Felipe Gomez

05/14/2024

BIDD 330A

Module 05

GitHub BIDD 330_Spring2024 Link: https://github.com/Phillips094/BIDD330_Spring2024

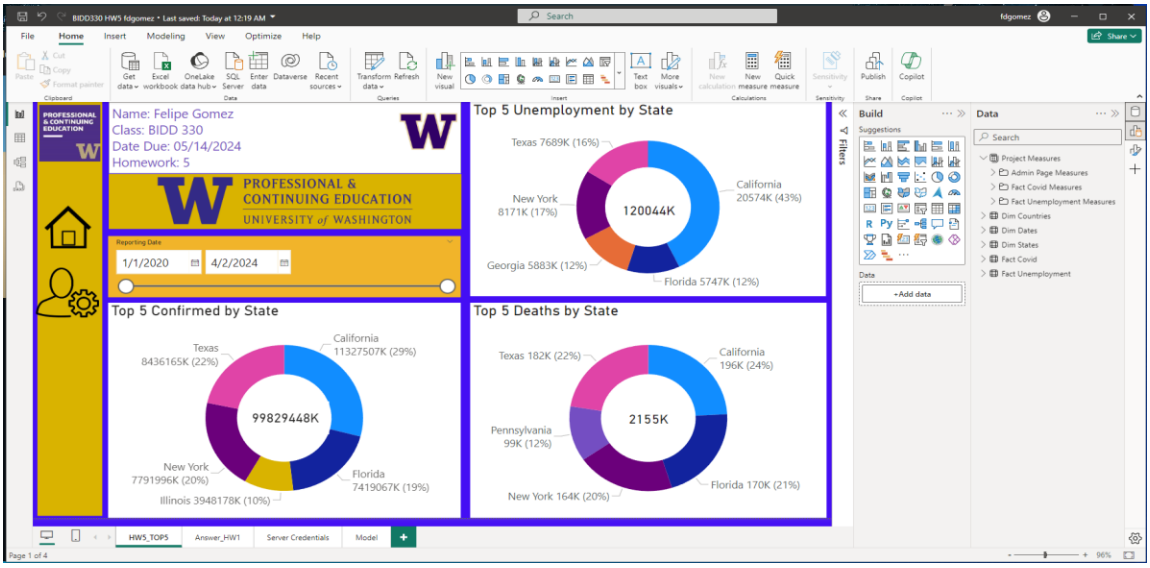
Intermediate Power BI and Jupyter Notebook

Introduction:

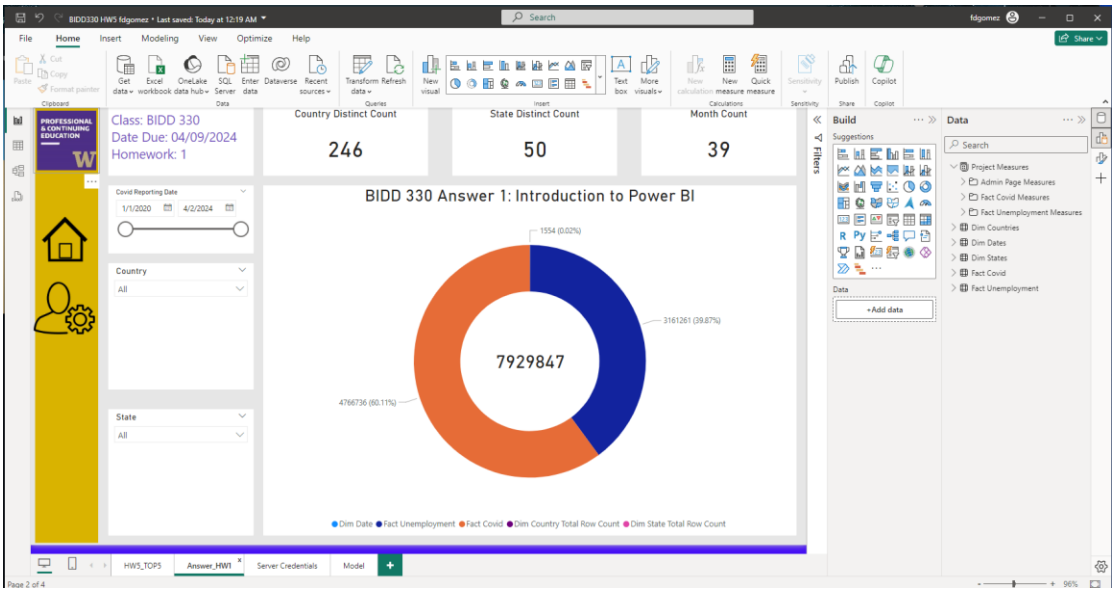
For module 05, we focus on developing an Intermediate Power BI report that builds from our previous HW1 Power BI assignment. We also focus on making updates from a jupyter notebook that is running python code to connect to our Black_Unemployment database and run data science experiments on our data. We copy over 2 images from our guest speaker's presentation and show them in our jupyter notebook. In our assignment, we perform "Intermediate" updates to our initial Power BI report, that demonstrate what someone who would be considered an intermediate at Power BI would be able to do.

For the Power BI updates, we focus on more difficult concepts that any BI Developer or Data Analyst would be required to learn. Some of the concepts included moving from materialized tables to views. We either create the views in our data warehouse and switch our connections to those views or we can write SQL code instead in our Power BI connection settings. We also focus on adding certain features like bookmarks to navigate through the report, we add Top 5 N charts for a certain category based on a certain value, slicers that control the report based on our Date Dimension, complex DAX calculations inside a measure table and finally publish to our UW PBI workspace.

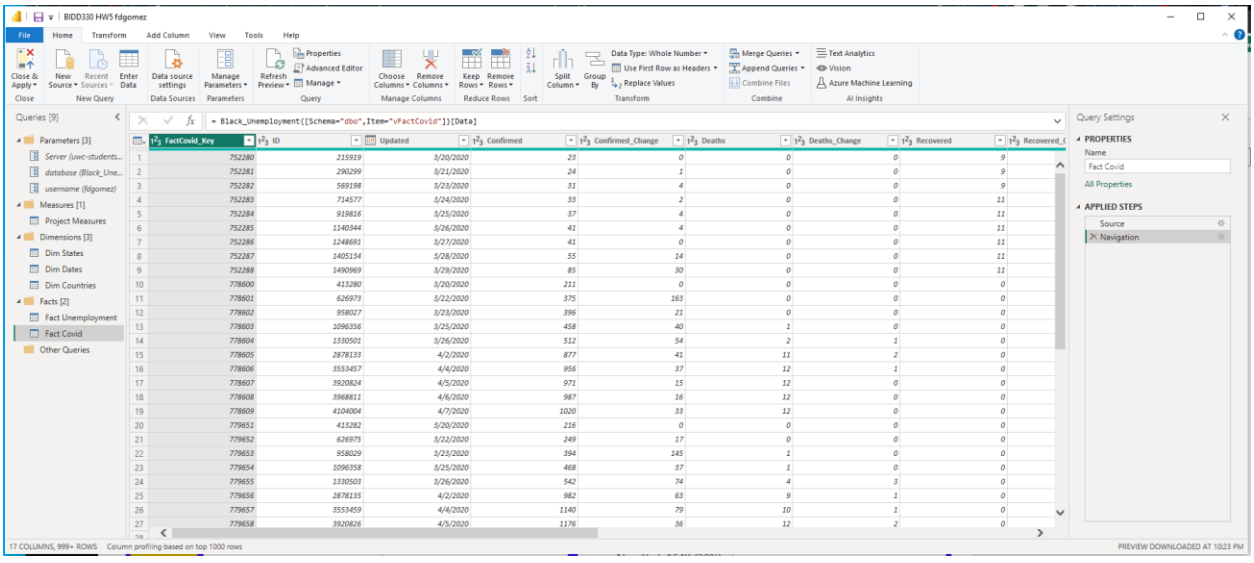
Our final Power BI intermediate report looks as below:



Our Power BI report focuses on TOP 5 for all our donut visualizations. We include our two navigation bookmarks, which are the “Home” and “Admin” buttons. They take us to different pages of our report. The “Home” bookmark is our main page shown above and our “Admin” bookmark gives us details about the row count from our dimensions and fact tables.



Lastly, for our Power BI report, we can see that we have our connections switched from our materialized tables to our views that were generated. For example, below is our updated Fact Covid table that is now pointing to our new view “vFactCovid”:



For our Jupyter Notebook assignment, we are given the task of either switching connections to in our Python script to our own connections or copying two charts from the script in the presentation.

We initially try to connect to our database using our credentials, so we make updates to our connection string for connecting to our data warehouse in SQL Server. Below is a screenshot of our updates comparing the old to our new and updated code:

jupyter CovidDataScience2024 Last Checkpoint: 2 minutes ago

File Edit View Run Kernel Settings Help Not Trusted

Code JupyterLab Python 3

2.Acquire Data

[ ]:
```
import pyodbc
import pandas as pd
# This ignores the warning in the pyodbc package
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore", category=UserWarning)

conn = pyodbc.connect('Driver={SQL Server};'
                      'Server=uwc-studentsql.continuum.uw.edu\uwcbiddsql;' #104.42.235.139;'
                      'Database=Black_Unemployment;' #bing_covid-19_database;'
                        'UID=fdgomez;' #UNtower01;'
                        'PWD=7U_}u4GkZb});' #ZLhGZ3tZKZyg;'
                        'Trusted_Connection=no;')

query = 'SELECT * FROM [dbo].[Raw_BingCovid]'
# Directly read your query into a DataFrame
df = pd.read_sql_query(query, con=conn)
# Make sure to close your connection
conn.close()
```

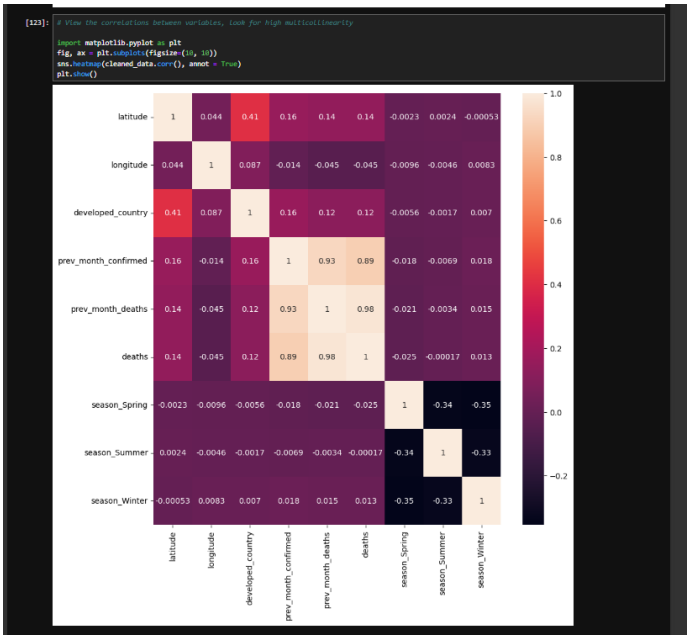
[*]:
```
import pyodbc
import pandas as pd
# This ignores the warning in the pyodbc package
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore", category=UserWarning)

conn = pyodbc.connect(
    'Driver={SQL Server};'
    'Server=uwc-studentsql.continuum.uw.edu\\uwcbiddsql;'
    'Database=Black_Unemployment;'
    'UID=fdgomez;'
    'PWD=7U_}u4GkZb});'
    'Trusted_Connection=no;'
)

query = 'SELECT * FROM [dbo].[Raw_BingCovid]'
# Directly read your query into a DataFrame
df = pd.read_sql_query(query, con=conn)
# Make sure to close your connection
conn.close()
```

We also update include screenshots of two of the visualizations provided by our guest speaker Matt in his Python script:

1)

2) 

These are two visualizations using the Seaborn API library in Python.

Summary:

In summary, we further investigate our Covid dataset by utilizing intermediate concepts in Power BI to develop meaningful visualizations that give us insights on the Covid data. Developing the intermediate level Power BI report requires grasping new concepts that take us to the next level, especially when it comes to the navigation panel and our new DAX equations. The jupyter notebook updates requires us to understand a bit of Python code and how to connect to our data warehouse using the api library pyodbc. Overall this assignment helped provide us with some insight into how to navigate through a jupyter notebook script and make updates to our Power BI reports.