Project Phase 2: Physical Design and Data Staging

CSI4142 - Fundamentals of Data Science

Winter 2023

uOttawa

Faculté de génie
Faculty of Engineering

School of Electrical Engineering and Computer Science
University of Ottawa

Professor Yazan Otoum

Group 10:
Mazharul Maaz - 300128179
Bill Battushig - 300109257
Xiao Meng Li - 300109886

Due date: Mar 24th, 2023

Table of Contents

A. Schematic of High-Level Data Staging Plan

1. Create and preprocess dimensions
    a. Date Dimension
        1. Generate date range
        2. Create a DataFrame with dates
        3. Convert dates to string format
    b. Country Dimension
        1. Load data, remove duplicate rows, check and convert data types and rename columns for consistency
        2. Remove non-alphanumeric values from the Country column for merging
    c. World Economic Indicator Dimension
        1. Load data, remove duplicate and Null rows, check and convert data types and rename columns for consistency
        2. Convert columns to appropriate data types
    d. Government Response Dimension
        1. Load data, remove duplicate rows, check and convert data types and rename columns for consistency
        2. Drop rows without valid vaccination info
        3. Keep only the latest vaccination info for each country
        4. Remove non-alphanumeric values from the Country column for merging
3. Create and preprocess fact tables
    a. Covid-19 Fact
        1. Load data, remove duplicate rows, check and convert data types and rename columns for consistency
    b. Emissions Fact
        1. Load data, remove duplicate rows, check and convert data types and rename columns for consistency
4. Merge dimensions and fact tables to create a final fact table
    a. Merge Dimensions and Facts into one table
5. Data normalization/scaling
    a. Scale World GDP to millions
    b. Scale Area from square miles to square kilometers
    c. Calculate emission changes
    d. Changing emissions so that changes are highlighted
6. Feature engineering
    a. Calculate the percentage of the population affected by Covid-19
    b. Ensuring the percentage caps at 100
7. Data transformation
    a. Normalize Emission Changes
    b. Normalize Covid-19 data
8. Final steps
    a. Make columns snake case
    b. Replace NaN values with "N/A"
    c. Generate a surrogate key

d. Reorder columns
e. Export the final dataset as a CSV file

B. Additional Details

1. We used Github to version control the source data sets.
2. All the columns that end with _n are normalized data columns.
3. We added texts in the Jupyter Notebook explaining the purpose of code blocks.

C. Data Quality Issues

1. Handling missing or noisy data
    - The fillna("N/A") function was used to replace NaN values in the final dataset with "N/A".
    - The data preprocessing includes dropping rows with missing values in columns such as total_deaths, new_deaths, new_cases, total_cases, average_CO2, average_CO4, and average_N2O.
2. Integrating data from different sources
    - The data for the project was gathered from multiple sources, such as CSV files containing country data, COVID-19 data, world economic indicator data, and environment data.
    - The data from these different sources were merged using the pd.merge() function to combine the relevant columns based on common keys like 'Country' and 'Year'.
3. Checking for duplicates and handling them
    - The duplicated().sum function was used to check for duplicate rows in the dataset, and the drop_duplicates() function was used to remove any duplicates found.
    - For the government response data, only the latest vaccination information for each country was retained using the drop_duplicates(subset="Country", keep="last", inplace=True) function.
4. Data cleaning and transformation
    - Data cleaning included removing non-alphanumeric characters from the 'Country' column using the str.replace() function in order to merge the columns later.
    - The 'Net migration' and 'Literacy (%)' columns were converted to float values by replacing commas with dots and casting the data type accordingly.
    - Data normalization and scaling were performed on several columns, such as emissions changes, COVID-19 cases, and vaccination information, using min-max normalization.
5. Feature engineering
    - New features were created, such as the percentage of the population affected by COVID-19, using the existing columns in the dataset.
6. Checking data types and converting them
    - The dtypes function was used to check the data types of each column, and the appropriate data types were assigned using the astype() function.

7. Renaming columns and reordering the dataset
    - Columns were renamed for clarity and to follow the snake_case naming convention.
    - The dataset columns were reordered, and a surrogate key column 'id' was added.

By implementing these steps, the data quality issues were effectively addressed, and the data from different sources were integrated into a single, clean dataset.

DBMS Screenshots

Uploading csv file to the dbms:

```
CREATE TABLE output_table (
    id                          int,
    Country_name               varchar(255),
    Total_cases                 int,
    New_cases                   int,
    Total_deaths                int,
    New_deaths                  int,
    Year                        int,
    Month                       int,
    Week                        int,
    Day                         int,
    Co2_change_n               float,
    Co4_change_n               float,
    N2o_change_n               float,
    Region                      varchar(255),
    Population                  int,
    Area_sq_km                 float,
    Net_migration_rate         float,
    Gdp_per_capita             float,
    Literacy_rate              float,
    Num_vaccinated_final       float,
    Num_fully_vaccinated_final float,
    Stringency_index           float,
    Unemployment_rate          float,
    World_gdp_millions         float,
    Covid_case_percent_n       float,
    Covid_death_percent_n      float,
    Vaccinated_final_percent_n float,
    Full_vaccinated_final_percent_n float
);

COPY output_table FROM 'C:\Users\Public\Output.csv' csv header;
```

Select all rows where country is Canada:

```
SELECT * FROM output_table o where o.country_name='Canada';
```

Data Output | Explain | Messages | Notifications

	id integer	country_name character varying (255)	total_cases integer	new_cases integer	total_deaths integer	new_deaths integer	year integer	month integer	week integer	day integer	co2_change_n double precision	co4_change_n double precision	n2o_change_n double precision	region character vary
1	14327	Canada	120	22	1	1	2020	3	10	8	0.445913462	0.084728033	0.04467354	NORTHERN
2	14328	Canada	144	24	2	1	2020	3	11	9	0.445913462	0.084728033	0.04467354	NORTHERN
3	14329	Canada	165	21	2	0	2020	3	11	10	0.445913462	0.084728033	0.04467354	NORTHERN
4	14330	Canada	228	63	2	0	2020	3	11	11	0.445913462	0.084728033	0.04467354	NORTHERN
5	14331	Canada	289	61	3	1	2020	3	11	12	0.445913462	0.084728033	0.04467354	NORTHERN
6	14332	Canada	386	97	5	2	2020	3	11	13	0.445913462	0.084728033	0.04467354	NORTHERN
7	14333	Canada	453	67	5	0	2020	3	11	14	0.445913462	0.084728033	0.04467354	NORTHERN
8	14334	Canada	549	96	5	0	2020	3	11	15	0.445913462	0.084728033	0.04467354	NORTHERN
9	14335	Canada	700	151	9	4	2020	3	12	16	0.445913462	0.084728033	0.04467354	NORTHERN
10	14336	Canada	854	154	12	3	2020	3	12	17	0.445913462	0.084728033	0.04467354	NORTHERN

Select the row where the highest percentage of the population contracted Covid-19:

```
1  SELECT * FROM output_table o where o.covid_case_percent_n=1;
```

Data Output | Explain | Messages | Notifications

	id integer	country_name character varying (255)	total_cases integer	new_cases integer	total_deaths integer	new_deaths integer	year integer	month integer	week integer	day integer	co2_change_n double precision	co4_change_n double precision	n2o_change_n double precision	region character varyin
1	7908	Bahrain	525171	1517	1458	0	2022	3	9	5	1	0.982217573	1	NEAR EAST