**Data Extraction, Transformation, and Cleaning Report**

# BIO Dataset

The BIO.csv dataset was cleaned and transformed using SQL to prepare it for analysis. Below is a detailed summary of the steps taken during the cleaning process:

## Extraction

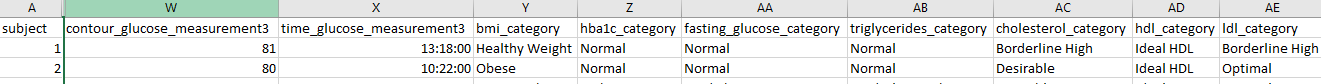
* Original File: bio.csv (45 rows × 24 columns)

Cleaned File: BIO\_Cleaned\_Data.csv (45 rows × 31 columns)

* The data was imported into a table named BIO. A schema was defined to ensure appropriate data types were assigned to each column. The ‘subject’ field was set as the primary key to uniquely identify each participant.

## Transformation

* **Renaming for Clarity:** Several columns were renamed to improve readability and consistency. For example: "Self-identify" was changed to race, "Body weight" became weight, "A1c PDL (Lab)" became HbA1c, "Fasting GLU - PDL (Lab)" became fasting\_glucose. Complex column names related to fingerstick glucose measurements and their timestamps were simplified (e.g., "#1 Contour Fingerstick GLU" became contour\_glucose\_measurement1)
* **Data Type Adjustments:** Converted collection\_time from a text field to a TIME type using SQL string parsing (SPLIT\_PART) and extracting ‘AM’ from the time value.
* **Derived Columns**: Computed new columns:  
   **\*** bmi\_category – Based on BMI ranges.  
   **\*** hba1c\_category – Based on A1C levels.  
   **\*** fasting\_glucose\_category, triglycerides\_category, etc. – Based on clinical guidelines.



* **Rounding Values:**
* Rounded BMI to two decimal places.
* Recalculated and rounded Cho/HDL Ratio for Subject 12 to one decimal place.

## Cleaning

* **Handling Outliers and Errors:** Subject 12 had unrealistic values:
* LDL was set to NULL due to invalid recalculation (high triglycerides).
* VLDL was recalculated using the standard formula triglycerides / 5.
* Cho/HDL Ratio was recalculated for accuracy.



* **Handling inconsistent naming conventions:** Renamed columns that had inconsistent naming conventions (e.g., columns starting with # or containing unnecessary descriptors).

The BIO cleaning supporting file: 

# 2. Gut Health Test Dataset

## Extraction

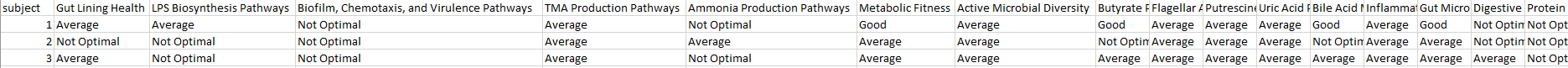
* Original File: gut\_health\_test.csv (47 rows × 23 columns)  
  Cleaned File: Gut\_Health\_Test\_Cleaned\_Data.csv (45 rows × 23 columns)
* The CSV file gut\_health\_test.csv is read into a Pandas DataFrame.
* This file contains multiple columns related to various gut microbiome functions such as, "Gut Lining Health", "Biofilm Pathways", "Gas Production", etc.
* Categorical values are encoded as integers (1, 2, 3).

## Transformation

* Converts numerical scales to meaningful categories (Good, Average, Not Optimal)
* Mapped numeric scores (1–3) to categorical labels:  
   - 1 → Not Optimal, 2 → Average, 3 → Good
* Converts all columns to string, then converts subject back to integer
* Changed data type from float to text
* Writes the transformed dataset to a new CSV for downstream tasks

## Cleaning

* Mapping Integer Codes to Descriptive Labels
* Standardizing Data Types
* Ensures **all columns are interpreted as strings**
* Prevents type inconsistencies (e.g., mixing int with NaN or "Not Applicable")
* Casting subject Column to Integer
* Required for performing numerical filtering and sorting, Helps with subject-level analysis or aggregation
* Removes specific subjects (24, 25) to clean outliers, Drop two rows of subject 24th and 25th
* Filtering Out Specific Subjects (Outlier Removal)
* Save Cleaned Dataset



The supporting file for cleaning: 

# Microbes Dataset:

## Extraction

* Original File: **‘microbes.csv’** (45 rows × 1980 columns) which has the data od presence or absence of 1980 count of microbes across 45 subjects as per ‘Scientific Dataset for personalized Nutrition and Diet monitoring’.
* Uploading the csv file into a new Jupyter notebook.
* Importing Pandas library and read the file.
* Print the data frame to view the loaded data.

### Transformation

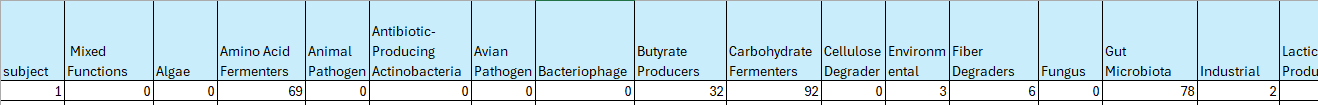
* **Define Functional Group Mapping-** Prepare a dictionary that maps microbial genera to functional groups (e.g., "Butyrate Producers", "Parasite", "Oral Microbiota", etc.).
* **Aggregated 1980 microbes**-Aggregated into 40 clinically meaningful categories:  
   Examples: Butyrate Producers, SCFA Producers, Pathogens, Probiotics, etc.
* Computed sum/count of presence for each category by subject.

### Cleaning:

* Removed unnecessary columns (e.g., unnamed or blank columns).
* Removed any whitespace from column names.
* Removed **square brackets** from values in the 'column\_name' column.
* Standardized genus names (e.g., lowercase conversion)
* From each column (representing a microbial feature), extracted the genus name for mapping.
* Matched each genus to its corresponding functional group.
* Mostly attempted to perform prefix-based matching.
* Transposed the DataFrame so each row represents a genus.
* Add new columns for genus name and functional group.
* Grouped by the functional group by using **Group by** function.
* Considered only numeric data for summing by converting the non-Numeric values to Nan/ missing and then replacing these Nan values to 0 by using **.fillna(0)** function.
* Calculated the sum of microbial abundances across all features within each functional group, per subject.
* Save the summarized, grouped dataset as ‘All\_Subjects\_Grouped\_Microbes\_Counts.CSV’ by using code given in the file below:

[](file:///C:\hp\Datathon_June_25\All_Subjects_Grouped_Microbes_Counts.ipynb)

And CSV file output looks like the screenshot below:



* Ensured alignment with subjects in BIO and Gut files.

# 4. CGMacros Dataset

The CGMacros dataset, originally comprised minute-level time-series data for 45 subject. The dataset included blood glucose levels, heart rate (HR), calories, Metabolic Equivalents (METs), nutritional details, and meal image paths.

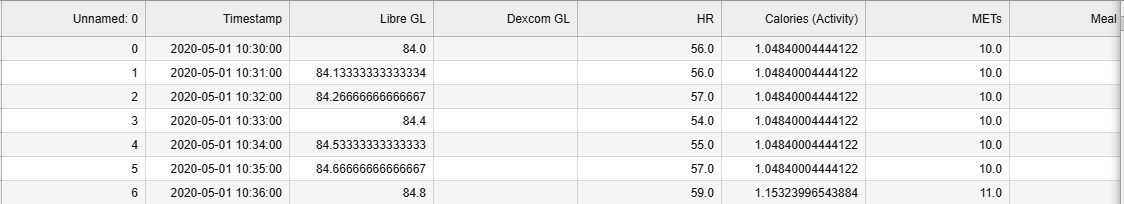
## Extraction

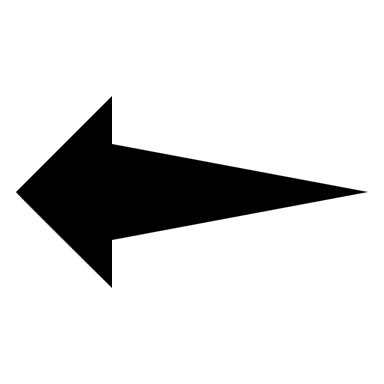
* **Source Files:** The dataset included 45 individual CGMacros files, each representing one participant and containing minute-by-minute physiological and nutritional measurements.
* **Technology Used:** Python was used to systematically load and process each file.

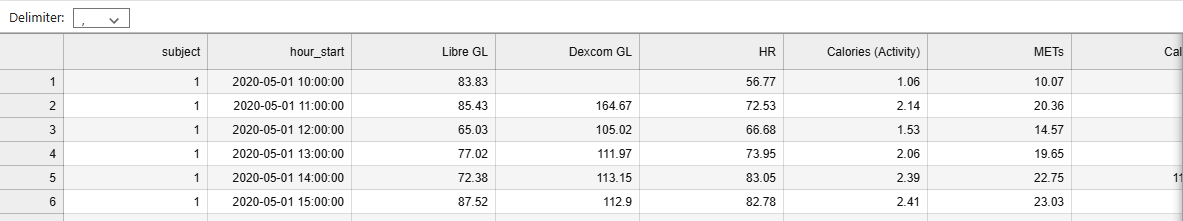
## Transformation

The raw minute-level data was transformed into a structured and analysis-ready hourly format through the following steps, all implemented in Python:

* **Timestamp Standardization:**  
  The original Timestamp column (minute-level) was replaced by a new ‘start\_hour’ column in the format YYYY/MM/DD HH:00:00, representing the beginning of each hourly segment.
* **Hourly Aggregation:**  
  For each participant, minute-level values for blood glucose, HR, calories(Activity), and METs were aggregated into hourly segments by calculating the mean of all values within each hour and rounded off the values.
* **Subject Identification:**  
  Each participant's subject number was extracted from the filename of their respective CGMacros file and added as a new ‘subject’ column in the data.
* **Removal of Redundant Columns:**  
  Any unnecessary columns such as ‘Unnamed’, present in some subjects CGMacros file were removed to clean up the dataset.
* **Unified Dataset Creation:**  
  Data from all 45 participants was combined into a single consolidated dataset ‘AllSubjectHourlyAveragedCGMacros.csv’, streamlining analysis and comparison.

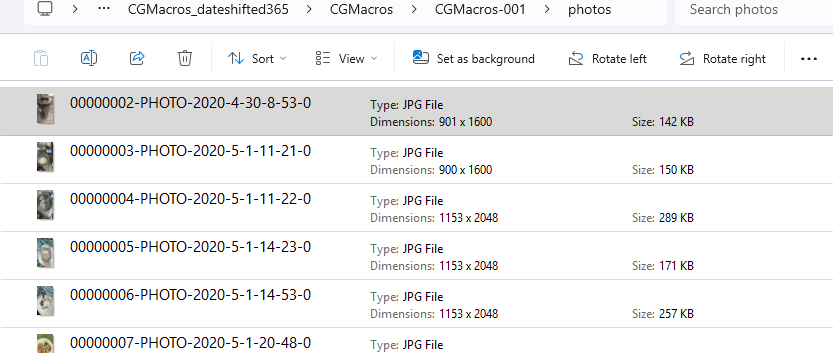


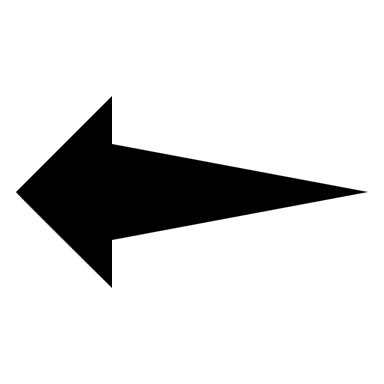


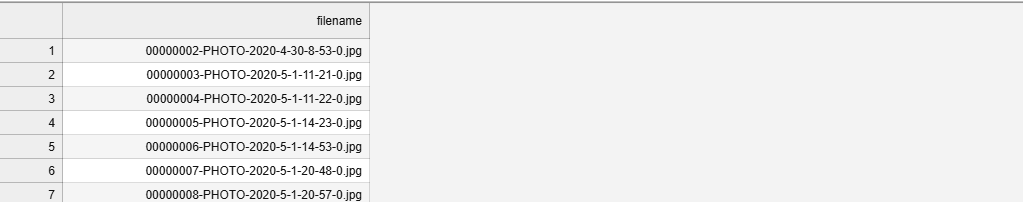


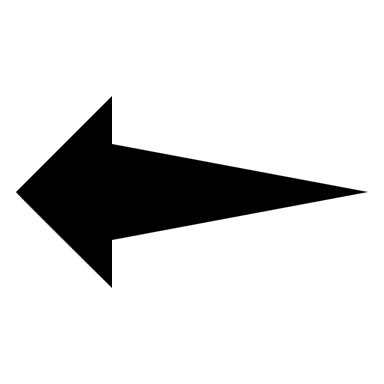
## Cleaning Cleaning was a critical phase to ensure data completeness, especially for missing image paths and nutritional details. This was also accomplished using Python.

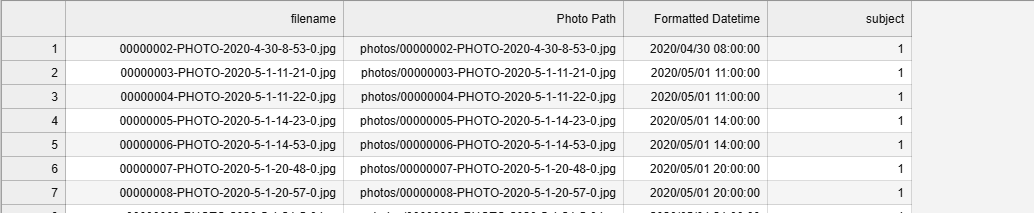
* **Image Path Reconstruction:**
* A list of all meal image filenames was generated from the folder structure.
* A new Photo Path column was created by prepending 'photos/' to each filename (e.g. photos/00000003-PHOTO-2020-5-1-11-21-0.jpg).
* The date and time were extracted from the filenames and formatted into a ‘Formatted Datetime’ column using the same format as ‘start\_hour’.
* The subject number was included with each image path record.



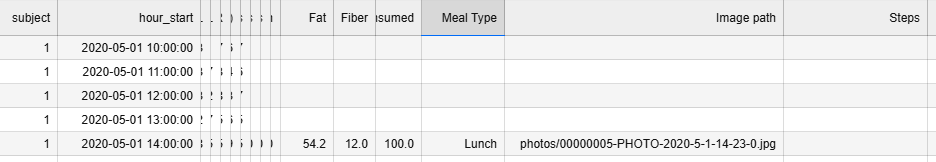


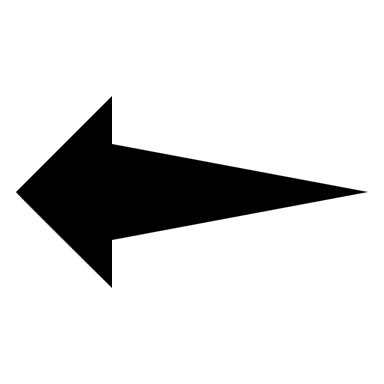


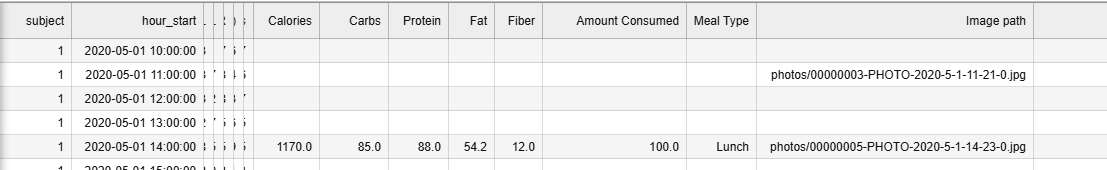




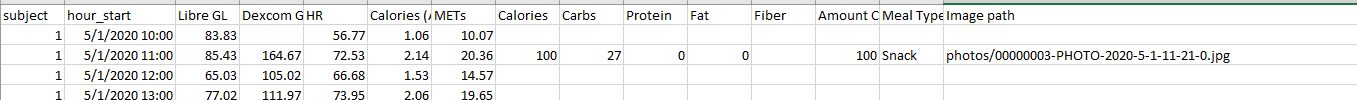
* **Imputation of Missing Image Paths:**  
  By matching subject and Formatted Datetime with ‘subject’ and ‘start\_hour’, missing or empty Image path entries in the main dataset were filled using the reconstructed ‘Photo Path’.







* **Nutritional Value Synchronization:**   
  Alongside image path insertion, associated nutritional values were updated to maintain consistency and completeness across the dataset.



* **Correction of Erroneous 'Amount Consumed' Values:**  
  The Amount Consumed column contained incorrect percentage values such as 300, 600, 900 etc. These values were standardized to 100%, based on visual confirmation from the end-meal images indicating that the meals were fully consumed.

The supporting files for above transformation and cleaning to AllSubjectHourlyAveragedCGMacros.csv:

## Enitity Relationship Diagram of the model:

The ERD defines ‘subject’ as the primary key in the BIO\_Cleaned\_Data table and a foreign key in related tables, enabling relational integrity across the dataset. It shows one-to-one relationships between BIO\_Cleaned\_Data and both All\_Subjects\_Grouped\_Microbes\_Counts and Gut\_Health\_Test\_Cleaned, meaning each subject has a single corresponding record in these tables. A one-to-many relationship is implied between BIO\_Cleaned\_Data and AllSubjectHourlyAveragedCGMacros, where each subject may have multiple hourly entries capturing activity and metabolic data. This design allows consistent integration of demographic, clinical, microbial, and behavioral data per subject.

