CITS5504 Data Warehousing

Project 1 Report

Yiren Wang (23794201)

2024-04-09

Contents

Introduction

When he Olympic torch was lightened in Athens, April 1896, Greece for the very first time, this huge international sports event, combining art and sport, was introduced to the public. Every four years, athletes from all over the world were prepared and joined for this game, showcasing not only their amazing sports skills to the audience but also their will to represent their nations with pride. In addition, hosting such huge events is beneficial to the host country in many ways. For instance, the Sydney Olympics helped Australia's economy, resulting in a staggering 6.1 billion Australian dollars more in GDP. Investigating the profound effects of the Olympics on public and organizations is becoming increasingly important, as is developing new tools for analyzing data and drawing conclusions about these matters.

Regardless of whether country hosted the Olympics or won any gold medals, the Olympic have a significant influence on countries all around the world. As a result, gathering information on the Olympics and its profound effects on society and institutions, as well as developing new tools for data analysis and conclusion drawing, are becoming more and more important.

The General Administration of Sport of China is the official organization In China for checking and regulating sports activities, which makes it the powerhouse of the Chinese sports development. It attempts to encourage public engagement in physical fitness activities while enhancing national sports capabilities and competitive ability, particularly in worldwide sports competition. The establishment of a data warehouse will provide the administration with a wealth of Olympic-related information, such as host nations, past results, and socioeconomic backgrounds, to help with these endeavors. With the use of this technology, the government will be able to customize investments and policies to increase public engagement in sports and support for Olympic athletes. These decisions will be based on a thorough examination of both economic and global sports trends.

Another client is a potent national public health agency in the United States called The Centers for Disease Control and Prevention (CDC), The primary goal of the CDC is to prevent and investigate disease, trauma, and disability to preserve the general public's health and well-being. It is beneficial to gain a deeper understanding of the relationships between many aspects, including mental health, medical standards, and the economy in different nations for CDC by evaluating and developing relationships among birth rates, sports levels, and health expenses created in a data warehouse. Using seven comma-separated value (CSV) files, this project aims to construct a data warehouse that will be utilized to simulate and analyze three business queries. To provide a thorough overview, this project is divided into eight parts. Firstly, the project assumes three hypothetical clients who might benefit from the data. The subsequent section will provide summaries and relevant information about these CSV files. The related Star Schema diagrams for these three business queries are introduced in the third section. The fourth session, which is the most important part of this project, includes the process explanation of the Fact table and Dimension table, and data warehouse modeling. Introduction to the process of ETL stages and data cleaning are provided as part of the fifth step. OLAP cube design will be explained in the sixth part. After that, data visualization for these business queries will be shown in the seventh part. Finally, the project will explain some questions regarding these queries.

Data Preparation

Concept Hierarchies

Firstly, we need to explore the Olympic dataset and create conceptual hierarchy for each dimension. In StarNet, each footprint represents a level of the dimention, culminating in an "All" at the top. Taking "region" as an example, its conceptual hierarchy is divided into three levels: total, continent, and country. These three levels are reflected in the StarNet model.

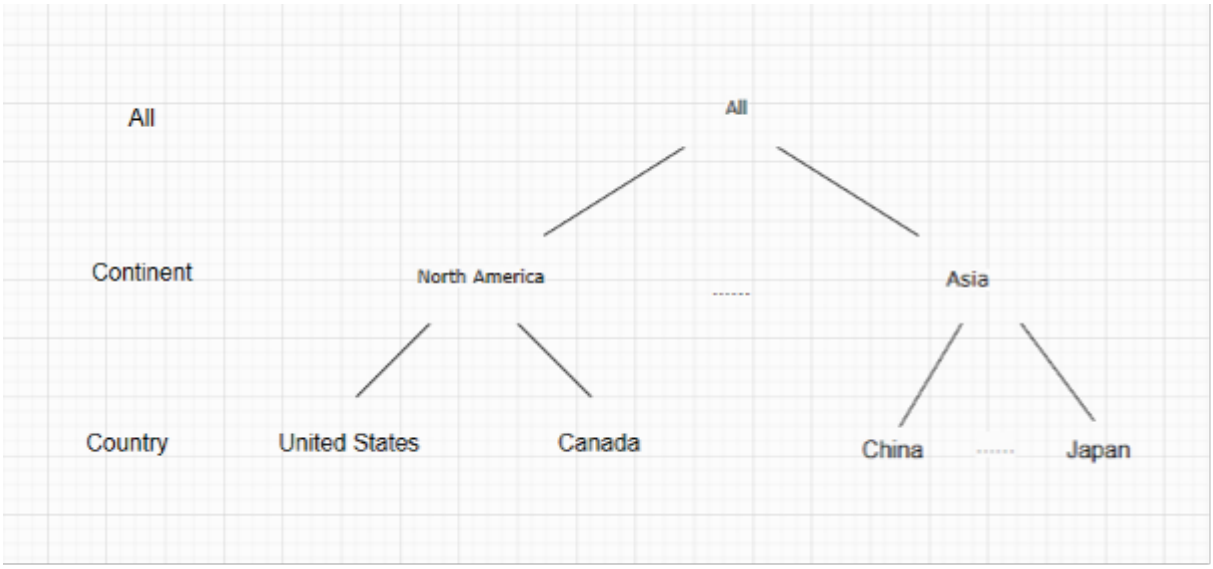

Figure 1: Concept Hierarchy

StarNet

In the StarNet model below, we can see that the chart is designed to analyze the relationship among the sports level of various countries, country size, wealth, and country. Each line in the chart represents these factors, with footprint on the lines indicating different levels of detail within the model. This means that the generality or range of data changes with the level of abstraction. The variation in detail depends on the OLAP process applied to the model. For example, in the "Region" dimension, users can generalize data from the country to the continent level through roll up, or access more detailed information from the state to the national level through roll down.

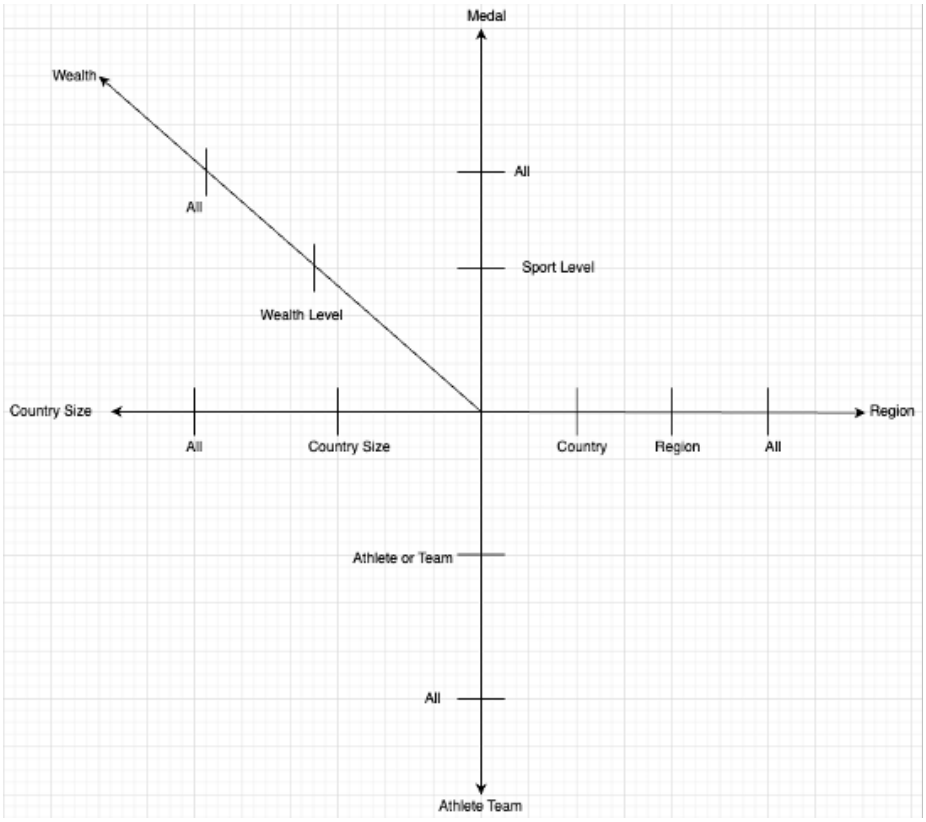

Figure 2: Client 1 StarNet

The StarNet for Client 2 below focuses on health and costs. The Starnet is connecting dimensions such as Country Size, Region, Country, Sports Level, Internet Level, and Host. These dimensions highlight the importance of analyzing health costs based on geographical, technological, and sports-related factors to understand how these factors affect medical expenditures or outcomes in different regions. The chart also suggests a relationship between the level of internet access and healthcare costs.

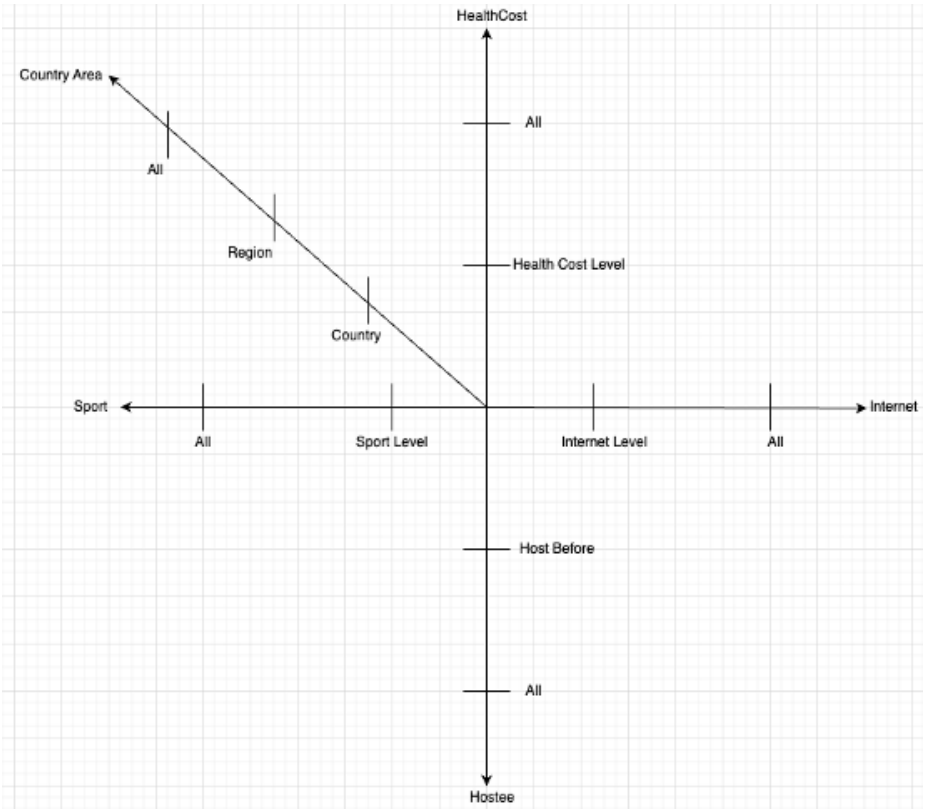

Figure 3: Client 2 StarNet

Business Queries with StarNet

Five business queries are created for each client, and the queries are listed as follows:

Client 1: The General Administration of Sport of China

1. How many times have countries with huge population host in total?

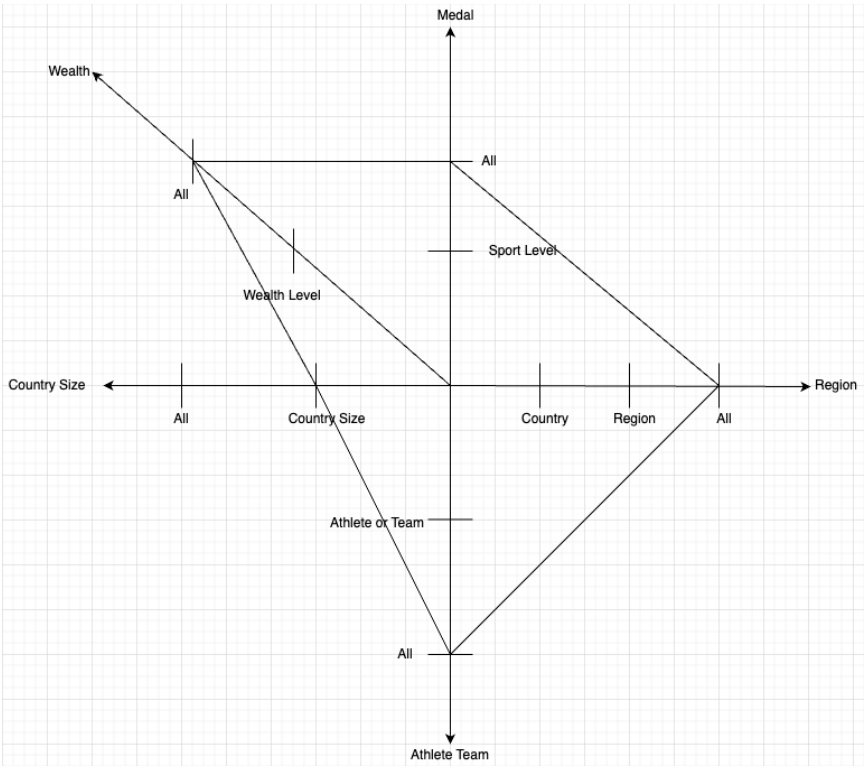

Figure 4: Client 1 Business Query 1

2. How many medals did the country with the top five highest GDP win in the 2021 Olympics?

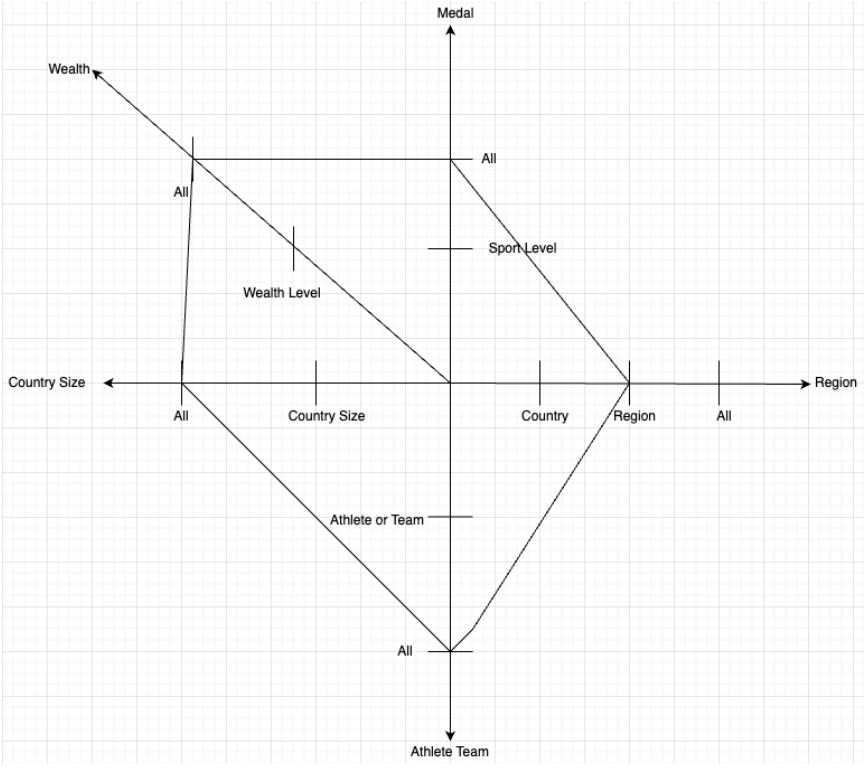

Figure 5: Client 1 Business Query 2

3. What is the GDP growth rate for the country won the most medals in Japan Olympic?

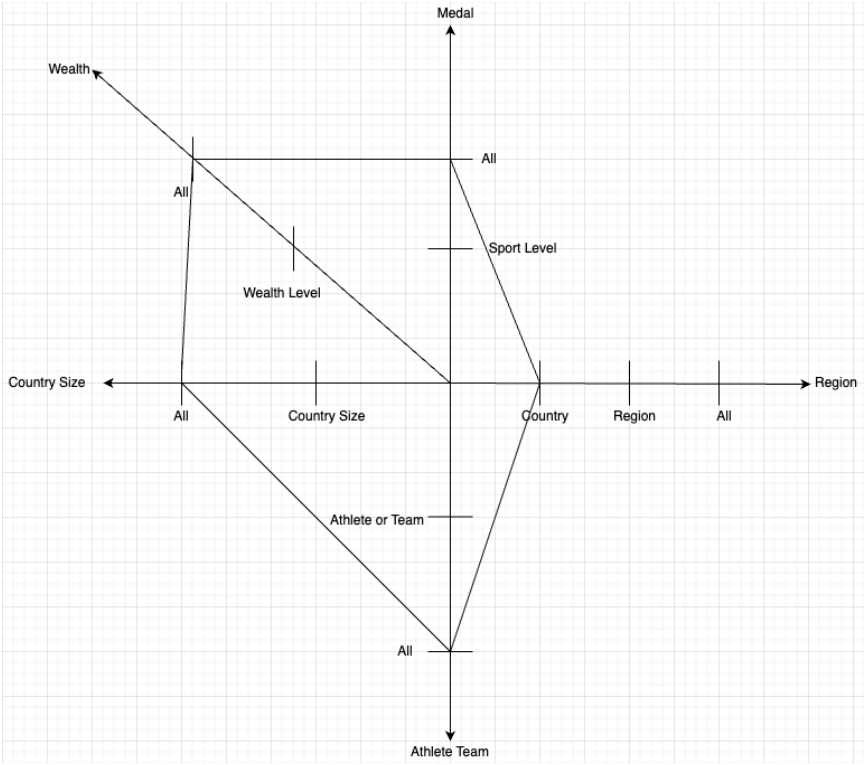

Figure 6: Client 1 Business Query 3

4. Do populous countries prefer individual or team sports?

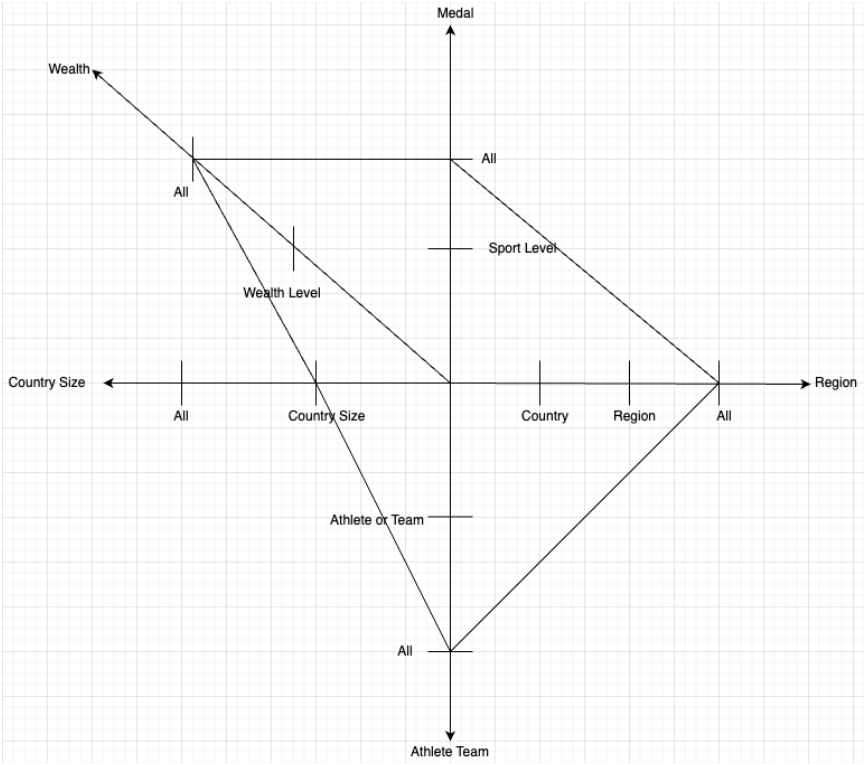

Figure 7: Client 1 Business Query 4

5. How many medals has the country that aces in individual sports won in their favorite sport?

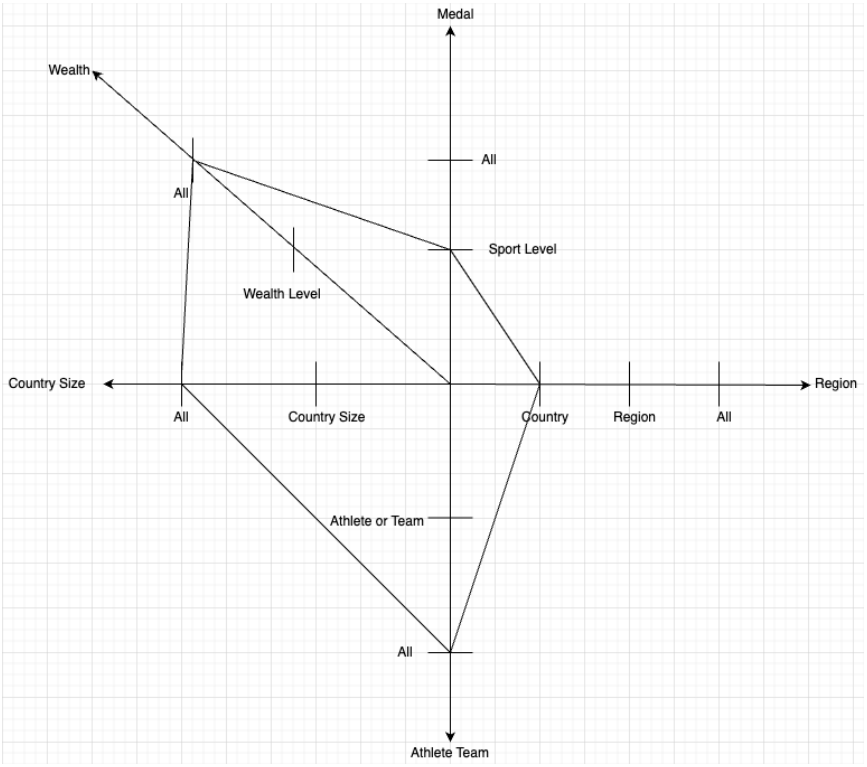

Figure 8: Client 1 Business Query 5

Client 2: The Centers for Disease Control and Prevention.

1. Which country have lower yearly population growth rate after 2000 comparing to one before 2000?

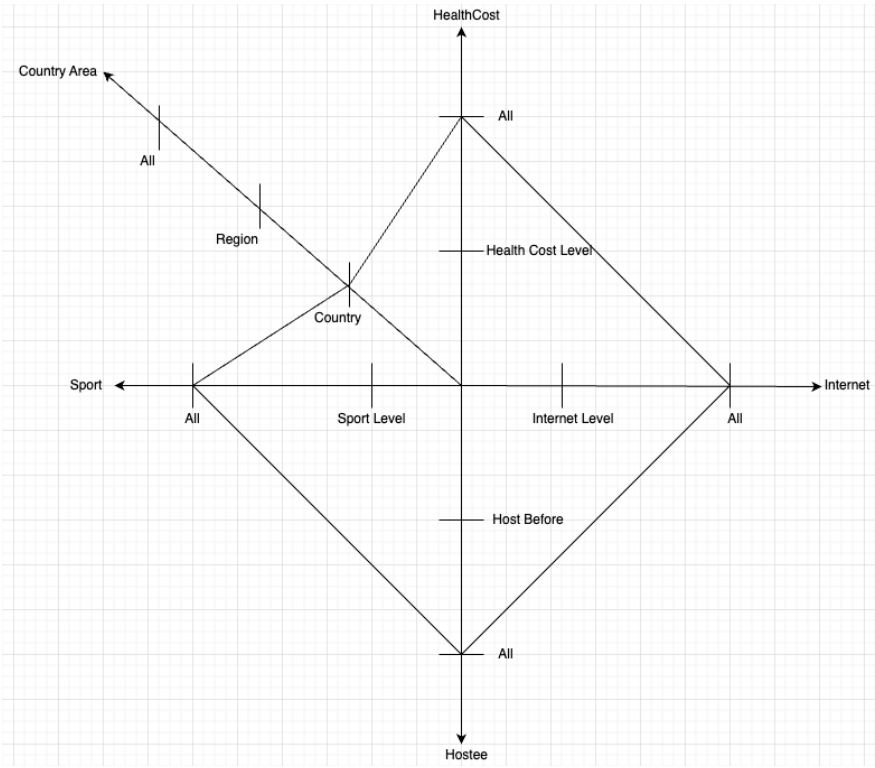

Figure 9: Client 2 Business Query 1

2. What is the different between the money the continent with the most strongest sports power country spend on health compared to the continent with the most average sports power country?

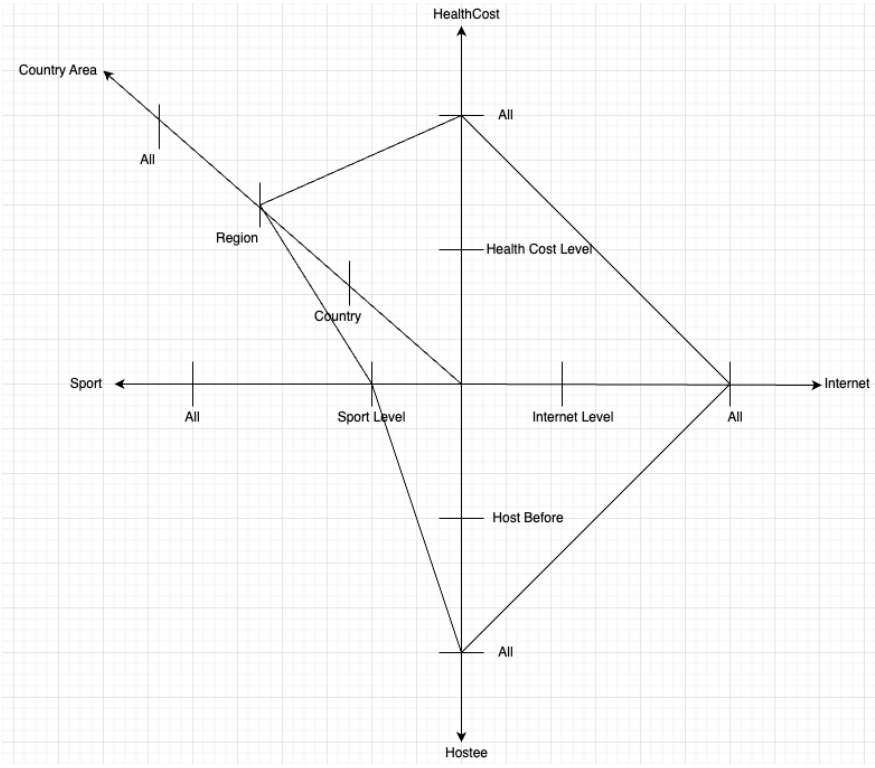

Figure 10: Client 2 Business Query 2

3. Do governments with middle health costs prefer to pay for their citizens' healthcare themselves, or do they let their citizens bear the costs?

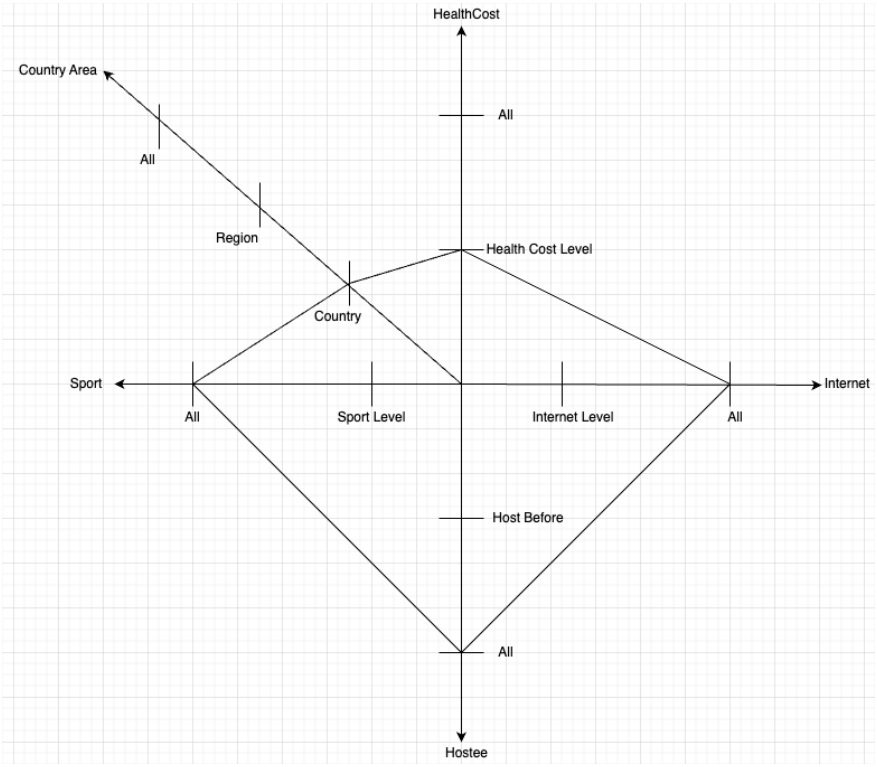

Figure 11: Client 2 Business Query 3

4. How does the GDP look like for middle countries with strong internet systems after hosting the Olympics?

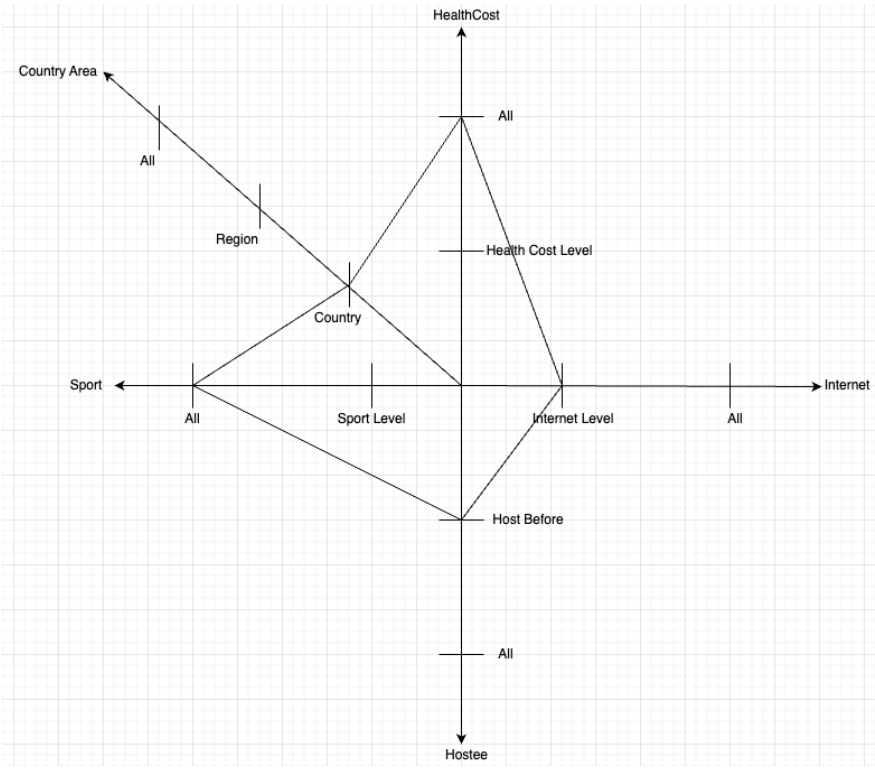

Figure 12: Client 2 Business Query 4

5. The change in the number depress disorder in all continents that not good at sports, specifically before and after the 2002 Olympics.

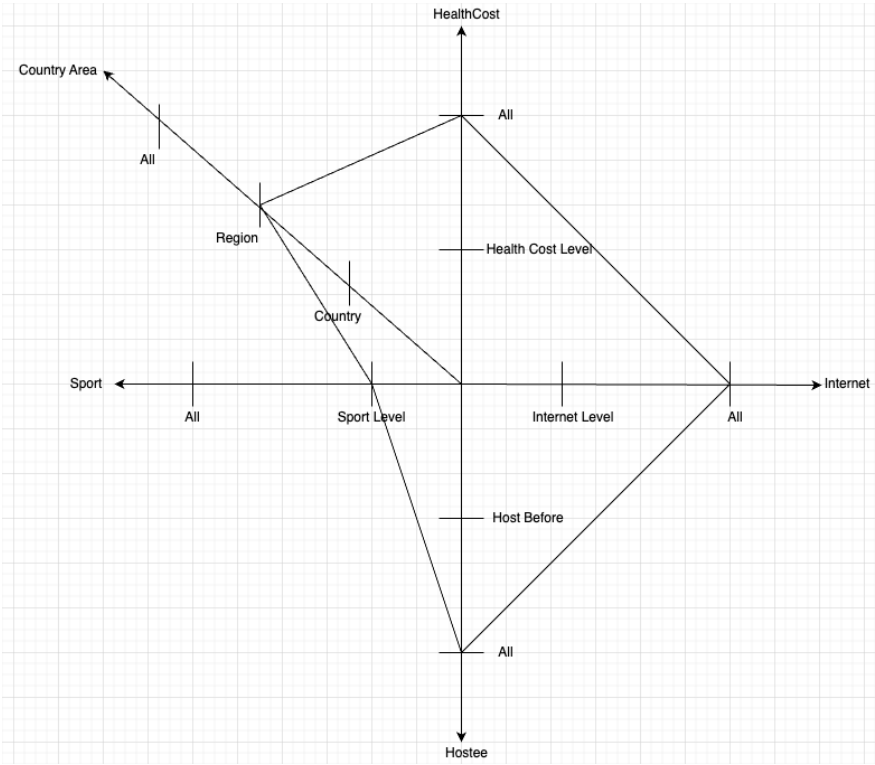

Figure 13: Client 2 Business Query 5

Star Schema

The databases of two clients, that are shown below, are both constructed by the star schema. For the first client, there are five dimension tables, including countrysize, region, sports level, economy level, and the size of gamers. The measuer used on the fact table the number of medals from the 2021 Olympics, GDP, GDP growth rate, the number of medals in sports each country good at, the number of times hosting the Olympics, and the average number of medals in individual and team competitions.

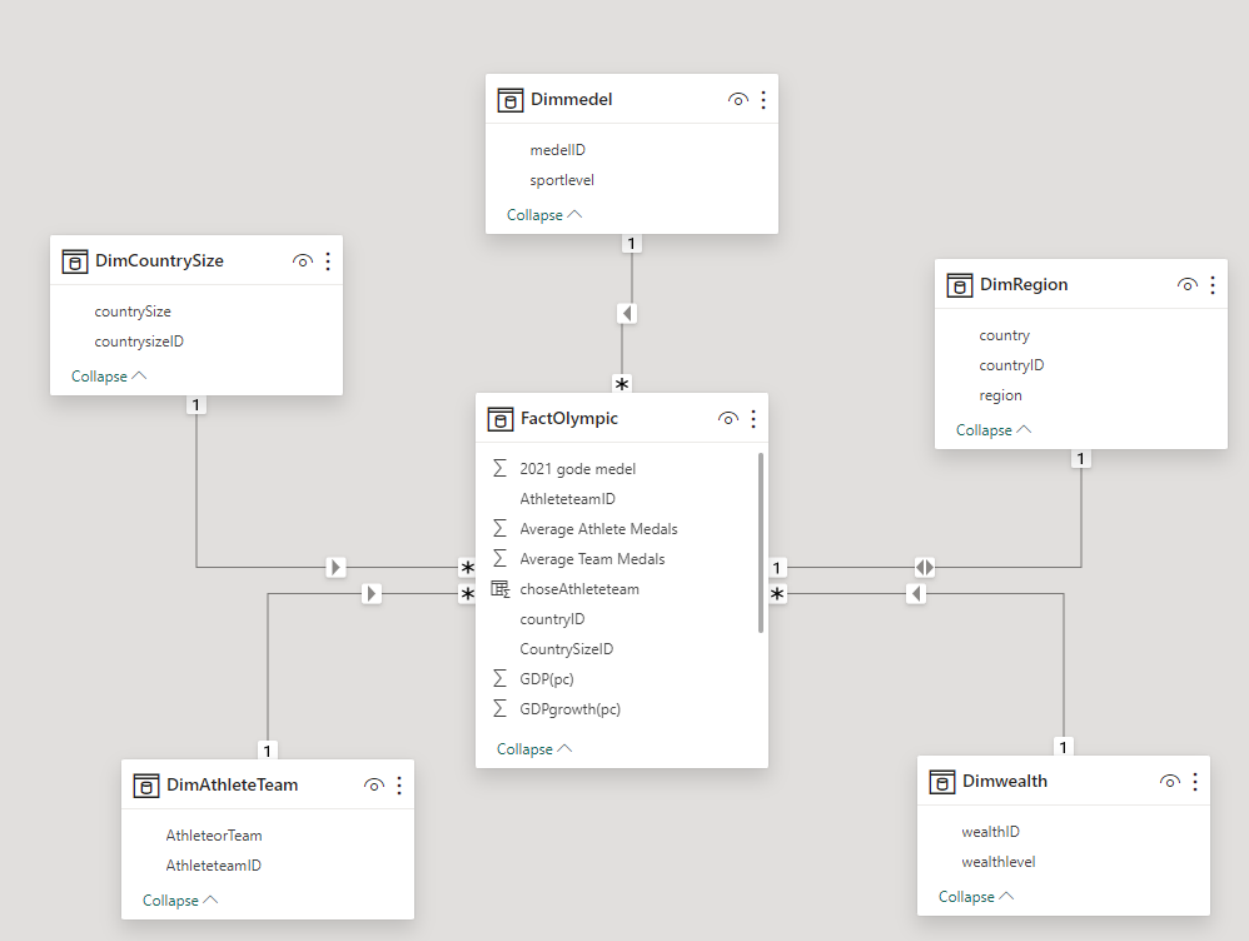

Figure 14: Star Schema For Client 1

In the second client, there are also five dimension tables surrounding the fact table, including health expenditure level, internet level, country/region, sports capability, and whether the Olympics have been hosted or not. On the other hand, the fact table contains the measures including the average yearly GDP growth rate, the change amount of depression disorder in 2002, GDP per capita, and the difference of health expenditures between 2002.

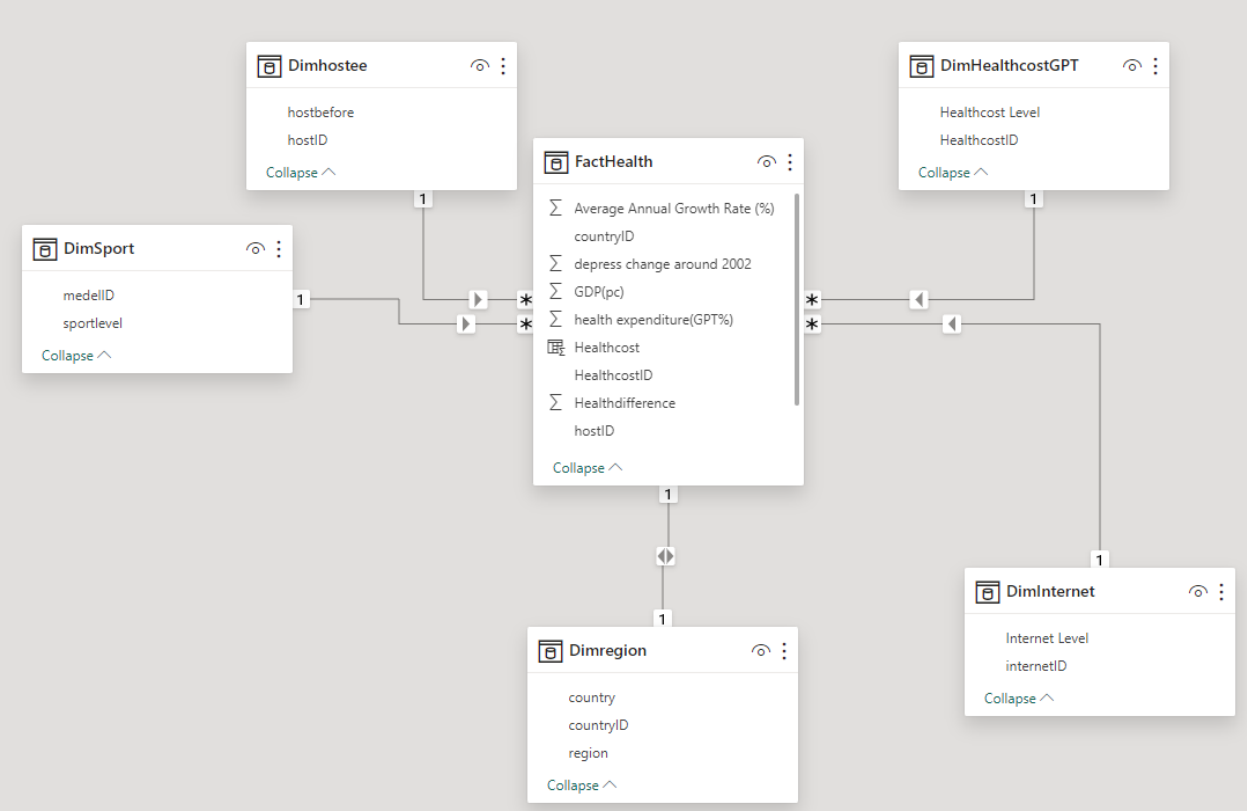

Figure 15: Star Schema For Client 2

Extract, Transform, and Load

We can see that, from the Olympic dataset, there are a lot of data issues and inconsistencies, such as missing values. Therefore, I used python, Microsoft VS code, and sql to extract, transform and load the cleaned and appropriate data, then using Jupyter notebook to create and populate the database in the pgadmin4 server.

The first step of the ETL process is to explore and analysis all the relevant data we are asked to answer the business queries. Based on the queries analysis above, we have to remodel and structure the data in an appropriate schema in order to extract the necessary information.

```
[2]: # upload Economic data.csv file
     Ecodata = pd.read_csv("Economic data.csv")
     Ecodata.head()
```

[2]:

	Time	Time Code	Country Name	Country Code	Poverty headcount ratio at $2.15 a day (2017 PPP) (% of population) [SI.POV.DDAY]	GDP per capita (current US$) [NY.GDP.PCAP.CD]	GDP per capita growth (annual %) [NY.GDP.PCAP.KD.ZG]	Secure Internet servers (per 1 million people) [IT.NET.SECR.P6]	Mortality rate, infant (per 1,000 live births) [SP.DYN.IMRT.IN]	Current expenditure [SH.XPD.CHEX.(
0	2020	YR2020	Afghanistan	AFG	..	516.8667974	-5.364665931	34.94796166	44.8	15.533
1	2020	YR2020	Albania	ALB	0	5343.037704	-2.745238678	884.8250911	8.4	
2	2020	YR2020	Algeria	DZA	..	3354.157303	-6.729941651	48.46764679	19.6	6.32
3	2020	YR2020	Andorra	AND	..	37207.222	-12.73507756	9665.379665	2.7	9.05
4	2020	YR2020	Angola	AGO	..	1502.950754	-8.672432129	19.7436402	48.7	2.911

Figure 16: Read csv File

Firstly, as the image shown above, I used the read_csv() statement from the pandas tookit to import all data into Python and assign them to different variables, also we will observe their data structure. These datasets include both numerical and textual data, which is a big advantage for our data analysis. However, before proceeding with the analysis, it is necessary to rename each column with long column names by using .rename() to benefit following analysis tasks.

```
mentalhealth.rename(columns={'Entity': 'Country'
                             , 'DALYs from depressive disorders per 100,000 people in, both sexes aged age-standardized': 'depressive disorders DALYS (m)'
                             , 'DALYs from schizophrenia per 100,000 people in, both sexes aged age-standardized': 'schizophrenia DALYS (m)'
                             , 'DALYs from bipolar disorder per 100,000 people in, both sexes aged age-standardized': 'bipolar disorder DALYS (m)'
                             , 'DALYs from eating disorders per 100,000 people in, both sexes aged age-standardized': 'eating disorders DALYS (m)'
                             , 'DALYs from anxiety disorders per 100,000 people in, both sexes aged age-standardized': 'anxiety disorders DALYS (m)'
                            }, inplace=True)
```

Figure 17: Rename Columns

To make sure the completeness and rationality of the data, it is essential to clean rows and columns and deal with missing values. In the Olympic data package, for example, all data contain a certain amount of missing values. Taking "Economic data.csv" as an example, we use the .isna().sum() statement to check the number of null values in each column. We find that most of the countries with missing values come from either poor countries or countries in wartime. Therefore, using the average value of the entire column is not a good choice to fill the blanks Instead, we choose to fill in the missing values with the minimum value from the list, selecting the most appropriate data without affecting the analysis.

```
Economyhea = pd.read_csv("./cleandata/countrycleanEconomydata.csv")
Economyhea['Personexpenditure_in_health'] = Economyhea['Personexpenditure_in_health'].fillna(Economyhea['Personexpenditure_in_health'].min())

Economyhea['Healthdifference'] = Economyhea['govermentpay_in_health'] - Economyhea['Personexpenditure_in_health']

# Select the relevant columns to display the result
differ = Economyhea[['Country', 'govermentpay_in_health', 'Personexpenditure_in_health', 'Healthdifference']]
```

Figure 18: Dealing With Missing Values And Extracting Columns

In the Olympic data, certain values for countries and regions need to be removed from the data frame, such as G7, Asia, and middle income. The data from these entities have a minimal impact on the Olympic games and are not helpful for our research at the country level. Storing data in this manner introduces a level of inequality. For instance, D7 and groups of low-income countries do not specify which countries they include or how to distinguish the standard for low income, leading to confusion.

Fixing country names is also a challenge in the ETL process. In all six files in the Olympic dataset, each table has its own country column, and the names are different to each other. To facilitate data analysis in future, we will use the country column in list-of-countries_areas-by-continent-2024.csv as the standard to modify other tables. For other tables, we will remove the names that are not considered as country, and the country name that no longer exist due to historical and political reasons. After that, for cases where the same country that has different names in different tables, we will use the .difference() statement as below to filter out incorrect names, and then correct and modify the names in VS Code.

```
countries_set = set(Countrylist['country'])
olympic_hosts_set = set(OlympicHost['game_location'].dropna())
olympic_medals_set = set(OlympicMedel['country_name'].dropna())
lifeset = set(cleancountry['Country'].dropna())

diff_olympic_hosts = olympic_medals_set.difference(olympic_hosts_set)
diff_olympic_medals = olympic_medals_set.difference(countries_set)
ohno = lifeset.difference(countries_set)
diff_olympic_hosts
```

Figure 19: Fixing Country Names

Next, we will proceed with deeper processing and transformation of the data. We will create a new column in the fact table to store calculated values based on population numbers. The detailed process is as follows: We use an if statement to go through the population data of each country and then compare each piece of data against three defined levels. The number 1 represents a population size of more than 40 million, 3 represents a population of less than 2 million, and for the rest of the population sizes, we use 2 to represent them. These numbers 1, 2, and 3 correspond to the classifications in DimCountrySize.

```
countrywithpopulation = countrywithmedel.merge(populationdata[['Country', 'population in 2021']], left_on='country', right_on='Country', how='left')

# Drop the duplicate country column
countrywithpopulation.drop(columns=['Country'], inplace=True)

countrywithpopulation['population in 2021'] = countrywithpopulation['population in 2021'].fillna(0).astype(int)

def countrymodel(x):
    return 1 if x > 40 else 3 if x < 3 else 2

countrywithpopulation['CountrySizeID'] = countrywithpopulation['population in 2021'].apply(countrymodel)
countrywithpopulation.drop(columns=['population in 2021'], inplace=True)
countrywithpopulation
```

Figure 20: Country Selection

In the fact tables of both clients, there are many examples like this, such as DimWealth. To categorize countries by different economic levels, we use If statement to go through the GDP of all countries, and then divide the highest GDP value by three. This allows us to establish three levels. Countries are then classified using the numbers 1, 2, and 3 according to this classification.

```
Econ = pd.read_csv("./cleandata/countrycleanEconomydata.csv")
countrywithGDP = countrywithpopulation.merge(Econ[['Country', 'GDP(pc)']], left_on='country', right_on='Country', how='left')
countrywithGDP.drop(columns=['Country'], inplace=True)
countrywithGDP['GDP(pc)'] = countrywithGDP['GDP(pc)'].fillna(0)
splitGDP = countrywithGDP['GDP(pc)'].max() / 3

def countrymodel(x):
    return 1 if x > 2 * splitGDP else 2 if x > splitGDP else 3

countrywithGDP['wealthID'] = countrywithGDP['GDP(pc)'].apply(countrymodel)

countrywithGDP
```

Figure 21: GDP Selection

For the measures of medals in the fact table and the dimension table, although the measures we need and the data corresponding to the dimension table in the olympic_medal.csv are in text format, we can still process and convert this data into numeric information. For example, in the fact table, if we need to calculate the total number of medals for each country, we can use the .groupby() statement to group by country, and then count the number of medals for each country. This processing not only transforms text data into a computable numeric form but also benefits future data analysis and decision-making.

```
medel = pd.read_csv("./cleandata/countrycleanmedel.csv")
countrywithmedel = DimRegion.merge(
    medel.groupby('country_name').size().reset_index(name='total_medals').sort_values(by='total_medals', ascending=False),
    left_on='country', right_on='country_name', how='left')

countrywithmedel['total_medals'] = countrywithmedel['total_medals'].fillna(0).astype(int)
countrywithmedel.drop(columns=['country_name'], inplace=True)
```

Figure 22: Calculation of Number of Medals

Implementation

Create table schema

After cleaning the Olympic data, my next step is to create database, tables, and their schema in pgAdmin 4. This process will be constructed through Python script, to ensure the uniqueness of the database and tables with no other database and tables with the same names. After that, we will create a database named 'Project1' and define the schemas of fact tables and dimension tables, including their primary and foreign key relationships. This is to follow the star schema data warehouse design, optimizing data analysis and query efficiency.

```
import psycopg2
from psycopg2 import OperationalError


def create_connection(db_name, db_user, db_password, db_host, db_port):
    connection = None
    try:
        connection = psycopg2.connect(
            database=db_name,
            user=db_user,
            password=db_password,
            host=db_host,
            port=db_port,
        )
        print("Connection to PostgreSQL DB successful")
    except OperationalError as e:
        print(f"The error '{e}' occurred")
    return connection
```

```
# Connection details
db_name = "project1"
db_user = "postgres"
db_password = "postgres"  # Update with your password
db_host = "pgdb"  # Update if your DB is hosted elsewhere
db_port = "5432"

# Create the connection
connection = create_connection(db_name, db_user, db_password, db_host, db_port)
```

Figure 23: Connection to pgAdmin

Next, I will use the SQL "CREATE TABLE" statement to construct the schema of mentioned tables above. Let the dimension table "dimRegion" as an example, I will define the column named "countryID" as the primary key to uniquely identify each country. The table will include other necessary columns, and each column clearly specify the data type, whether null values are allowed, and other constraints to ensure data integrity and accuracy.

```
create_tables_queries = [
    '''
    CREATE TABLE IF NOT EXISTS DimHealthcost ( HealthcostID INTEGER PRIMARY KEY, "Healthcost Level" VARCHAR(255));

    ''',
    '''
    CREATE TABLE IF NOT EXISTS Dimhostee ( hostID INTEGER PRIMARY KEY, "hostbefore" VARCHAR(255));

    ''',
    '''
    CREATE TABLE IF NOT EXISTS Dimcountryarea ( country TEXT NOT NULL, region TEXT NOT NULL, countryID INTEGER PRIMARY KEY );

    ''',
    '''
    CREATE TABLE IF NOT EXISTS DimInternet ( internetID INTEGER PRIMARY KEY, "Internet Level" VARCHAR(255));

    ''',
    '''
    CREATE TABLE IF NOT EXISTS DimSport ( medelID INTEGER PRIMARY KEY, sportlevel VARCHAR(255));

    ''',
    '''
    CREATE TABLE IF NOT EXISTS FactHealth (
        countryID bigint PRIMARY KEY,
        "health expenditure(GPT%)" NUMERIC,
        HealthcostID INTEGER,
        InternetID INTEGER,
        "Average Annual Growth Rate (%)" NUMERIC,
        Healthdifference NUMERIC,
        "depressive disorders rate of Change" NUMERIC,
        medelID INTEGER,
        hostID INTEGER,
        "GDP(pc)" NUMERIC
    );
```

Figure 24: Create Table

Finally, each dimension table will be constructed with a primary key to follow the star schema, and the fact table will link to them through foreign keys. This step ensures that the fact table and dimension tables can be connected efficiently.

```
    ,
'''
ALTER TABLE FactHealth ADD CONSTRAINT FK_countryID FOREIGN KEY (countryID) REFERENCES Dimcountryarea(countryID)
''',
'''
ALTER TABLE FactHealth ADD CONSTRAINT FK_HealthcostID FOREIGN KEY (HealthcostID) REFERENCES DimHealthcost(HealthcostID)
''',
'''
ALTER TABLE FactHealth ADD CONSTRAINT FK_InternetID FOREIGN KEY (InternetID) REFERENCES DimInternet(internetID)
''',
'''
ALTER TABLE FactHealth ADD CONSTRAINT FK_medelID FOREIGN KEY (medelID) REFERENCES DimSport(medelID)
''',
'''
ALTER TABLE FactHealth ADD CONSTRAINT FK_hostID FOREIGN KEY (hostID) REFERENCES Dimhostee(hostID)
'''
```

Figure 25: Define Foreign Keys Relationship

Upload the data

Completing the construction of the table schema in pgadmin 4, we will use Python scripts to import the cleaned data from .csv files into the dimension and fact tables in pgAdmin. This process refers to the code from Lab 3. By using the 'read_csv' statement, we can read cleaned data from csv files and insert them into the schema The directory path where the CSV files are stored is set to ./projectdata/Client1 or ./projectdata/Client2.

Data analyze and Data Warehouse

Cube

Star schema in PGadmin

After successfully connecting to pgAdmin using python and creating two fact tables and dimensional tables for two clients, we new continue to use Python to line the data to pgAdmin and create data cubes for both clients. Two following plots are the data relationship diagrams between the fact table dimension tables. Th appearance of data cube enhance the possibilities and operability of data analysis and organization, which makes it better than other sql database.

This is the data relationship diagram for Client 1.

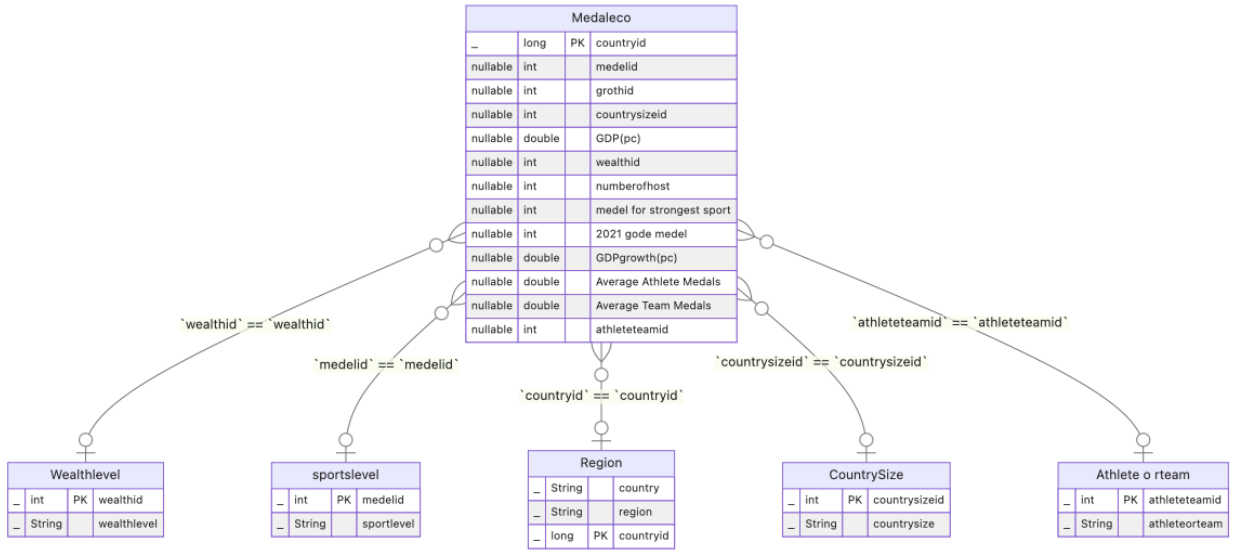

Figure 26: Star Schema For Client 1

This is the data relationship diagram for Client 2.

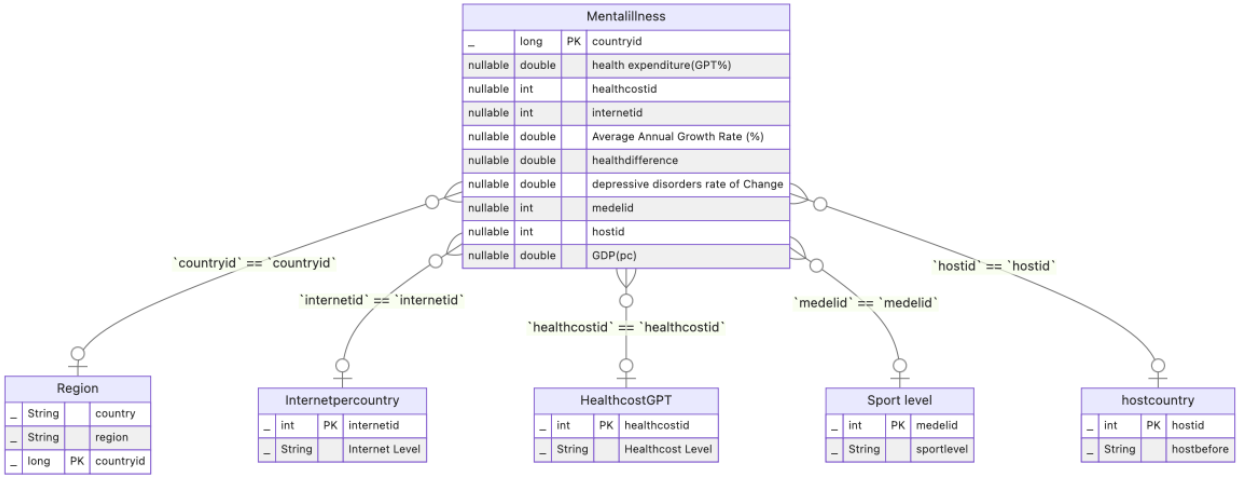

Figure 27: Star Schema For Client 2

Data Cube Hierarchy

Data cubes play a significant role in data warehouse by creating meaningful concept hierarchies to establish relationships among data. Consider the concept hierarchy for Client 1 below as an example, after we establish a hierarchical structure between continents and countries, we can use OLAP operations such as drill-down to obtain detailed information about each country in a continent, or to explore which countries are associated with specific continent. Once the hierarchical structure of countries is successfully constructed, we can delve into the relationships between features by performing roll-up and drill-down operations.

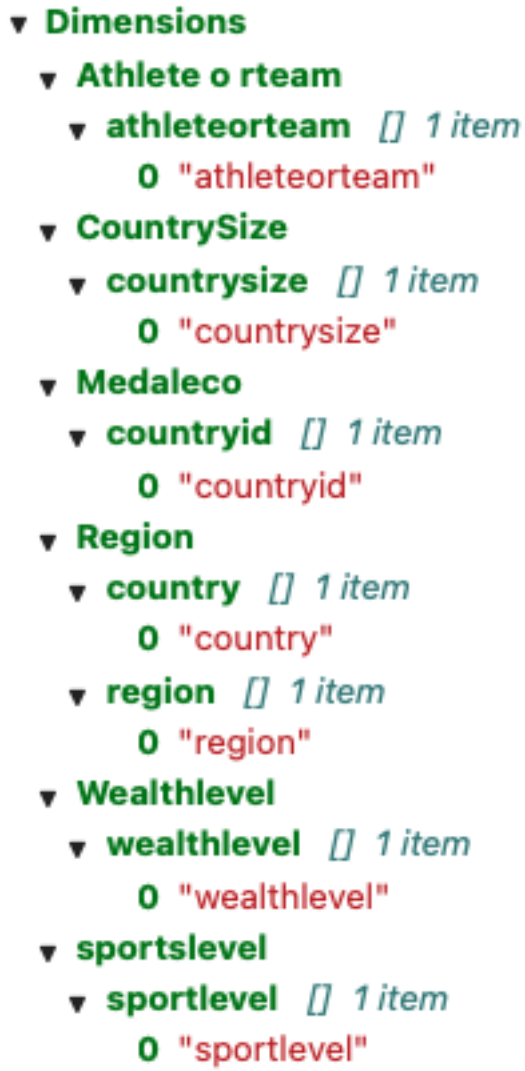

Figure 28: Dimension Hierarchy

Business Queries Virtualization

After finishing the data preprocessing mentioned above, I constructed OLAP operations, including slice, roll down, and roll up, to clean hierarchies and measures. Through these processes, I obtained cleaned and organized data, allowing me to effectively answer the queries posed by two clients.

Client 1

1. How many times have countries with huge population hosted in total?

In this query, I selected the dimensions of countrysize and the measure of the number of hold, with the following data:

countrysize	numberofhost.SUM
Large	35
Medium	18
Small	0

Figure 29: Cube Measure Client 1 Query 1

And the plot is as follows:

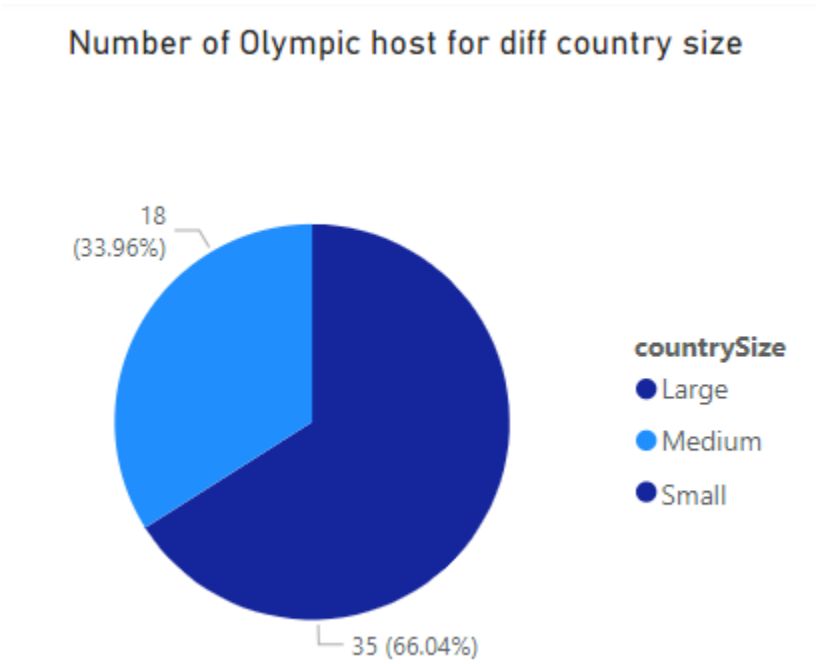

Figure 30: Client 1 Query 1

2. How many medals did the country with the top five highest GDP win in the 2021 Olympics?

In this query, I selected the dimensions of continent and the measure of the number of '2021 gode medel' and GDP per capita with the following data:

region	GDP(pc).SUM	2021 gode medel.SUM
Africa	112,350.34	11
Asia	488,640.15	126
Europe	1,529,554.80	134
North America	302,493.15	63
Oceania	154,009.00	29
South America	81,330.90	11

Figure 31: Cube Measure Client 1 Query 2

And the plot is as follows:

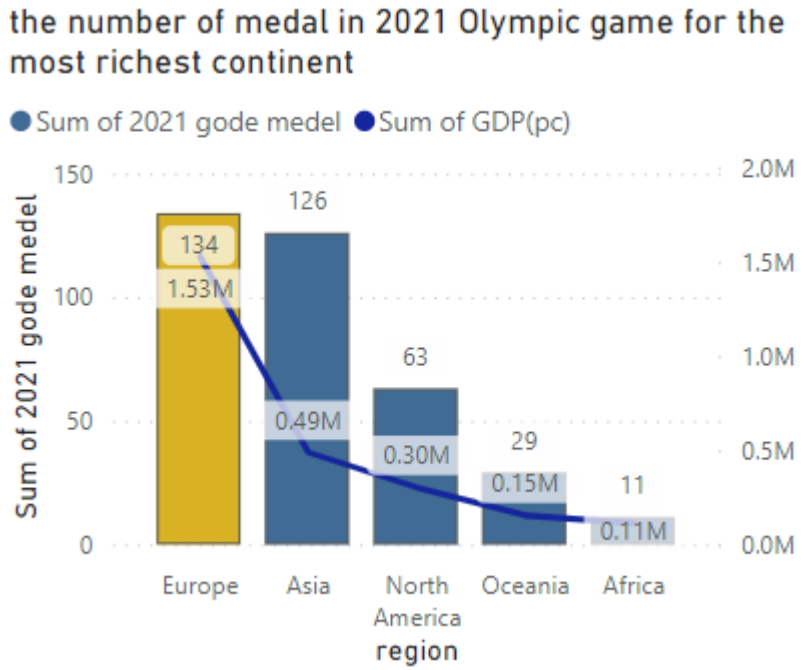

Figure 32: Client 1 Query 2

3. What is the GDP growth rate for the country won the most medals in Japan Olympic?

In this query, I selected the dimensions of continent and the measure of the number of '2021 gode medel' and GDP growth per capita with the following data:

region	country	GDPgrowth(pc).SUM	2021 gode medel.SUM
Africa	Algeria	-6.73	0
	Angola	-8.67	0
	Benin	.95	0
	Botswana	-10.40	0
	Burkina Faso	-.77	0
...	...	...	...
South America	Paraguay	-2.15	0
	Peru	-12.15	0
	Suriname	-16.91	0
	Uruguay	-6.28	0
	Venezuela	.00	1

Figure 33: Cube Measure Client 1 Query 3

And the plot is as follows:

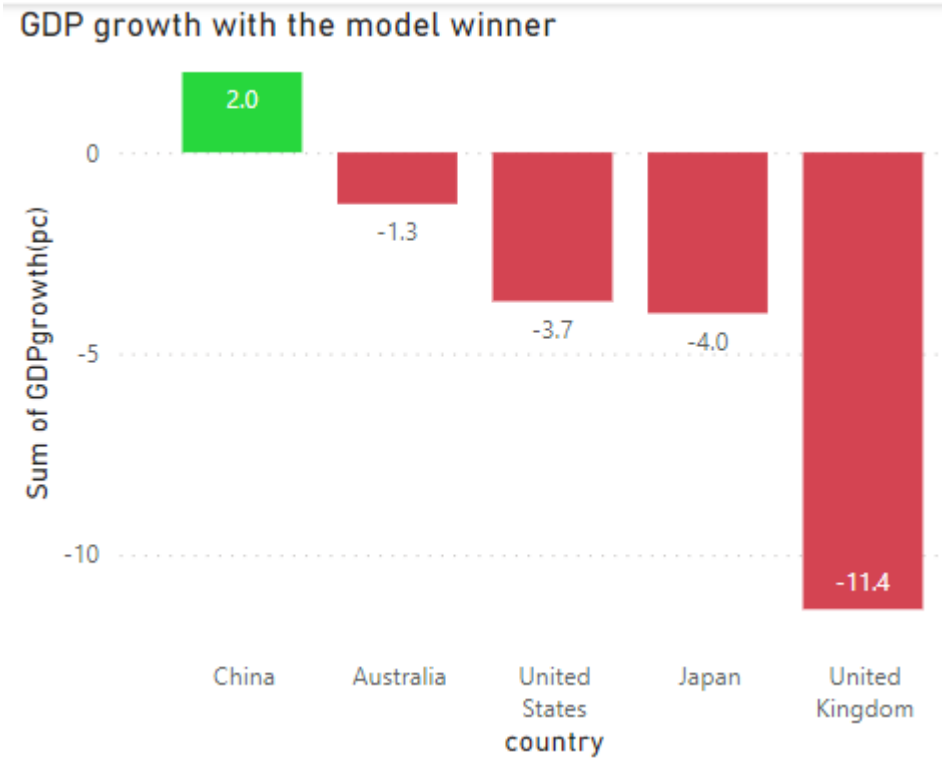

Figure 34: Client 1 Query 3

4. Do populous countries prefer individual or team sports?

In this query, I selected the dimensions of countrysize and the measure of Average Athlete Medals and Average Team Medals with the following data:

countrysize	Choiseing Athelete
Large	137.82
Medium	120.53
Small	38.46

Figure 35: Cube Measure Client 1 Query 4

And the plot is as follows:

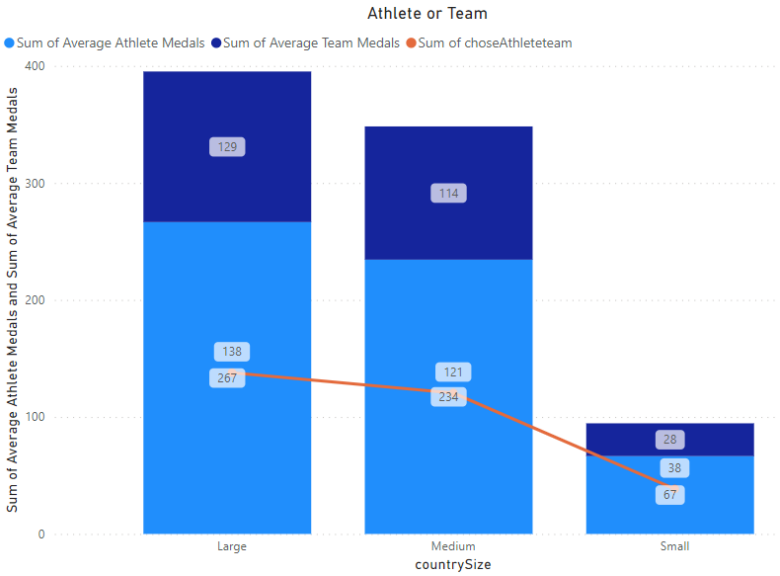

Figure 36: Client 1 Query 4

5. How many medals has the country that aces in individual sports won in their favorite sport?

In this query, I selected the dimensions of continent and sportlevel and the measure of medel for strongest sport with the following data:

region	country	sportlevel	medel for strongest sport.SUM
Africa	Algeria	Poor	9
	Angola	Poor	0
	Benin	Poor	0
	Botswana	Poor	2
	Burkina Faso	Poor	1
...	...	...	...
South America	Paraguay	Poor	1
	Peru	Poor	3
	Suriname	Poor	2
	Uruguay	Poor	6
	Venezuela	Poor	6

Figure 37: Cube Measure Client 1 Query 5

And the plot is as follows:

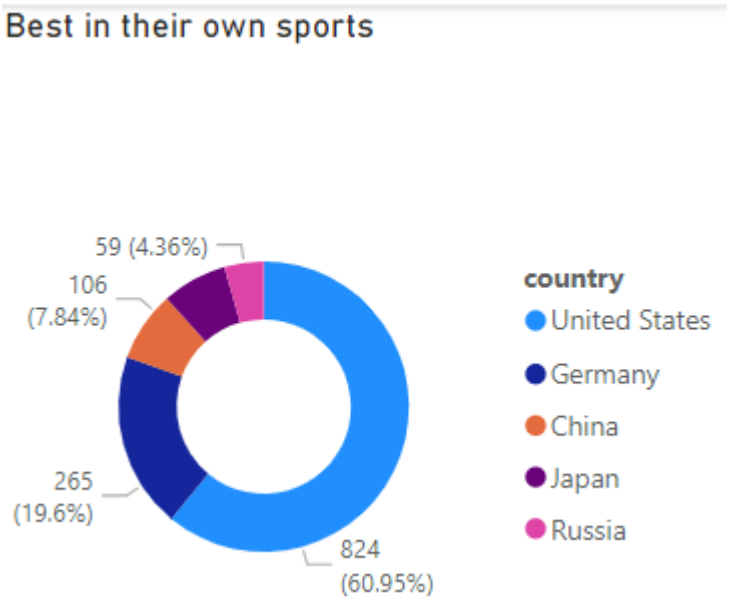

Figure 38: Client 1 Query 5

Client 2

1. Which country have lower yearly population growth rate after 2000 comparing to one before 2000?

In this query, I selected the dimensions of region and the measure of Average Annual Growth Rate (%) with the following data:

region	country	Average Annual Growth Rate (%).SUM
Africa	Algeria	1.83
	Angola	3.63
	Benin	3.00
	Botswana	1.96
	Burkina Faso	3.02
...	...	...
South America	Paraguay	1.60
	Peru	1.28
	Suriname	1.29
	Uruguay	.27
	Venezuela	.70

Figure 39: Cube Measure Client 2 Query 1

And the plot is as follows:

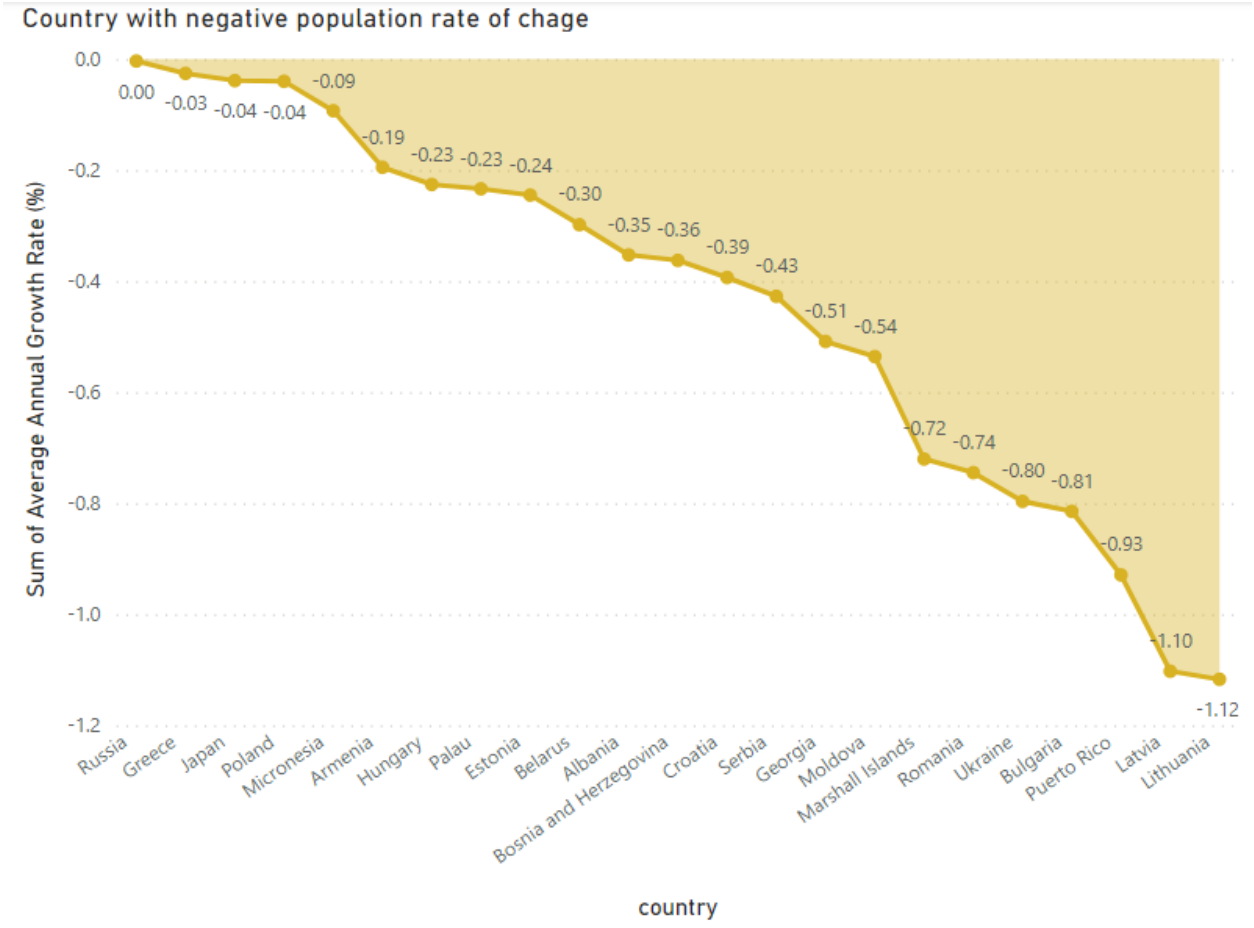

Figure 40: Client 2 Query 1

2. What is the different between the money the continent with the most strongest sports power country spend on health compared to the continent with the most average sports power country?

In this query, I selected the dimensions of region and sportlevel in GDP and the measure of health expenditure(GPT%) and GDP per capita with the following data:

		healthcost
region	sportlevel	
Africa	Poor	31,442,313.40
Asia	Poor	123,507,133.43
Europe	Medium	2,159,869.97
	Poor	509,242,202.41
North America	Poor	38,468,346.74
	Strong	1,195,343.76
Oceania	Poor	20,653,963.61
South America	Poor	7,686,333.88

Figure 41: Cube Measure Client 2 Query 2

And the plot is as follows:

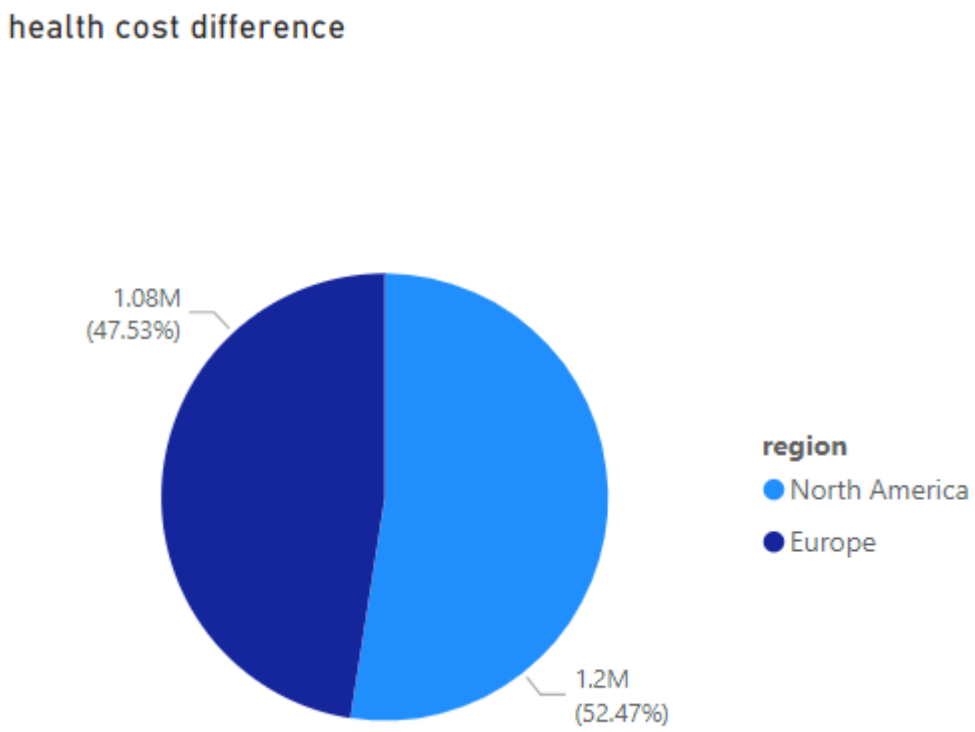

Figure 42: Client 2 Query 2

3. Do governments with middle health cost prefer to pay for their citizens' healthcare themselves, or do they let their citizens bear the costs?

In this query, I selected the dimensions of country and health cost in GDP and the measure of health cost difference of Change with the following data:

region	country	Healthcost Level	healthdifference.SUM	GDP(pc).SUM
Africa	Algeria	Low	54.17	3,354.16
	Angola	Low	-5.94	1,502.95
	Benin	Low	-4.13	1,237.95
	Botswana	Low	199.78	5,875.07
	Burkina Faso	Low	2.75	833.24
...	...	...	...	...
South America	Paraguay	Medium	35.18	5,353.35
	Peru	Low	140.06	6,063.63
	Suriname	Low	139.41	4,796.53
	Uruguay	Medium	622.08	15,650.50
	Venezuela	Low	-17.13	.00

Figure 43: Cube Measure Client 2 Query 3

And the plot is as follows:

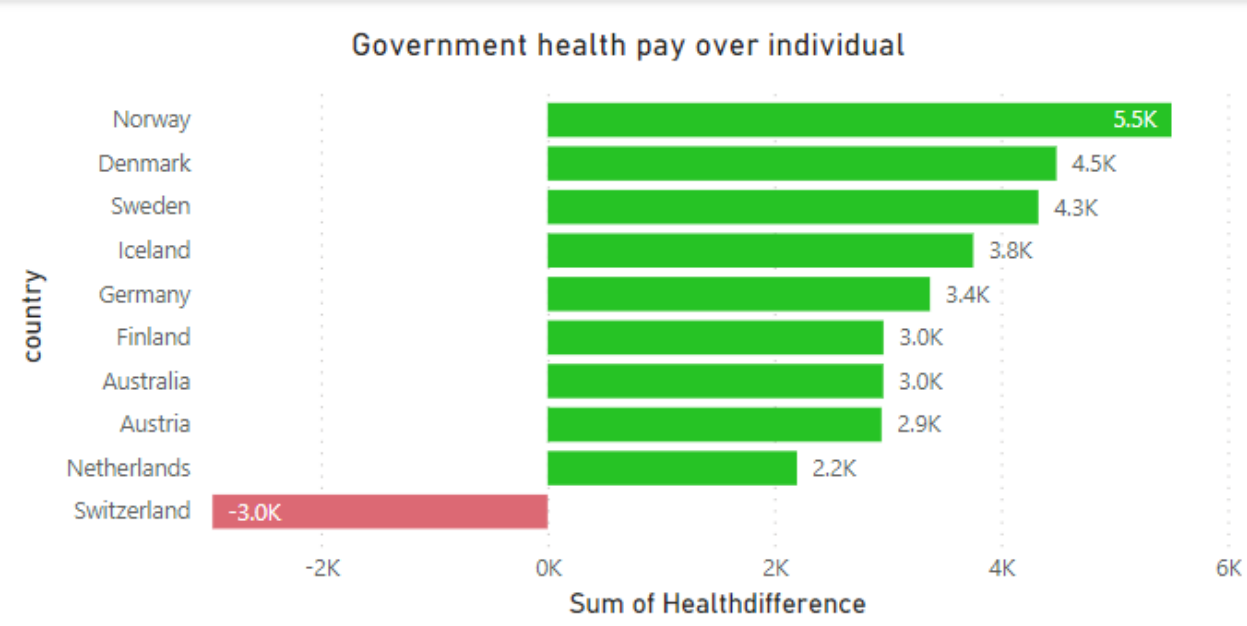

Figure 44: Client 2 Query 3

4. How does the GDP look like for middle countries with strong internet systems after hosting the Olympics?

In this query, I selected the dimensions of region, whether host before and Internet level and the measure of GDP per capita of Change with the following data:

region	country	hostbefore	Internet Level	GDP(pc).SUM
Africa	Algeria	none	Poor	3,354.16
	Angola	none	Poor	1,502.95
	Benin	none	Poor	1,237.95
	Botswana	none	Poor	5,875.07
	Burkina Faso	none	Poor	833.24
...	...	...	...	...
South America	Paraguay	none	Poor	5,353.35
	Peru	none	Poor	6,063.63
	Suriname	none	Poor	4,796.53
	Uruguay	none	Poor	15,650.50
	Venezuela	none	Poor	.00

Figure 45: Cube Measure Client 2 Query 4

And the plot is as follows:

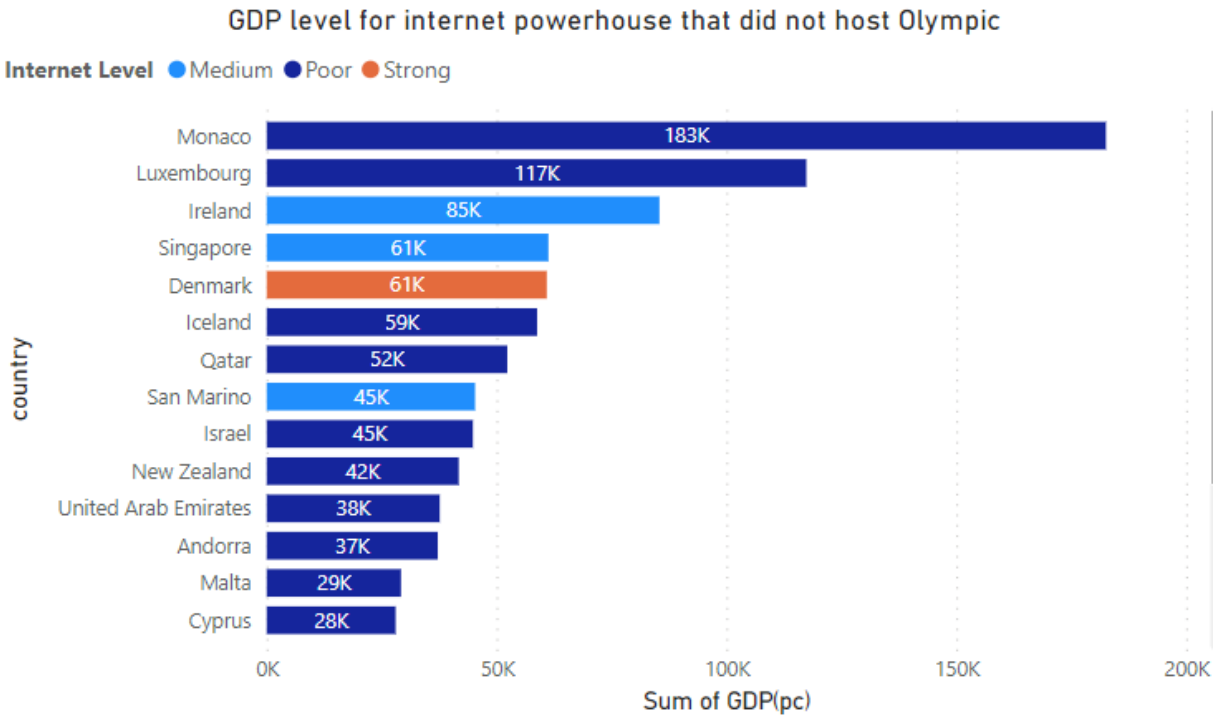

Figure 46: Client 2 Query 4

5. The change in the number depress disorder in all continents that not good at sports, specifically before and after the 2002 Olympics.

In this query, I selected the dimensions of region and sport level and the measure of depressive disorders rate of Change with the following data:

		depressive disorders rate of Change.SUM
region	sportlevel	
Africa	Poor	68.74
Asia	Poor	-1.90
Europe	Medium	6.99
	Poor	-78.54
North America	Poor	-2.60
	Strong	7.42
Oceania	Poor	9.15
South America	Poor	17.20

Figure 47: Cube Measure Client 2 Query 5

And the plot is as follows:

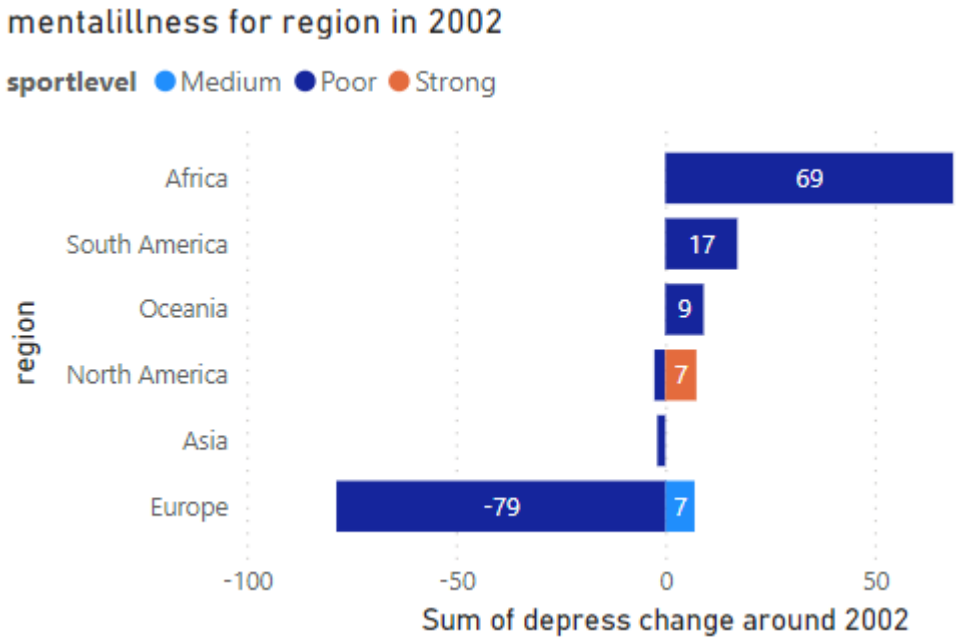

Figure 48: Client 2 Query 5

Association Rule Mining

These is a data analysis method in Data Warehouse is called association rule mining. The main purpose of ARM is to find significant correlations among variables in a large dataset. In order to figure out interesting relationships between features in the data, ARM will first identify the feature that happens with the most frequency and generating strong association rules from these features. After that it will figure out interesting connections between these features in the data. For instance, ARM will help the retail sector to identify the most frequently purchased product combinations by examining shopping baskets from the customer. In order to identify complex patterns and trends in the data, ARM is widely used not only in market basket research but also in fields like recommendation systems, bioinformatics, and so on.

During the analysis process of AMR, one of the most important algorithms is the Apriori algorithm. The Apriori algorithm, as a classic method in data mining for discovering association rules, is focused on recognizing the frequently occurring features and association rules between them in the dataset and figure out the important patterns and trends behind the data. The background lock behind this algorithm is based on one assumption, which is that a superset of a set with small frequent is also less frequent. Based on this assumption, the search process will be optimized and reduces the computational complexity, which allows the algorithm to handle the larger dataset more efficiently. The Apriori can not only identify the feature with high frequency but also create strong association rules based on these features by calculating support level, lift level, and confidence level.

```
# transfor code into integer type
onehotdf = onehotdf.astype(int)

# Calculate support for each items
supportnumber = onehotdf.sum() / len(onehotdf)

# create a list the rules
rules = []

factors = onehotdf.columns
for i in range(len(factors)):
    for j in range(i + 1, len(factors)):
        eventa, eventb = factors[i], factors[j]

        # Support calculations
        supta = supportnumber[eventa]
        suptb = supportnumber[eventb]
        suptab = (onehotdf[eventa] & onehotdf[eventb]).mean()

        # Confidence
        confiab = suptab / supta if supta > 0 else 0
        confiba = suptab / suptb if suptb > 0 else 0

        # Lift
        liftab = confiab / suptb if suptb > 0 else 0
        liftba = confiba / supta if supta > 0 else 0

        # Append the rules, supports, confidences, and lifts to the list
        rules.append([eventa, eventb, suptab, confiab, liftab])
        rules.append([eventb, eventa, suptab, confiba, liftba])

# Convert the list of rules to a DataFrame
rules = pd.DataFrame(rules, columns=['Antecedent', 'Consequent', 'Support', 'Confidence', 'Lift'])

# delete minimum support, confidence, and lift values
rules = rules[(rules['Support'] > 0.01) &
                            (rules['Confidence'] > 0.1) &
                            (rules['Lift'] > 1)]
```

Figure 49: Association Rule Mining Process

Based on the conclusions below, we can summary the association roles in the dataset from different perspectives, which can figure out some valuable information behind the data. The following is some conclusion based on support, confidence, and lift:

1. Based on the result of support value, we find out that the highest support value is for the rule including "participant_type_Athlete" and "event_gender_Men," with a support of 47.17%. This indicates that almost half of the records shows that athlete join the Olympic are male.

2. From the confidence, we can see that making "event_gender_Men" as the one of the features, the confidence is 73.48%. This indicates a significant correlation from male to other features, such as athletes.

3. Analyzing the lift, all rules have a lift above 1, with the highest is 1.053375. This suggests that there is a statistically significant positive correlation between the antecedent and consequent.

	Antecedent	Consequent	Support	Confidence	Lift
2951563	participant_type_Athlete	event_gender_Men	0.471707	0.676239	1.053375
2951562	event_gender_Men	participant_type_Athlete	0.471707	0.734776	1.053375
2952734	medal_type_BRONZE	participant_type_Athlete	0.246423	0.710162	1.018089
2952735	participant_type_Athlete	medal_type_BRONZE	0.246423	0.353272	1.018089
2951556	event_gender_Men	medal_type_BRONZE	0.224638	0.349917	1.008421
...	...	...	...	...	...
389552	slug_game_munich-1972	medal_type_BRONZE	0.010154	0.349206	1.006372
246174	discipline_title_Swimming	country_name_Australia	0.010154	0.124787	4.231051
246175	country_name_Australia	discipline_title_Swimming	0.010154	0.344288	4.231051
98886	discipline_title_Diving	event_gender_Women	0.010108	0.476087	1.631330
2953201	country_name_Norway	medal_type_GOLD	0.010016	0.368421	1.124730

Figure 50: Association Rule Mining result

Suggestions based on the result:

1. Based on the results above, we can clearly see that there exists a strong connection between male athletes and Olympic games. This relationship not only shows the dominant position of men in sports events but also highlights the potential issues of unequal opportunities for female athletes. To solve this problem, we can design programs that support and promote the development of female athletes. These measures could include providing training resources tailored for female athletes and increasing the number of women-specific competitions.

2. Consider the significant representation of male athletes in sports events, General Administration of Sport of China can analyze the data in detail and design a customized training programs for male athletes. This design will not only help in using resources more efficiently but also plays an important role in enhancing athletes' performance. These training programs can effectively enhance their skills and physical fitness by addressing specific needs and requirements.

3. Also, we need to include deeper analysis of other dimensions such as nationality and socioeconomic background, which is essential for truly understanding the full information of the Olympics. Through analysis from different perspectives, GASC can design more inclusive and equitable sports programs, to ensure that all Olympic athletes have the opportunity to show their talents.

What if analysis

It is essential to consider the COVID pandemic's influence on the world economy in 2020 when analysis which country had the greatest GDP. This influence had caused economic downturns in several countries, even leading to economic collapse in some countries. However, if we discount the pandemic's impact and presume that the world economy was steady and did not decline. Then, based on their level of economic strength, which country has won the most gold medals in 2021.

For answering this question, we modified and change through the database. First, we extracted some relevant data from the fact table of Client 1 with python. Then, we modified the GDP data for countries with declining economies using the following formula, GDP decided by 1 + negative growth rate. For those countries that did not have economic decline, their GDP data remained the same.

The image below shows the process of processing the data.

```
# modify the data
countrywithGDPgrowth['GDP(pc)'] = countrywithGDPgrowth.apply(
    lambda row: row['GDP(pc)'] / (1 - row['GDPgrowth(pc)']) if row['GDPgrowth(pc)'] < 0 else row['GDP(pc)'] * 1.02,
    axis=1)

countrywithGDPgrowth.to_csv('./Client1/FactOlympicwithoutCOVID.csv', index=False, encoding='utf-8-sig')
```

Figure 51: What if analysis process

Also the result are as follows

	countryID	medelID	grothID	CountrySizeID	GDP(pc)	wealthID	numberofhost	medel for strongest sport	2021 goude medel	GDPgrowth(pc)	Average Athlete Medals	Average Team Medals	AthleteteamID
0	0	3	2	1	32.049599	3	0	12	1	-6.726292	2.555556	1.000000	1
1	1	3	1	1	10616.893502	3	2	106	72	1.995558	30.619048	19.470588	1
2	2	1	2	1	2874.734812	2	8	824	40	-3.700953	45.038462	15.040000	1
3	3	3	2	1	258.091260	3	0	31	2	-2.885094	3.250000	2.625000	1
4	4	3	2	1	83.885979	3	0	8	0	-2.970295	1.000000	1.000000	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
229	229	3	3	3	0.000000	3	0	0	0	0.000000	0.000000	0.000000	0
230	230	3	3	3	0.000000	3	0	0	0	0.000000	0.000000	0.000000	0
231	231	3	3	3	0.000000	3	0	0	0	0.000000	0.000000	0.000000	0
232	232	3	3	3	0.000000	3	0	0	0	0.000000	0.000000	0.000000	0
233	233	3	3	3	0.000000	3	0	0	0	0.000000	0.000000	0.000000	0

Figure 52: What if analysis result

After completing the data cleaning, we analyzed the GDP and medals through data visualization. Below are the charts that resulted from our visualization of the data.

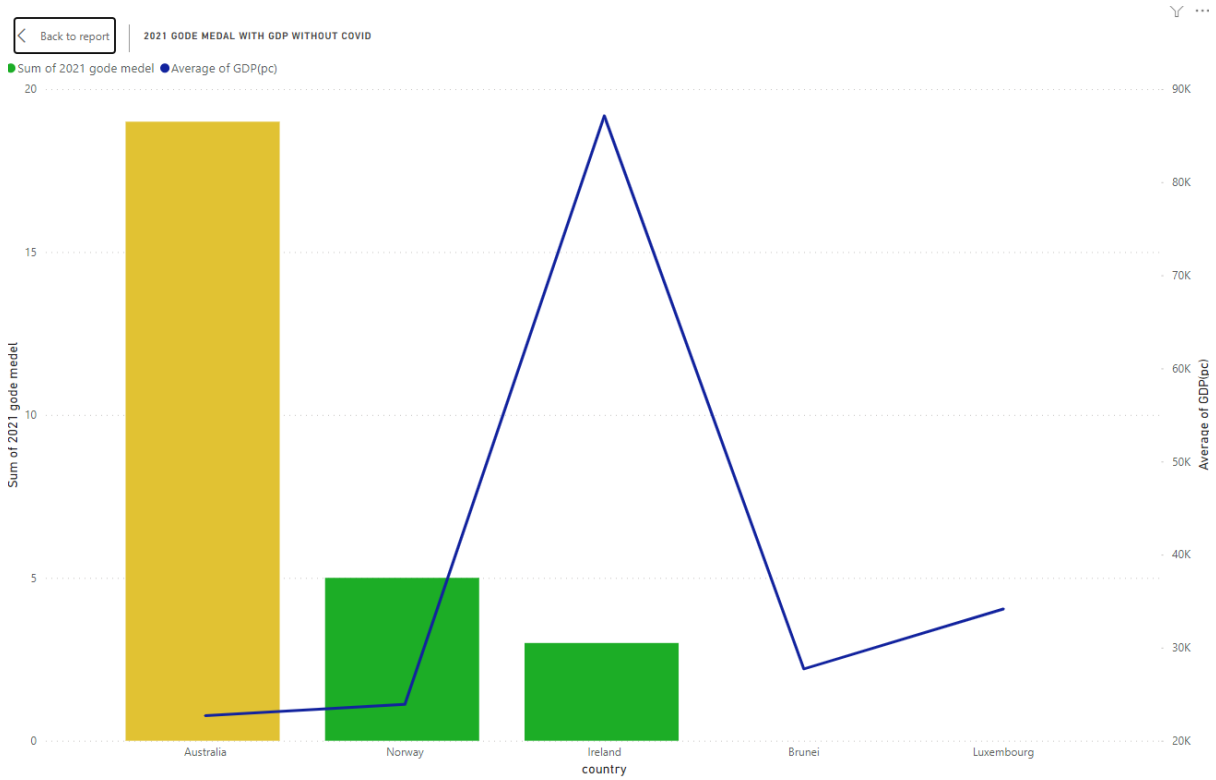

Figure 53: What if analysis plot

The data cube is an outdated technology?

For multidimensional data analysis, Data Cube, also referred to as cube for Online Analytical Processing, is a powerful and efficient tool. Based on its capacity to pre-calculate and store aggregated data, data cubes' primary goal is to improve the speed at querying and analyzing large data sets. This allows them to make fast responses to even complex queries containing multiple dimensions and hierarchies, which is critical for business environments that require immediate analysis of results to support decision-making. For example, Amazon, the world's leading e-commerce giant, uses data cubes to drill down into its sales data to capture customer behavior and market demand more accurately.[5]

In a wide range of complex data processing tasks, data cube has demonstrated their unparalleled value and effectiveness. Recently, due to their effective performance and adaptable data handling capabilities, innovative technologies such as columnar storage and in-memory computing [4] have garnered significant industry attention and have been widely utilized in various data analysis scenarios. Some are concerned about the development of data cubes and their data processing capability in future work. However, data cubes still possess irreplaceable advantages when dealing with certain types of data analysis tasks. Especially data cubes can provide more intuitive and efficient analysis capabilities when dealing with complex multidimensional analysis needs and hierarchical data structures.

Data cubes offer a powerful and user-friendly method for data organization and analysis in the field of multidimensional data analysis. Data cubes streamline and optimize the process of evaluating data from many angles to allow users to "slice and dice" [3] data along various dimensions. The ability to handle and analyze multidimensional data flexibly is crucial in different business and analytical scenarios. For example, comply like Amazon and Walmart may need to analyze the sales performance of different product categories across different regions and time periods [2].

Moreover, data cubes are also preferred due to their scalability and performance. Data cubes can manage huge and complicated datasets with fast query response times because they pre-aggregate data and store it in highly efficient forms. For businesses who need to process large volumes of data quickly and with analytical results, this functionality is essential[1].

Technologies like columnar storage and in-memory computing [4] do not always directly replace data cubes, even though they offer enticing benefits and could be appropriate alternatives in some situations. Data cubes continue to provide significant advantages, particularly in situations involving hierarchical data structures and complex multidimensional analysis.

In conclusion, data cubes continue to be useful and user-friendly tool for scenarios requiring multi-dimensional analysis, pre-aggregated data, and smooth connection with current BI tools and processes, despite the emergence of competing technologies that offer key functionalities.

Reference

[1]Chen, C. P., & Zhang, C. Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. Information sciences, 275, 314-347.

[2]Fu, L., & Hu, W. C. (2008). Online Analytical Processing and Data-Cube Technologies. In Handbook of Research on Public Information Technology (pp. 627-635). IGI Global.

[3] Kirkgoze, R., Katic, N., Stolba, M., & Tjoa, A. M. (1997, September). A security concept for OLAP. In Database and Expert Systems Applications. 8th International Conference, DEXA'97. Proceedings (pp. 619-626). IEEE.

[4] Schaffner, J., Eckart, B., Schwarz, C., Brunnert, J., Jacobs, D., Zeier, A. (2011). Towards Analytics-as-a-Service Using an In-Memory Column Database. In: Agrawal, D., Candan, K.S., Li, WS. (eds) New Frontiers in Information and Software as Services. Lecture Notes in Business Information Processing, vol 74. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-19294-4_11

[5] S. Chaudhuri, U. Dayal and V. Ganti, "Database technology for decision support systems," in Computer, vol. 34, no. 12, pp. 48-55, Dec. 2001, doi: 10.1109/2.970575.