ISOM 671 Group Assignment 3

Team 5: Prince Musonerwa, Jingyi Huang, Foster Mosden, Sunday Nwanyim

We have written PySpark code for processing streaming data from wearable sensors on activity trackers. Leveraging PySpark's structured streaming, the code enables real-time data ingestion, transformation, and analysis. The data is loaded from JSON files in an S3 bucket, and the Arrival_Time column is transformed into a timestamp. The streaming process involves windowed aggregation, computing counts, and averages for different activities over specific intervals (5, 15, 30 minutes).

The setup_streaming_query function configures a streaming query to perform aggregations based on a specified window duration. It groups the data by time windows and activity type, calculating counts and mean values for sensor readings (x, y, z).

The main loop iterates over different window sizes (5, 15, and 30 minutes), setting up streaming queries and displaying the results. One SQL query identifies the maximum count of a specific activity type ("gt") within a 15-minute window, providing recommendations based on certain conditions. The other query analyzes the change in distance over time, calculating the movement recommendations by comparing the average distances between consecutive windows.

The proposed Apache Spark framework for activity tracking is a comprehensive solution that can be used for real-time data processing and anomaly detection while delivering specific insights to the user. Utilizing Spark’s streaming module, the code processes live data like steps and heart rate, enabling immediate feedback. It features robust anomaly detection and diagnostic algorithms for early health issue identification and activity analysis. Engineered for scalability, it integrates with various databases for efficient data management and historical trend analysis. Incorporating Spark's machine learning, the framework offers personalized recommendations.