Report Laboratory of Data Science

Francesco Lorenzo Carone, Vincenzo Rocchi

January 4, 2024

1 Project Assignment - Part 1

In the initial phase of our project, our objective was to construct and populate a database using a variety of source files. The primary data source was *Police.csv*, complemented by *dates.xml* for mapping each `date_id`, along with three distinct *.json* files correlating participant age, type, and status to numerical identifiers.

1.1 Assignment 0: Database Design and Implementation

Methodology: Assignment 0 was executed using SQL Server Management Studio, adhering to the schema outlined in the project guidelines. Our team's primary contribution involved deciding which attributes to incorporate into the tables. The following list delineates the attributes added:

- **Geography**:
  - geo_id (PK),
  - latitude DECIMAL,
  - longitude DECIMAL,
  - city NVARCHAR,
  - state NVARCHAR,
  - continent NVARCHAR;
- **Gun**:
  - gun_id (PK),
  - is_stolen BIT,
  - gun_type NVARCHAR;
- **Date**:
  - date_id (PK),
  - date DATE,
  - day INT,

  - month INT,
  - year INT,
  - quarter INT,
  - day_of_week NVARCHAR;

- **Incident**:
  - Incident_id (PK);

- **Participant**:
  - participant_id (PK),
  - age_group INT,
  - gender NVARCHAR,
  - status INT,
  - type INT;

The design of the *custody* table followed the specifications provided in the project documentation. Additionally, we implemented Foreign Key constraints to establish relational links between the IDs of the *Custody* table and the corresponding IDs in other tables.

1.2 Assignment 1: Data Segmentation and Table Population

Methodology: Assignment 1 presented several challenges requiring critical decision-making and innovative solutions. Our primary task was to segment the comprehensive *Police.csv* file into six distinct CSVs, corresponding to each table in our database. The following steps outline our approach for each table:

- **Date Table:** We extracted each `date_pk` present in *Police.csv* and mapped the actual dates from *dates.xml*. Additionally, we computed the corresponding quarter for each date.

- **Geography Table:** Utilizing the `reverse_geocoder` library, we converted a list of coordinates into associated cities. We faced a challenge with coordinates near the US/Mexico border, where some US coordinates were incorrectly mapped to Mexican cities. To resolve this, we replaced the library's default CSV with a US-specific one. Each city was inserted uniquely, leading to complexities in creating the Custody CSV, discussed later.

- **Gun Table:** We created a unique row and ID for each distinct combination of `(gun_stolen_bit, gun_type)`.

- **Incident Table:** This involved adding each distinct `IncidentID` from *Police.csv*.

- **Participant Table:** We processed data from *Police.csv* using dictionaries provided in the JSON files. A unique `participant_id` was generated for each distinct combination of participant attributes.

- **Custody Table:** The `crime_gravity` was calculated as per the formula in the project guidelines. We leveraged the CSVs created earlier to link various gun and participant IDs. The participant and gun IDs were matched by searching for their respective attribute combinations in the corresponding CSVs. However, a challenge arose with the Geography CSV; instead of having an ID for each (latitude, longitude), we only had one for each (city, state, continent). We devised a function to match each coordinate to the nearest one in the Geography CSV and assigned the corresponding `geo_id` in our Custody CSV.

This assignment required meticulous data handling and problem-solving skills, particularly in the accurate segregation of the *Police.csv* and the effective use of external libraries and custom functions to overcome data mapping challenges.

1.3 Assignment 2: Efficient Data Importing Techniques

Methodology: In Assignment 2, our focus was on developing efficient methods for populating our database with the segmented data. We achieved this by creating two specialized functions:

1. **CSV-to-Table Population Function:** This function is designed to automate and streamline the process of transferring data from a CSV file into the corresponding table in our database. It starts by determining the number of columns present in each table's CSV file. The function then efficiently performs batch INSERT operations, transferring the data from each row of the CSV into the database. This approach not only ensures accuracy in data transfer but also significantly improves the efficiency of the database population process.

2. **Batch Execution Function:** The second function we developed employs the `cursor.exceutemany` method. By passing a SQL query and the corresponding data, this function enables us to populate the database at a much faster rate compared to using a standard execute method. This optimization is particularly beneficial for handling large datasets, as it minimizes the time required for database population while maintaining data integrity.

These functions played a crucial role in effectively managing the bulk data import. Their implementation not only facilitated a smoother and faster data transfer but also minimized potential errors, thereby ensuring a reliable and robust database setup for our project.

2 Project Assignment - Part 2

2.1 Assignment 0 - Analysis of Custody Data Using SSIS

Introduction: This segment of the project focused on computing the annual total of custodies using SQL Server Integration Services (SSIS). The aim was to showcase our proficiency in data manipulation and analysis, particularly in aggregating custody data on an annual basis.

Methodology:

- **Initiation of the SSIS Project:** We began by starting a new SSIS project in Microsoft Visual Studio, chosen for its extensive data integration capabilities and SQL Server compatibility.
- **Data Flow Task Creation:** A Data Flow Task was added in the Control Flow. This step was essential for defining our data extraction, transformation, and loading (ETL) workflow.
- **Data Extraction:** The project involved connecting to our existing group database to fetch the required data, building upon our earlier database interactions.
- **Data Joining with Lookup:** We used a Lookup transformation to merge data from the 'date' and 'custody' tables, based on the 'date_id'. This allowed us to extract 'custody_id' and 'year' from the respective tables.
- **Aggregating Data:** The data was then aggregated using an Aggregate Transformation, counting the instances of 'custody_id' and grouping by 'year'.
- **Data Storage:** The aggregated data was temporarily stored in a flat file, using a Flat File Destination for its simplicity and ease of access.
- **Results:** This process successfully generated a clear and accessible dataset showing the annual counts of custodies, essential for analyzing trends over time.

2.2 Assignment 1 - States with Youth Criminal Problem

Introduction The objective of this analysis is to identify states with a youth criminal problem, defined as having the age group under 18 with the highest overall crime gravity.

Methodology :

1. **Data Extraction:** Extracted relevant data from the custody table, considering rows with crime gravity greater than 1.

2. **Data Joining:** Joined the extracted data with the participant table based on participant_id for obtaining the age group of each custody record.

3. **Data Splitting:** Divided the dataset into two groups: under 18 and over 18, using a conditional split.

4. **Overall Crime Gravity Calculation:** Calculated the highest overall crime gravity for each state in both under 18 and over 18 age groups.

5. **Boolean Column Derivation:** Derived a boolean column indicating whether the highest overall crime gravity for the under 18 age group is greater than that for the over 18 age group.

Issue with Crime Gravity Calculation: An issue arose during the calculation of the highest overall crime gravity, particularly related to the age groups in the dataset. The Crime Gravity calculation incorporated information from the `participant_age.json` dictionary, which was structured as follows: {"Adult 18+": 1, "Teen 12-17": 3, "Child 0-11": 6}.

It became apparent that utilizing this structure introduced an inherent bias into the analysis. The bias stemmed from the assigned crime gravity values, where the age groups "Teen 12-17" and "Child 0-11" had significantly higher gravity values than the "Adult 18+" group.

Looking at the results, this predisposition is evident. The data suggests that, due to the disproportionate weight given to younger age groups in the Crime Gravity calculation, almost every state appears to have a youth criminal problem.

2.3 Assignment 2 - Ratio Calculation Results

Introduction: The objective was to compute the ratio between the total gravity of crimes with a stolen gun and the total gravity of crimes with a not-stolen gun for each incident. Due to challenges in calculating a ratio between two totals for each incident, we adjusted our approach and instead computed the ratio between the total crime gravity of incidents involving stolen guns and not-stolen guns.

Methodology:

1. **Data Extraction:** Relevant data was extracted from the custody table, focusing on rows with crime gravity greater than 1.

2. **Data Joining:** The extracted data was joined with the gun table using the gun_id as the key.

3. **Column Derivation:** Two columns were derived to represent the crime gravity of incidents with stolen guns and not-stolen guns.

4. **Total Calculation:** The total crime gravity was calculated for incidents with stolen and not-stolen guns using aggregation.

5. **Ratio Computation:** A new column was added to calculate the ratio between the total crime gravity of incidents with stolen guns and not-stolen guns.

3 Project Assignment - Part 3

3.1 Assignment 0 - Building a Data Cube Using SSAS

Introduction: The process of constructing a data cube from the tables in our database involved several critical steps, ensuring the creation of appropriate hierarchies and measures to facilitate insightful analytics.

Methodology :

1. **Initialization:** The initial phase of the project commenced with the setup of a new SSAS project in Visual Studio. This step was crucial for establishing the foundational elements of our data cube. A data source was added to form a connection with the university server, and the deployment configuration was meticulously set up to align with project requirements.

2. **Data Views and dimensions:** Subsequently, our focus shifted to creating views and dimensions within the database. This stage was pivotal in laying down the structural framework of the cube. Two key dimensions that required immediate attention were geography and date. For the date dimension, we introduced several named calculations such as Month Name, Day Name, Week, and Quarter. These calculations were instrumental in retrieving specific date-related details, which were further utilized to organize the data in a non-alphabetical, but chronologically relevant manner.

3. **Hierarchy construction:** The construction of hierarchies within the date dimension was a nuanced process. We developed four distinct hierarchies to cater to different granularities: yearly, weekly, quarterly, and daily. To ensure accuracy and consistency in relationships, we employed composite keys and rigid relationships. This approach was vital in maintaining the integrity of the data across different levels of granularity.

4. **Geography dimension:** In addressing the geography dimension, a state hierarchy was established. This hierarchy allowed for the categorization of cities within each U.S. state, providing a geographical context to our data. A continent hierarchy was deemed unnecessary as our data was confined to a single continent.

5. **Other dimensions and measures:** Further advancements in the project involved the addition of other dimensions such as Incident, Participant, and Gun, specifically for Gun we created the GunType hierarchy and for the Incident we added the IncidentId hierarchy. These dimensions were

complemented by the introduction of specific measures from the custody fact table. These measures, namely the count of custodies and the sum of crime gravity, were essential for our analytical computations.

6. **Conclusion:** The final phase of the assignment involved deploying the cube on the analysis server. This step was critical in bringing the theoretical aspects of the cube into a practical, usable state. Post-deployment, rigorous testing was conducted to ensure the functionality and accuracy of the cube. The successful deployment and testing marked the completion of a meticulously crafted data cube, poised to support our analytical needs with reliability and precision.

3.2 Assignment 1 - Percentage Increase or Decrease in Total Crime Gravity

Introduction: In this analysis, we aimed to quantify the percentage increase or decrease in total crime gravity concerning the previous year for each state. The following steps were taken to achieve this:

Methodology:

- **MDX Query Definition:** We defined a set of MDX queries to calculate various metrics related to crime gravity, specifically focusing on the percentage change from the previous year.
- **Member Definitions:** Utilizing MDX, we created three key members:
  - `YearCrimeGravity`: Represents the total crime gravity for the current year.
  - `YearCrimeGravityPrev`: Represents the total crime gravity for the previous year.
  - `DiffCrimeGravity`: Represents the difference in crime gravity between the current and previous years.
  - `DiffPercCrimeGravity`: Represents the percentage change in crime gravity, calculated as the difference divided by the crime gravity of the previous year.
- **MDX Query Execution:** The MDX query was executed on the specified database (`DB_183_DS_NEW`) to retrieve the required data.
- **Results Presentation:** The results are presented in a tabular format, with each row corresponding to a specific year and each column providing information on the percentage change, absolute difference, total crime gravity for the current year, and total crime gravity for the previous year.

3.3 Assignment 2 - Total Crime Gravity Percentage for Each Gun

Introduction: The goal of this analysis was to showcase the total crime gravity for each gun in percentage terms relative to the total crime gravity of all guns. The following steps outline the process and the MDX query used:

Methodology:

- **MDX Query Definition:** We defined an MDX query to calculate the total crime gravity for each gun type and the corresponding percentage relative to the total crime gravity of all guns.
- **Member Definitions:** Two key members were created in the MDX query:
  - `TotCrimeGravity`: Represents the total crime gravity for all gun types.
  - `PercCrimeGravity`: Represents the percentage of crime gravity for each gun type, calculated as the crime gravity of the specific gun type divided by the total crime gravity of all guns.
- **MDX Query Execution:** The MDX query was executed on the specified database (`DB_183_DS_NEW`) to retrieve the necessary data.
- **Results Presentation:** The results are presented in a tabular format, with each row corresponding to a specific gun type. Columns provide information on the absolute crime gravity and the percentage of crime gravity for each gun type.

3.4 Assignment 3 - Incidents with Total Gravity Score Above State Average

Introduction: In this analysis, the objective was to identify incidents where the total gravity score was greater or equal to the average gravity score for each state. We tried again to understand and compute what was requested, but we failed. Every time, as is evident from the code, we calculated the average and the total crime gravity, respectively, for the state and the incident. Then, we filtered the results, picking only the IDs of the incidents where the total crime gravity was greater than the average. In the select part of the MDX code, we had to put the filter on rows and the total gravity score of the incident and the average of the state on columns. However, the results were set on the incident ID only. Having said that, this is what we came up with.

Methodology:

- **MDX Query Definition:** We defined an MDX query to calculate the average gravity score for each state and filter incidents where the total gravity score was greater or equal to the state's average.

- **Member Definition:** A new member, `AverageGravityScore`, was created to represent the average gravity score for each state.
- **SET Definition:** A set, `FilteredIncidents`, was established using the `Filter` function, which identifies incidents meeting the condition of having a total gravity score greater or equal to the average gravity score.
- **MDX Query Execution:** The MDX query was executed on the specified database (`DB_183_DS_NEW`) to retrieve the necessary data.
- **Results Presentation:** The results are presented in tabular format, showcasing the total gravity score and the average gravity score for incidents.

3.5 Assignment 4 - Development of a Geographical Dashboard in Power BI for Analyzing Crime Gravity by Age Group

Introduction: Visual analytics play a crucial role in decision support systems, particularly in understanding complex datasets. This assignment documents the process of designing an interactive dashboard in Power BI to analyze the geographical spread of crime gravity in different age groups.

Methodology :

1. **Initialization:** The dashboard was developed following these key steps: Initiating a new Power BI session and loading data into memory, bypassing the limitations of direct query mode and optimizing data handling for efficiency and standalone functionality.
2. **Dashboard Development:** Data was imported from the constructed data cube, focusing on geographic distribution and age-related crime statistics.
3. **Visualization Components:** The dashboard comprises several interactive elements:

   Geographic Map with Pie Charts: This feature displays the total crime gravity in each state, segmented into pie charts representing different age groups. It provides an at-a-glance view of how crime gravity is distributed geographically and demographically.

   Age Group Column Chart: A side column chart offers a clearer representation of crime gravity divided solely by age group. This aids in comparing the impact of crime across ages without the geographic element.

   State Selection List: An interactive list allows users to select individual or multiple states. This feature is useful for focusing on specific areas and comparing the total crime gravity among them.

4. **Implementation and Performance:** The decision to use in-memory data loading over direct query was driven by performance efficiency and the need for a standalone operational capability. This approach ensures faster data processing and the flexibility to move the entire dataset as a cohesive unit for cloud-based project saving and sharing.

5. **Conclusion:** The dashboard effectively integrates and visualizes critical data, offering insightful views into the geographical and age-based distribution of crime gravity. Its interactive features enhance the decision-making process by allowing users to explore data from multiple perspectives.

3.6 Assignment 5 - Interactive Visualization of Crime Trends Using Date Hierarchies and Demographic Filters in Power BI

Introduction: In Decision Support Systems, the ability to visualize data effectively can unearth valuable insights. This assignment focuses on the creation of a Power BI dashboard that uses various visualization techniques to explore crime gravity in relation to time and demographic factors.

Methodology :

1. **Dashboard development:** Utilizing the date hierarchy created during the data cube process. Integrating demographic filters to enhance data analysis. Selecting visualization types that best represent the data complexities.

2. **Dashboard Components:** Data was imported from the constructed data cube, focusing on crime gravity, time and demographic statistics.

3. **Ribbon Charts for Time and Demographic Analysis:** The dashboard comprises several interactive elements, two ribbon charts were created:

    - **Day and Month Analysis:** The first ribbon chart displays the splits of crime gravity across days of the week and months of the year, integrated with a gender filter. This allows for an analysis of how crime gravity varies over time and between genders.
    - **Custody Count by Age and Gun Type:**The second ribbon chart illustrates the count of custodies, split by age group and type of gun. This visualization helps in understanding potential correlations between age, gun types, and crime incidences.

4. **Pie Chart for Age Group Analysis:** A pie chart was included to present a clear view of the distribution of crime gravity across different age groups. This visualization complements the ribbon charts by offering a straightforward representation of age-related data.

5. **Analysis and Insights:** The dashboard facilitates multi-dimensional analysis, revealing patterns and trends in crime data related to time, gender, age, and gun type. For instance, the ribbon charts can highlight specific days or months with higher crime gravity, while the pie chart provides a demographic breakdown of the involved age groups.

6. **Conclusion:** This Power BI dashboard serves as a powerful tool for analyzing complex crime data. Its interactive and diverse visual components allow users to delve into various aspects of the data, aiding in informed decision-making processes.