IEBS | Digital School

Innovation & Entrepreneurship Business School

Master in Data Science and Big Data

Machine Learning Models for Cardiorespiratory Fitness Prediction using Biometric Data

Marcos Jiménez Juliana

PhD, Molecular Biology

Master's Thesis

Supervised by: Zaira Adame
Director of the Master in Data Science and Big Data
Director of Strategic Innovation and Business Intelligence IEBS
Madrid, 2025

The structure and length of this Master’s Thesis strictly follow the guidelines provided in the template for the preparation of a Global Project issued by the school. All code and data used in this work is available in a GitHub repository under an MIT license. The link to the repository can be found in the section "Proposed Solution -Code Availability". This document has been written in LaTeX.

Madrid, 2025

Index

List of Figures

List of Tables

Abstract

Cardiorespiratory Fitness (CRF), commonly quantified through maximal oxygen uptake (VO_2max), is a key predictor of cardiovascular and overall health. Direct measurement through treadmill-based cardiopulmonary exercise testing is the gold standard, but its use is limited in large-scale or clinical contexts due to cost, time, and safety concerns. This project proposes machine learning as a scalable alternative for VO_2max estimation using biometric, clinical, and lifestyle data. Three algorithms were implemented and compared: Multiple Linear Regression, Random Forest, and XGBoost. XGBoost achieved the best predictive performance, followed by MLR. The final model was deployed through a Streamlit web application, enabling user-friendly CRF estimation and categorical fitness classification. The results demonstrate that machine learning offers a cost-effective, non-invasive, and practical tool to complement direct VO_2max measurement, supporting broader CRF assessment in both clinical and epidemiological settings.

Introduction

1 Cardiorespiratory Capacity

1.1 Cardiorespiratory Fitness

Cardiorespiratory fitness (CRF) is defined as the capacity of the respiratory and circulatory systems to efficiently transport molecular oxygen (O_2) from the atmosphere to skeletal muscle cells during physical exercise (Caspersen *et al.*, 1985, reviewed in Raghuveer *et al.*, 2020). The efficiency of oxygen transport depends on multiple genetic factors such as cardiac size and function, lung volume, muscle fibers, blood vessel elasticity, mean red blood cell hemoglobin, affinity of hemoglobin for oxygen, as well as environmental factors such as age, exercise, diet or smoking (reviewed in Klevjer *et al.*, 2023).

Low CRF is a strong risk factor for cardiovascular diseases (CVD) (Lakka *et al.*, 1994; Kodama *et al.*, 2009), while high CRF has been shown to reduce the risk of several conditions such as diabetes (Lee *et al.*, 2009), metabolic syndrome (Earnest *et al.*, 2013), or dementia (DeFina *et al.*, 2013). For these reasons, CRF is considered a powerful predictor of overall health (Kodama *et al.*, 2009), and its incorporation as a risk factor into the European Systematic Coronary Risk Evaluation (SCORE) algorithm has been shown to improve the reclassification of CVD mortality risk by up to 43% (Laukkanen *et al.*, 2007; Liu *et al.*, 2023).

1.2 Maximal Oxygen Uptake (VO_2max)

The gold standard metric for quantifying CRF is maximal oxygen uptake (VO_2max). This term was originally introduced by Hill, Lupton, *et al.*, 1923, who suggested the existence of an upper limit to the amount of oxygen that can be delivered to the muscles (reviewed in Kaminsky *et al.*, 2019). Thus, VO_2max reflects an individual's oxygen consumption capacity at maximal exercise intensity.

The assessment of VO_2max can be performed using both direct and indirect methods, which vary in accuracy, required equipment, and applicability across different populations (Lang *et al.*, 2018). Direct assessment through cardiopulmonary exercise testing, most commonly known as the treadmill exercise stress test remains the standard method for measuring VO_2max (Foster *et al.*, 1984; of Sports Medicine *et al.*, 2013). This approach enables the accurate quantification of CRF, establishing it as a reliable indicator of both disease risk and overall health status (reviewed in Kaminsky *et al.*, 2019).

However, cohort studies and health surveys rarely measure VO_2max due to the time, cost, and resources required to conduct treadmill exercise stress testing, as well as the potential risks associated with maximal physical exertion in certain patients (Liu *et al.*, 2023). Consequently, there is a notable lack of VO_2max data in large-scale population-based studies, which limits the accuracy of disease risk prediction and the evaluation of overall health status. As an alternative, several submaximal field tests and non-exercise prediction models have been developed during recent years (Lang *et al.*, 2018; Liu *et al.*, 2023).

2 Alternatives to the Treadmill Exercise Stress Test

2.1 Submaximal Exercise Tests

Submaximal exercise tests are designed to estimate VO_2max without requiring participants to exercise until exhaustion. These tests rely on the linear relationship between heart rate and oxygen consumption at different exercise intensities. Common protocols include the Astrand-Ryhming cycle ergometer test (Sartor *et al.*, 2013) or field-based assessments such as the 6-minute walk test (Hong *et al.*, 2019) or step test variants (Beutner *et al.*, 2015; Bohannon *et al.*, 2015; Ho *et al.*, 2025). Although less accurate than direct cardiopulmonary testing, submaximal protocols are a widely accepted alternative with minimal trained personnel and equipment needs. However, these tests are still rarely implemented in large-scale cohort studies and health surveys, which limits the accuracy of disease risk prediction and the evaluation of overall health status. For this reason, nonexercise-based alternatives have emerged as a practical solution.

2.2 Nonexercise-Based Algorithms

To address the logistical barriers of exercise testing, several nonexercise algorithms have been developed since the past century (Jackson *et al.*, 1990). These algorithms estimate VO_2max based on easily obtainable information such as age, sex or body mass index and propose equations based on linear relationships (reviewed in Ashfaq *et al.*, 2022). Although they offer practical and reasonably accurate estimates, they present several limitations. A key limitation is that many of these algorithms were developed using very small cohorts, leading to suboptimal performance in other populations (reviewed in Ashfaq *et al.*, 2022; Liu *et al.*, 2023). Furthermore, they are mostly based on linear relationships, which prevents them from capturing nonlinear interactions between variables. Moreover, their construction often relied on a limited selection of health indicators, overlooking additional variables that could be relevant to CRF (Liu *et al.*, 2023).

2.3 Machine Learning Approaches

Recent advances in data science have introduced machine learning (ML) methods as a promising tool for CRF estimation. By integrating diverse datasets, including demographic, biometric, lifestyle, and wearable device data, ML algorithms can address the challenges outlined in the previous section. They are capable of generalizing predictions using larger cohorts, incorporating variables not intuitively associated with CRF, and identifying complex nonlinear relationships that traditional regression models may overlook (reviewed in Ashfaq *et al.*, 2022, Liu *et al.*, 2023).

The section "State of the Art" briefly discusses some of the ML models proposed by researchers prior to this work.

3 Proposed Solution in this Work

This work proposes the training of several Machine Learning regression models with the aim of estimating Cardiorespiratory Fitness by predicting the continuous target variable VO_2max as accurately as possible using public, large-scale biometric data without the need to perform physical tests.

3.1 Overview

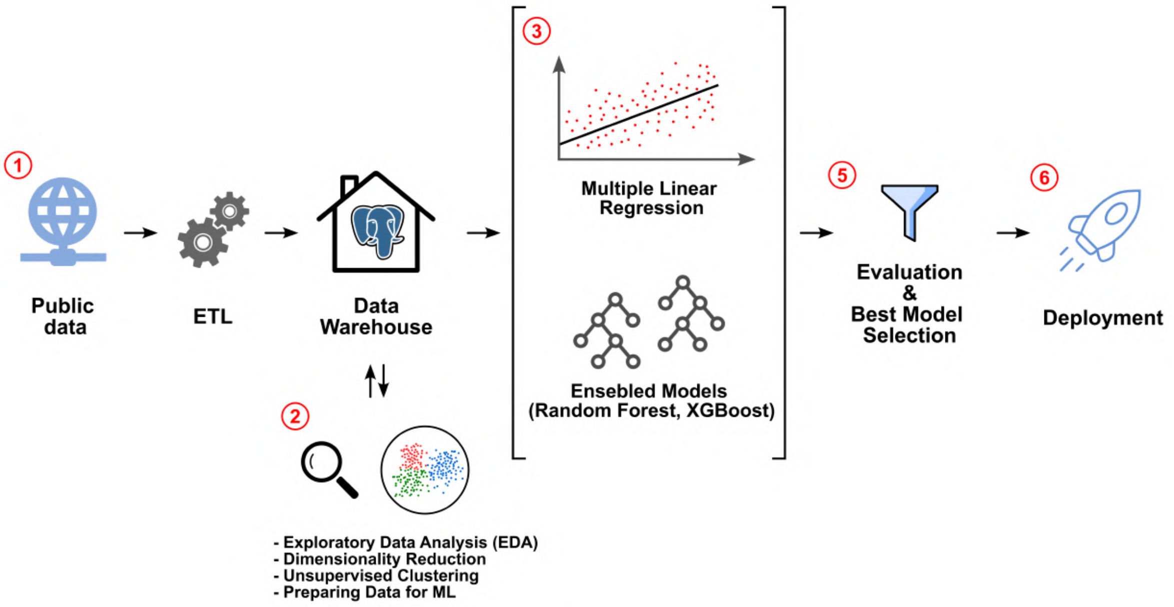
1
Public data
ETL
Data Warehouse
2
- Exploratory Data Analysis (EDA)
- Dimensionality Reduction
- Unsupervised Clustering
- Preparing Data for ML
3
Multiple Linear Regression
Ensebled Models (Random Forest, XGBoost)
5
Evaluation & Best Model Selection
6
Deployment

Figure 1. Overview of the proposed workflow for cardiorespiratory fitness estimation. The workflow is divided into 5 sections: **1.** Data acquisition from public sources and ETL into a PostgreSQL Data Warehouse. **2.** Exploratory Data Analysis (EDA), dimensionality reduction, clustering, and feature preparation form ML. **3.** Training of regression models (e.g., Multiple Linear Regression, Random Forest, XGBoost). **4.** Model evaluation and best model selection. **5.** Deployment of the selected model for practical use.

Figure 1 provides a visual representation of the proposed workflow carried out in this work. The work is divided into 5 sections or sub-projects that have been explained in detail in the section "Proposed Solution" of this document. In summary:

1. **Building a Biometric Data Warehouse in PostgreSQL (ETL)**: cleaning and integrating raw

NHANES `.xpt` files into a centralized PostgreSQL database, ensuring consistency and accessibility for downstream analysis.

2. **Exploratory Data Analysis and Data Preparation for Modeling**: statistical profiling, dimensionality reduction, clustering and feature engineering to reveal hidden patterns and prepare data for ML models.

3. **Aerobic Fitness Prediction Using Ensemble Machine Learning**: training regression models (Multiple Linear Regression, Random Forest and XGBoost) to estimate VO_2max.

4. **Evaluation and Model Selection**: assessing performance metrics (e.g., RMSE, R^2) across all trained models to select the most robust predictor.

5. **Deployment**: deploying the best-performing model as an interactive web application using `Streamlit`. Users will be able to input their biometric data (e.g., age, weight, height), to obtain an estimated VO_2max value and receive a clear interpretation of what that value means in terms of their aerobic fitness.

3.2 Data Source and Study Polulation Limitations

For this purpose, data from the National Health and Nutrition Examination Survey (NHANES) were used. NHANES is conducted by the National Center for Health Statistics (NCHS) and collects data about the health of adults and children in the United States, being the only national health survey that includes health exams, laboratory tests, and dietary interviews for participants of all ages. Since 1999, NHANES has been conducted as a continuous program with two-year cycles (cohorts). The NHANES project typically includes between 1,200 and 2,000 variables stored in tabular format across separate files for each type of data (demographic, dietary, examination, laboratory and questionnaire data) and for each survey cycle.

Data from the NHANES cohorts 1999-2000, 2001-2002, and 2003-2004 (available at NHANES Cohorts) will be used, since this was the only period where experimental estimates of VO_2max were measured, making these cohorts exceptionally valuable.

As indicated in the CRF manual (for Health Statistics, 2005), participants eligible for CRF measurements (VO_2max) were selected according to their age, medical conditions, and physical limitations. Additionally, the NHANES study was limited to people aged between 16-49 years. Thus, the predictions of this work are also limited to this age range.

State of the Art

Previous Machine Learning Algorithms for CRF prediction

As explained in the introduction, some of the Machine Learning (ML) models for CRF (VO_2max) prediction proposed by previous researchers prior to this work are presented in this section. The following **Table 1** summarizes studies conducted between 2016 and 2021. The data have been extracted and reproduced from the review Ashfaq *et al.*, 2022.

Study	Population Study Size	ML Model	Number of Predictor Variables	R	SEE	RMSE
Abut *et al.*, 2016	100	SVM	4	0.94		2.92
		MLP Neural Network	4	0.93		3.14
		TB	4	0.92		3.38
Dinçer *et al.*, 2016	26	MLR	10	0.79		4.22
Kaya *et al.*, 2016	48	SVM	5	0.72	8.03	
		MLP	5	0.56	9.58	
		SDT	5	0.38	10.67	
Beltrame *et al.*, 2016	10	Neural Network	6	0.97		
M. F. Akay *et al.*, 2017	98	SVM	8	0.77	4.87	
		GRNN	8	0.81	4.51	
		RBFN	8	0.51	7.24	
		DTF	8	0.70	5.62	
M. Akay *et al.*, 2017	62	MLR	7	0.93	5.14	
Beltrame *et al.*, 2017	16	Random Forest	1	0.87		
M. Akay *et al.*, 2017	18	MLR	7	0.88	3.49	
M. F. Akay *et al.*, 2018	333	MLR	6		3.95	
M. Akay *et al.*, 2018	97	SVM with Relief-F	9	0.785	6.415	
		RBF	9	0.661	7.740	
		TB	9	0.662	7.771	
Przednowek *et al.*, 2018	308	MLP	18			4.78
		ANN with RBF	18			4.07
Borror *et al.*, 2019	12	ANN	5	0.91	3.34	
Abut *et al.*, 2019	185	SVM	7	0.86		2.91
		GRNN	7	0.81		3.37
		SDT	7	0.64		4.51
Zignoli *et al.*, 2020	7	RNN	6	0.94		
Shandhi *et al.*, 2020	17	SLR	3	0.64		4.3

Table 1. Summary of small population size studies for CRF prediction conducted between 2016 and 2021. The table has been simplified from Ashfaq *et al.*, 2022. *SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; TB: Tree Bagging; GRNN: Generalized Regression Neural Network; RBF/RBFN: Radial Basis Function Network; DTF: Decision Tree Forest; ANN: Artificial Neural Network; RNN: Recurrent Neural Network; SLR: Simple Linear Regression. R: Correlation coefficient; SEE: Standard Error of the Estimate; RMSE: Root Mean Squared Error*

As shown in **Table 1**, although several efforts have been made in recent years to predict cardiorespiratory fitness (CRF), these algorithms were generally developed using only few variables and study-specific, small-sized cohorts. This limitation leads to suboptimal performance when applied to other populations and makes it difficult to compare the algorithms with one another. Consequently, the table provides only a general overview of their performance rather than a precise benchmark (reviewed in Ashfaq *et al.*, 2022).

By contrast, **Table 2** presents non-exercise algorithms that were developed using larger data cohorts. The data in this table have been simplified and extracted from Liu *et al.*, 2023.

Study	**Population Study Size**	**ML Model**	**Number of Predictor Variables**	**RMSE [95% CI]**	**R^2 [95% CI]**	**P value***
Liu *et al.*, 2023	NHANES (1999–2004): 5689	Extended LightGBM model	49	8.26 [7.44–9.09]	0.32 [0.26–0.38]	
	NHANES (1999–2004): 5689	Parsimonious LightGBM model	40	8.51 [7.73–9.33]	0.28 [0.23–0.33]	.003
Jackson *et al.*, 2012	11365	MLR	7	9.67 [8.81–10.66]	0.22 [0.20–0.31]	<.001
			7	9.71 [8.94–10.62]	0.25 [0.17–0.26]	<.001
			7	9.80 [9.03–10.71]	0.22 [0.17–0.27]	<.001
			7	10.94 [10.10–11.88]	0.26 [0.21–0.32]	<.001
Matthews *et al.*, 1999	799	MLR	6	10.05 [9.36–10.81]	0.20 [0.15–0.25]	<.001
Heil *et al.*, 1995	374	MLR	5	10.38 [9.54–11.03]	0.23 [0.18–0.29]	<.001
Jurca *et al.*, 2005	ACLS participants: 46190	MLR	5	10.24 [9.50–11.04]	0.20 [0.16–0.26]	<.001
Jurca *et al.*, 2005	NASA Employees: 1863	MLR	5	10.57 [9.83–11.36]	0.20 [0.15–0.25]	<.001
Wier *et al.*, 2006, BMI version	NASA Employees: 2081	MLR	4	11.36 [10.87–12.62]	0.16 [0.12–0.21]	<.001
Wier *et al.*, 2006, WC version		MLR	4	11.14 [10.87–12.64]	0.18 [0.13–0.23]	<.001
Wier *et al.*, 2006, %fat version		MLR	4	10.87 [10.77–12.31]	0.20 [0.15–0.26]	<.001
Jackson *et al.*, 1990, BMI version	NASA Employees: 1543	MLR	5	11.50 [10.76–12.30]	0.18 [0.14–0.23]	<.001
Jackson *et al.*, 1990, %fat version		MLR	5	11.70 [10.87–12.62]	0.23 [0.18–0.29]	<.001

Table 2. Summary of non-exercise algorithms for CRF prediction conducted in larger population cohorts. The table is a simplified version of the summary presented by Liu *et al.*, 2023. *MLR: Multiple Linear Regression; RMSE: Root Mean Squared Error; CI: Confidence Interval;* R^2*: Coefficient of Determination.*

Although these non-exercise algorithms were developed using much larger data cohorts, they all share the limitation of relying exclusively on multiple linear regression models and on a relatively small number of predictor variables. In contrast, the only study conducted with both a substantial dataset and a larger set of predictor variables, while also employing a more advanced machine learning model, was carried out by Liu *et al.*, 2023. Moreover, this study used NHANES 1999–2004 data, the same source employed in the present work, which makes it an appropriate benchmark for comparison with the models trained here.

In addition, it is important to note that even in their best-performing model, the authors were unable to explain more than 32% of the variance in VO_2max (R^2 reported in Liu *et al.*, 2023). This highlights that both the nature of the data and the outcome variable pose a challenge that remains far from ideal conditions.

Finally, the novelty of the present work lies in two main aspects:

- Attempting to improve the previous models reported in **Table 1** and **Table 2** while employing a broader range of predictive models, including Multiple Linear Regression and ensembled methods based on bagging (Random Forest) and boosting (XGBoost).
- Exploring a practical application of these predictions by deploying the best-performing model as an interactive web application using `Streamlit`. Users will be able to input their biometric data (e.g., age, weight, height) to obtain an estimated VO_2max value and receive a clear interpretation of what that value implies in terms of their aerobic fitness.

Objectives

The general objective of this Master's Thesis is to design, implement, and validate a data-driven system capable of accurately predicting cardiorespiratory fitness using VO$_2$max as a target variable from non-exercise biometric data (NHANES), addressing the limitations of previous studies that relied on small cohorts, restricted variable sets, and exclusively linear regression models.

To achieve this general objective, the following specific objectives were defined:

1. **Data Integration and Warehousing:** build a centralized biometric data warehouse in PostgreSQL by cleaning, transforming, and integrating raw NHANES files into a consistent and accessible format for downstream analysis.

2. **Data Analysis and Preparation:** conduct exploratory data analysis (EDA), apply dimensionality reduction and clustering techniques, and perform feature engineering to enhance data quality and prepare meaningful inputs for predictive modeling.

3. **Predictive Modeling:** train and compare multiple machine learning models, including Multiple Linear Regression, ensemble methods (Random Forest and XGBoost), and a neural network, to capture both linear and nonlinear relationships between biometric data and VO$_2$max.

4. **Model Evaluation and Deployment:** evaluate models using performance metrics such as RMSE and R^2, select the best-performing predictor, and deploy it as an interactive web application in `Streamlit`, enabling users to input their biometric data and receive both a VO$_2$max estimate and an interpretation of their aerobic fitness.

Proposed Solution

Code Availability

All code and data used in this work is available in a GitHub repository under an MIT license.

Name of the Repository: crf_prediction Link to the repository: https://github.com/mjimenezj/crf_prediction

Structure of the repository:

```
CRF_prediction/
|-- data/
|   |-- raw/               # Original NHANES .xpt files (raw source data)
|   |-- processed/         # Cleaned and transformed datasets
|-- images/                # Images, figures and plots
|-- notebooks/             # Jupyter Notebooks documenting the workflow
|   |-- 1_ETL.ipynb
|   |-- 2_EDA_and_Clustering.ipynb
|   |-- 3_ML_models.ipynb
|-- src/                   # Source code for modular implementation
|   |-- etl.py             # Python script for ETL operations
|   |-- eda.py             # Python script for EDA operations
|-- app/                   # Streamlit application for deployment
|   |-- app.py             # streamlit app python script
|   |-- scaler.joblib      # scaler for prediccionts
|   |-- xgb_model.joblib   # best model for predictions
|-- .env                   # Env variables (sensitive information, not tracked by Git)
|-- .env.example           # Template for environment variables, safe to share
|-- .gitignore             # Git configuration to exclude unnecessary files
|-- README.md              # Project documentation and usage instructions
|-- requirements.txt       # Python dependencies required to run the project
```

The general scheme of the proposed solution can be found in the section "Proposed Solution in this Work – Overview" of the Introduction. The following sections provides a detailed description of all the workflow:

1 Building a Biometric Data Warehouse in PostgreSQL (ETL)

The code explained in this section can be found in the Jupyter Notebook `notebooks/1_ETL.ipynb` and in `src/etl.py`.

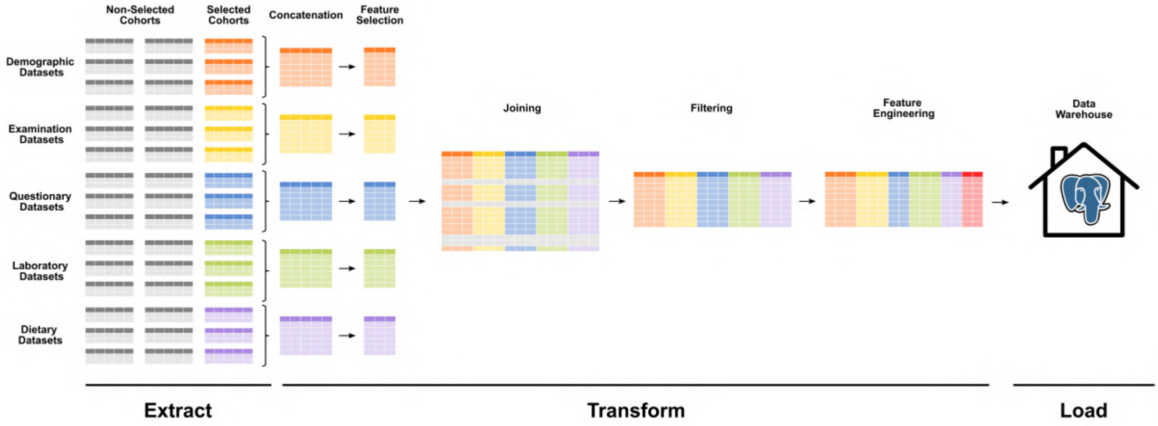

Figure 2. Pipeline followed during the ETL (Extract Transform Load).

1.1 Selecting Variables

As described in the introduction, the NHANES project typically includes between 1,200 and 2,000 variables stored in tabular format across separate files for each type of data (demographic, dietary, examination, laboratory, and questionnaire data) and for each survey cycle. To build effective machine learning models for aerobic fitness estimation, the first step consisted of selecting the most relevant features, since it would be unfeasible to use all the variables available in NHANES. Variable selection can be carried out using two complementary approaches:

1. **Domain knowledge:** leverage expertise in the field, the problem, the business or the sector to perform an initial filter of relevant versus irrelevant variables.

2. **Model-based approaches:** as the first approach may be biased even for highly experienced experts, machine learning methods can be used to identify the most important variables. Examples include:

   - **Using decision trees to rank variables:** decision tree-based models can naturally provide a ranking of variable importance. These models evaluate how much each feature contributes to reducing prediction error, allowing identification of the variables that have the greatest impact on the target outcome.

   - **Univariate models to assess predictive power:** univariate models analyze each predictor independently to evaluate its relationship with the target variable (VO_2max). This method helps to identify features that individually carry meaningful information, serving as a preliminary filter before applying multivariate models.

- **Permutation importance to measure variable impact on model performance:** permutation importance is a model-agnostic technique that measures the effect of shuffling a feature's values on the predictive performance of a trained model. If randomizing a variable significantly decreases model accuracy, it indicates that the variable is important. This method provides a robust way to assess feature relevance and can complement rankings obtained from tree-based models or univariate analysis (Altmann *et al.*, 2010).

In this work, both domain knowledge based on genetic and environmental variables affecting CRF (reviewed in Klevjer *et al.*, 2023), and variable ranking from previously developed models (Liu *et al.*, 2023) have been used. It should be noted that the most precise approach would combine domain expertise with ML-based variable selection as mentioned above. However, selecting a few variables from thousands using ML is an ambitious task that exceeds the scope and objectives of this work. **A subset of 35 NHANES variables has been selected**. The chosen variables are listed in **Table S1** in Supplementary Material.

1.2 Extracting and Transforming the Data

Since NHANES does not provide a REST API for data access, there are two ways to retrieve the datasets: via web scraping or manually. Given that the data for the cohorts of interest rarely undergo updates and will only be downloaded once, the manual approach was chosen. This also avoids potential legal issues related to web scraping the NHANES website. The total size of the raw data is under 100 MB, so storage is not a concern. These datasets, in their original `.xpt` format were stored in the **`data/raw/`** folder of the repository.

1.2.1 Loading Data

The first step consisted of loading the data from `.xpt` files into pandas `dataframes`. For this purpose, a function called `xpt_to_df` was defined, which opens NHANES `.xpt` files (SAS format) as Pandas DataFrames and returns a `{df_names: dfs}` dictionary. 39 `.xpt` files have been opened as pandas `dataframes`. It was verified that the identifier variable `SEQN` follows a sequential order across NHANES cohorts A (1999-2000), B (2001-2002), and C (2003-2004). **A maximum of 31,126 unique records were extracted from the 1999-2004 period**.

1.2.2 Extracting Selected Variables

The previously selected variables (see **Table S1**) were extracted for each cohort, and the cohorts were concatenated. For this purpose, a function called `extract_selected_variables` was defined, which extracts specific NHANES variables from cohort datasets (e.g., `DEMO_A`, `HIQ_B`), including participant ID (`SEQN`) and survey cycle (`SDDSRVYR`), and concatenates them into one DataFrame per source file (e.g., `DEMO`, `HIQ`, etc.). Variables were successfully selected, and 39 dataframes corresponding to 3 cohorts were concatenated into 12 dataframes.

In summary, for example, the demographic datasets `DEMO_A` (cohort 1999-2000), `DEMO_B` (cohort 2001-2002), and `DEMO_C` (cohort 2003-2004), which contain all demographic variables (around 150 variables, as shown in the DEMO oficial documentation), were concatenated into a single dataset called `DEMO`, which contains only the selected demographic variables (see **Table S1**). The same procedure was applied to all other types of data (examination, laboratory, questionnaire, etc.).

1.2.3 Validating Concatenation

To ensure that the concatenation was successful, it is good practice to check that the resulting dataframes have the expected size. For this purpose, a function called `validate_concatenation` was defined, which measures the shape of individual cohort DataFrames (A, B, C) and validates the concatenated DataFrame. The log shows that no rows were lost and that the concatenation was correct. The concatenated dataframes are expected to have at most 31,126 rows, as this is the number of unique IDs detected previously (see section 1.2.1). However, it can be noticed that the dataframe labeled `PAQIAF` (*Physical Activity - Individual Activities*) contains 35,572 rows instead of the expected maximum of 31,126 records. As explained in the NHANES PAQIAF documentation, this occurs because the dataframe includes multiple rows for the same individual if they report engaging in more than one type of activity (basketball, running, swimming, etc.). One row per activity. This highlights the importance of reviewing the processes, as otherwise it would have been difficult to detect.

In order to avoid unnecessarily duplicating rows or data, a new variable was created to combine the three previous ones before joining all the datasets. The variable `physical_activity_time` was created using the variables `PADLEVEL`, `PADTIMES`, and `PADDURAT`. These variables represent the intensity level of the physical activity (1: moderate, 2: vigorous), the number of times the activity was performed in the past 30 days, and the duration of the activity per session, respectively. Using this information, the new variable **`physical_activity_time`** measures the total activity performed per month in minutes. For simplification, it is assumed that 1 minute of vigorous activity is equivalent to 2 minutes of moderate activity, as previously considered by other researchers (Liu *et al.*, 2023).

1.2.4 Joining Datasets

Unique dataframes are now available for each data source (demographic, examination, laboratory, questionnaire, etc.) for the 1999-2004 period with the variables of interest. In order to merge all of them (12) into a single dataframe, a function called `merge_datasets` was defined, which joins all dataframes on `SEQN` (unique ID column) to form a single unified dataframe. Additionally, a function called `descriptors` was defined, which provides an initial report of the information contained in each variable: count, mean, std, min, 25% percentile, 50% percentile, 75% percentile, max, number of unique values, and number of null values.

1.2.5 Filtering the Dataset

The main objective of this project is to predict the VO_2max variable (called `CVDESVO2` in the dataframe, according to NHANES nomenclature). Therefore, **the dataframe has been filtered by VO_2max** and only the records with non-null values have been kept, in total **8,324** records. Although this represents a drastic reduction of the data, it is important to remember that in previous efforts (see **Table 1** in the section "State of the Art") VO_2max was experimentally measured in only dozens or hundreds of cases, making the NHANES project dataset still extremely valuable.

1.2.6 Feature Engineering

1.2.6.1 Creating New Variables

Before storing the data in the data warehouse, the columns will be renamed and some variables will be transformed.

1. The new variable `physical_activity_time` was already created (see section 1.2.3) using the variables `PADLEVEL`, `PADTIMES`, and `PADDURAT`.

2. New variable `education_level` has been created by combining `DMDEDUC2` and `DMDEDUC3`. The original coding of these variables can be found in NHANES DEMO documentation. This new variable is coded as: `0`: *less than high school*; `1`: *high school*; `2`: *greater than high school*.

3. New variable `weight_diff` has been created as the difference between current self-reported weight in pounds (`WHD020`) and self-reported weight 1 year ago in pounds (`WHD050`).

4. New variable `smoker` has been created from questionnaire variables `SMQ620`, `SMQ020`, `SMQ040`, `SMQ680`. Classification: `no`: *individuals who never smoked cigarettes or smoked less than 100 cigarettes in their lifetime*; `yes`: *individuals who smoke now, or used tobacco/nicotine in the last*

5 days; `former` : *individuals who stated they had smoked cigarettes, or smoked at least 100 cigarettes in their lifetime, but currently do not smoke or did not use tobacco/nicotine in the last 5 days*.

5. All variables have been renamed to names that are easier to identify.

1.2.6.2 Re-Encoding Invalid or Special Values

In many datasets, certain numeric codes are used to represent non-informative responses such as "Refused to answer" or "Don't know". These codes are not inherently missing values but should be treated as such to ensure consistent handling during data preprocessing. In this step, these special values have been re-encoded (e.g., `77` , `99`) as proper missing values (`NaN`) to prepare the dataset for downstream imputation or analysis.

1.2.6.3 Handling Missing Data

It is most advisable to handle missing data during the ETL process, rather than simply leaving them as "missing" in the data warehouse." Doing so during ETL has several advantages regarding data homogeneity and quality. There are different ways to handle missing values: **(1). Imputation**: to replace missing values with the mean, median, most frequent value, or similar; **(2). Explicit marking**: to explicitly set 'missing' or -1 for numeric values when imputation is not possible; **(3). Record removal**: when many fields are missing or if those records are not useful.

In this case, it has been decided to impute the values whenever possible using a multivariate method based on the machine learning algorithm K-Nearest Neighbor (KNN): the **KNN Imputer**. The KNN Imputer is a technique for imputing missing values by utilizing the values of the nearest neighbors in the feature space.

- It works by finding the k most similar samples (neighbors) to the missing data point based on the feature values and imputing the missing value using the average (or weighted average) of these neighbors' corresponding values. The default value of k is 5.
- This method is particularly useful, as it captures the patterns and correlations in the data. In this case, the KNN Imputer has been applied to variables with missing values to preserve the structure and correlations of the dataset, improving the quality of the imputation over simpler methods like mean or median imputation.

This approach was chosen for all cases except for `weight_diff` , where it was decided to replace missing values with the mean value, i.e., 0, assuming there was no weight difference over the year. The method selected for preprocessing each variable has been listed in **Table S2** (see Supplementary Material).

1.3 Data Warehouse in PostgreSQL

The objective of this section is to create a clean, structured PostgreSQL database to store biometric data, intended for future use in training various machine learning algorithms. The stored data should serve as a single source of truth.

1.3.1 Connection to PostgreSQL

To interact with the PostgreSQL database, the psycopg2 library has been used, which provides a direct interface between Python and PostgreSQL. A connection was established by providing the required parameters: database name, user, password, host, and port. Once connected, a cursor object was created to execute SQL commands.

1.3.2 Storing the Data

A table named `biometrics` has been created to hold the cleaned dataset. The schema was defined using standard SQL types (e.g., `FLOAT`, `INTEGER`, `TEXT`). After defining the table structure, the `cleaned_df` DataFrame from pandas was converted to a list of tuples and inserted into the table using `psycopg2.extras.execute_values`, which efficiently inserts multiple rows at once. At the end, 8324 rows have been inserted into the 'biometrics' table. The final column names, their descriptions, and the mapping of each variable to the original NHANES variable names can be found in **Table S3** in the Supplementary Material.

In addition to storing the data in PostgreSQL, the cleaned dataset has also been saved as a **`.csv`** file in the **`data/processed/`** folder. This ensures that the dataset is easily accessible directly from the repository for analysis, visualization, or testing purposes, without requiring a connection to the database. To maintain security and reproducibility, database credentials are stored in a separate `.env` file, while a template file `.env.example` is provided in the repository. This allows other users to recreate the PostgreSQL database locally by supplying their own credentials without exposing sensitive information.

1.3.3 Storing Metadata

To enhance the usability and clarity of the dataset, a secondary table named `biometrics_metadata` has been created to store metadata. This table includes information about each variable in the `biometrics` table: the column name, description, data type, additional comments, and source.Thanks to this structured storage in PostgreSQL, the biometric data is now safely stored and accesible for the next sections.

2 Exploratory Data Analysis and Clustering

Before training the machine learning models, it is crucial to understand the structure and relationships within the data, identify key patterns, and transform the dataset into a form optimal for the algorithms. The code of this section can be found in `notebooks/2_EDA_and_Clustering.ipynb` and in `src/eda.py`.

2.1 Importing Data from PostgreSQL Data Warehouse

First of all, the data previously stored in the DataWarehouse NHANES database in the table called `biometrics` is imported using Python's `SQLAlchemy` library. Alternatively, the data can be obtained from the cleaned `biometrics.csv` file in the repository.

2.2 Exploratory Data Analysis (EDA)

2.2.1 Univariate Data Distribution

A visual analysis of all variables based on box-plot and histograms has been performed. The results are exposed in **Figures S1-S4** in Supplementary Material. In addition, a statistical analysis of the continuous numerical variables distributions was performed using the `distribution_analysis_df` function defined in `eda.py` file stored in `src/` folder of the repository. The results can be observed in **Table S4** (Supplementary Material).

2.2.2 Target Variable Distribution (VO2Max)

Based on the statistical analysis, the target variable exhibits a positively skewed (right-tailed) distribution with a high kurtosis, indicating heavy tails (leptokurtic) (see **Figure S5** in Supplementary Material). Additionally, results from the Shapiro-Wilk, Kolmogorov-Smirnov, and D'Agostino tests all strongly reject the null hypothesis of normality. The best-fitting distribution among common candidates is the log-normal, while among all tested distributions, it fits best to a generalized hyperbolic distribution.

Given this significant deviation from normality and the presence of heavy skew, a logarithmic transformation will be applied later in the pipeline to better normalize the distribution and improve the performance of machine learning models.

2.2.3 Multivariate Relationships

The Pearson correlation coefficient has been calculated for the numerical variables of the dataset. The Pearson correlation coefficient measures the strength and direction of the linear relationship between two numerical variables ranging from -1 to 1. The correlation matrix is displayed in **Figure S6** in Supplementary Material. It reveals several important patterns in the relationships between variables:

- `Age` is positively correlated with `education_level`, `blood_pressure`, `cholesterol`, `BMI`, and `waist_perimeter`, suggesting that these health and socioeconomic indicators tend to increase with age.
- `Gender` shows a strong positive correlation with `height` and `VO_2max`, reflecting typical physiological differences.
- Socioeconomic factors like `education_level` and `poverty_ratio` are moderately correlated, as expected, and both show positive associations with health metrics such as `cholesterol` and `blood_pressure`.
- Strong intercorrelations are observed among body composition variables: `BMI`, `waist_perimeter`, and `weight` are highly correlated with each other, indicating they capture overlapping aspects of body size.
- Additionally, there is notable correlation among hematological measures: `red_blood_cell_count`, `hemoglobin`, and `hematocrit`.

Overall, these patterns suggest meaningful relationships that will be useful for predictive modeling. Where multicollinearity might arise—especially among anthropometric measures—dimensionality reduction or variable selection strategies may be necessary.

2.2.4 Linear Dimensionality Reduction (PCA)

Principal Component Analysis (PCA) helps reduce the dimensionality of data by transforming original variables into a smaller set of uncorrelated components that capture most of the variance in the dataset. This simplification improves model efficiency, reduces noise and multicollinearity, and often enhances predictive performance. By visualizing the cumulative explained variance, the minimum number of components needed to retain a desired level of information (e.g., 95%) can be selected, making the dataset easier to analyze and interpret while preserving its essential structure.

As shown in **Figure S7**, in this case the PCA captures very little variance in just two components, indicating that the data's structure is not well represented by linear combinations of variables alone. Therefore, we will explore nonlinear dimensionality reduction techniques like t-SNE, which are better suited to capture complex patterns and relationships in the data for tasks such as clustering or outlier detection.

2.2.5 Non-linear Dimensionality Reduction (t-SNE)

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a non-linear dimensionality reduction technique particularly effective at preserving local structure in high-dimensional data. It maps complex relationships into a low-dimensional space suitable for visualization. In this case (see **Figure S8** in Supplementary Material), t-SNE has been used to project the standardized dataset into two dimensions, revealing potential groupings that are not captured by linear methods like PCA.

2.2.6 Clustering Analysis (DBSCAN)

To further investigate the structure revealed by t-SNE, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) has been applied, a clustering algorithm that identifies groups based on data density without requiring a predefined number of clusters. DBSCAN successfully identifies 2 distinct clusters in the t-SNE projection. These clusters may correspond to meaningful subgroups in the data. To interpret the components more deeply, the average feature values within each cluster have been evaluated, which helps to understand which variables are most influential in shaping the structure observed in the t-SNE plot. The clustering analysis can be found in **Figure S9** in Supplementary Material. As observed in the Figure, the variables `gender` and `poverty_ratio` appear to have a strong influence on the structure of the components. These visualizations highlight distinct patterns and separations in the 2D embedding, suggesting that these demographic features are key drivers of variation within the dataset. This insight can guide further analysis and help explain the underlying composition of the identified clusters.

3 Aerobic Fitness Prediction Using Ensemble Machine Learning Models

The code explained in this section can be found in `notebooks/3_ML_models.ipynb`

3.1 Data Preparation for Modeling

In this section, the data is prepared for subsequent use in ML. To achieve this:

- Informative variables used only for data storage but irrelevant for modeling, such as the ID or cohort number, have been removed.

- One-hot encoding has been performed using pandas' `get_dummies()` method to transform categorical variables.

Based on the EDA results (Skewness, Kurtosis and Distribution Fit), some data transformations has been applied (see **Table S5**) in Supplementary Material).

In addition, before applying machine learning models, the data should be normalized using Z-score scaling. This means that each numerical feature was transformed by subtracting its mean and dividing by its standard deviation, resulting in features with a mean of 0 and a standard deviation of 1.This normalization is important because it prevents features with larger scales from dominating the modeling process. It also helps many algorithms converge faster and improves overall model performance and stability. The Z-score distribution of numeric variables before and after normalization have been displayed in supplementary **Figure S10**.

The dataset now is suitable for Regression Machine Learning modeling for several reasons:

- **Normalized numerical variables:** All numerical features have a mean close to 0 and a standard deviation close to 1, which facilitates training and improves convergence in many machine learning models.
- **Encoded categorical variables:** Categorical variables have been transformed into binary (one-hot) format, which is the appropriate format for use in regression.
- **Sufficient sample size:** With over 8,000 rows, the dataset provides a solid foundation to train regression models with multiple predictors.
- **Variety of features:** The dataset includes demographic, biometric, and lifestyle variables, which can help explain the target variable well (assuming it is properly defined).

3.2 Machine Learning Models

Before applying any machine learning models, the dataset is split into training and testing subsets. It is crucial to perform this split prior to feature scaling to avoid *data leakage*, a situation where information from the test set inadvertently influences the model during training. If scaling were applied to the entire dataset before splitting, statistical properties of the test set (such as the mean and standard deviation) would contribute to the transformation of the training data, artificially inflating model performance and reducing the validity of evaluation metrics. By splitting first, scaling parameters are computed exclusively on the training set and then applied to the test set, ensuring that the model is evaluated on truly unseen data.

In this study, three types of regression models were applied: a traditional Multiple Linear Regression, a Bagging ensemble model (Random Forest), and a Boosting ensemble model (XGBoost). Each model was trained following the following pipeline to ensure reproducibility, fairness, and robustness of results:

3.2.1 Splitting the Dataset

The dataset was divided into training and testing subsets to evaluate the performance of the machine learning models on unseen data. **A split ratio of 80% for training and 20% for testing was adopted**, ensuring that the models have sufficient data for learning while retaining a representative portion for unbiased evaluation.

3.2.2 Feature scaling

In order to avoid data leakage, after splitting the dataset, the numerical features were standardized using Z-score normalization, resulting in a mean close to 0 and a standard deviation close to 1 for each variable. This transformation ensures that all features contribute equally to the modeling process and facilitates faster convergence and improved stability of machine learning algorithms.

3.3 Selected Machine Learning Algorithms

As discussed in previous sections of this document, three machine learning algorithms were selected to predict Cardiorespiratory Fitness (CRF) through the estimation of VO_2max. The selected models cover both traditional regression and ensemble learning approaches, providing a solid foundation for comparison in terms of predictive performance and generalization ability.

3.3.0.1 Multiple Linear Regression

First, a Multiple Linear Regression (MLR) model was trained to serve as a baseline. MLR is one of the most classical approaches in regression modeling, based on fitting a linear function of the explanatory variables that best predicts the target variable by minimizing the residual sum of squares. Although MLR has limited flexibility compared to more advanced models, it provides interpretability and allows for an initial benchmark to evaluate improvements obtained with ensemble methods.

3.3.0.2 Random Forest

Secondly, a Random Forest model was trained. Random Forest is an ensemble method based on bagging, where multiple decision trees are constructed using random subsets of features and samples. The predictions of these trees are then aggregated, typically by averaging, to produce a more stable and robust final prediction. Random Forest is known for its ability to handle high-dimensional data, reduce overfitting compared to single decision trees, and capture complex nonlinear relationships.

3.3.0.3 XGBoost

Finally, an XGBoost model was implemented. XGBoost is a gradient boosting framework that builds decision trees sequentially, where each tree attempts to correct the errors of its predecessors. It includes several optimizations such as regularization, shrinkage, and efficient handling of sparse data, making it one of the most powerful algorithms for structured/tabular data. XGBoost often achieves state-of-the-art results in regression and classification tasks, albeit at the cost of higher complexity and more careful hyperparameter tuning.

3.3.1 Coarse-to-Fine Hyperparameter Tuning Search

The coarse-to-fine hyperparameter tuning strategy was applied consistently across the three algorithms. In the coarse phase, a broader parameter space was explored using RandomizedSearchCV, allowing for efficient sampling of potential configurations. This approach has allowed for significant computational time savings given the limited computational resources available. In the fine phase, a narrower search was conducted around the best parameters identified in the coarse search using GridSearchCV.

3.3.2 Models Training

Each model was trained on the training set using the best hyperparameters obtained from the fine search. This ensured that the models were optimized for predictive performance while avoiding overfitting. The training was performed exclusively on the training data, while the test set was kept aside for the final evaluation to provide an unbiased measure of generalization.

4 Deployment

After selecting the best-performing model based on the evaluation results, a deployment stage was carried out to make the prediction system accessible to end-users. For this purpose, a web-based application was developed using *Streamlit*, an open-source Python framework designed for building interactive data applications with minimal overhead. Streamlit provides a simple yet powerful way to transform Python scripts into shareable web interfaces, which makes it particularly well-suited for rapid prototyping and deployment of machine learning solutions.

In this project, Streamlit was used to create a user-friendly interface where individuals can input their biometric and lifestyle data to obtain an estimation of their VO_2max. The application incorporates the same preprocessing steps and transformations that were applied during model training, ensuring consistency between training and inference. Once the input features are collected, they are transformed, scaled, and passed through the trained XGBoost model to generate the prediction. The output is then back-transformed to the original VO_2max scale and presented to the user alongside a categorical fitness interpretation (e.g., Poor, Fair, Good, Excellent, Superior).

The use of Streamlit ensures that the predictive model is not limited to a research environment but can be easily accessed and interacted with in real time. This approach enables practical usage, fosters reproducibility, and provides a clear demonstration of how machine learning models can be translated into actionable tools for end-users.

Evaluation

The evaluation of the models was carried out on the test set using multiple regression metrics: the coefficient of determination (R^2), the Root Mean Squared Error (RMSE), the Mean Squared Error (MSE), the Mean Absolute Error (MAE), and the Mean Absolute Percentage Error (MAPE). The following provides a brief explanation of each metric, while the actual evaluation results are presented in the **Results** section.

- **Coefficient of determination (R^2):** Measures the proportion of variance in the target variable explained by the model. It provides an indication of overall fit, with values closer to 1 indicating better explanatory power.
$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

- **Root Mean Squared Error (RMSE):** Represents the square root of the average squared differences between predicted and actual values. RMSE is expressed in the same units as the target variable and is useful for interpreting the magnitude of prediction errors.
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- **Mean Squared Error (MSE):** Calculates the average of the squared differences between predicted and actual values. It penalizes larger errors more heavily, making it sensitive to outliers.
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- **Mean Absolute Error (MAE):** Computes the average of the absolute differences between predicted and actual values. Unlike MSE, it is less sensitive to outliers and provides a more robust measure of average prediction error.
$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- **Mean Absolute Percentage Error (MAPE):** Expresses the prediction error as a percentage of the actual values, offering an intuitive interpretation of the model's accuracy in relative terms. However, it may be unstable when actual values are close to zero.
$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

Although all metrics were used to provide a comprehensive assessment of model performance, the MAE was specifically used during the coarse-to-fine hyperparameter grid search. This choice was motivated by the objective of minimizing the average absolute deviation between predicted and actual

VO_2max values, thereby ensuring robust performance across the entire range of observations. Unlike MSE or RMSE, MAE is less sensitive to extreme outliers, which is particularly important in biomedical datasets where atypical physiological measurements can occur. By optimizing for MAE, the models were guided to produce predictions that are consistently close to the true values, improving interpretability and practical utility for end-users, especially in applications such as individualized fitness assessment through the deployed Streamlit service.

Results

1 Model's Hyperparameter Search - Results

As defined in the previous section "Proposed Solution - Coarse-to-Fine Hyperparameter Tuning Search", cross-validation was employed for the hyperparameter search, using a coarse-to-fine hyperparameter tuning approach. **Table 3** presents the best combination of hyperparameters identified for the three models.

Model	Parameters	Mean Test cross-val MAE
MLR	`fit_intercept: True`, `positive: False`	0.147014
Random Forest	`n_estimators: 450`, `max_features: 0.8`, `max_depth: 25`, `min_samples_leaf: 20`, `min_samples_split: 9`	0.147246
XGBoost	`colsample_bytree: 1`, `gamma: 0.25`, `learning_rate: 0.012`, `max_depth: 4`, `n_estimators: 900`, `reg_alpha: 0.01`, `reg_lambda: 3.5`, `subsample: 0.8`	0.145229

Table 3. Comparison of three models. Each row shows the parameter settings and performance metrics for a model. Metrics include mean and standard deviation of test R^2, negative MSE, negative MAE, and MAPE.

2 Model Performance Results

After training the three selected machine learning models using the optimized hyperparameters from the coarse-to-fine search, their performance was evaluated on the test set. As explained in the Evaluation Section, the evaluation metrics include R^2, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), providing a comprehensive view of predictive accuracy and error distribution. **Table 4** presents the results for the three models.

Model	Test Set Metrics
MLR	R^2: **0.2953** MSE: **0.0378** RMSE: **0.1944** MAE: 0.1479 MAPE: 0.0398
Random Forest	R^2: 0.2927 MSE: 0.0379 RMSE: 0.1947 MAE: 0.1480 MAPE: 0.0399
XGBoost	R^2: 0.2901 MSE: 0.0381 RMSE: 0.1951 MAE: **0.1464** MAPE: **0.0393**

Table 4. Test set evaluation metrics for the three selected machine learning models. Multiple Linear Regression (MLR), Random Forest (RF), and XGBoost (XGB). The metrics reported are R^2, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), providing a comprehensive view of predictive performance. Higher R^2 values indicate better model fit, while lower values of the error metrics indicate higher predictive accuracy. Best values among the three models are highlighted in bold.

These results provide a detailed overview of the predictive performance of each model on the test set. A comprehensive discussion and interpretation of these metrics, including comparisons with previous studies and an analysis of the strengths and limitations of each modeling approach, is presented in the Discussion Section.

Discussion, Future Work and Conclusions

Discussion

In this work, a comprehensive datawarehouse was generated through an ETL process using data from the NHANES project. Subsequently, three machine learning models for regression (Multiple Linear Regression (MLR), Random Forest (RF), and XGBoost (XGB)) were trained to predict Cardiorespiratory fitness by predicting VO_2max.

The performance of the three machine learning models was evaluated on the test set, with metrics summarized in **Table 4** of the Results section. All three models achieved similar predictive performance, with R^2 values ranging from 0.2927 to 0.2953. The error metrics (MSE, RMSE, MAE, and MAPE) were also closely aligned, indicating comparable predictive accuracy across models. Among them, MLR slightly outperformed the ensemble methods in terms of R^2, while XGBoost achieved the lowest MAE and MAPE, suggesting marginally better predictive precision for individual observations.

Notably, the use of ensemble methods (RF and XGB) did not substantially outperform the simpler MLR model, which indicates that the relationships between the predictor variables and VO_2max may be predominantly linear in nature, or that the available predictors capture only a limited portion of the underlying variance. Nonetheless, XGBoost slightly reduced the mean absolute error, which could be advantageous in applications requiring more precise individual-level predictions.

When compared with previous non-exercise algorithms (see **Table 2** and Liu *et al.*, 2023), our models exhibit comparable performance despite using a smaller dataset of approximately 8,000 participants. Liu et al.'s best-performing LightGBM model explained 32% of the variance in VO_2max, slightly higher than our models ($R^2 \sim 0.29$). However, such comparisons should be interpreted with caution, as slight differences in data preprocessing can influence R^2, even when the source dataset is the same. The relatively modest R^2 values observed in both this study and prior literature highlight the inherent difficulty of predicting VO_2max from non-exercise variables alone.

An analysis of feature importance from the XGBoost model provides additional insights into the predictors of VO_2max (see **Figure S11** in Supplementary Materials). The most influential variable was *gender* (importance = 0.234), followed by *body_fat_percent* (0.143) and *pulse_rate* (0.038). Other notable contributors included *waist_perimeter*, *education_level*, and *bmi*, among others. These results suggest that both physiological factors (e.g., body composition and cardiovascular indicators) and demographic or lifestyle variables (e.g., gender, education level, ethnicity, and smoking status) play significant roles in predicting cardiorespiratory fitness. Understanding the relative importance of these features can help guide future studies and interventions aimed at improving VO_2max, as well as inform the selection of variables for simplified predictive models.

Finally, it is worth to mention that a key limitation of this study is the relatively small dataset. Although processing approximately 8,000 patients represents a considerable effort, it remains insufficient to reach ideal predictive performance. As a result, the R^2 values remain far from theoretical upper limits, emphasizing the challenge of modeling VO_2max with limited information. Additionally, ensemble methods did not substantially outperform linear regression, further underscoring the constraints imposed by dataset size and variable selection.

Future Work

Future research could focus on several avenues to improve predictive performance:

- Expanding the hyperparameter search space and performing also a grid search for pre-processing steps (scaling methods, feature selection, etc.), which was not feasible in the current study due to computational limitations.
- Identifying larger datasets or encouraging health organizations to release or consolidate additional data to enhance model training and generalization.
- Enhancing the Streamlit service by integrating more contextual information related to VO_2max, such as recommended exercise regimes, health risk interpretation, or lifestyle guidance tailored to the predicted fitness level.
- Exploring additional non-linear or interaction effects among predictor variables, or incorporating new predictors such as physiological or wearable sensor data to capture more variance.

Conclusions

The main conclusions of this work are summarized as follows:

- The creation of a centralized data warehouse in PostgreSQL proved successful and greatly facilitated the machine learning pipeline.
- Among the three models, XGBoost achieved the best performance, although differences with MLR and RF were minimal, indicating that simpler linear relationships dominate the prediction of VO_2max in the available dataset.
- The best-performing model, XGBoost, has been deployed in a Streamlit service to allow end-users to input their biometric data and obtain an estimated VO_2max value with a clear interpretation of its meaning in terms of aerobic fitness.
- Overall, this study demonstrates the feasibility of estimating VO_2max using non-exercise variables, the practical utility of deploying the model for end-users, and the potential for further improvements through larger datasets and more extensive model exploration.

References

References

Abut, F., Akay, M. F., & George, J. (2016). Developing new VO2max prediction models from maximal, submaximal and questionnaire variables using support vector machines combined with feature selection. Computers in biology and medicine, 79, 182–192. https://doi.org/https://doi.org/10.1016/j.compbiomed.2016.10.018

Abut, F., Akay, M. F., & George, J. (2019). A robust ensemble feature selector based on rank aggregation for developing new vo\textsubscript {2} max prediction models using support vector machines. Turkish Journal of Electrical Engineering and Computer Sciences, 27(5), 3648–3664. https://doi.org/https://journals.tubitak.gov.tr/elektrik/vol27/iss5/27/

Akay, M. F., Bozkurt, O., Cetin, E., & Yarim, I. (2018). Multiple linear regression-based physical fitness prediction models for Turkish secondary school students. New Trends and Issues Proceedings on Humanities and Social Sciences, 5(4), 58–64.

Akay, M. F., Çetin, E., Yarım, İ., Bozkurt, Ö., & Özçiloğlu, M. M. (2017). Development of novel maximal oxygen uptake prediction models for turkish college students using machine learning and exercise data. 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN), 186–189. https://doi.org/https://ieeexplore.ieee.org/document/8319382/

Akay, M., Cetin, E., Yarim, I., & ÖZÇİLOĞLU, M. (2017). New prediction models for the maximal oxygen uptake of college-aged students using non-exercise data. New Trends and Issues Proceedings on Humanities and Social Sciences, 4(4).

Akay, M., Ozciloglu, M., Çetin, E., Yarim, I., & Daneshvar, S. (2018). Estimating the maximal oxygen uptake with new prediction models for college-aged students using feature selection algorithm. New Trends and Issues Proceedings on Humanities and Social Sciences, 5(4).

Altmann, A., Toloşi, L., Sander, O., & Lengauer, T. (2010). Permutation importance: A corrected feature importance measure. Bioinformatics, 26(10), 1340–1347. https://doi.org/https://doi.org/10.1093/bioinformatics/btq134

Ashfaq, A., Cronin, N., & Müller, P. (2022). Recent advances in machine learning for maximal oxygen uptake (VO2 max) prediction: A review. Informatics in Medicine Unlocked, 28, 100863. https://doi.org/https://doi.org/10.1016/j.imu.2022.100863

Beltrame, T., Amelard, R., Villar, R., Shafiee, M. J., Wong, A., & Hughson, R. L. (2016). Estimating oxygen uptake and energy expenditure during treadmill walking by neural network analysis of easy-to-obtain inputs. Journal of Applied Physiology, 121(5), 1226–1233. https://doi.org/https://doi.org/10.1152/japplphysiol.00600.2016

Beltrame, T., Amelard, R., Wong, A., & Hughson, R. L. (2017). Prediction of oxygen uptake dynamics by machine learning analysis of wearable sensors during activities of daily living. Scientific reports, 7(1), 45738. https://doi.org/https://doi.org/10.1038/srep45738

Beutner, F., Ubrich, R., Zachariae, S., Engel, C., Sandri, M., Teren, A., & Gielen, S. (2015). Validation of a brief step-test protocol for estimation of peak oxygen uptake. European journal of preventive cardiology, 22(4), 503–512. https://doi.org/https://doi.org/10.1177/2047487314533216

Bohannon, R. W., Bubela, D. J., Wang, Y.-C., Magasi, S. S., & Gershon, R. C. (2015). Six-minute walk test vs. three-minute step test for measuring functional endurance. The Journal of Strength & Conditioning Research, 29(11), 3240–3244. https://doi.org/https://doi.org/10.1519/JSC.0000000000000253

Borror, A., Mazzoleni, M., Coppock, J., Jensen, B. C., Wood, W. A., Mann, B., & Battaglini, C. L. (2019). Predicting oxygen uptake responses during cycling at varied intensities using an artificial neural network. Biomedical Human Kinetics, 11(1), 60–68.

Caspersen, C. J., Powell, K. E., & Christenson, G. M. (1985). Physical activity, exercise, and physical fitness: Definitions and distinctions for health-related research. Public health reports, 100(2), 126. https://doi.org/https://pmc.ncbi.nlm.nih.gov/articles/PMC1424733/

DeFina, L. F., Willis, B. L., Radford, N. B., Gao, A., Leonard, D., Haskell, W. L., Weiner, M. F., & Berry, J. D. (2013). The association between midlife cardiorespiratory fitness levels and later-life dementia: A cohort study. Annals of internal medicine, 158(3), 162–168. https://doi.org/https://doi.org/10.7326/0003-4819-158-3-201302050-00005

Dinçer, Ö., AKAY, M., ÇETİN, E., YARIM, İ., & DANESHVAR, S. (2016). New Prediction Equations for Estimating the Maximal Oxygen Consumption of College aged Students Using Hybrid Data. ÇÜ Mühendislik Mimarlık Fak. Dergisi.

Earnest, C. P., Artero, E. G., Sui, X., Lee, D.-c., Church, T. S., & Blair, S. N. (2013). Maximal estimated cardiorespiratory fitness, cardiometabolic risk factors, and metabolic syndrome in the aerobics center longitudinal study. Mayo Clinic Proceedings, 88(3), 259–270. https://doi.org/https://doi.org/10.1016/j.mayocp.2012.11.006

for Health Statistics, N. C. (2005). The NHANES Cardiovascular Fitness Procedure Manual. 2005. https://doi.org/https://wwwn.cdc.gov/nchs/data/nhanes/public/2005/manuals/CV.pdf

Foster, C., Jackson, A. S., Pollock, M. L., Taylor, M. M., Hare, J., Sennett, S. M., Rod, J. L., Sarwar, M., & Schmidt, D. H. (1984). Generalized equations for predicting functional capacity from treadmill performance. American heart journal, 107(6), 1229–1234. https://doi.org/https://doi.org/10.1016/0002-8703(84)90282-5

Heil, D. P., Freedson, P. S., Ahlquist, L. E., Price, J., & Rippe, J. M. (1995). Nonexercise regression models to estimate peak oxygen consumption. Medicine and science in sports and exercise, 27(4).

Hill, A. V., Lupton, H., et al. (1923). Muscular exercise, lactic acid, and the supply and utilization of oxygen. QJ Med, 16(62), 135–71. https://doi.org/http://www.jstor.org/stable/81203

Ho, C.-A., Yeh, H.-C., Lau, H.-T., Chang, E.-Y., Hsu, C.-W., Chang, C.-H., Huang, C.-C., Chien, W.-S. C., & Ho, C.-S. (2025). Establish VO2max prediction models based on exercise and body parameters from the step test. International Journal of Medical Sciences, 22(11), 2676. https://doi.org/https://doi.org/10.7150/ijms.109977

Hong, S. H., Yang, H. I., Kim, D.-I., Gonzales, T. I., Brage, S., & Jeon, J. Y. (2019). Validation of submaximal step tests and the 6-min walk test for predicting maximal oxygen consumption in young and healthy participants. International journal of environmental research and public health, 16(23), 4858. https://doi.org/https://doi.org/10.3390/ijerph16234858

Jackson, A. S., Blair, S. N., Mahar, M. T., Wier, L. T., Ross, R. M., & Stuteville, J. E. (1990). Prediction of functional aerobic capacity without exercise testing. Medicine and science in sports and exercise, 22(6), 863–870. https://doi.org/https://doi.org/10.1249/00005768-199012000-00021

Jackson, A. S., Sui, X., O'Connor, D. P., Church, T. S., Lee, D.-c., Artero, E. G., & Blair, S. N. (2012). Longitudinal cardiorespiratory fitness algorithms for clinical settings. American journal of preventive medicine, 43(5), 512–519. https://doi.org/https://doi.org/10.1016/j.amepre.2012.06.032

Jurca, R., Jackson, A. S., LaMonte, M. J., Morrow Jr, J. R., Blair, S. N., Wareham, N. J., Haskell, W. L., van Mechelen, W., Church, T. S., Jakicic, J. M., et al. (2005). Assessing cardiorespiratory fitness without performing exercise testing. American journal of preventive medicine, 29(3), 185–193. https://doi.org/https://doi.org/10.1016/j.amepre.2005.06.004

Kaminsky, L. A., Arena, R., Ellingsen, Ø., Harber, M. P., Myers, J., Ozemek, C., & Ross, R. (2019). Cardiorespiratory fitness and cardiovascular disease-the past, present, and future. Progress in cardiovascular diseases, 62(2), 86–93. https://doi.org/https://doi.org/10.1016/j.pcad.2019.01.002

Kaya, K., AKAY, M., ÇETİN, E., & YARIM, İ. (2016). Development of new prediction models for maximal oxygen uptake using artificial intelligence methods.

Klevjer, M., Nordeidet, A. N., & Bye, A. (2023). The genetic basis of exercise and cardiorespiratory fitness–relation to cardiovascular disease. Current Opinion in Physiology, 33, 100649. https://doi.org/https://doi.org/10.1016/j.cophys.2023.100649

Kodama, S., Saito, K., Tanaka, S., Maki, M., Yachi, Y., Asumi, M., Sugawara, A., Totsuka, K., Shimano, H., Ohashi, Y., et al. (2009). Cardiorespiratory fitness as a quantitative predictor of all-cause mortality and cardiovascular events in healthy men and women: A meta-analysis. Jama, 301(19), 2024–2035. https://doi.org/https://jamanetwork.com/journals/jama/fullarticle/1108396

Lakka, T. A., Venalainen, J. M., Rauramaa, R., Salonen, R., Tuomilehto, J., & Salonen, J. T. (1994). Relation of leisure-time physical activity and cardiorespiratory fitness to the risk of acute myocardial infarction in men. New England Journal of Medicine, 330(22), 1549–1554. https://doi.org/https://www.nejm.org/doi/full/10.1056/NEJM199406023302201

Lang, J. J., Phillips, E. W., Orpana, H. M., Tremblay, M. S., Ross, R., Ortega, F. B., Silva, D. A. S., & Tomkinson, G. R. (2018). Field-based measurement of cardiorespiratory fitness to evaluate physical activity interventions. Bulletin of the World Health Organization, 96(11), 794. https://doi.org/https://pmc.ncbi.nlm.nih.gov/articles/PMC6239007/pdf/BLT.18.213728.pdf

Laukkanen, J., Rauramaa, R., Salonen, J., & Kurl, S. (2007). The predictive value of cardiorespiratory fitness combined with coronary risk evaluation and the risk of cardiovascular and all-cause death. Journal of internal medicine, 262(2), 263–272. https://doi.org/https://doi.org/10.1111/j.1365-2796.2007.01807.x

Lee, D.-c., Sui, X., Church, T. S., Lee, I.-M., & Blair, S. N. (2009). Associations of cardiorespiratory fitness and obesity with risks of impaired fasting glucose and type 2 diabetes in men. Diabetes care, 32(2), 257–262. https://doi.org/https://doi.org/10.2337/dc08-1377

Liu, Y., Herrin, J., Huang, C., Khera, R., Dhingra, L. S., Dong, W., Mortazavi, B. J., Krumholz, H. M., & Lu, Y. (2023). Nonexercise machine learning models for maximal oxygen uptake prediction in national population surveys. Journal of the American Medical Informatics Association, 30(5), 943–952. https://doi.org/https://doi.org/10.1093/jamia/ocad035

Matthews, C. E., Heil, D. P., Freedson, P. S., & Pastides, H. (1999). Classification of cardiorespiratory fitness without exercise testing. Medicine and science in sports and exercise, 31(3), 486–493. https://doi.org/https://doi.org/10.1097/00005768-199903000-00019

of Sports Medicine, A. C., et al. (2013). ACSM's guidelines for exercise testing and prescription.

Przednowek, K., Barabasz, Z., Zadarko-Domaradzka, M., Przednowek, K. H., Nizioł-Babiarz, E., Huzarski, M., Sibiga, K., Dziadek, B., & Zadarko, E. (2018). Predictive modeling of VO2max based on 20 m shuttle run test for young healthy people. Applied Sciences, 8(11), 2213. https://doi.org/https://doi.org/10.3390/app8112213

Raghuveer, G., Hartz, J., Lubans, D. R., Takken, T., Wiltz, J. L., Mietus-Snyder, M., Perak, A. M., Baker-Smith, C., Pietris, N., Edwards, N. M., et al. (2020). Cardiorespiratory fitness in youth: An important marker of health: A scientific statement from the American Heart Association. Circulation, 142(7), e101–e118. https://doi.org/https://doi.org/10.1161/CIR.0000000000000866

Sartor, F., Vernillo, G., De Morree, H. M., Bonomi, A. G., La Torre, A., Kubis, H.-P., & Veicsteinas, A. (2013). Estimation of maximal oxygen uptake via submaximal exercise testing in sports, clinical, and home settings. Sports medicine, 43(9), 865–873. https://doi.org/https://doi.org/10.1007/s40279-013-0068-3

Shandhi, M. M. H., Bartlett, W. H., Heller, J. A., Etemadi, M., Young, A., Plötz, T., & Inan, O. T. (2020). Estimation of instantaneous oxygen uptake during exercise and daily activities using a wearable cardio-electromechanical and environmental sensor. IEEE journal of biomedical and health informatics, 25(3), 634–646. https://doi.org/https://ieeexplore.ieee.org/abstract/document/9143414

Wier, L. T., Jackson, A. S., Ayers, G. W., & Arenare, B. (2006). Nonexercise models for estimating vo2max with waist girth, percent fat, or bmi. Medicine and science in sports and exercise, 38(3), 555–561. https://doi.org/https://doi.org/10.1249/01.mss.0000193561.64152

Zignoli, A., Fornasiero, A., Ragni, M., Pellegrini, B., Schena, F., Biral, F., & Laursen, P. B. (2020). Estimating an individual's oxygen uptake during cycling exercise with a recurrent neural network trained from easy-to-obtain inputs: A pilot study. PLoS One, 15(3), e0229466. https://doi.org/https://doi.org/10.1371/journal.pone.0229466

Supplementary Material

Number	NHANES Variable Name	Description	Source Dataset (File)
1	`SEQN`	Unique ID. Participant's identifier number	All
2	`SDDSRVYR`	Cohort. Survey cycle (1: 1999-2001; 2: 2001-2003;...)	Demographics (DEMO.xpt)
3	`RIDAGEYR`	Participant's age in years	Demographics (DEMO.xpt)
4	`RIAGENDR`	Gender - Biological sex	Demographics (DEMO.xpt)
5	`RIDRETH1`	Self-reported race/ethnic group	Demographics (DEMO.xpt)
6	`INDFMPIR`	Income-to-poverty threshold ratio	Demographics (DEMO.xpt)
7	`INDFMINC`	Annual family income	Demographics (DEMO.xpt)
8	`DMDEDUC2`	Highest level of education achieved (Adults 20+)	Demographics (DEMO.xpt)
9	`DMDEDUC3`	Highest level of education achieved (Children/Youth 6–19)	Demographics (DEMO.xpt)
10	`HID010`	Coverage by any health insurance	Questionnaire (HIQ.xpt)
11	`SMQ620`	Ever tried cigarette smoking (Y/N)	Questionnaire (SMQMEC.xpt)
12	`SMQ020`	Smoked at least 100 cigarettes in life	Questionnaire (SMQ.xpt)
13	`SMQ040`	Smoke cigarettes (Y/N)	Questionnaire (SMQ.xpt)
14	`SMD680`	Used tobacco/nicotine last 5 days? (Y/N)	Questionnaire (SMQMEC.xpt)
15	`SMQ680`	Used tobacco/nicotine last 5 days? (Y/N) (since 2001)	Questionnaire (SMQMEC.xpt)
16	`PADLEVEL`	Activity level	Questionnaire (PAQIAF.xpt)
17	`PADTIMES`	Number of times did activity in past 30 days	Questionnaire (PAQIAF.xpt)
18	`PADDURAT`	Average duration of activity (minutes)	Questionnaire (PAQIAF.xpt)
19	`WHD020`	Current self-reported weight (pounds)	Questionnaire (WHQ.xpt)
20	`WHD050`	Self-reported weight 1 year ago (pounds)	Questionnaire (WHQ.xpt)
21	`BPXSY1`	Systolic blood pressure (mm Hg)	Examination (BPX.xpt)
22	`BPXDI1`	Diastolic blood pressure (mm Hg)	Examination (BPX.xpt)
23	`BPXPLS`	Pulse rate (beats per minute)	Examination (BPX.xpt)
24	`BMXBMI`	Body Mass Index (kg/m^2)	Examination (BMX.xpt)
25	`BMXWAIST`	Waist circumference (cm)	Examination (BMX.xpt)
26	`BMXWT`	Weight (kg)	Examination (BMX.xpt)
27	`BMXHT`	Height (cm)	Examination (BMX.xpt)
28	`BIDPFAT`	Estimated percent body fat	Examination (BIX.xpt)
29	`LBXTC`	Total serum cholesterol (mg/dL)	Laboratory (L13.xpt)
30	`LBXRBCSI`	Red cell count SI	Laboratory (L25.xpt)
31	`LBXHGB`	Hemoglobin (g/dL)	Laboratory (L25.xpt)
32	`LBXHCT`	Hematocrit (%)	Laboratory (L25.xpt)
33	`LBXMCHSI`	Mean cell hemoglobin (pg)	Laboratory (L25.xpt)
34	`LBXRDW`	Red cell distribution width (%)	Laboratory (L25.xpt)
35	`CVDESVO2`	**VO2max** (ml/kg/min)	Examination (CXV.xpt)

Table S1. Selected NHANES variables for CRF prediction. The table includes the NHANES variables selected for predicting VO$_2$max. More information is available at available at NHANES.

Column	Type	Selected Method
`id`	Identifier	None
`cohort`	Ordinal categorical	None
`age`	Continuous	None
`gender`	Binary	None
`ethnic_group`	Categorical	None
`poverty_ratio`	Continuous	**KNN Imputer**
`family_income`	Ordinal discrete	**KNN Imputer** + discrete transformation
`health_insurance`	Binary	**KNN Imputer** + binary transformation
`physical_activity_time`	Continuous	**KNN Imputer**
`systolic_bp`	Continuous	**KNN Imputer**
`diastolic_bp`	Continuous	**KNN Imputer**
`pulse_rate`	Continuous	None
`bmi`	Continuous	**KNN Imputer**
`waist_perimeter`	Continuous	**KNN Imputer**
`body_fat_percent`	Continuous	**KNN Imputer**
`weight`	Continuous	**KNN Imputer**
`height`	Continuous	**KNN Imputer**
`cholesterol`	Continuous	**KNN Imputer**
`red_blood_cell_count`	Continuous	**KNN Imputer**
`hemoglobin`	Continuous	**KNN Imputer**
`hematocrit`	Continuous	**KNN Imputer**
`mean_cell_hemoglobin`	Continuous	**KNN Imputer**
`red_cell_distribution_width`	Continuous	**KNN Imputer**
`vo2max`	Continuous	None
`education_level`	Ordinal discrete	None
`weight_diff`	Continuous	**Assign 0 to missing values**

Table S2. Selected NHANES variables for preprocessing. Column names, type of data, and selected methods to handle missing values or transformations.

Number	Variable Name	Description	NHANES Variable(s) Name
1	`id`	Participant's unique identifier number	`SEQN`
2	`cohort`	Survey cycle	`SDDSRVYR`
3	`age`	Participant's age in years	`RIDAGEYR`
4	`gender`	Biological sex	`RIAGENDR`
5	`ethnic_group`	Self-reported race/ethnic group	`RIDRETH1`
6	`poverty_ratio`	Income-to-poverty threshold ratio	`INDFMPIR`
7	`family_income`	Annual family income	`INDFMINC`
8	`education_level`	Highest level of education achieved (adult/adolescent)	`DMDEDUC2`, `DMDEDUC3`
9	`health_insurance`	Coverage by any health insurance	`HID010`
10	`smoker`	Multiple variables on tobacco use history	`SMQ620`, `SMQ020`, `SMQ040`, `SMD680`
11	`physical_activity_time`	Total time of moderate-intensity activity per week	`PAD200`, `PAD320`, `PADLEVEL`, `PADTIMES`, `PADDURAT`
12	`weight_diff`	Weight difference from last year	`WHD020`, `WHD050`
13	`systolic_bp`	Systolic blood pressure (mm Hg)	`BPXSY1`
14	`diastolic_bp`	Diastolic blood pressure (mm Hg)	`BPXDI1`
15	`pulse_rate`	Pulse rate (beats per minute)	`BPXPLS`
16	`bmi`	Body Mass Index (kg/m2̂)	`BMXBMI`
17	`waist_perimeter`	Waist circumference (cm)	`BMXWAIST`
18	`body_fat_percent`	Estimated body fat percentage	`BIDPFAT`
19	`weight`	Weight (kg)	`BMXWT`
20	`height`	Height (cm)	`BMXHT`
21	`cholesterol`	Total serum cholesterol (mg/dL)	`LBXTC`
22	`red_blood_cell_count`	Red blood cell count	`LBXRBCSI`
23	`hemoglobin`	Hemoglobin (g/dL)	`LBXHGB`
24	`hematocrit`	Hematocrit (%)	`LBXHCT`
25	`mean_cell_hemoglobin`	Mean cell hemoglobin (pg)	`LBXMCHSI`
26	`red_cell_distribution_width`	Red cell distribution width (%)	`LBXRDW`
27	`vo2max`	Predicted maximal oxygen uptake (ml/kg/min)	`CVDESVO2`

Table S3. Summary of variables in the cleaned dataset after ETL. This table shows all the variables included in the cleaned dataset, their descriptions and the mapping of each variable to the original NHANES variable names.

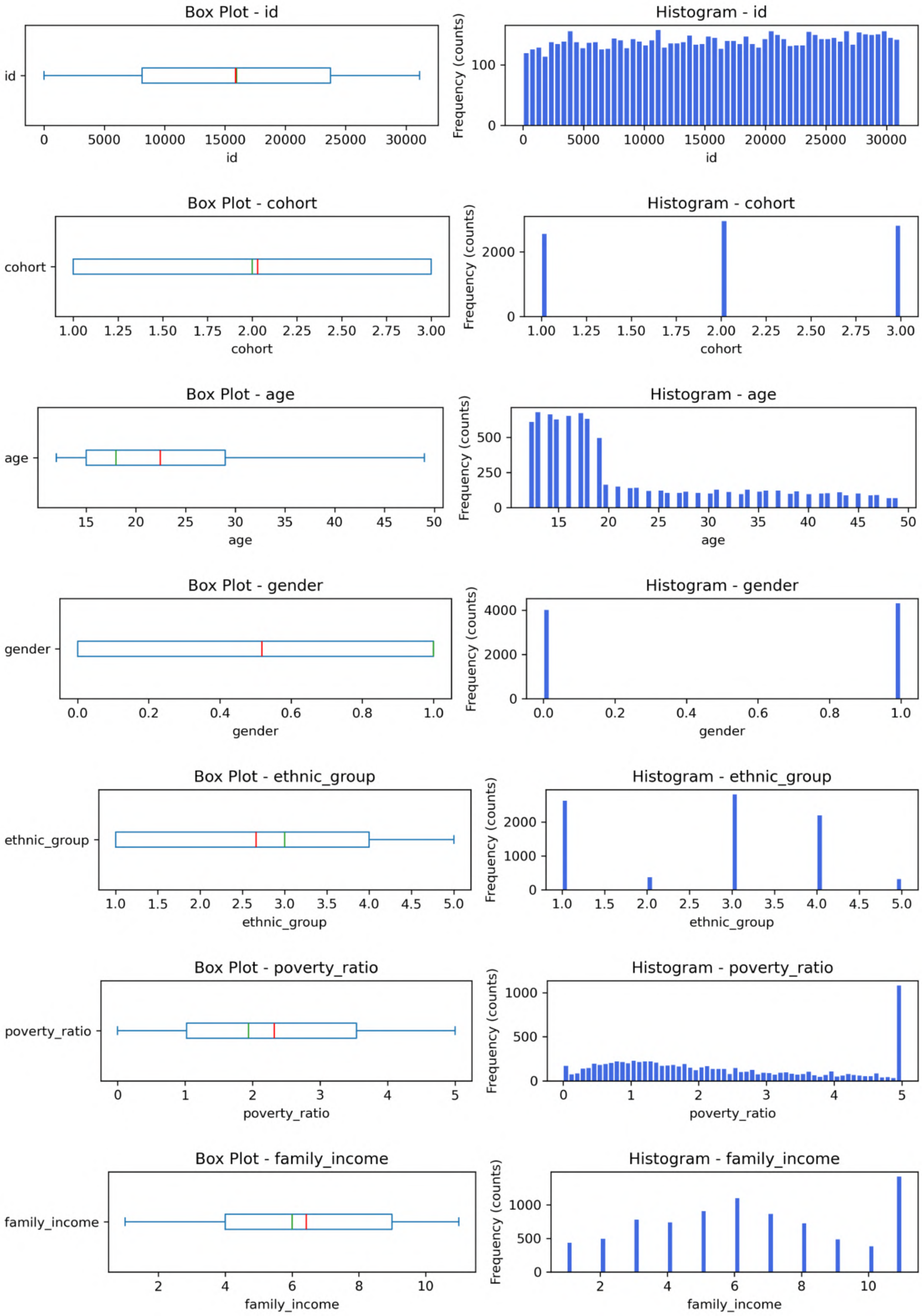

Figure S1. Visual data distribution of numeric variables 1. Each subplot shows a *box plot* (left) highlighting median (green line), quartiles, and mean (red line), and a *histogram* (right) showing the frequency distribution.

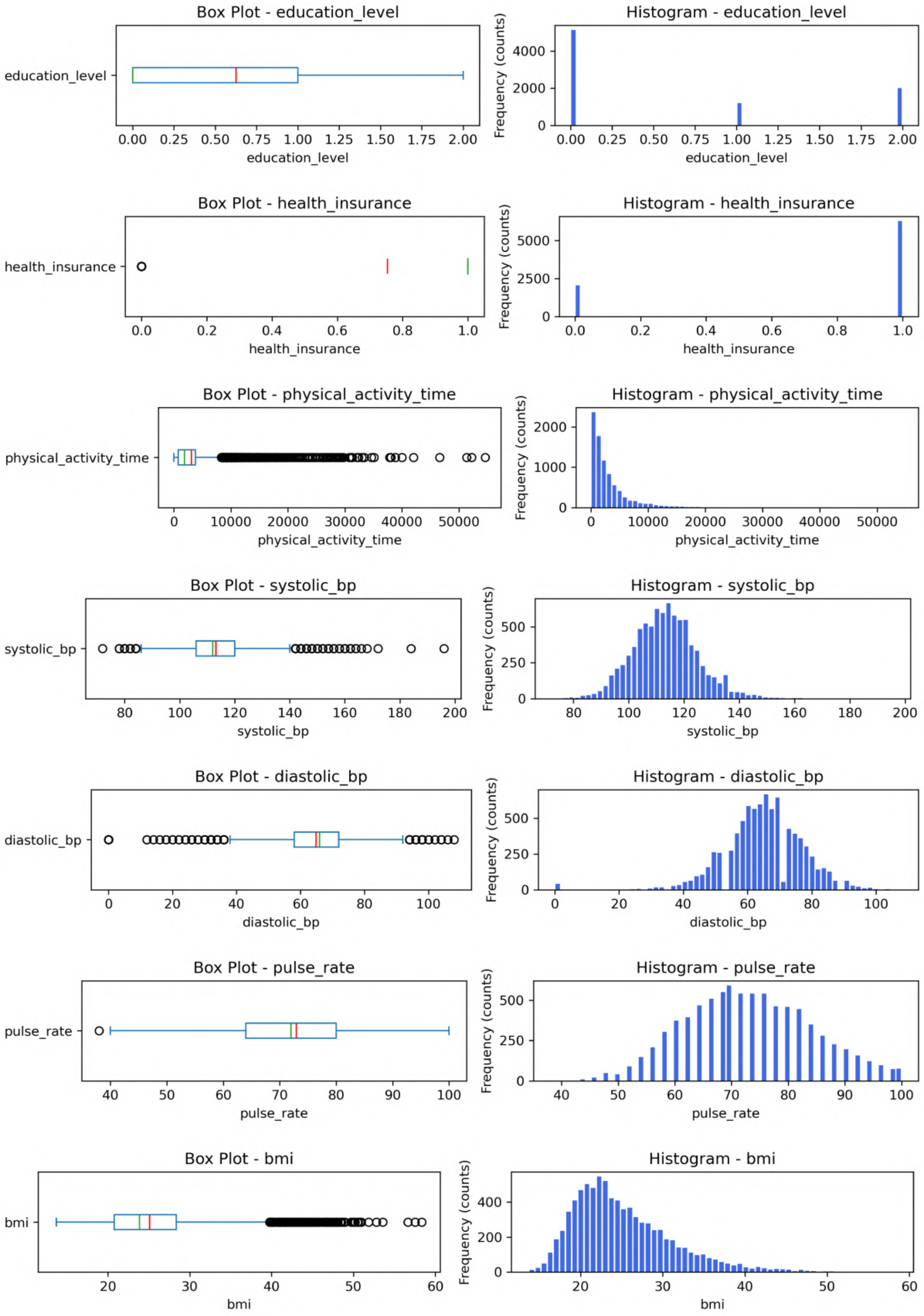

Figure S2. Visual data distribution of numeric variables 2. Each subplot shows a *box plot* (left) highlighting median (green line), quartiles, and mean (red line), and a *histogram* (right) showing the frequency distribution.

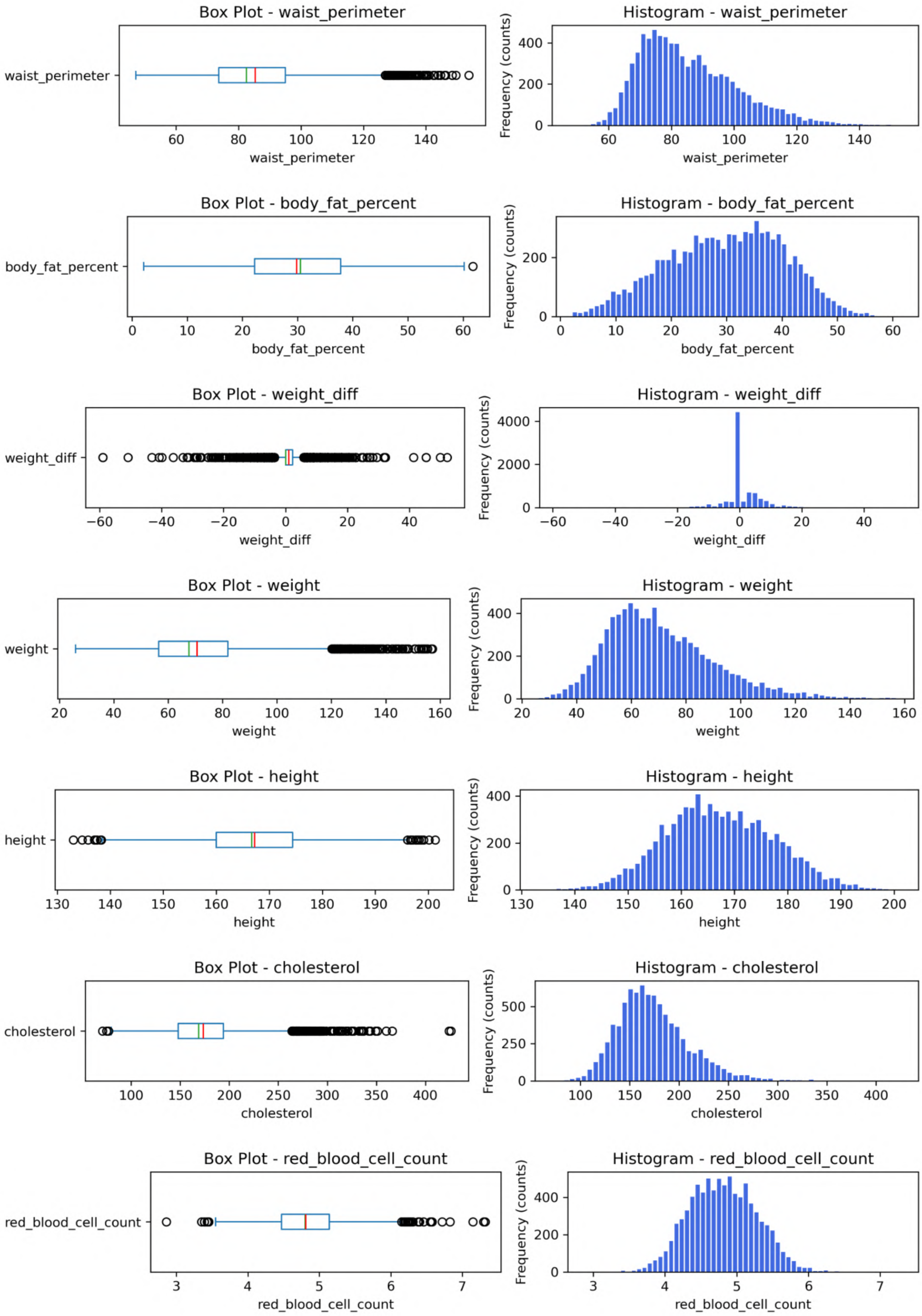

Figure S3. Visual data distribution of numeric variables 3. Each subplot shows a *box plot* (left) highlighting median (green line), quartiles, and mean (red line), and a *histogram* (right) showing the frequency distribution.

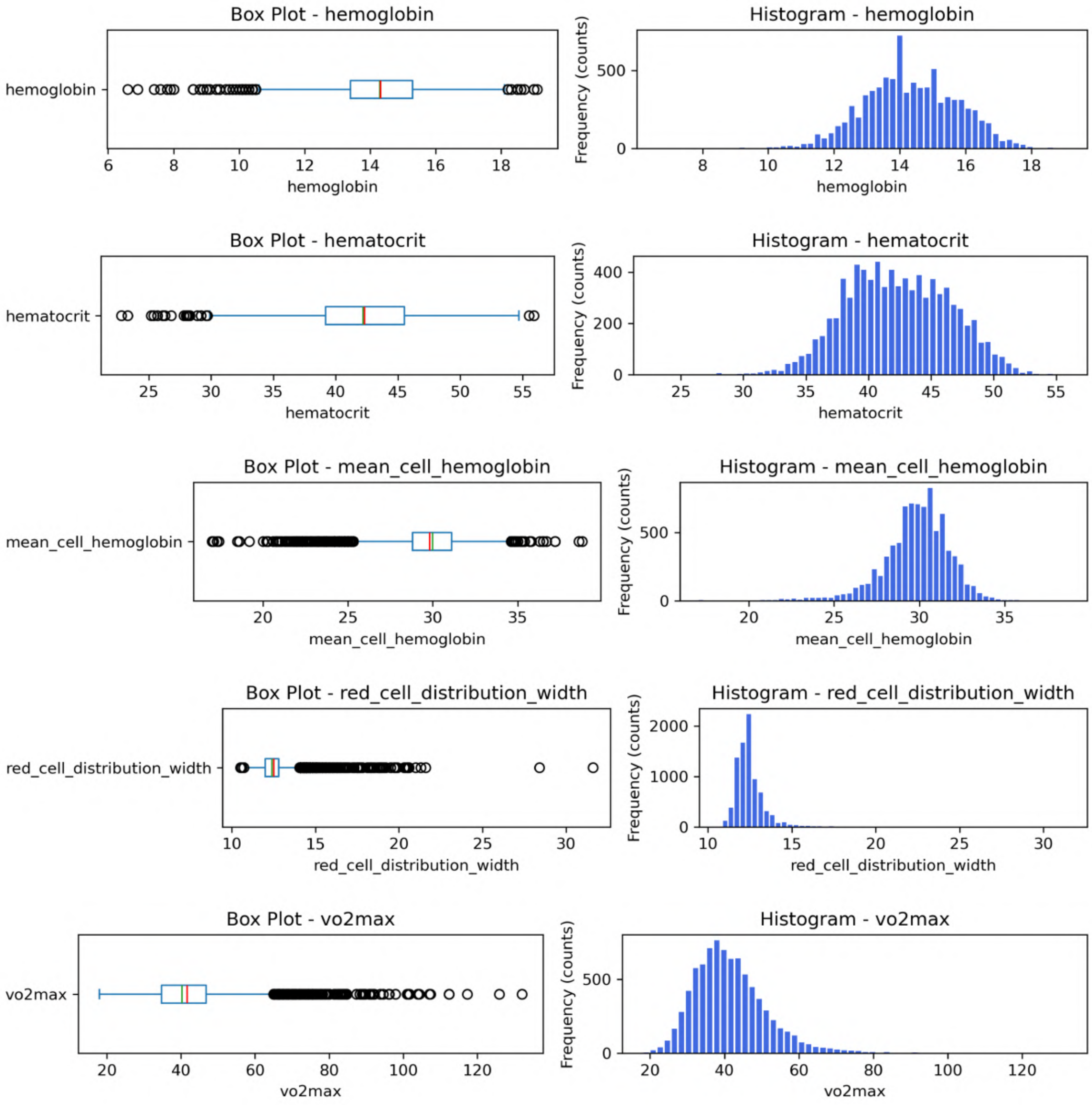

Figure S4. Visual data distribution of numeric variables 4. Each subplot shows a *box plot* (left) highlighting median (green line), quartiles, and mean (red line), and a *histogram* (right) showing the frequency distribution.

Variable	**Skewness**	**Kurtosis**	**Shapiro-Wilk**	**Kolmogorov-Smirnov**	**D'Agostino**	**Best Fit (common)**
id	-0.034293	-1.203046	0.0	0.0	0.0	beta
cohort	-0.053791	-1.445915	0.0	0.0	0.0	chi2
age	1.055482	-0.167185	0.0	0.0	0.0	chi2
gender	-0.071646	-1.994867	0.0	0.0	0.0	chi
ethnic_group	-0.168039	-1.302859	0.0	0.0	0.0	chi
poverty_ratio	0.467745	-1.060340	0.0	0.0	0.0	beta
family_income	0.035898	-1.059492	0.0	0.0	0.0	beta
education_level	0.794797	-1.136969	0.0	0.0	0.0	chi
health_insurance	-1.178560	-0.610997	0.0	0.0	0.0	beta
physical_activity_time	3.865170	24.876823	0.0	0.0	0.0	beta
systolic_bp	0.467102	1.274095	0.0	0.0	0.0	logistic
diastolic_bp	-0.808325	3.672225	0.0	0.0	0.0	logistic
pulse_rate	0.144753	-0.419012	0.0	0.0	0.0	beta
bmi	1.082575	1.466438	0.0	0.0	0.0	lognorm
waist_perimeter	0.784018	0.388234	0.0	0.0	0.0	lognorm
body_fat_percent	-0.164252	-0.592768	0.0	0.0	0.0	beta
weight_diff	-0.172712	11.823981	0.0	0.0	0.0	logistic
weight	0.830100	0.847024	0.0	0.0	0.0	lognorm
height	0.112414	-0.284065	0.0	0.000001	0.0	beta
cholesterol	0.887262	1.746818	0.0	0.0	0.0	lognorm
red_blood_cell_count	0.166687	0.011558	0.0	0.000001	0.0	beta
hemoglobin	-0.155960	0.298141	0.0	0.0	0.0	gamma
hematocrit	-0.049396	-0.230510	0.0	0.000002	0.0	chi2
mean_cell_hemoglobin	-0.980948	3.270486	0.0	0.0	0.0	logistic
red_cell_distribution_width	4.136816	41.830918	0.0	0.0	0.0	logistic
vo2max	1.421569	5.078667	0.0	0.0	0.0	lognorm

Table S4. Descriptive statistics and distribution fits for cleaned dataset variables. Skewness, kurtosis, normality test results, and best-fitting to common distribution ('norm', 'lognorm', 'gamma', 'expon', 'beta', 'chi', 'chi2' and 'logistic').

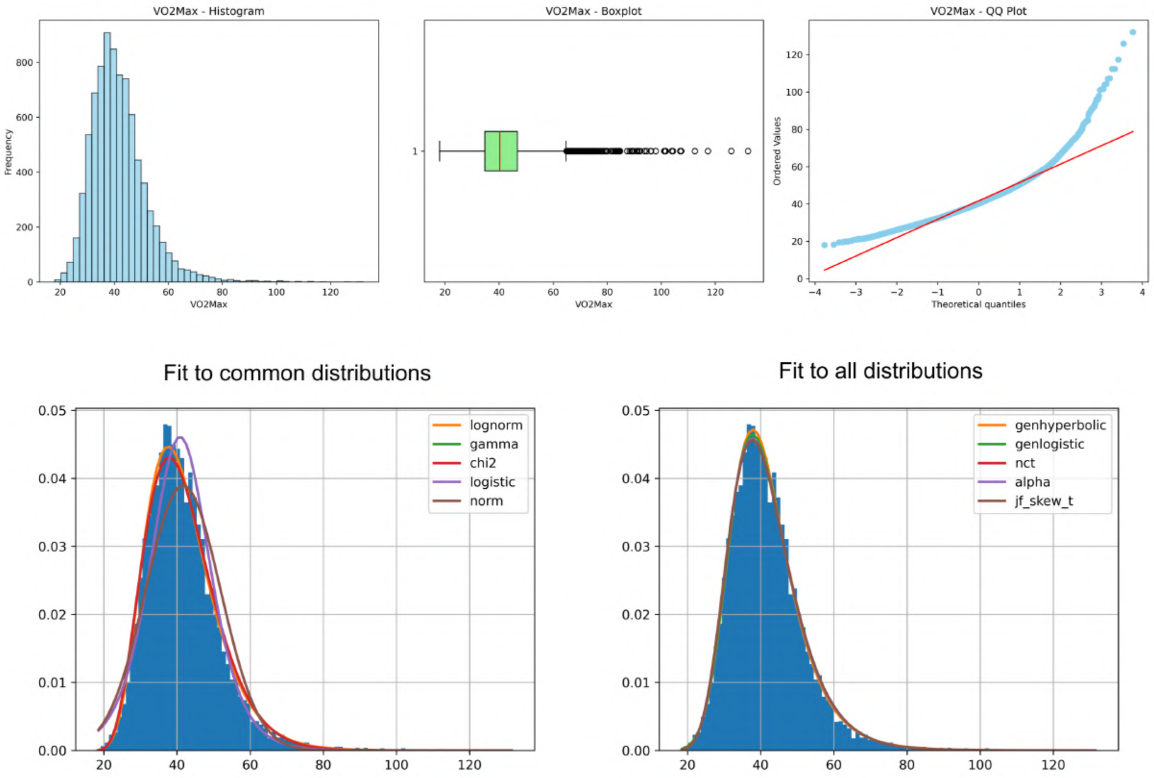

Figure S5. Comprehensive distribution analysis for VO$_2$max. The figure combines three visualizations: (1) a *histogram* showing the frequency distribution, (2) a *box plot* highlighting median, quartiles, and outliers, and (3) a *QQ plot* assessing normality against a theoretical normal distribution. These plots, along with skewness, kurtosis, normality tests, and distribution fits (common ((’norm’, ’lognorm’, ’gamma’, ’expon’, ’beta’, ’chi’, ’chi2’ and ’logistic’)) and all tested distributions), provide a complete view of the variable’s distribution characteristics.

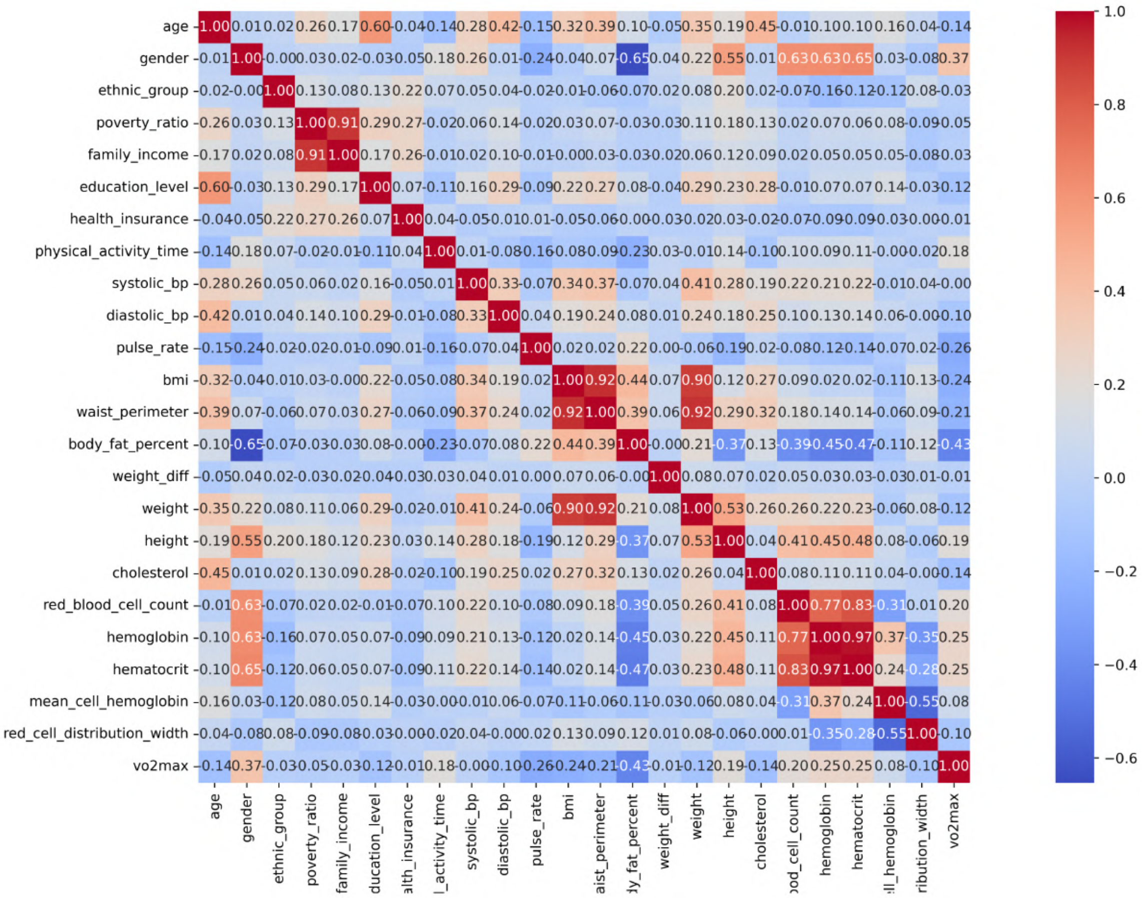

Figure S6. Correlation matrix of the dataset variables. The heatmap displays pairwise Pearson correlations between all numeric variables (excluding `id`, `cohort`, and `smoker`). Strong positive and negative correlations are highlighted in red and blue, respectively. Notable correlations include `Age` with `education_level`, `blood_pressure`, `cholesterol`, `BMI`, and `waist_perimeter`, as well as physiological associations between `Gender`, `height`, and `VO_2max`. Additionally, hematological measures `red_blood_cell_count`, `hemoglobin`, and `hematocrit` are strongly intercorrelated.

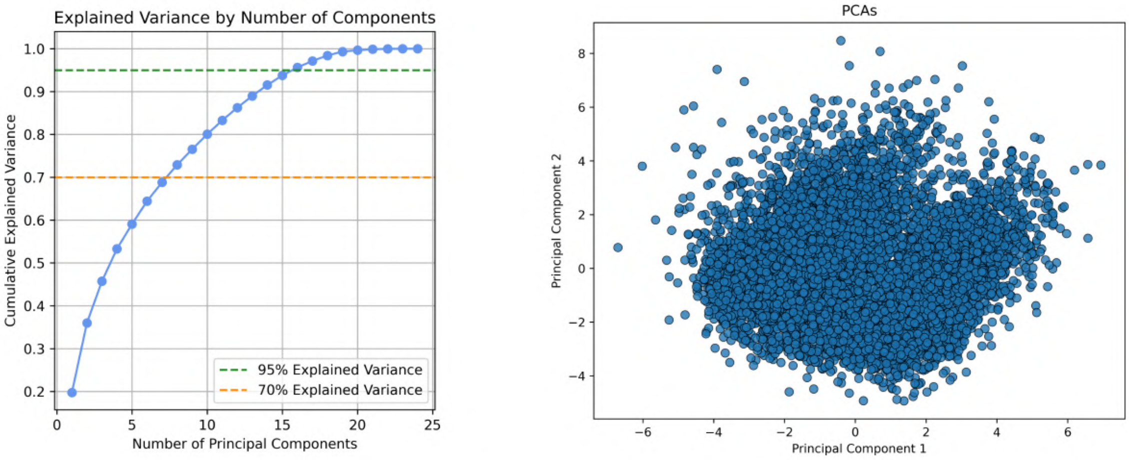

Figure S7. Principal Component Analysis (PCA) visualizations. The left panel shows the cumulative explained variance as a function of the number of principal components, with reference lines indicating 70% and 95% thresholds. The right panel displays a scatter plot of the data projected onto the first two principal components, illustrating the spread and relative positioning of observations in the reduced 2-dimensional space.

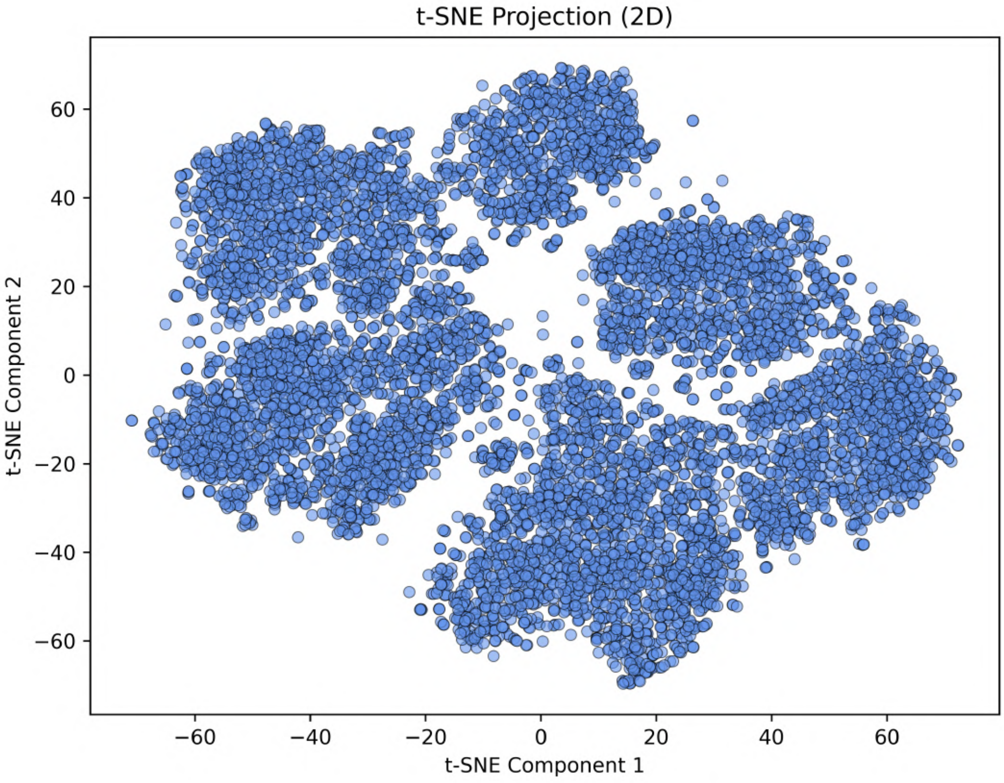

Figure S8. t-SNE 2-dimensional projection of the dataset. The scatter plot displays the observations projected onto the first two t-SNE components, showing the relative distances and overall structure in a reduced 2D space. Points are colored uniformly and include edge outlines for visual clarity.

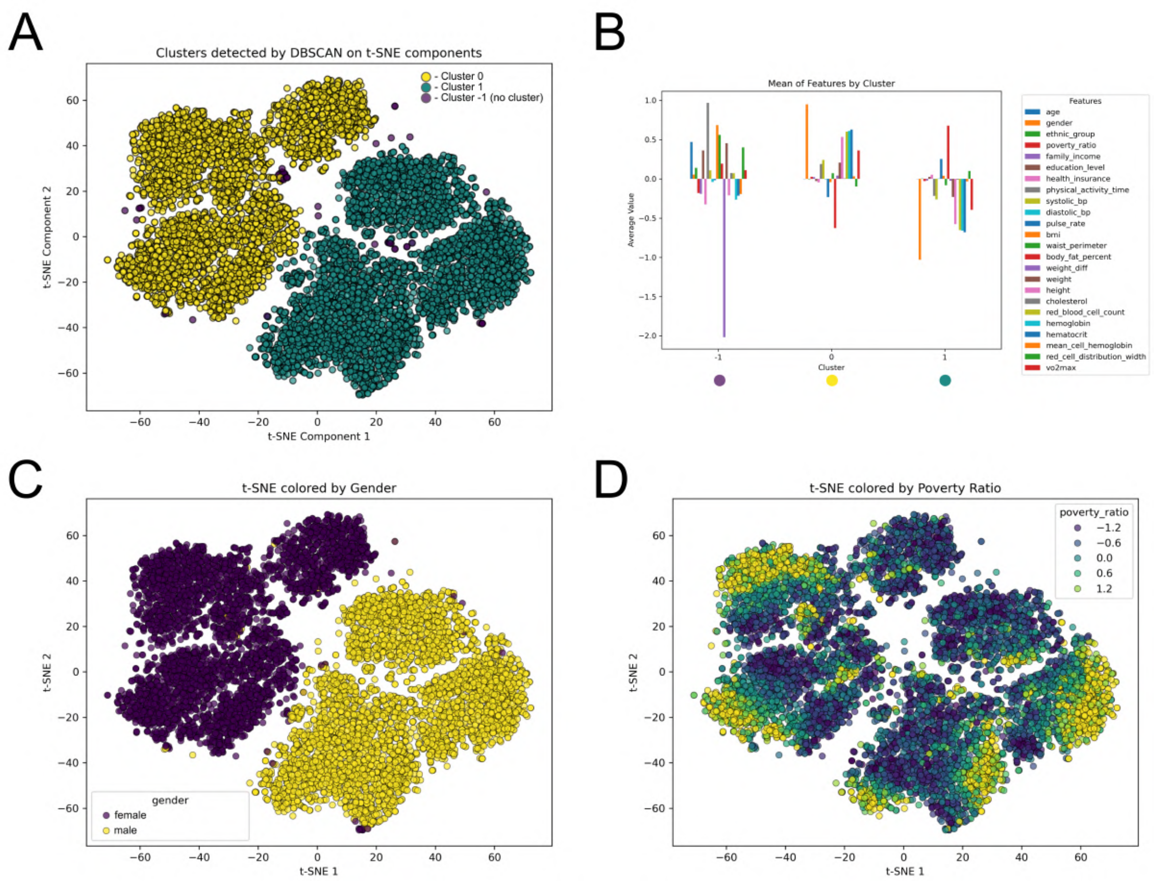

Figure S9. DBSCAN clustering and t-SNE visualizations. (**A**) Scatter plot of observations in the 2D t-SNE space colored by DBSCAN cluster labels. (**B**) Bar plot showing the mean values of numeric features for each DBSCAN cluster. (**C**) t-SNE projection colored by `gender`, highlighting patterns associated with this variable. (**D**) t-SNE projection colored by `poverty_ratio`, illustrating differences related to socioeconomic status.

Variable	Skewness	Kurtosis	Best Fit	Suggested Transformation	Reasoning
physical_activity_time	3.91	25.35	Exponential	`np.log1p(physical_activity_time)`	Extreme right-skew and heavy tails: log1p reduces range safely.
weight_diff	-0.17	11.82	Logistic	`np.sign(x) * np.sqrt(abs(x))`	Heavy tails with both signs: signed sqrt handles asymmetry.
poverty_ratio	0.46	-1.06	Beta	`np.log1p(poverty_ratio)`	Mild skew, but better fit and stability with log transform.
bmi	1.08	1.47	Log-normal	`np.log(bmi)`	Positive skew and lognormal fit suggest logarithmic transform.
waist_perimeter	0.78	0.38	Log-normal	`np.log(waist_perimeter)`	Moderate skew with log-normal fit: log transform is suitable.
weight	0.83	0.85	Log-normal	`np.log(weight)`	Positive skew and good log-normal fit: log transform recommended.
cholesterol	0.88	1.75	Log-normal	`np.log(cholesterol)`	Skewed and leptokurtic: log reduces tail effects.
vo2max	1.42	5.08	Log-normal	`np.log(vo2max)`	Highly skewed and heavy-tailed: log transform stabilizes variance.

Table S5. Suggested transformations for skewed variables. Skewness, kurtosis, best-fitting distributions, and recommended transformations to stabilize variance and reduce skewness.

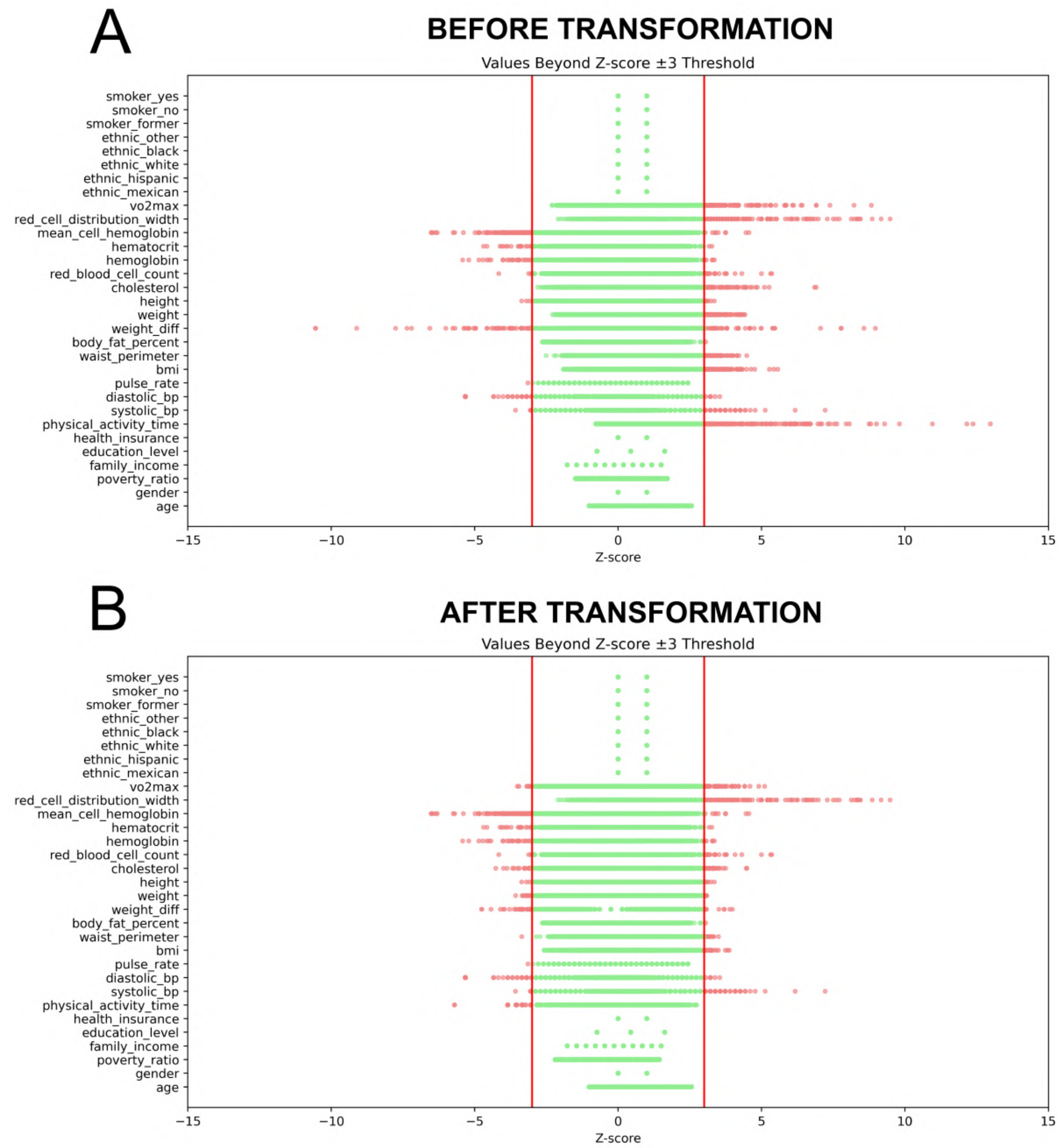

Figure S10. Standardized values (Z-scores) across variables before and after transformations. Panel (A) shows the distribution of raw standardized values where points beyond the ± 3 Z-score threshold are highlighted in red, indicating potential outliers. Panel (B) displays the same visualization after applying variable-specific transformations, including logarithmic and symmetric s uare root adjustments, which reduce extreme skewness and heavy tails. Green points indicate values within the ± 3 Z-score range, while red points indicate values outside this range.

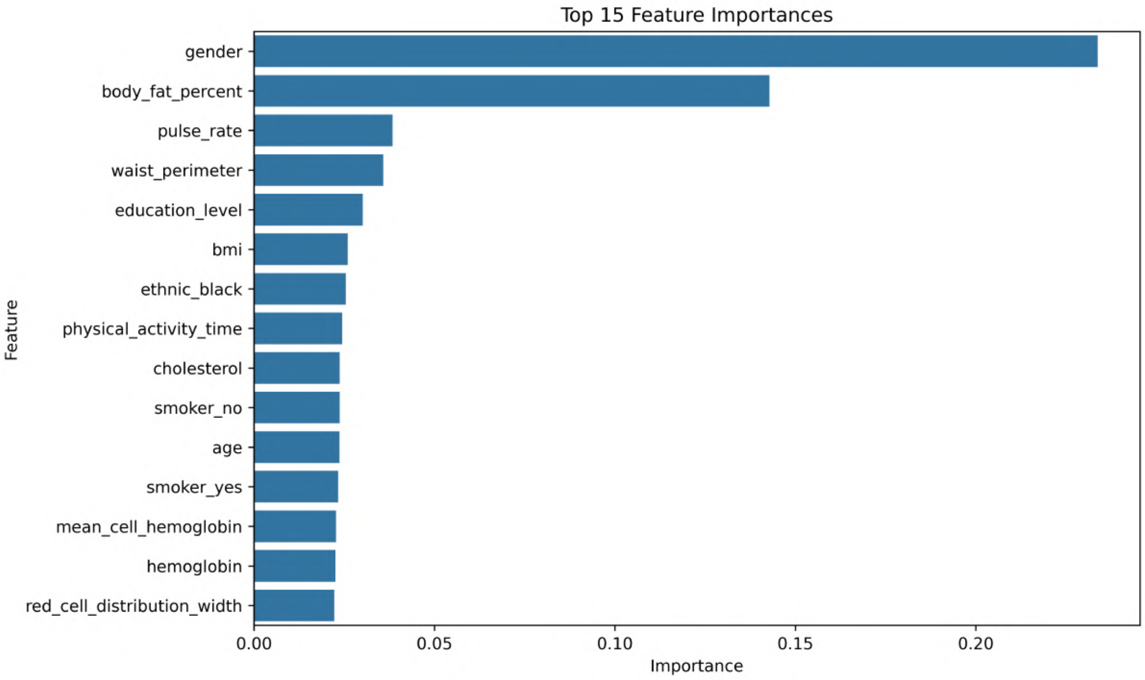

Figure S11. Feature importance of predictor variables from the XGBoost model. The bar plot shows the relative contribution of each variable in predicting VO_2max. *Gender* and *body_fat_percent* were the most influential features, followed by *pulse_rate*, *waist_perimeter*, *education_level*, and *bmi*. Variables with lower importance still provide incremental predictive value in combination with others.