BI APPLICATION

GROUP 3

DATASET: Olympics Games Data

Team members:
Akash Kumar 23030141003
Romaan Halai 23030141025
Swapnil Kothari 23030141034
Afrin Mahasaniya 23030141038
Rhea Shree S 23030141050
Swarn Prabha 23030141069

TABLE OF CONTENTS

INTRODUCTION

The Olympic Games are the world's foremost sports competition, featuring thousands of athletes from around the globe. This project focuses on analyzing historical data from various Olympic events, athletes, and participating countries. The primary objective is to utilize MongoDB's advanced features to store, manage, and analyze this vast dataset, providing valuable insights and visualizations that can be further explored in Tableau also utilization of a few machine learning algorithms on the dataset for insights generation.

SCOPE OF THE PROJECT

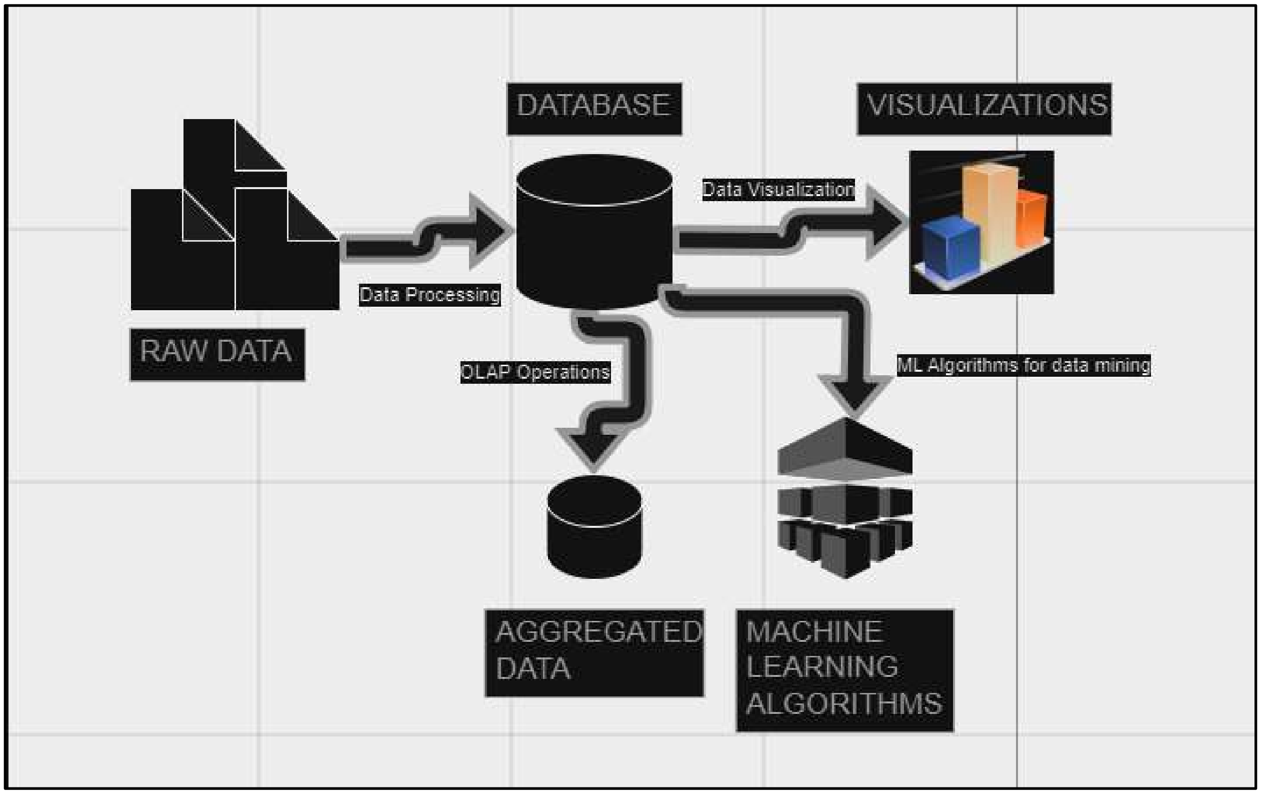
The Olympic Games represent the pinnacle of international sports competitions, showcasing thousands of athletes from across the world. This project aims to:

- Analyze historical data related to Olympic events, athletes, and participating nations.
- The goal is to leverage MongoDB's advanced capabilities for storing, managing, and analyzing this extensive dataset.
- By doing so, the project will provide meaningful insights and visualizations, which can be further explored using Tableau.
- Additionally, the project will incorporate several machine learning algorithms to generate deeper insights from the dataset.

OBJECTIVES

- **Efficient Data Storage:** Implement a robust database structure in MongoDB to store Olympic data, including athlete details, event results, and country participation.
- **Advanced Data Analysis:** Utilize MongoDB's querying capabilities to extract meaningful insights from the data also to use different aggregation stages to perform slicing, dicing and other operations on database.
- **Data Visualization:** Connect MongoDB with Tableau to create dynamic and interactive dashboards.
- **Implementation of Unique MongoDB Features:** Explore and implement features such as full-text search.
- **Machine Learning:** Use of different machine learning algorithm to get insights using the processed data.

ARCHITECTURE OF THE SYSTEM



- <u>**Data Collection**</u>
  - **Source:** Data is initially sourced from the Kaggle website, consisting of six CSV files.
  - **Contents:** These files encompass a range of historical Olympic data, including details on events, athletes, and participating countries.

- <u>**Data Processing**</u>
  - **Tool:** The raw data is imported into Excel.
  - **Tasks:** In Excel, the data is cleaned to remove any inconsistencies or errors and is transformed to meet the requirements of subsequent analysis. This includes handling missing values, correcting data formats, and standardizing entries.

- <u>**Data Storage**</u>
  - **Database:** The cleaned data is then stored in a MongoDB database.

- **Operations:** MongoDB's advanced features are utilized for performing aggregation operations, which include summarizing, grouping, and filtering the data to extract meaningful patterns and insights.

- **<u>Data Visualization</u>**

  - **Tool:** Tableau is used to create visualizations from the processed data.
  - **Purpose:** These visualizations include charts, graphs, and dashboards that represent various aspects of the Olympic data, making it easier to analyze trends and patterns.

- **<u>Machine Learning</u>**

  - **Algorithms:** Machine learning techniques are applied to the dataset.
  - **Predictive Analysis:** Algorithms are used to forecast medal counts based on historical data.
  - **Clustering:** Athletes are clustered based on attributes such as height, weight, and other relevant factors to identify patterns and groupings within the data.

DATASET OVERVIEW

The Olympic project consists of six primary datasets, each containing detailed information about various aspects of the Olympic Games.

1. **Athlete:** This dataset provides detailed information about the athletes.

Fields:

- code: A unique code for the athlete.
- name_short: The athlete's short name.
- gender: The gender of the athlete.
- function: The role or function of the athlete (e.g., competitor, coach).
- country_code: The code representing the athlete's country.
- country: The full name of the country.
- height: The height of the athlete.
- weight: The weight of the athlete.
- disciplines: The disciplines the athlete participates in.
- events: The events the athlete is involved in.
- birth_date: The birth date of the athlete.
- occupation: The athlete's occupation.
- coach: The name of the athlete's coach.
- influence: Key influences in the athlete's career.
- ritual: Pre-event rituals of the athlete.

2. **Athlete_bio:** This dataset contains biographical data for athletes.

Fields:

- athlete_id: The unique identifier for the athlete.
- name: The full name of the athlete.
- sex: The gender of the athlete.
- born: The birth date of the athlete.
- height: The height of the athlete.
- weight: The weight of the athlete.
- country: The country the athlete represents.
- description: A description of the athlete's career and achievements.
- special_notes: Additional notes or highlights about the athlete.

3. **Athelete_event_results:** This dataset records the performance of athletes in different Olympic events.

Fields:

- edition: The specific edition of the Olympics.
- edition_id: A unique identifier for the Olympic edition.

- country_noc: National Olympic Committee (NOC) code representing the country.
- sport: The sport in which the event took place.
- event: The specific event in the sport.
- result_id: A unique identifier for the event result.
- athlete: Name of the athlete.
- athlete_id: A unique identifier for the athlete.
- pos: Position achieved by the athlete in the event.
- medal: Type of medal won (if any).
- isTeamSport: Indicates whether the event was a team sport.

4. **Games:** This dataset provides information about the different Olympic Games editions.

Fields:

- edition.2: The edition of the Olympics (secondary identifier).
- year: The year in which the Olympics took place.
- city: The host city of the Olympics.
- country_noc: The NOC code of the host country.
- start_date: The starting date of the Olympic Games.
- end_date: The ending date of the Olympic Games.
- competition_date: The dates on which the competitions were held.
- edition_id: A unique identifier for the Olympic edition.

5. **Games_medal_tally:** This dataset summarizes the medal tally for different editions of the Olympics.

Fields:

- edition: The specific edition of the Olympics.
- edition_id: A unique identifier for the Olympic edition.
- year: The year of the Olympics.
- country: The country whose medal tally is recorded.
- gold: The number of gold medals won.
- silver: The number of silver medals won.
- bronze: The number of bronze medals won.
- total: The total number of medals won.

6. **Olympic_results:** This dataset records detailed results for various Olympic events.

Fields:

- result_id: A unique identifier for the event result.
- event_title: The title of the event.
- edition: The specific edition of the Olympics.
- edition_id: A unique identifier for the Olympic edition.
- sport: The sport associated with the event.

- result_date: The date when the result was recorded.
- result_location: The location where the event took place.
- result_participants: The number of participants in the event.
- result_format: The format of the event result (e.g., time-based, point-based).
- result_detail: Detailed information about the event result.
- result_description: A description of the event outcome.

PROCESS

DATA SELECTION

For this project, the selection of data is methodically approached to cover an extensive range of elements related to the Olympic Games. We have incorporated **six fundamental datasets** that together provide a comprehensive view of the Olympics. These datasets include in-depth **information about athletes, detailing their profiles, physical characteristics, and roles, as well as their involvement and results** in various Olympic events. Additionally, the biographical data offers valuable context about the **athletes' careers and notable achievements**. Information on different Olympic Games editions is included, capturing details such as **the host city, country, and the specific dates** of each Games. To further enrich the analysis, we have added a **dataset summarizing the medal tallies** for each edition, which outlines how medals were distributed among different countries. Detailed event results are also part of the selection, providing insights into the specifics of each competition. By integrating these varied datasets, the project is designed to provide **a deep and well-rounded analysis of Olympic performances, historical trends, and the impact of the Games on both athletes and nations.**

DATA CLEANING AND PREPROCESSING

During the data cleaning and preprocessing phase, we carried out several detailed tasks to ensure the datasets were accurate, consistent, and ready for comprehensive analysis. **A key task was standardizing date formats across all datasets.** Initially, dates were recorded in various formats, including DD/MM/YYYY, MM-DD-YYYY, and YYYY/MM/DD. To enable smooth temporal analysis and maintain uniformity, we converted all dates to a consistent format of YYYY-MM-DD. This was essential for accurate tracking and comparison of events over time.

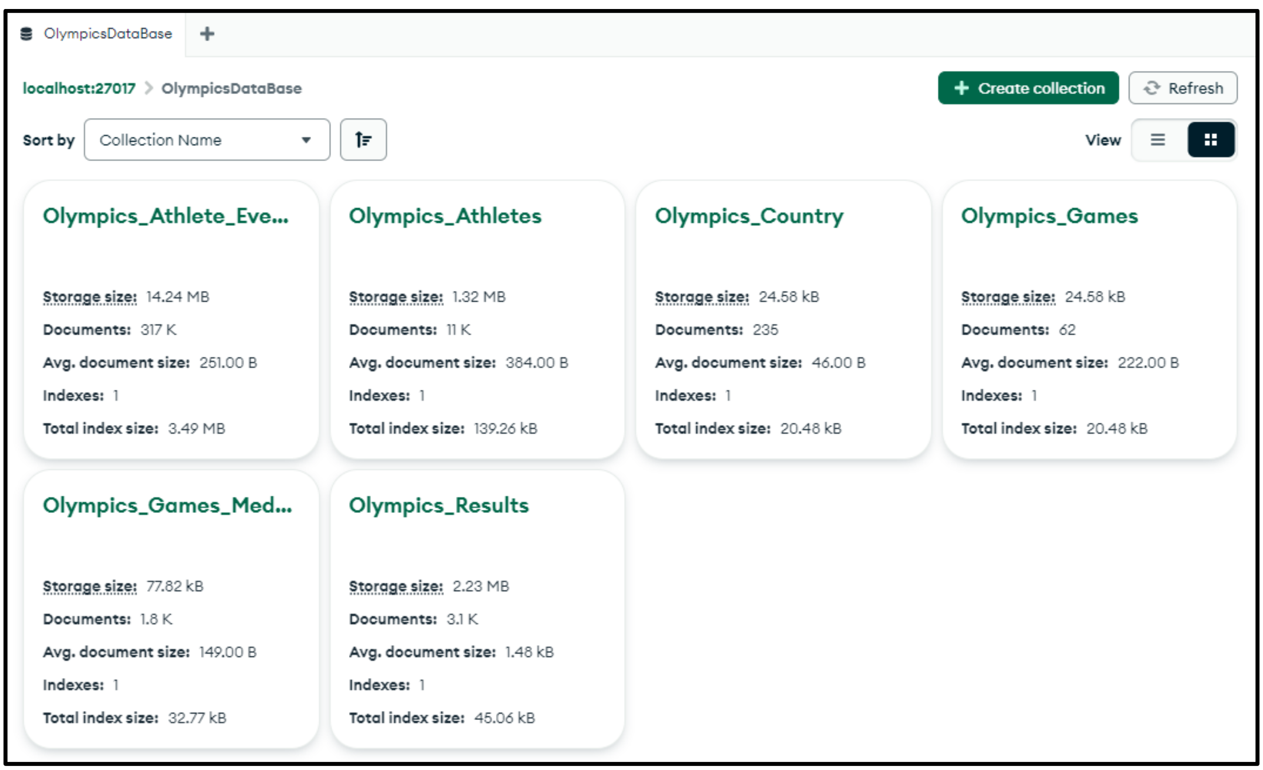
We also **addressed junk characters present in the datasets**. These **unwanted symbols and malformed text** often resulted from data entry errors or formatting issues. We systematically identified and removed these characters to prevent any potential distortion in the analysis, thus maintaining data integrity.

Furthermore, we tackled character **encoding issues caused by UTF-8 encryption**. UTF-8 encoding can lead to character representation problems, such as **garbled or misinterpreted text**. To resolve this, we reviewed and corrected text fields where encoding errors occurred, ensuring accurate display of all characters and symbols.

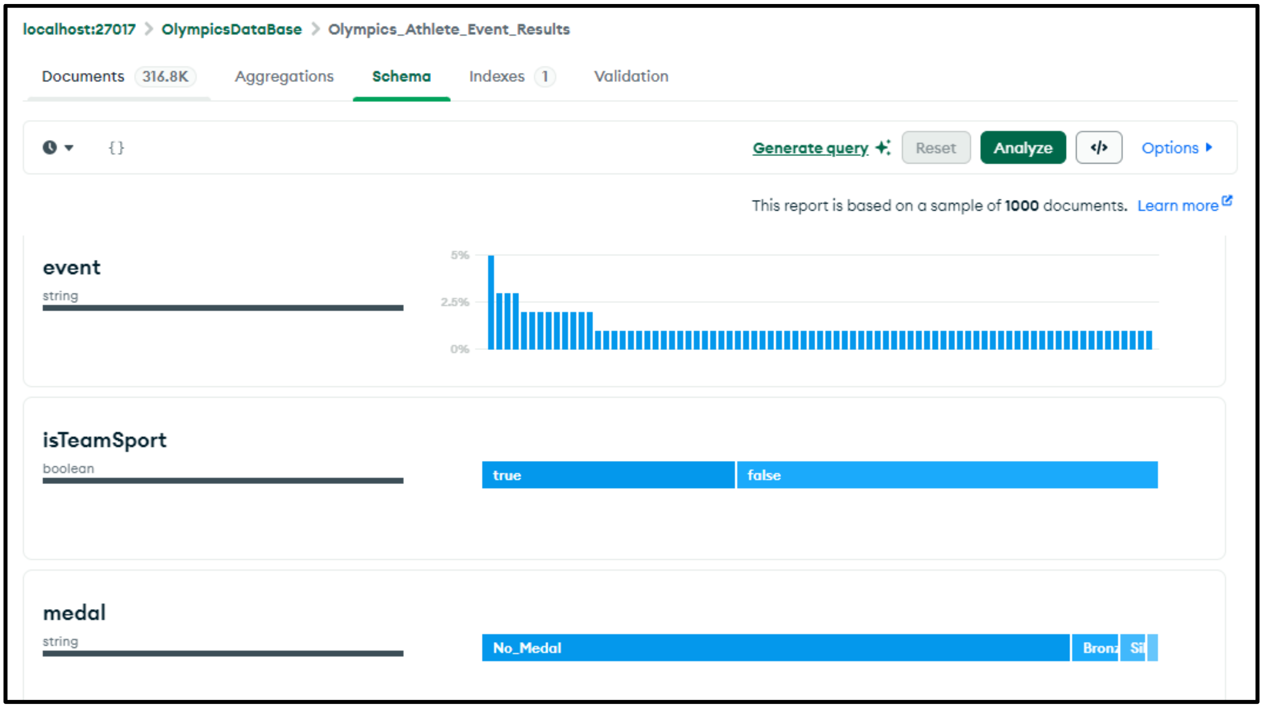
Additionally, we conducted validation checks to identify and correct discrepancies or inconsistencies. This included verifying that **unique identifiers (like athlete IDs and event IDs)** were consistently applied and free from duplication. We also ensured that mandatory fields were complete and there were no missing or null values that could affect the analysis.

DATA STORAGE AND MANAGEMENT

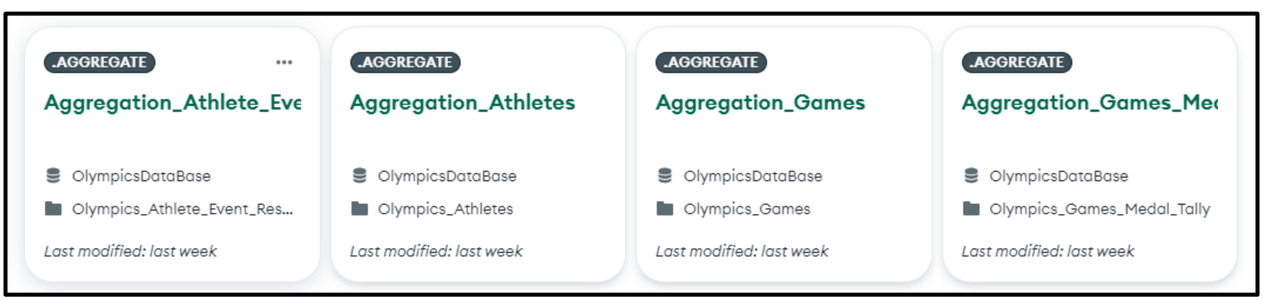
After preprocessing the datasets, we proceeded to **store the refined data in MongoDB**. Each of the six datasets, now cleaned and formatted, was imported into **separate collections** within a MongoDB database. This separation ensured that the data from **each CSV file was kept distinct and organized according to its specific context**—such as athlete details, event results, and medal tallies. We used MongoDB's robust import tools to handle the data insertion, which allowed for efficient management and retrieval of the datasets. During this process, we verified the integrity of the imported data to ensure that it matched the preprocessed formats and contained no discrepancies. By storing the data in MongoDB, we leveraged its flexible schema and scalable architecture to support future queries and analyses. This approach also facilitated the seamless integration of data from various sources while maintaining high performance and reliability.



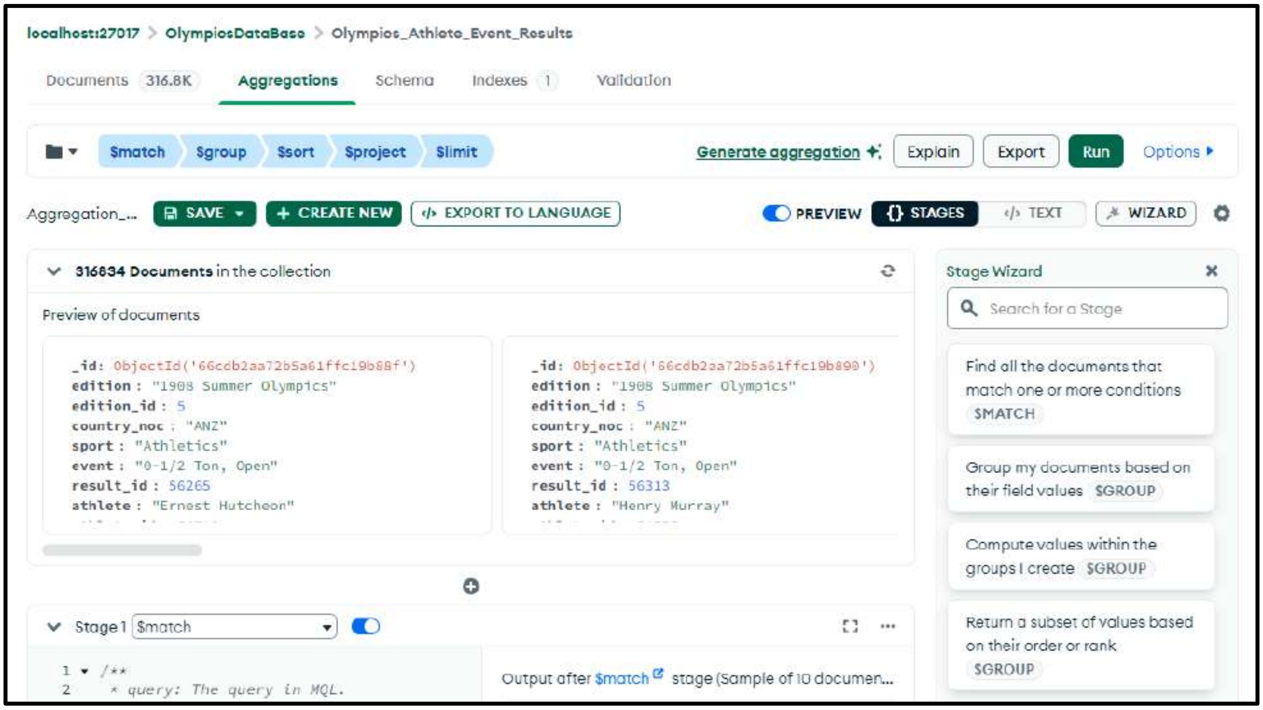
MongoDB employs a flexible schema that accommodates a wide range of data types and structures, making it ideal for our diverse datasets. Unlike traditional relational databases, MongoDB collections do not require a fixed schema, allowing for easy adjustments as data evolves. Each collection in our MongoDB database corresponds to one of the datasets, such as Athlete, Athlete_bio, or Games, and contains documents that represent individual records. These documents are stored in a BSON format, which extends JSON to include additional data types. This flexibility enables us to store varied fields and structures without rigid constraints, facilitating quick adaptations and updates to the data. Indexing and querying are optimized within this schema-less environment, ensuring that data retrieval remains efficient even as datasets grow in size and complexity.

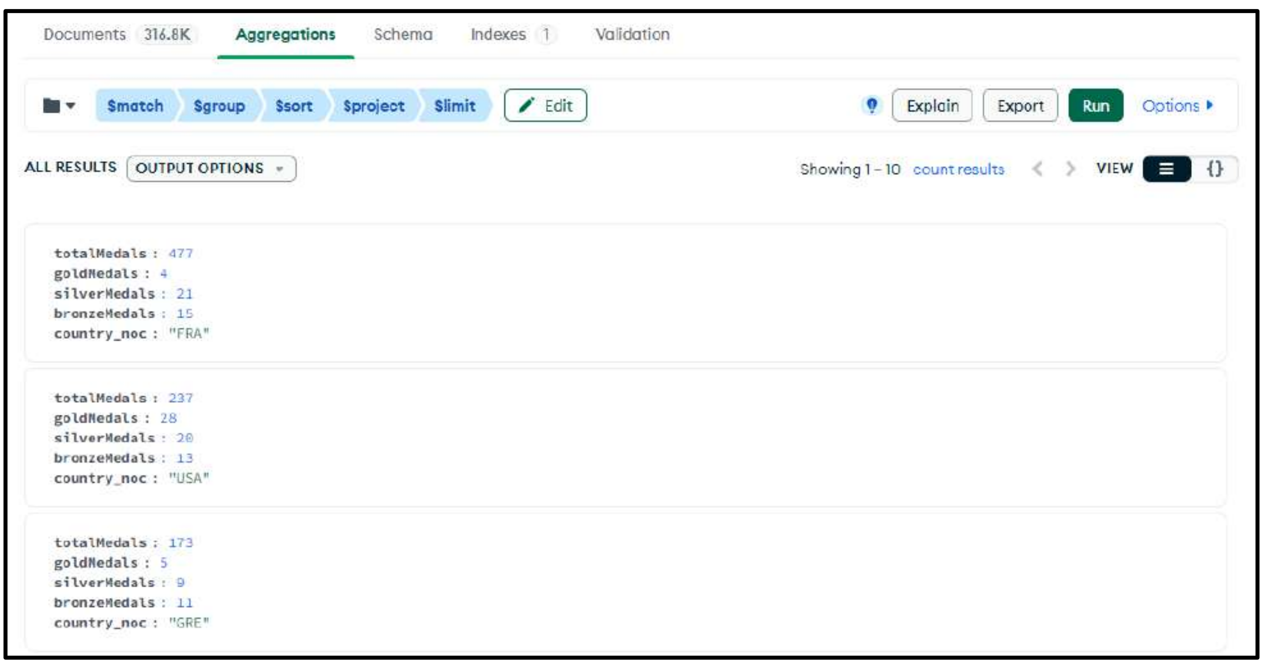


OLAP OPERATIONS – AGGREGATION



To analyze and derive insights from the datasets, we utilized MongoDB's aggregation framework. This powerful feature allows us to perform complex data transformations and computations directly within the database. We designed aggregation pipelines to handle tasks such as summarizing medal counts by country, calculating average athlete heights, and correlating event results with Olympic editions. Each pipeline consists of multiple stages, including filtering, grouping, and sorting, which are executed sequentially to produce the desired results. For instance, an aggregation pipeline might filter data by specific Olympic editions, group results by country, and then sort them by medal counts to generate a comprehensive medal tally. By leveraging MongoDB's aggregation capabilities, we were able to perform sophisticated data analyses efficiently and derive meaningful insights that would be time-consuming and complex to compute outside the database environment.

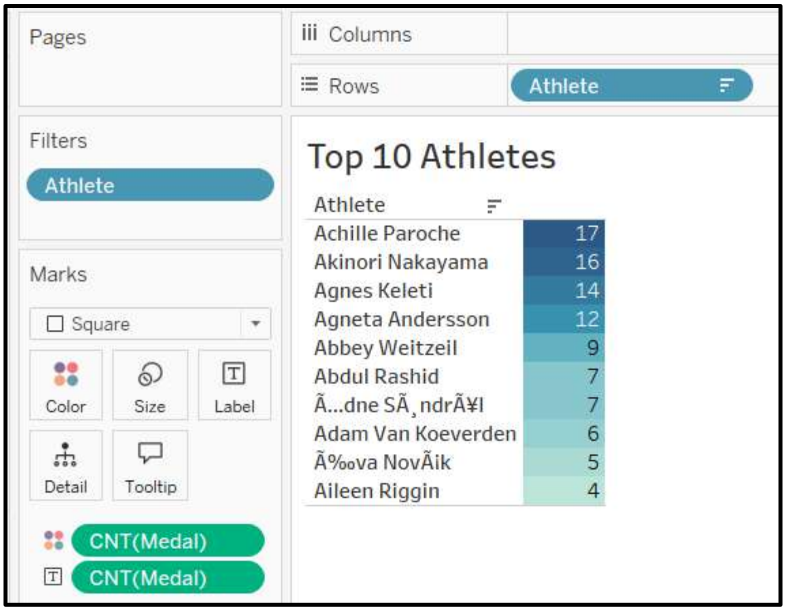




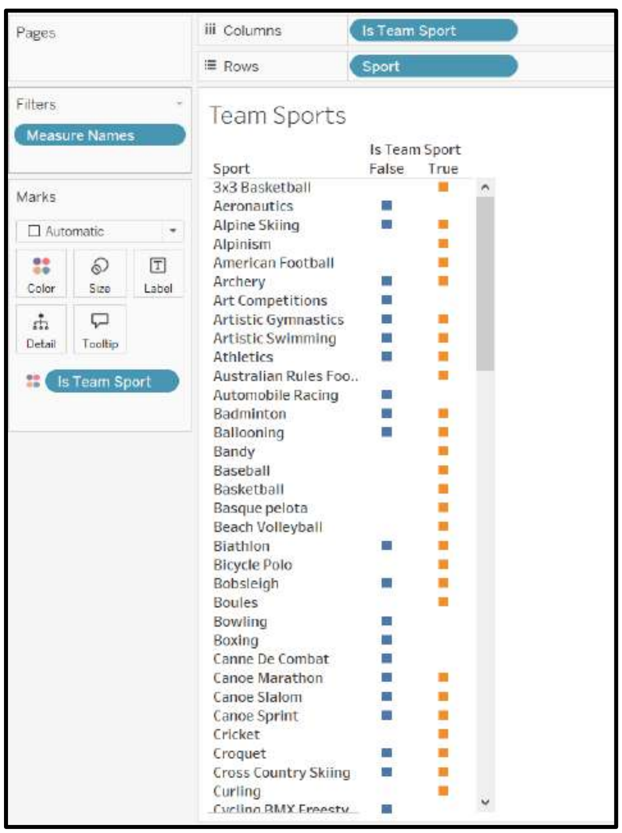
DATA VISUALIZATION

We developed a dashboard that consolidated various visual insights, including a ranking of the Top 10 athletes, which highlighted their achievements and standings. To illustrate the nature of different sports, we created a classification chart distinguishing between team and individual sports. We also designed a heat map to represent the yearly medal counts, providing a clear view of medal distribution across different years. For a detailed comparison of countries, we generated a bar graph showing the medal count distribution by country, and another bar graph to depict the age-wise distribution of medals. These visualizations not only made complex data more accessible but also facilitated interactive exploration and deeper understanding of Olympic trends and patterns.

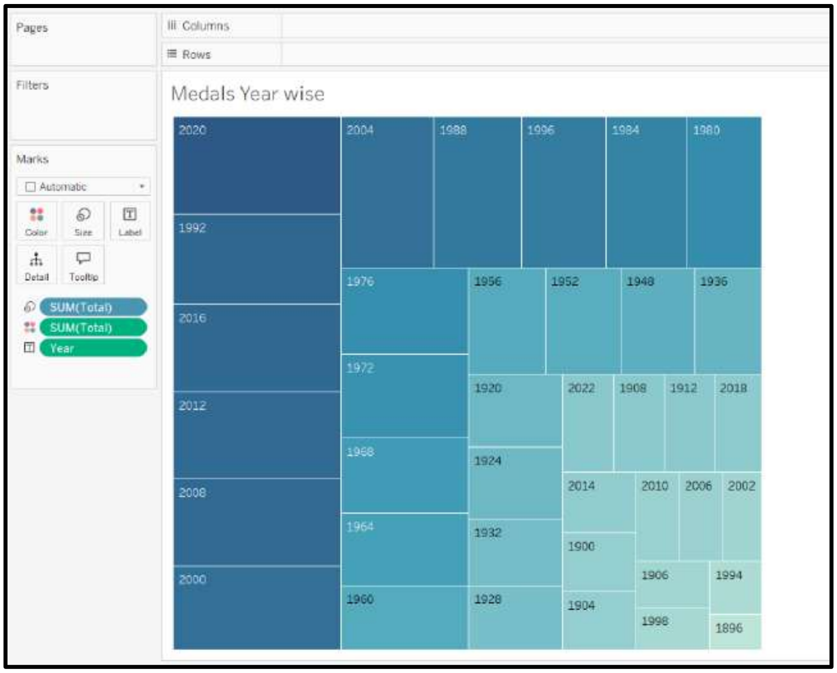
DASHBOARD



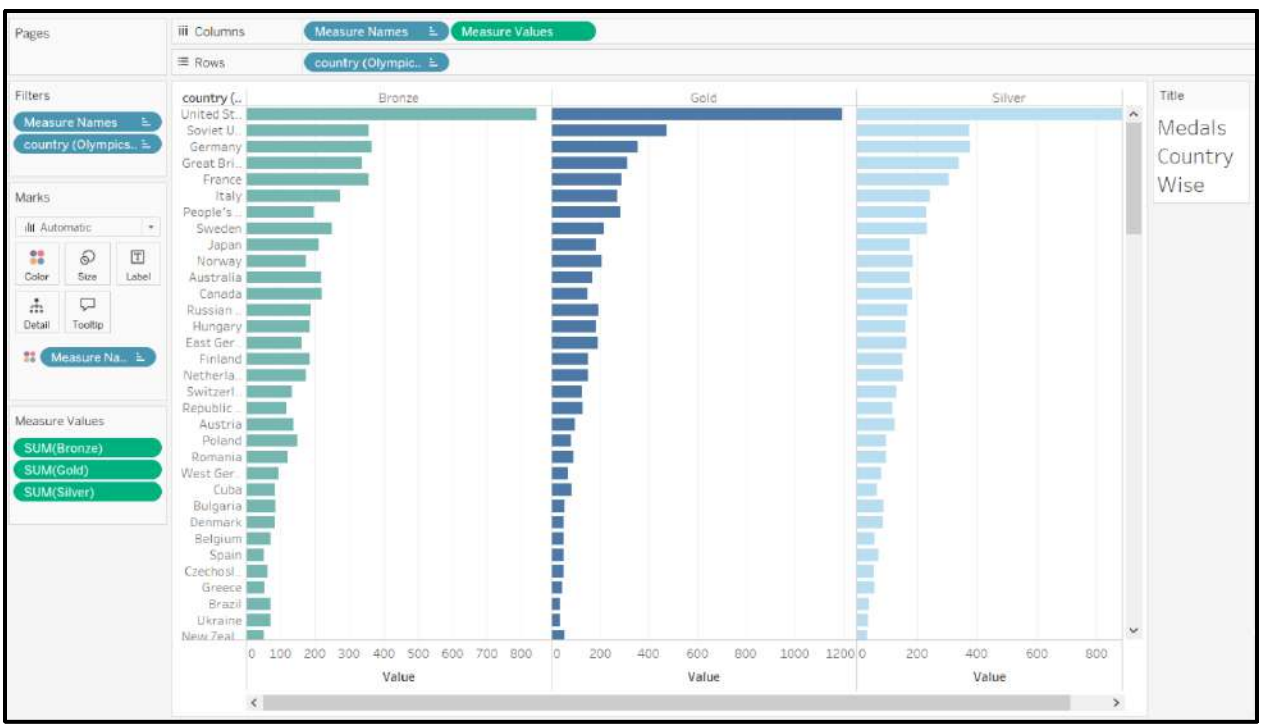
TOP 10 ATHELTES



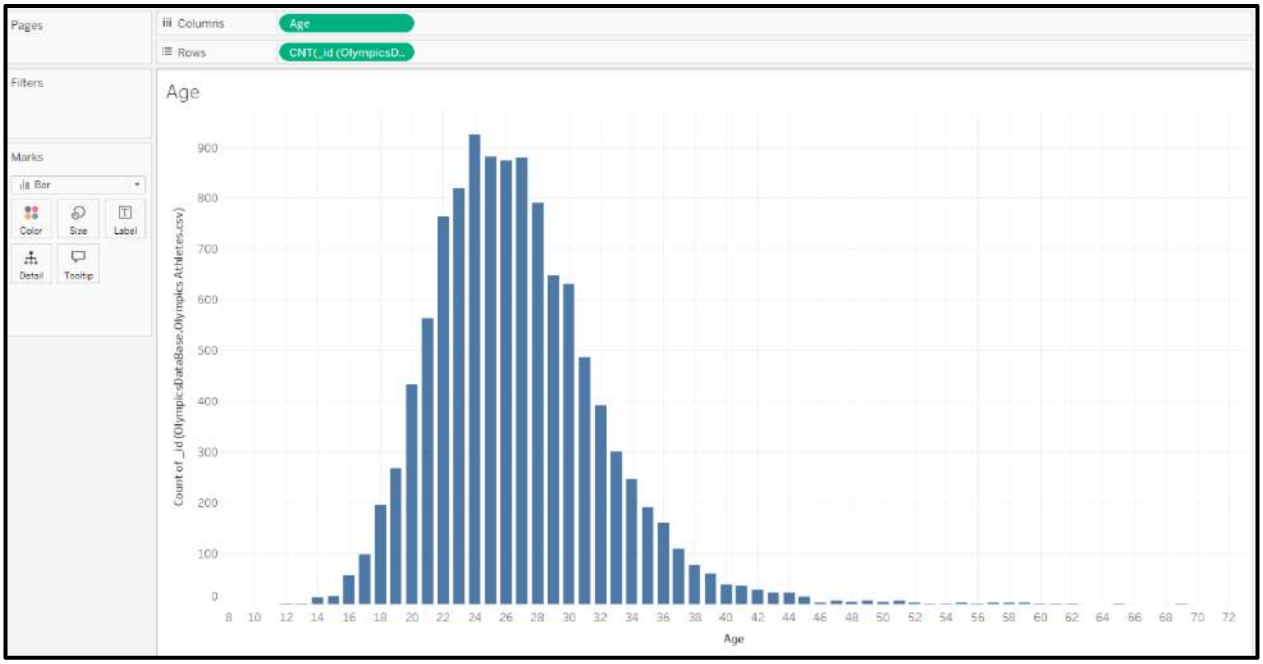
SPORT CLASSIFICATION BASED ON TEAM OR INDIVIDAL



YEARWISE MEDAL COUNT – HEAT MAP



COUNTRYWISE MEDAL COUNT DISTRIBUTION – BAR GRAPH



AGEWISE MEDAL COUNT DISTRIBUTION – BAR GRAPH

MACHINE LEARNING ALGORITHM APPLICATION

Two distinct machine learning algorithms were utilized to enhance the analysis of Olympic data. The first algorithm was applied **to predict medal counts**, leveraging historical data on country-wise and event-wise medal tallies to forecast future outcomes. By training on past performances, this predictive model provided insights into which countries might excel in upcoming events based on their historical success rates. This approach offered valuable foresight into medal distribution trends and potential future results.

Code:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from sklearn.preprocessing import StandardScaler

medal_tally = pd.read_csv("Olympic_Games_Medal_Tally.csv")

medal_tally['previous_medals'] = medal_tally.groupby('country')['total'].shift(1).fillna(0)

features = ['year', 'gold', 'silver', 'bronze', 'previous_medals']
X = medal_tally[features]
y = medal_tally['total']

scaler = StandardScaler()
```

```
X_scaled = scaler.fit_transform(X)

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)

model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"Mean Squared Error: {mse}")
print(f"Mean Absolute Error: {mae}")
print(f"R-squared: {r2}")

medal_tally['predicted_medals'] = model.predict(X_scaled)
medal_tally.to_csv("Olympic_Games_Medal_Tally.csv", index=False)
```

Output:

	A	B	C	D	E	F	G	H	I
1	edition	edition_id	year	country	gold	silver	bronze	total	predicted_medals
2	1896 Summer Olympics	1	1896	United States	11	7	2	20	19.98
3	1896 Summer Olympics	1	1896	Greece	10	18	19	47	46.7
4	1896 Summer Olympics	1	1896	Germany	6	5	2	13	13
5	1896 Summer Olympics	1	1896	France	5	4	2	11	11.21
6	1896 Summer Olympics	1	1896	Great Britain	2	3	2	7	7.04
7	1896 Summer Olympics	1	1896	Hungary	2	1	3	6	6
8	1896 Summer Olympics	1	1896	Austria	2	1	2	5	5.03
9	1896 Summer Olympics	1	1896	Australia	2	0	0	2	2.01
	1896 Summer	1	1896	Denmark	1	2	3	6	6

Olympic_Games_Medal_Tally

The second algorithm was employed for **clustering athletes**, considering attributes such as height, weight, discipline, and birth date. This clustering model grouped athletes into distinct categories based on these factors, revealing patterns and similarities among them. Analysis of these clusters provided insights into how different physical characteristics and disciplines relate to athletic performance. The inclusion of birth date data allowed for additional understanding of age-related trends and potential impacts on performance. Both machine learning models proved to be powerful analytical tools, enhancing the understanding of Olympic data and supporting deeper insights into athletic performance and medal predictions.

Code:

```
        import pandas as pd
        from sklearn.feature_extraction.text import CountVectorizer
        from sklearn.cluster import KMeans
        from sklearn.preprocessing import StandardScaler
        import numpy as np

        file_path = 'Olympic_Athletes.csv'
        data = pd.read_csv("Olympic_Athletes.csv")
        vectorizer = CountVectorizer()
        X_disciplines = vectorizer.fit_transform(data['disciplines'].fillna(''))
data['birth_date'] = pd.to_datetime(data['birth_date'], format='%d-%m-%Y', errors='coerce',
dayfirst=True)
data['birth_date'] = data['birth_date'].apply(lambda x: x.timestamp() if pd.notnull(x) else np.nan)
        X_birth_date = data[['birth_date']].fillna(0).values
        X = np.hstack([X_disciplines.toarray(), X_birth_date])
        scaler = StandardScaler()
        X_scaled = scaler.fit_transform(X)
        num_clusters = 5
        kmeans = KMeans(n_clusters=num_clusters, random_state=0)
        clusters = kmeans.fit_predict(X_scaled)
        data['cluster'] = clusters
        data.to_csv(file_path, index=False)
        print("Clustering completed and results saved to the CSV file.")
```

Output:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	code	name_short	gender	function	country_code	country	nationality	height	weight	disciplines	events	birth_date	occupation	coach	influence	ritual	cluster
2	1532872	ALEKSANYAN A	Male	Athlete	ARM	Armenia	Armenia	0	0	['Wrestling']	["Men's Greco-Roman 9		Athlete	Gevorg Al	His father, Gevorg Alek		2
3	1532873	AMOYAN M	Male	Athlete	ARM	Armenia	Armenia	0	0	['Wrestling']	["Men's Greco-Roman 77kg"]						2
4	1532874	GALSTYAN S	Male	Athlete	ARM	Armenia	Armenia	0	0	['Wrestling']	["Men's Greco-Roman 67kg"]			Personal: Martin Alekhanyan (ARI			2
5	1532944	HARUTYUNYAN	Male	Athlete	ARM	Armenia	Armenia	0	0	['Wrestling']	["Men's Freestyle 57kg"]		Athlete	National: Habetnak Kurghinyan			2
6	1532945	TEVANYAN V	Male	Athlete	ARM	Armenia	Armenia	0	0	['Wrestling']	["Men's Freestyle 65kg"]		Athlete	National: Habetnak Kurghinyan (A			2
7	1532951	ARENAS L	Female	Athlete	COL	Colombia	Colombia	162	0	['Athletics']	["Women's 20km Race W		Athlete	Personal: Brent Vallanc	In compet		2
8	1533112	McKENZIE A	Male	Athlete	JAM	Jamaica	Jamaica	0	0	['Judo']	['Men -60 kg']		Athlete	Personal a	"My coach Luke Preston		2
9	1533136	BASS BITTAYE G	Female	Athlete	GAM	Gambia	Gambia	161	0	['Athletics']	["Women's 100m", "Wo		Athlete, police	Personal: Christophe Belliard (FRA			2
10	1533176	CAMARA E	Male	Athlete	GAM	Gambia	Gambia	178	0	['Athletics']	["Men's 100m"]		Athlete, prisoi	Personal: Momodou Lamin Kujab			2
11	1533188	RUEDA SANTOS	Female	Athlete	MEX	Mexico	Mexico	0	0	['Triathlon']	["Women's Individual"]		Athlete	Personal: Eugenio Chimal (MEX)			2
12	1533189	TAPIA VIDAL RI	Female	Athlete	MEX	Mexico	Mexico	0	0	['Triathlon']	["Women's Individual"]		Athlete	Luis Miguel ChÃ¡vez RincÃ³n			2
13	1533190	GRAJALES C	Male	Athlete	MEX	Mexico	Mexico	0	0	['Triathlon']	["Men's Individual"]		Athlete	Personal: Eugenio Chimal (MEX)			2
14	1533208	MAAROUFOU H	Male	Athlete	COM	Comoros	Comoros	183	0	['Athletics']	["Men's 100m"]		Athlete	Personal: Franck Ne (FRA)			2
15	1533209	SAADI M	Female	Athlete	COM	Comoros	Comoros	0	0	['Swimming']	["Women's 50m Freesty		Athlete, stude	Personal: Marc-Olivier Froger			0
16	1533230	DIOSDADO N	Female	Athlete	MEX	Mexico	Mexico	0	0	['Artistic Swin	['Team', 'Duet']		Athlete, sport	National: .	Her parents		0
17	1533231	JIMENEZ J	Female	Athlete	MEX	Mexico	Mexico	0	0	['Artistic Swin	['Team', 'Duet']		Athlete, sport	National: .	Fictional boxer Rocky B		0
18	1533232	SOBRINO J	Female	Athlete	MEX	Mexico	Mexico	0	0	['Artistic Swin	['Team']		Athlete	National: Adriana Loftus (MEX)			0
19	1533234	ALFEREZ R	Female	Athlete	MEX	Mexico	Mexico	0	0	['Artistic Swin	['Team']		Athlete	National: Adriana Loftus (MEX)			0

Olympic_Athletes

INSIGHTS

- **Medal Forecasting**: Utilizing historical data on country and event medal tallies enables accurate predictions for future Olympic medal outcomes, providing valuable insights into potential future results.

- **Performance Trends by Country:** Analysis highlighted consistent performance patterns among countries in specific sports, allowing for predictions about which nations may excel in upcoming events.

- **Athlete Grouping Insights:** Clustering athletes by height, weight, discipline, and birth date revealed significant patterns, showing how certain physical traits and age groups correlate with performance.

- **Age and Performance:** Incorporating birth date data into the clustering analysis provided insights into how an athlete's age affects performance, uncovering age-related success trends.

- **Visualization of Medal Counts:** Heat maps and bar charts illustrating medal counts offered a visual representation of how medals are distributed over different years and countries, highlighting patterns and outliers.

- **Comprehensive Data Insights:** Integrating predictive models, clustering, and visualizations enhanced the overall comprehension of Olympic data, offering a detailed view of performance trends and athlete characteristics.

CHALLENGES

- **Data Inconsistencies:** Variations in event names across different Olympic editions required extensive data cleaning and standardization.
- **Data Gaps:** Missing or incomplete data within the athlete and athlete_bio, such as incomplete birth dates or missing athlete descriptions, limited the depth of analysis, particularly for earlier editions of the Olympics.
- **Handling Diverse Data Types:** Managing different data types across the fields in the games and olympic_results files, such as dates, categorical data, and numerical data, required careful preprocessing to ensure compatibility with analysis tools.
- **Scalability Issues:** The large volume of data in files like athlete_event_results and olympic_results caused performance bottlenecks during data loading and querying in MongoDB.

LEARNING AND EXPERIENCE

- Utilizing MongoDB: MongoDB's adaptable schema and scalable design proved highly effective for storing and managing the diverse Olympic datasets. Its capability to accommodate various data formats facilitated efficient data integration and access.

- Mastering Data Visualization: The use of Tableau for creating visual representations of the data greatly enhanced the understanding and presentation of complex information. Interactive dashboards and various chart types made data trends more accessible and easier to interpret.

- Machine Learning Applications: Implementing machine learning algorithms for predicting medal counts and clustering athletes highlighted the effectiveness of predictive analytics and pattern detection. These methods offered valuable insights into future medal prospects and athlete groupings.

- Integration of Models and Tools: Combining machine learning models with MongoDB and Tableau demonstrated how different technologies can work together to provide a richer analysis of the data. This integration allowed for comprehensive insights into Olympic performance.

- Value of Teamwork: Effective teamwork played a critical role in managing and analyzing the extensive datasets. Collaborative efforts ensured the successful completion of data processing, visualization, and machine learning tasks.

- Significance of Data Cleaning: The project emphasized the necessity of meticulous data cleaning and preprocessing. Standardizing formats and resolving encoding issues were vital for ensuring the accuracy and reliability of the analysis.

- Holistic Data Approach: Combining various techniques—data storage, visualization, and machine learning—proved invaluable for gaining a thorough understanding of the datasets. This integrated approach enriched the analysis and provided deeper insights.