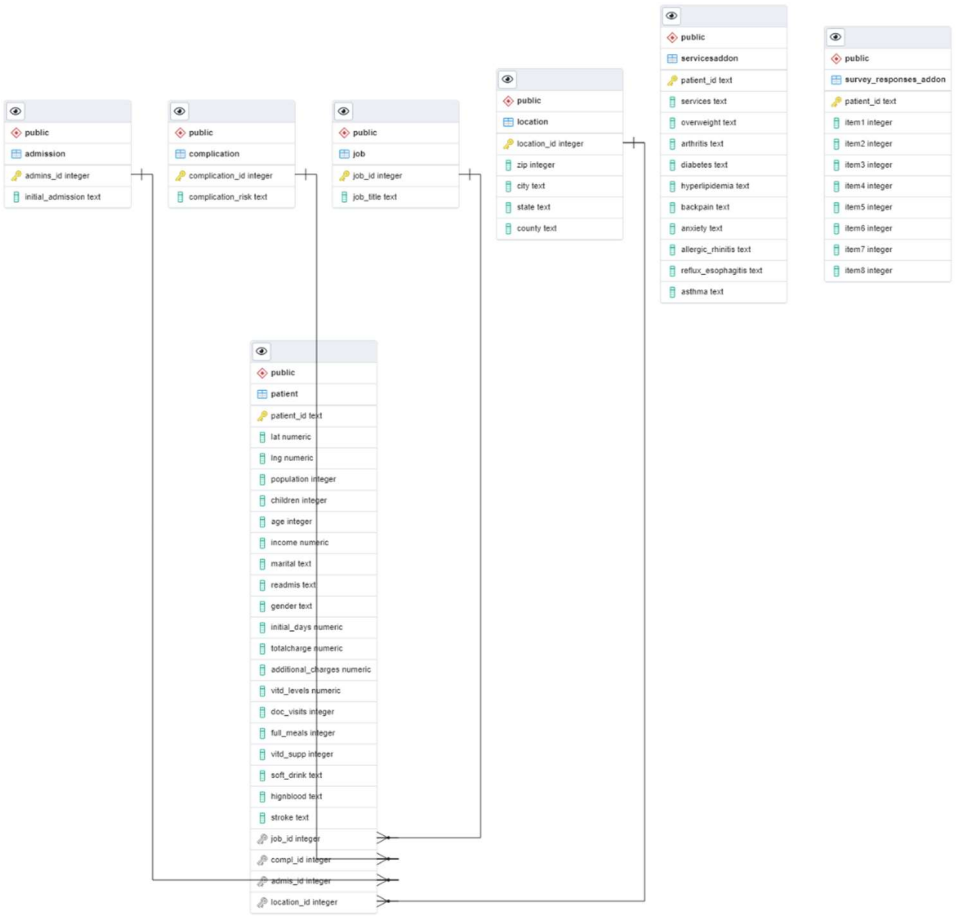
Please see the attached Tableau dashboards: KMoikTabLive.twbx and KMoikTabExtract.twbx. The Live version will require an active connection to the databases used to visualize the data. The Extract file has extracted the data accessed during the live connection so it can be viewed by anyone, even if they are not connected to the databases used.

Additionally, a Tableau Story with the dashboards/visualizations I created can be viewed on Tableau Public at: D211 - Advanced Data Acquisition | Tableau Public

A1. Both Data Sets

Please see the attached WGU provided medical data tables used: patient and location, and the third-party data set used titled Original DMC data. The 3rd party data set is the same I used in D210 and was obtained via Kaggle: Hospital Admissions Data: Two Year Hospital Admissions and Discharge Data from Hero DMC Heart Institute (Sahani, 2022). While the datasets are not necessarily an apples-to-apples comparison, they have enough data and variables to satisfactorily complete this assignment. The WGU dataset will be referred to as WGU and the 3rd party dataset will be referred to as DMC.

Below is the ERD for the WGU medical_data:



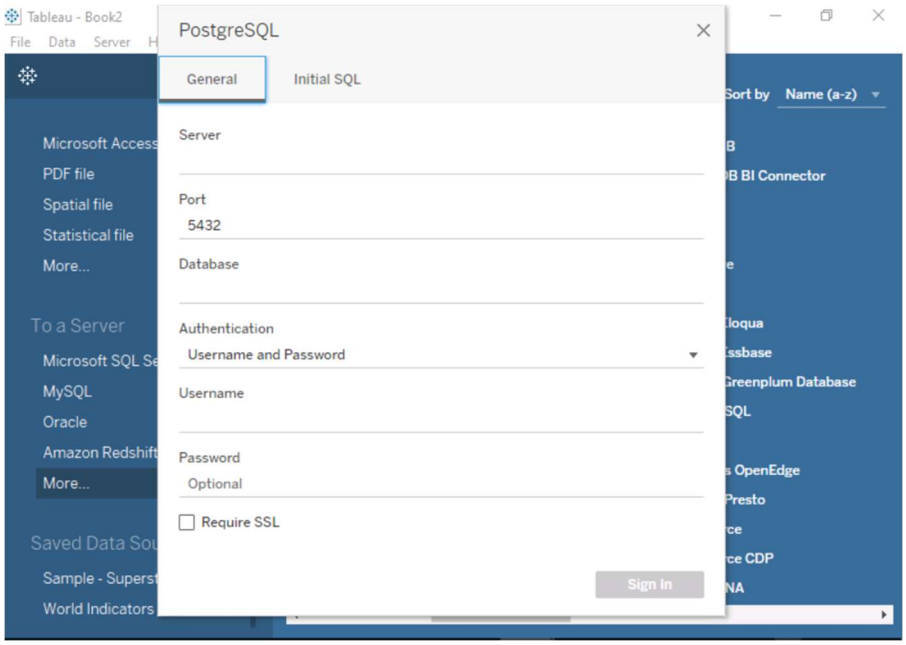
A2. Dashboard Installation

The easiest method to view the dashboards, is to not install anything. Viewers can click the below Tableau Public link to view the dashboard in a free and accessible manner:

D211 - Advanced Data Acquisition | Tableau Public

Otherwise, users should open a new Labs on Demand session in the Virtual Machine. Once loaded, double click on the desktop shortcut called pgAdmin4.

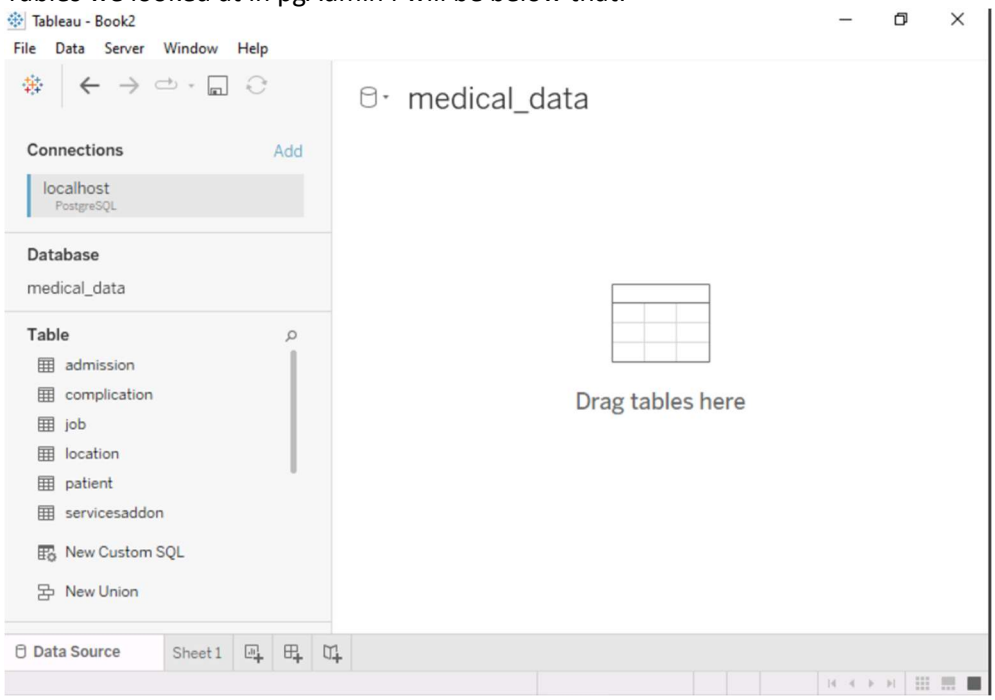
1. Once pgAdmin4 is loaded, there should be a column on the left side of the program. In that left column, click on the right arrow next to servers. This should open the menu below and PostgresSQL13 should be showing under servers, and below that there should be a tab that says Databases
2. Under the Databases tab, click on the right arrow next to medical_data. This is the WGU provided medical data set.
    a. Scroll down to Schemas and click on the right arrow next to it. Another drop down menu should open called public
    b. Under public, click on the right arrow next to Tables. Seven tables should appear titled: admission, complication, job, location, patient, servicesaddon, and survey_responses
        i. You can click the right arrows next to each table name to look at the columns, constraints, and other options
3. Looking at the table called patient and its columns, it appears that table has a lot of the information I have been using in my analysis and has just been broken down into multiple tables rather than being one large file with all of the data.
    a. On the top left, to the right of Browser, there is a database symbol with a play symbol on top. Click on that to open a new SQL query. Now I can write some SQL queries to view the data in the tables provided.
4. Once I confirm which tables I want to use in my analysis, go back to the desktop (but do not exit out of pgAdmin4. Double click on the desktop shortcut called Tableau 2021.4 that is located on the desktop of the Virtual Machine. This will open a new workbook in Tableau.
5. After Tableau opens, on the left side there is a blue Connect column. Scroll down to the option “To a Server”. If PostgreSQL is not already populated, click on the “more” button under “To a Server” and scroll to the right until you see PostgreSQL. Click on PostgreSQL.
6. A box will pop up to log in to PostgreSQL. Under the tab called General:



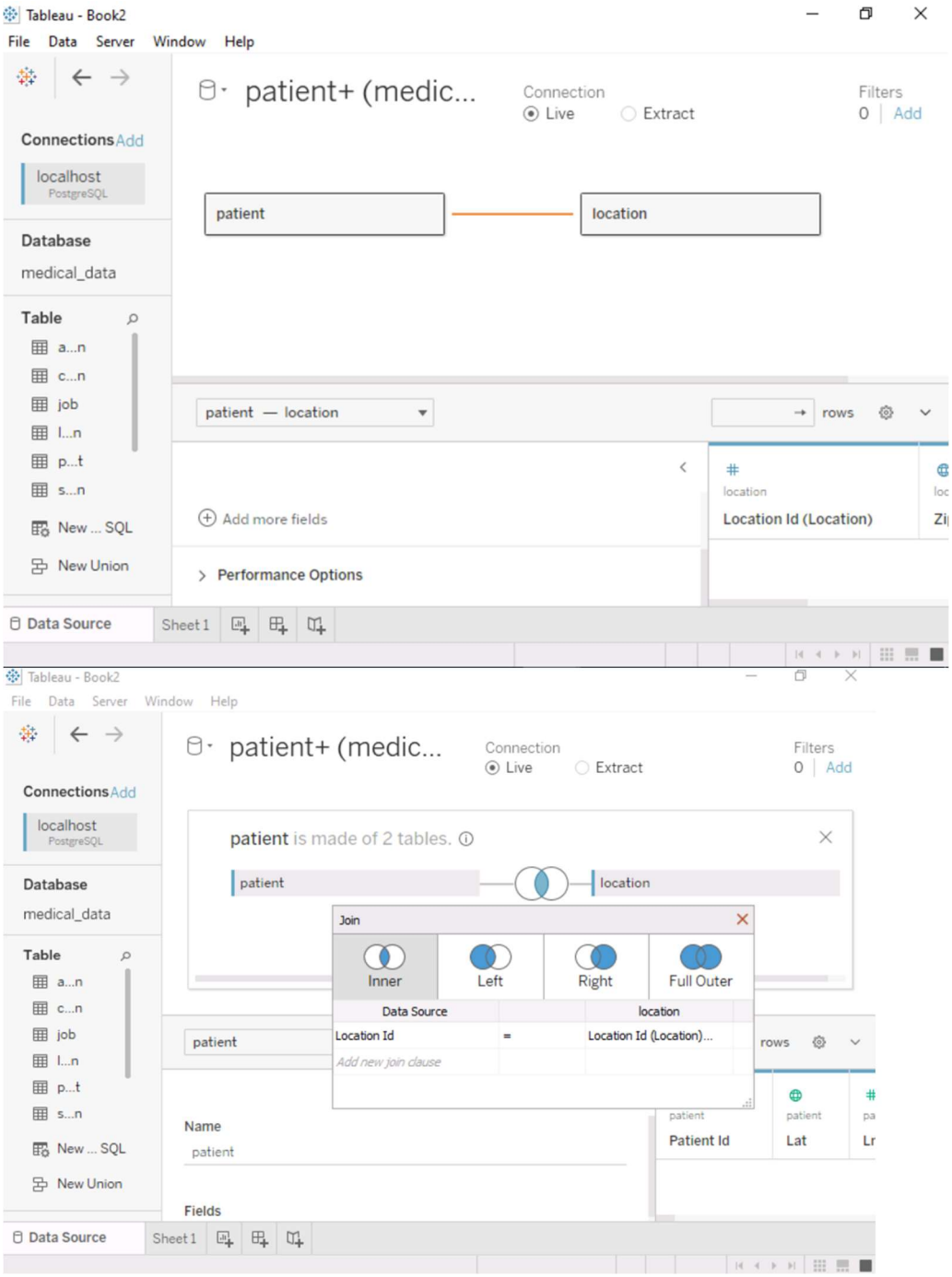
For Server type: localhost
Port: 5432
Database: medical_data
Authentication: Username and Password
Username: postgres
Password: Passw0rd!

a. Once you have filled out the above parameters, click sign in

7. Once the sign in process has completed, localhost with PostgreSQL underneath should show up under Connections. The Database will list our medical_data database from pgAdmin4, and the Tables we looked at in pgAdmin4 will be below that.



8. Now that we have connected Tableau to PostgreSQL, drag the Table called Patient to the right where it says Drag Tables Here. Then drag the Table Location to the right. A line should appear between the 2 tables. Double click on one of the tables and a Join box will appear. Choose the appropriate join, in this case, I chose Inner Join. Then choose Location ID = Location ID for both tables to join them on that column.

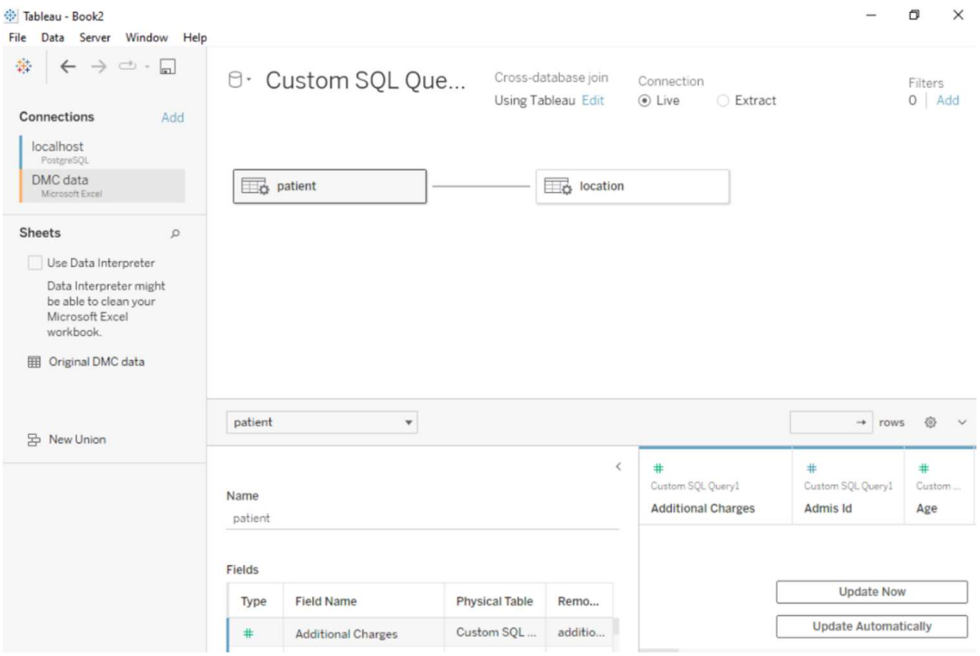


9. Now when you right click on patient and click Edit Custom SQL, you can see the INNER JOIN in the SQL code:

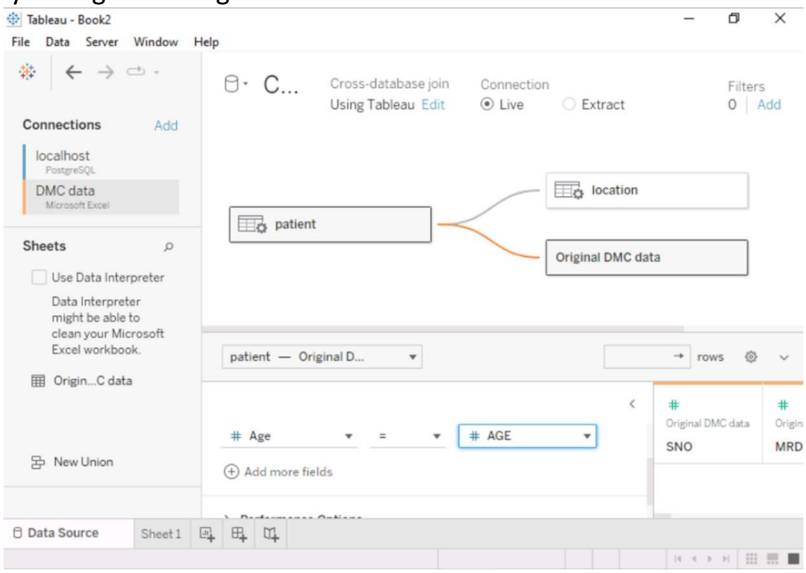
```
SELECT "patient"."additional_charges" AS "additional_charges",
  "patient"."admis_id" AS "admis_id",
```

```
 "patient"."age" AS "age",
 "patient"."children" AS "children",
CAST("location"."city" AS TEXT) AS "city (location)",
 "patient"."compl_id" AS "compl_id",
 CAST("location"."county" AS TEXT) AS "county (location)",
 "patient"."doc_visits" AS "doc_visits",
 "patient"."full_meals" AS "full_meals",
 CAST("patient"."gender" AS TEXT) AS "gender",
 CAST("patient"."hignblood" AS TEXT) AS "hignblood",
 "patient"."income" AS "income",
 "patient"."initial_days" AS "initial_days",
 "patient"."job_id" AS "job_id",
 "patient"."lat" AS "lat",
 "patient"."lng" AS "lng",
 "location"."location_id" AS "location_id (location) #1",
 "patient"."location_id" AS "location_id",
 CAST("patient"."marital" AS TEXT) AS "marital",
 CAST("patient"."patient_id" AS TEXT) AS "patient_id",
 "patient"."population" AS "population",
 CAST("patient"."readmis" AS TEXT) AS "readmis",
 CAST("patient"."soft_drink" AS TEXT) AS "soft_drink",
 CAST("location"."state" AS TEXT) AS "state (location)",
 CAST("patient"."stroke" AS TEXT) AS "stroke",
 "patient"."totalcharge" AS "totalcharge",
 "patient"."vitd_levels" AS "vitd_levels",
 "patient"."vitd_supp" AS "vitd_supp",
 "location"."zip" AS "zip (location)"
FROM "public"."patient" "patient"
 INNER JOIN "public"."location" "location" ON ("patient"."location_id" =
"location"."location_id")
```

10. We still need to connect to our 3rd party data, which is an Excel sheet. Please load the provided XLSX titled original DMC data to the Users -> Public folder on the Virtual Machine. If you rename it, please take note of what you called it so you can find it.
11. Go back to our open Tableau, next to Connections, click the blue add, click Microsoft Excel since it is an Excel spreadsheet
    a. File explorer will open, navigate to where you saved the spreadsheet and click on it and then click open. You should now see the 3rd party data source listed under connections and under sheets in the left column of Tableau.



12. Drag DMC data to where the Patient and Location Tables are. A line should appear between patient and DMC data. Set patient age = DMC data AGE to create the connection. We are now ready to begin building our visualizations and dashboards.



A3. Dashboard Navigation

If loading my dashboards from scratch (vs using the above provided Tableau Public link), once you have confirmed the above connections to PostgreSQL and the 3rd party data set, you can now go to File -> Open in Tableau. The file explorer will open, and you will need to navigate to where you saved my provided twbx file for the Live version. Click open once you have located that file.

Tableau should open my 4 worksheets, 2 dashboards, and 1 story. For ease of access, I have edited the bottom navigation tab in Tableau so the story is the first tab next to Data Source, then my 2 dashboards,

and finally the 4 worksheets I used to create the aforementioned. By clicking on the D211 Story tab, viewers can see the story I created with the dashboards. The first page of the story is Patient Demographics. The main visualization is a map of the world with the US highlighted. As WGU data has patient information for Americans, that is the part of the map that shows the patient counts based on the provided data. A smaller visual on the right side shows a count of patients from the DMC data set per the area type they reside in, either Rural or Urban. As the DMC data set did not provide addresses for the patient information provided, there is no way to know where these patients resided to create a map. The legends provided above the DMC visualization explains the use of color.

Using the top navigation buttons on the story, viewers can then click on the right arrow to view the 2nd sheet titled Comparison of Patient Age and Number of Initial Admission Days. The top visualization on this sheet is titled WGU Age and Days. There are accompanying slide filters on the right of the visualization which allows viewers to change or view ages and/or initial days to see how the data may change and view relationships that might appear. The bottom visualization is titled DMC Age and Days. This visualization also has sliders on the right to change Ages and/or the Duration of Stay to see how the data changes and see any relationships that might occur.

Now that we have viewed the story, we can use the navigation bar at the bottom of Tableau to view the next tab: Area Dashboard. This dashboard makes up the first page of the story we just looked at. The dashboard allows for viewers to reposition the visualizations and legends, if desired. The next tab is the dashboard titled Age and Initial Days, which was used to create the 2nd sheet in our story. Again, viewers are able to manipulate the placement of the visualizations, filters, and legends.

The next objects in our navigation bar are our sheets. The first sheet is the first visualization: WGU Patient Count Map. This is where I created the map visualization. If desired, viewers can edit this visualization and it will update on the dashboard and story. The next sheet is WGU Age and Days, then DMC Age and Days, and finally DMC Area and Days. Viewers can edit all of these sheets to change the visualizations I have created and it will update on the dashboards and story.

A4. SQL Code

The SQL code for the WGU provided tables are below:

Patient table from WGU medical_data:

```
-- Table: public.patient

-- DROP TABLE public.patient;

CREATE TABLE public.patient
(
    patient_id text COLLATE pg_catalog."default" NOT NULL,
    lat numeric,
    lng numeric,
    population integer,
    children integer,
    age integer,
```

```
    income numeric,
    marital text COLLATE pg_catalog."default",
    readmis text COLLATE pg_catalog."default",
    gender text COLLATE pg_catalog."default",
    initial_days numeric,
    totalcharge numeric,
    additional_charges numeric,
    vitd_levels numeric,
    doc_visits integer,
    full_meals integer,
    vitd_supp integer,
    soft_drink text COLLATE pg_catalog."default",
    hignblood text COLLATE pg_catalog."default",
    stroke text COLLATE pg_catalog."default",
    job_id integer,
    compl_id integer,
    admis_id integer,
    location_id integer,
    CONSTRAINT patient_pkey PRIMARY KEY (patient_id),
    CONSTRAINT patient_admis_id_fkey FOREIGN KEY (admis_id)
        REFERENCES public.admission (admins_id) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION,
    CONSTRAINT patient_compl_id_fkey FOREIGN KEY (compl_id)
        REFERENCES public.complication (complication_id) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION,
    CONSTRAINT patient_job_id_fkey FOREIGN KEY (job_id)
        REFERENCES public.job (job_id) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION,
    CONSTRAINT patient_location_id_fkey FOREIGN KEY (location_id)
        REFERENCES public.location (location_id) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
)

TABLESPACE pg_default;

ALTER TABLE public.patient
    OWNER to postgres;
```

Location table from WGU medical_data:

Location table:

```
-- Table: public.location
```

```
-- DROP TABLE public.location;

CREATE TABLE public.location
(
    location_id integer NOT NULL,
    zip integer,
    city text COLLATE pg_catalog."default",
    state text COLLATE pg_catalog."default",
    county text COLLATE pg_catalog."default",
    CONSTRAINT location_pkey PRIMARY KEY (location_id)
)

TABLESPACE pg_default;

ALTER TABLE public.location
    OWNER to postgres;
```

The SQL code for adding PostgreSQL to Tableau:

```
SELECT "patient"."additional_charges" AS "additional_charges",
  "patient"."admis_id" AS "admis_id",
  "patient"."age" AS "age",
  "patient"."children" AS "children",
  "patient"."compl_id" AS "compl_id",
  "patient"."doc_visits" AS "doc_visits",
  "patient"."full_meals" AS "full_meals",
  CAST("patient"."gender" AS TEXT) AS "gender",
  CAST("patient"."hignblood" AS TEXT) AS "hignblood",
  "patient"."income" AS "income",
  "patient"."initial_days" AS "initial_days",
  "patient"."job_id" AS "job_id",
  "patient"."lat" AS "lat",
  "patient"."lng" AS "lng",
  "patient"."location_id" AS "location_id",
  CAST("patient"."marital" AS TEXT) AS "marital",
  CAST("patient"."patient_id" AS TEXT) AS "patient_id",
  "patient"."population" AS "population",
  CAST("patient"."readmis" AS TEXT) AS "readmis",
  CAST("patient"."soft_drink" AS TEXT) AS "soft_drink",
  CAST("patient"."stroke" AS TEXT) AS "stroke",
  "patient"."totalcharge" AS "totalcharge",
  "patient"."vitd_levels" AS "vitd_levels",
  "patient"."vitd_supp" AS "vitd_supp"
FROM "public"."patient" "patient"
```

Joining the WGU tables in Tableau:

```
SELECT "patient"."additional_charges" AS "additional_charges",
  "patient"."admis_id" AS "admis_id",
  "patient"."age" AS "age",
  "patient"."children" AS "children",
  CAST("location"."city" AS TEXT) AS "city (location)",
  "patient"."compl_id" AS "compl_id",
  CAST("location"."county" AS TEXT) AS "county (location)",
  "patient"."doc_visits" AS "doc_visits",
  "patient"."full_meals" AS "full_meals",
  CAST("patient"."gender" AS TEXT) AS "gender",
  CAST("patient"."hignblood" AS TEXT) AS "hignblood",
  "patient"."income" AS "income",
  "patient"."initial_days" AS "initial_days",
  "patient"."job_id" AS "job_id",
  "patient"."lat" AS "lat",
  "patient"."lng" AS "lng",
  "location"."location_id" AS "location_id (location) #1",
  "patient"."location_id" AS "location_id",
  CAST("patient"."marital" AS TEXT) AS "marital",
  CAST("patient"."patient_id" AS TEXT) AS "patient_id",
  "patient"."population" AS "population",
  CAST("patient"."readmis" AS TEXT) AS "readmis",
  CAST("patient"."soft_drink" AS TEXT) AS "soft_drink",
  CAST("location"."state" AS TEXT) AS "state (location)",
  CAST("patient"."stroke" AS TEXT) AS "stroke",
  "patient"."totalcharge" AS "totalcharge",
  "patient"."vitd_levels" AS "vitd_levels",
  "patient"."vitd_supp" AS "vitd_supp",
  "location"."zip" AS "zip (location)"
FROM "public"."patient" "patient"
  INNER JOIN "public"."location" "location" ON ("patient"."location_id" = "location"."location_id")
```

To create the 1st Tableau Table: WGU Patient Count Map:

```
SELECT CAST("Custom SQL Query"."state" AS TEXT) AS "state"
FROM (
  SELECT CAST("location"."city" AS TEXT) AS "city",
    CAST("location"."county" AS TEXT) AS "county",
    "location"."location_id" AS "location_id (location)",
    CAST("location"."state" AS TEXT) AS "state",
    "location"."zip" AS "zip"
  FROM "public"."location" "location"
) "Custom SQL Query"
GROUP BY 1
```

```
SELECT SUM(1) AS "cnt:patient_1F27295BA06C47689B64EA954319A647:ok",
```

```
  CAST("Custom SQL Query"."state" AS TEXT) AS "state"
FROM (
  SELECT "patient"."additional_charges" AS "additional_charges",
    "patient"."admis_id" AS "admis_id",
    "patient"."age" AS "age",
    "patient"."children" AS "children",
    CAST("location"."city" AS TEXT) AS "city (location)",
    "patient"."compl_id" AS "compl_id",
    CAST("location"."county" AS TEXT) AS "county (location)",
    "patient"."doc_visits" AS "doc_visits",
    "patient"."full_meals" AS "full_meals",
    CAST("patient"."gender" AS TEXT) AS "gender",
    CAST("patient"."hignblood" AS TEXT) AS "hignblood",
    "patient"."income" AS "income",
    "patient"."initial_days" AS "initial_days",
    "patient"."job_id" AS "job_id",
    "patient"."lat" AS "lat",
    "patient"."lng" AS "lng",
    "location"."location_id" AS "location_id (location) #1",
    "patient"."location_id" AS "location_id",
    CAST("patient"."marital" AS TEXT) AS "marital",
    CAST("patient"."patient_id" AS TEXT) AS "patient_id",
    "patient"."population" AS "population",
    CAST("patient"."readmis" AS TEXT) AS "readmis",
    CAST("patient"."soft_drink" AS TEXT) AS "soft_drink",
    CAST("location"."state" AS TEXT) AS "state (location)",
    CAST("patient"."stroke" AS TEXT) AS "stroke",
    "patient"."totalcharge" AS "totalcharge",
    "patient"."vitd_levels" AS "vitd_levels",
    "patient"."vitd_supp" AS "vitd_supp",
    "location"."zip" AS "zip (location)"
  FROM "public"."patient" "patient"
    INNER JOIN "public"."location" "location" ON ("patient"."location_id" = "location"."location_id")
) "Custom SQL Query1"
  LEFT JOIN (
  SELECT CAST("location"."city" AS TEXT) AS "city",
    CAST("location"."county" AS TEXT) AS "county",
    "location"."location_id" AS "location_id (location)",
    CAST("location"."state" AS TEXT) AS "state",
    "location"."zip" AS "zip"
  FROM "public"."location" "location"
) "Custom SQL Query" ON ("Custom SQL Query1"."location_id" = "Custom SQL Query"."location_id
(location)")
GROUP BY 2
```

To create the 2nd Tableau table: WGU Age and Gender:

```
SELECT "Custom SQL Query1"."age" AS "age",
  "Custom SQL Query1"."initial_days" AS "initial_days"
FROM (
  SELECT "patient"."additional_charges" AS "additional_charges",
    "patient"."admis_id" AS "admis_id",
    "patient"."age" AS "age",
    "patient"."children" AS "children",
    CAST("location"."city" AS TEXT) AS "city (location)",
    "patient"."compl_id" AS "compl_id",
    CAST("location"."county" AS TEXT) AS "county (location)",
    "patient"."doc_visits" AS "doc_visits",
    "patient"."full_meals" AS "full_meals",
    CAST("patient"."gender" AS TEXT) AS "gender",
    CAST("patient"."hignblood" AS TEXT) AS "hignblood",
    "patient"."income" AS "income",
    "patient"."initial_days" AS "initial_days",
    "patient"."job_id" AS "job_id",
    "patient"."lat" AS "lat",
    "patient"."lng" AS "lng",
    "location"."location_id" AS "location_id (location) #1",
    "patient"."location_id" AS "location_id",
    CAST("patient"."marital" AS TEXT) AS "marital",
    CAST("patient"."patient_id" AS TEXT) AS "patient_id",
    "patient"."population" AS "population",
    CAST("patient"."readmis" AS TEXT) AS "readmis",
    CAST("patient"."soft_drink" AS TEXT) AS "soft_drink",
    CAST("location"."state" AS TEXT) AS "state (location)",
    CAST("patient"."stroke" AS TEXT) AS "stroke",
    "patient"."totalcharge" AS "totalcharge",
    "patient"."vitd_levels" AS "vitd_levels",
    "patient"."vitd_supp" AS "vitd_supp",
    "location"."zip" AS "zip (location)"
  FROM "public"."patient" "patient"
    INNER JOIN "public"."location" "location" ON ("patient"."location_id" = "location"."location_id")
) "Custom SQL Query1"
GROUP BY 1,
  2
```

To create the 3rd Tableau table DMC Age and Days:

```
SELECT "Original DMC data A1:BD10000"."AGE" AS "AGE",
  "Original DMC data A1:BD10000"."DURATION OF STAY" AS "DURATION OF STAY"
FROM "TableauTemp"."Original DMC data$_A0000000001:BD0000010000" "Original DMC data
A1:BD10000"
GROUP BY 1,
  2
```

To create the 4th Tableau table DMC Area and Days:

```
DMC Area and Days
SELECT "Original DMC data A1:BD10000"."DURATION OF STAY" AS "DURATION OF STAY",
  "Original DMC data A1:BD10000"."RURAL" AS "RURAL"
FROM "TableauTemp"."Original DMC data$_A0000000001:BD0000010000" "Original DMC data
A1:BD10000"
GROUP BY 1,
  2
```

The SQL code obtained from the Tableau visualizations were found using direction from Thomas Spicer (2017).

Part 2: Demonstration

B: Panopto Video

The link for my Panopto video is:
https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=eae6f069-e5c1-47dd-90dd-b12e00f9a704

Part 3: Report

C1. Dashboard Alignment

Part of the needs outlined in the WGU-provided Data Dictionary is to identify trends and key metrics in order to provide actionable business insights. The dashboards included in my Tableau story analyze via comparison and visualization key metrics in order for me to provide insight into the data that can be used to make data-driven decisions. My dashboard successfully connects to multiple data sources to explore the data, find relationships, and create visualizations. This is useful as often times we have to source data from different places so it is important to know how to access those different sources and, in this case, view the data all in one place. Creating my dashboard meets the business needs of healthcare providers in obtaining information regarding their patients as well as seeing how they compare to other healthcare providers in order to observe trends and help make business decisions that can achieve goals such as lowering costs and readmission rates.

C2. Business Intelligence Tool

The two business intelligence tools I used are PostgreSQL and Tableau. PostgreSQL is an advanced and open-source relational database that allows users to store, scale, and even use other programming languages to build, view, and manipulate data (PostgreSQL, n.d.).

The tool I used most was Tableau. Tableau is a useful program primarily used for visualizing data, but which also provides light data cleaning capabilities, analytics, and provides easy access to users to create and share visualizations and findings in their data (Tableau, n.d.). Tableau also provides free version, in addition to paid versions.

Both of these tools were useful in accessing, storing, manipulating, and visualizing my datasets in order to understand the data, find trends, and meet business needs. Tableau, especially, makes it easy to find relationships with the ease and variety of data visualizations that are possible.

C3. Data Cleaning

Using both of these datasets in D210 allowed me to have additional insight as to what, if anything the datasets needed before visualization in Tableau. PostgreSQL allowed me to view the WGU-provided tables using Select * From statements to observe any issues before connecting the database to Tableau. Additionally, in Tableau, I was able to filter results so no NULLS were shown and could easily rename variables to make them more descriptive for viewers to understand.

Using Tableau, I was able to visualize and compare many of the variables to find trends and determine which visualizations were the most important and best showed the data I wanted to present.

C4. Dashboard Creation

Once the data sources are connected in Tableau, we are ready to create our dashboard.

1. First, use the navigation buttons on the bottom of Tableau on the left side. To the right of Data Source, click on Sheet 1 to move to a new worksheet.
    a. On the left column, all 3 tables we connected as our Data Sources should be listed with their respective columns listed below each table
2. For our first sheet, we are going to visualize the location and count of patients in the WGU data set
    a. Find State under the Location Table and drag it to the Columns field in Tableau. Tableau will automatically change it so that Longitude (generated) is in the columns field and Latitude (generated) is in the rows field.
    b. The Show Me tab on the right should have Symbol Map highlighted as the visual, but if not, click on Symbol Map
    c. Now, look at the Patient Table on the left, and at the bottom, a patient (Count) column should be present. Drag that to the Marks Box and onto the box that says Color
    d. Click on the color box, then click edit colors, use the drop down to choose Orange-Blue Diverging, which it is color blind friendly. Click okay. You should now have a map of the United States showing the count of patients in each state.
    e. Right click on Sheet1 in the bottom navigation bar and rename the sheet to WGU Patient Count Map so we know what this sheet shows. This should also change the name above our map, but if not you can rename it by double clicking on the Sheet1 label above our map and rename it to the same thing as our navigation bar
3. Now, on the bottom navigation tab, to the right of our current sheet, there is a button right next to it to create a new worksheet. Click that button. We now have a new sheet called Sheet 2

4. Next we want to look at Initial Days. From the Patient Table, drag Initial Days to the columns field. Tableau automatically changes it to SUM(Initial days), which we do not want. Once Initial Days is in the column field, click the arrow on the right of Initial days in the column field and click on Dimension. Now we just have our Initial days without a sum of them.
    a. Next, take Age from the Patient Table and drag it to the Rows field. Tableau again changes it to SUM, so click the arrow on the right and click Dimensions.
    b. We can also drag Initial days to the Filters box, click attribute and then click on next. It is now showing as ATTR(Initial Days). Click on the arrow on the right of ATTR(Initial Days) and click show filter. Do the same thing for Age from the Patient Table. We can now filter our Initial days and Ages.
    c. Drag Initial Days from the Patient Table to the Color box in the Marks field. Click the arrow on the right of Initial Days to change it to Dimension and not SUM. Then click the color box, edit colors, from the drop down choose orange-blue divergent, and hit ok.
    d. Just as we did with Sheet 1, we're going to rename this sheet WGU Age and Days so we know what it is from the navigation bar.
5. For our 3rd visualization, create another new worksheet. We'll look at Initial days and Ages in the DMC data
6. On the left column, locate the DMC data Table and find Duration of Stay
    a. Drag it to the Columns field. Click on the right down arrow and change Duration of Stay to dimension so it is no longer the sum
    b. Next find, AGE on the DMC data table and drag it to the Rows field. This also changes to SUM, so click the arrow on the right on AGE and change it to dimension.
    c. Drag Duration of Stay from under the DMC data table to the Color box in the Marks column. If it changes to SUM of duration of stay, click on the arrow on the right and change it to Dimension. Then click the color box, edit colors, and from the drop down choose orange-blue divergent, and click ok.
    d. Next drag Duration of Stay from the DMC Data Table to Filters. Click Attribute and next and then ok. Then click on the right arrow and click show filter. Do the same with Age
    e. Right click on Sheet 3 on the bottom navigation bar and rename it to DMC Age and Days
7. Create a 4th worksheet. The DMC data set does not have exact locations like the WGU dataset, but we can compare Rural and Urban areas with Duration of Stay
    a. Under the DMC data table, drag Duration of Stay to the Columns field. It changes to SUM, so click on the arrow on the right and change it to Dimension
    b. Then drag Rural to the Rows field
    c. Drag Rural from the table data on the left to the Color box in the Marks column. It should automatically set R as blue and U as orange. Duration of Stay will also show up as Detail and SUM. Click on the right arrow on Duration of Stay and change it to dimension.
    d. Our Visualization should have a label Rural and underneath R and U. Right click on R, click edit alias, and change it to Rural. Click ok. Right click on U, click edit alias, and change it to Urban. Click ok. This makes our visualization more readable with a description of what R and U mean.
    e. Right click on the worksheet in the bottom navigation bar and rename worksheet 4 as DMC Area and Days
8. To create dashboards with our visualizations, next to the new worksheet button on the bottom navigation, there is a new dashboard button. Click that to create a new dashboard. This Dashboard will be for our Area Visualization so I will right click on the dashboard and rename it to Area Dashboard

   a. On the left side under sheets, drag the WGU Patient Count Map to the dashboard. Then do the same with the DMC Area and Days sheet. I left the WGU sheet as tiled and changed the DMC sheet to floating and dragged it to the right side of the page under the legends.
   b. Double click the title of the top legend and change it to WGU count of patients. Click ok. Double click on the title of the next legend for Rural. Rename it to DMC Area Legend. Click Ok
9. Create a 2nd Dashboard. Rename it Age and Initial Days
   a. From the sheets on the right, drag WGU Age and Days to the dashboard. Do the same with DMC Age and Days. You can choose where you place it. I chose to place it on the bottom of the dashboard. Both sheets were added as Tiled so this caused the WGU sheet to shorten so it is on the upper half of the dashboard and the DMC data is below it
   b. Double click the title Age on the WGU Age slider and rename it to WGU Age. Hit ok. Double click on Initial Days title in the WGU initial days slider and also add WGU in front of Initial Days. Hit ok. Do the same for the DMC sliders but add DMC in front of the title. Right click on the DMC slider filters and change them to floating and drag them down the page so they are in line with the DMC sheet that was added to the bottom half of the dashboard. I also made the Legend floating and placed it between the sliders for the WGU and DMC visualizations.
10. In order to add my dashboards as one document to Tableau Public, I went back to the navigation tab at the bottom and clicked create new story. I renamed the story to D211 Story.
   a. Drag the Area Dashboard to the first page of the story. Change the caption to say Patient Demographics.
   b. On the left column, under new story point, click Blank to create a second page. Once that 2nd page is added, rename it to Comparison of Patient Age and Number of Initial Admission Days.
   c. On the 2nd story sheet, which is still blank, drag the Age and Initial Days dashboard to the page.
   d. Now that my Dashboards are on one story, click Server, Tableau Public, then save to Tableau Public as… This will bring up a box to name this, I chose D211 Story. I logged in to my free Tableau Public account and published the story.

C5. Data Analysis Results

By visualizing a map of the number of patients per state from the WGU dataset, shareholders can easily see which states have the most patients. Knowing this information would make it possible to update the marketing department in the best areas, understanding where expansion might be possible, as well as take a deeper dive into the patients that make up these numbers. Additionally, I found it interesting that WGU patients were consistently admitted for far longer at all ages than DMC patients. Knowing this, it tells shareholders that it would be helpful to explore this trend further to provide additional business insights. Understanding what caused the high amount of admission days would allow a provider to determine if that number is reasonable, what factors affect, and how we can use that information to be more efficient and potentially profitable.

C6. Analysis Limitations

One of the most noticeable limits of my analysis is that the DMC dataset provides information for patients from a hospital in India, whereas the WGU provided medical_data is from patients in the United States, so things like admission days may not be completely comparable across the two countries. Additionally, while the visualizations make it easy to see potential trends, it would be helpful to use Tableau with other tools such as predictive modeling or other more in-depth analytics to verify the visuals with statistical findings. On a more specific note, regarding the limitations of the WGU data set, much of the information in the DMC data set would be helpful to have in the WGU data set. Overall, the WGU dataset could be made better if it had more observations but also more data in general. Variables such as initial diagnoses and comorbidities would be helpful in better understanding the patients and how these variables may relate to each other.

D. Web Sources

No third-party code was used in this task.

E. Sources

Sahani, Ashish. (2022). *Hospital Admissions Data: Two Year Hospital Admissions and Discharge Data from Hero DMC Heart Institute*. Kaggle. https://www.kaggle.com/datasets/ashishsahani/hospital-admissions-data

Spicer, Thomas. (2017). *7 Steps to Export SQL Statements From Tableau*. OpenBridge. https://blog.openbridge.com/7-steps-to-export-sql-statements-from-tableau-7e51a2fd4277

PostgreSQL. (n.d.). *What is PostgreSQL?* PostgreSQL. https://www.postgresql.org/about/

Tableau. (n.d.). *What is Tableau?* Tableau. https://www.tableau.com/why-tableau/what-is-tableau#:~:text=Tableau%20is%20a%20visual%20analytics,See%20How%20Tableau%20Works

F. Professional Communication