**DSCI 4412: Introduction to Big Data Technologies Spring 2024**

**Project Paper**

# Fatigue Prediction Using Machine Learning  and Big Data Technologies

Maya Hany Elshweikhy- 900204233 - mayahanyy1@aucegypt.edu.

Ahmed Hassanin- 900215405 - ahmed.fikry@aucegypt.edu

Supervised by Dr Sally Elghamrawy

**Table of contents**

## Abstract

Fatigue is a prevalent symptom in various physical illnesses, often exacerbated by inadequate monitoring of vital signs such as heart rate and sleep patterns. Leveraging data obtained from Fitbit Fitness Trackers, this study employs big data and machine learning techniques to predict and understand fatigue levels. The project integrates resources such as Hadoop, Spark, and H2O, with a focus on H2O's capability to build predictive models efficiently and at scale. H2O's user-friendly interface and extensive algorithm suite enable rapid model development and customization. Its scalability ensures optimal performance, crucial for analyzing large and diverse datasets.

The dataset, sourced from 30 Fitbit users, comprises minute, hour, and day-level activity, heart rate, and sleep data. Five machine learning models including Gradient Boosting Machine, Deep Learning, Random Forest, XGBoost and Generalized Linear Model are compared and applied. Hyperparameter tuning is conducted to optimize model performance, evaluated using Mean Squared Error and R2-score metrics. The research not only analyzes algorithm performance but also examines interpretability and scalability, contributing to the understanding and prediction of fatigue in diverse contexts.

**Keywords**: Fatigue prediction, Big data analytics, Machine learning, Fitbit Fitness Trackers, H2O, Scalability, Algorithm comparison, Hyperparameter tuning, Mean Squared Error, R2-score.

# Fatigue Prediction Using Machine Learning  and Big Data Technologies

## 1. Main Topic Exploration: Big Data Techniques

Fatigue is one of the most common symptoms of many physical illnesses. Many of the people who suffer from fatigue, sleep deprivation, or any other related symptoms suffer from that due to lack of recording of their physical vitals such as heart rate, sleep recordings, and others. Through this dataset obtained by Fitbit Fitness Trackers, records of users' vitals are obtained, and the data collected could be used for several predictive models that would be useful, such as fatigue detection and sleep analysis, among others.

The concept of the project is to use big data with machine learning techniques to predict fatigue, and the reasons behind it, by utilizing records of people's vitals and reported fatigue levels. This will be achieved through utilizing several resources, such as Hadoop, Spark, and H2O.

H2O is a pivotal component in our arsenal of tools for tackling the complexities of fatigue prediction and analysis within our big data project. Its inclusion is driven by its prowess in harnessing the power of machine learning with big data combined.

One of the primary reasons for leveraging H2O is its ability to streamline the process of building predictive models. With its user-friendly interface and comprehensive suite of algorithms, H2O empowers us to swiftly construct, and fine-tune models tailored to our specific objectives. This agility is crucial in our quest to predict fatigue accurately and identify its underlying causes.

Moreover, H2O's scalability aligns seamlessly with the demands of big data analytics in the context of fitness analytics which include second-by-readings. As the dataset continues to grow in size and complexity, H2O ensures the maintenance of high-performance levels without compromising on accuracy. By harnessing distributed computing capabilities, H2O enables us to leverage the full potential of our computing infrastructure, maximizing resource utilization and minimizing processing times.

## 2. Problem Statement Identification

Fatigue and stress are one of the main symptoms for many people. Also, with the rise of sensor technology and wearable vital measurement devices, there is huge data that could be useful to derive better solutions to better optimize, fix, and control stress/fatigue levels, and everything that comes with it such as pain management, sleep issues, and cognitive functions.

Therefore, the decision was to tackle this problem from a big data perspective by utilizing the big data provided by sports watches and other fitness products that provide measurements for vitals to solve this issue.

This will be done through using big data technology such as Spark and H2O. These frameworks provide machine learning tools that are suitable for big data such as the dataset which will be vital for fast and efficient processing of the data. Solutions will be provided such as more accurate classification models for fatigue and better optimized data pipelines that would add to the reference paper.

## 3. Big Dataset Selection for Problem Analysis

### Dataset Source:

M&ouml;bius, “Fitbit Fitness Tracker Data,” Kaggle,
https://www.kaggle.com/datasets/arashnic/fitbit (accessed Mar. 12, 2024).

This dataset was collected by surveying 30 eligible users of a certain fitness tracker product. Thirty users consented to submit their personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. The data in some instances also are second-level, hour-level, or day-level. Even though the data is generated from 30 users only, there are millions of data records due to the nature of trackers recording techniques, which vary from second level to hour- or day-level sometimes.

The data is divided into 3 main sections. 1) Activity, where activity variables are collected such as calories burned, sedentary time, number of steps, etc. 2) Heart rate, which only includes heart rate time series and resting heart rate. Finally, 3) Sleep, which includes many variables including duration, stages duration, and efficiency, to name a few.

For the machine learning models and techniques, the paper compared and applied six models. Decision Tree (DT), Random Forest (RF), XGBoost (XGB), k-Nearest Neighbor (kNN), Fully-Connected Neural Network (FCNN) and LSTM (Long ShortTerm Memory) neural network.

Our research will include the analysis of these algorithms. The algorithms Fully Connected Neural Network (FCNN) and LSTM (Long ShortTerm Memory) neural network are deep learning pipeline algorithms. The hyperparameters must be tuned to create models with high-performance. This is done by running multiple experiments with different configurations and using a small part of the training data as validation data.

To evaluate the performance of the models we use the Mean Squared Error (MSE) where n is the number of samples, y is the actual values and ˆy is the predicted values. The MSE is a common error metric for regression models and is typically used when training models since it measures the difference between the predictions and the ground truth.

We also use R2-score, often referred to as the coefficient of determination The R2-score represents the ratio between the variance explainable by the model and the total variance. A perfect fit will give an R2-score of 1, while a score of 0 will indicate that the model performs equally to predicting the mean of the actual observations for any input. This error metric has the advantage of being interpretable independent of the input variables, unlike MSE, where the magnitude of the error depends on the scale of the input data.

**Table 1: Available variables from the data set collected with the Fitbit fitness tracker. The sleep stages include: Deep sleep, light sleep, REM sleep and awake.**

| **Type** | **Variable** | **Granularity** |
|---|---|---|
| Activity | Calories burned | Daily |
| | Number of floors | Daily |
| | Sedentary minutes | Daily |
| | Lightly active minutes | Daily |
| | Fairly minutes | Daily |
| | Very minutes | Daily |
| | Number of steps | Daily |
| | Distance walked | Daily |
| Heart rate | Heart rate time series | 1 second |
| | Resting heart rate | Daily |
| Sleep | Duration | Daily |
| | Efficiency | Daily |
| | Start time | Daily |
| | End time | Daily |
| | Main sleep or nap | Daily |
| | Sleep stage duration | 1 second |
| | Number of occurences of sleep stage | Daily |

Table 2: 26 Features extracted from the Fitbit fitness tracker data

| Feature name | Data Type | Unit | Description |
|---|---|---|---|
| Sleep total duration | numeric | minutes | Total duration of sleep in minutes |
| Sleep efficiency score | numeric | 0-100 | Score representing the efficiency of sleep, ranging from 0 to 100 |
| Deep sleep duration | numeric | minutes | Duration of deep sleep in minutes |
| Light sleep duration | numeric | minutes | Duration of light sleep in minutes |
| REM sleep duration | numeric | minutes | Duration of REM (rapid eye movement) sleep in minutes |
| Awake in bed duration | numeric | minutes | Duration spent awake while in bed in minutes |
| Deep sleep count | numeric | count | Count of occurrences of deep sleep |
| Light sleep count | numeric | count | Count of occurrences of light sleep |
| REM sleep count | numeric | count | Count of occurrences of REM sleep |
| Awake in bed count | numeric | count | Count of occurrences of being awake while in bed |
| Sedentary minutes | numeric | minutes | Total duration of sedentary activities in minutes |
| Lightly active minutes | numeric | minutes | Total duration of lightly active activities in minutes |

| | | | |
|---|---|---|---|
| Fairly active minutes | numeric | minutes | Total duration of fairly active activities in minutes |
| Very active minutes | numeric | minutes | Total duration of very active activities in minutes |
| Average heart rate | numeric | beats per min | Average heart rate measured in beats per minute |
| Minimum heart rate | numeric | beats per min | Minimum heart rate measured in beats per minute |
| Maximum heart rate | numeric | beats per min | Maximum heart rate measured in beats per minute |
| Resting heart rate | numeric | beats per min | Resting heart rate measured in beats per minute |
| Calories burned | numeric | kcal | Total calories burned |
| Steps | numeric | count | Total number of steps taken |
| Distance | numeric | meters | Total distance traveled in meters |
| Age | numeric | years | Age of the individual in years |
| Gender | categorical | female/ male | Gender of the individual (female or male) |
| Weight | numeric | kg | Weight of the individual in kilograms |
| Height | numeric | cm | Height of the individual in centimeters |
| Body Mass Index | numeric | kg/m^2 | Body Mass Index (BMI), calculated as weight (kg) divided by height (m) squared |

## 4. Literature Review Summary: Analysis of 10-15 Indexed Papers

### [1] E. Husom *et al.*, "Machine learning for fatigue detection using Fitbit Fitness Trackers," *Proceedings of the 10th International Conference on Sport Sciences Research and Technology Support*, 2022. doi:10.5220/0011527500003321

Patil et al. conducted a review on stress and fatigue detection using machine learning techniques, emphasizing the importance of accurate detection methods in healthcare and industrial settings. The paper provides insights into various machine-learning approaches employed for stress and fatigue detection, highlighting their strengths and limitations. There were six different ML algorithms applied and compared to create a model that can estimate the FAS score (Fatigue Assessment Scale) based on fitness tracker data: Decision Tree (DT), Random Forest (RF), XGBoost (XGB), k-Nearest Neighbor (kNN), Fully-Connected Neural Network (FCNN) and LSTM (Long Short- Term Memory) neural network. The purpose of implementing these algorithms is to establish a relationship between a given set of input features X, which in this scenario represents the data gathered from the Fitbit tracker, and an output target y. In our context, this output target y corresponds to the FAS-score. The various ML algorithms have several hyper-parameters that control the configuration of the al- algorithm and the training process. These hyper- parameters were tuned in order to create models with high performance. In addition, grid search and manual search of hyper-parameters was used for the best combination.

By analyzing existing literature, the authors offer a comprehensive overview of the advancements in this field, addressing challenges such as data acquisition, feature selection, and model performance evaluation. The authors began by constructing neural networks with initially limited layers and nodes, gradually increasing their complexity while monitoring error metrics until achieving promising performance. Eventually, they settled on specific configurations for deep learning methods: a fully connected neural network (FCNN) with one hidden layer comprising 8 nodes, each employing Rectified Linear Unit (ReLU) activation; and a Long Short-Term Memory (LSTM) network with one layer featuring 8 hidden units, each using sigmoid activation. To assess model performance, Mean Squared Error (MSE) and R2-score were employed. MSE calculates the average squared difference between predicted and actual values, providing a common error metric for regression models, while R2-score, or coefficient of determination, evaluates the proportion of variance explained by the model's predictions. These metrics serve as indicators of model accuracy and suitability for the task at hand.

**[2] R. Hooda, V. Joshi, and M. Shah, "A comprehensive review of approaches to detect fatigue using machine learning techniques,"** ***Chronic Diseases and Translational Medicine*****, vol. 8, no. 1, pp. 26–35, Feb. 2022. doi:10.1016/j.cdtm.2021.07.002**

The authors present a comprehensive review of approaches to detect fatigue using machine learning techniques. The paper categorises existing methodologies based on data sources, feature extraction methods, and classification algorithms. Through a detailed analysis of recent studies, the authors discuss the potential applications of machine learning in fatigue detection across various domains, including healthcare, transportation, and occupational safety. In the study of Fatigue detection using ML the following models K‐nearest neighbors, SVM, Linear Regression, and ANN were used for comparison. The model's performance was tested using the accuracy score. The highest accuracy scores were ANN with 83.6% and Linear Regression with 90%. The review identifies key challenges and future research directions, emphasizing the need for robust and reliable fatigue detection systems to enhance human performance and well-being.

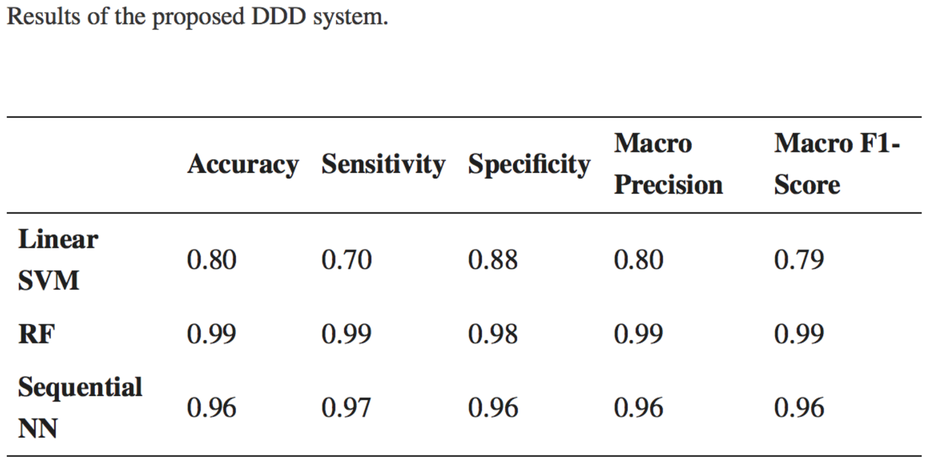
**[3] C.-A. Cos *et al.*, "Enhancing mental fatigue detection through physiological signals and machine learning using contextual insights and efficient modelling,"** ***Journal of Sensor and Actuator Networks*****, vol. 12, no. 6, p. 77, Nov. 2023. doi:10.3390/jsan12060077**

The authors propose a multimodal system for detecting driver fatigue using wearable sensors to improve road safety by identifying fatigue-related changes in physiological signals. The paper describes the integration of electroencephalography (EEG) and gyroscopic data with image processing techniques to detect driver fatigue in real-time. Through experiments conducted on a driving simulator, the authors demonstrate the effectiveness of their proposed system in accurately detecting fatigue-induced impairments in driver performance. The study contributes to the development of advanced driver assistance systems (ADAS) capable of mitigating the risks associated with driver fatigue.

The feature selection process utilized in this study takes into account both statistical and contextual considerations regarding the relevance of features. Statistical relevance was determined by conducting variance and correlation analyses between independent features and the dependent variable, which in this case is the fatigue state. On the other hand, contextual analysis relied on insights gained from the experimental design and features' characteristics. The ML models included random forest, decision tree, support vector machine, k-nearest neighbors, and gradient boosting. The findings revealed that random forest had the highest performance, achieving an average accuracy and F1-score of 96% in classifying three levels of mental fatigue.

**[4] Y. Albadawi, A. AlRedhaei, and M. Takruri, "Real-time machine learning-based driver drowsiness detection using visual features,"** ***Journal of Imaging*****, vol. 9, no. 5, p. 91, Apr. 2023. doi:10.3390/jimaging9050091**

This paper uses machine learning techniques to review the detection of fatigue and its impact on health and safety. The paper provides an overview of the physiological and behavioural markers of fatigue, emphasizing the importance of early detection and intervention to prevent adverse outcomes. Through a comprehensive analysis of existing literature, the authors discuss various machine learning approaches employed for fatigue detection, including feature extraction, classification algorithms, and model validation techniques. The review highlights the potential applications of machine learning in assessing fatigue levels across different populations and environments, underscoring the need for interdisciplinary collaboration to address this significant public health concern. The ML algorithms applied were random forest, sequential neural network, and linear support vector machine classifiers. The dataset used was NTHUDDD video dataset. Evaluations of the proposed system over the National Tsing Hua University driver drowsiness detection dataset showed that it can successfully detect and alarm drowsy drivers with an accuracy up to 99%. Here is the table showing all classifires with their model performance scores:



Results of the proposed DDD system.

| | Accuracy | Sensitivity | Specificity | Macro Precision | Macro F1-Score |
|---|---|---|---|---|---|
| **Linear SVM** | 0.80 | 0.70 | 0.88 | 0.80 | 0.79 |
| **RF** | 0.99 | 0.99 | 0.98 | 0.99 | 0.99 |
| **Sequential NN** | 0.96 | 0.97 | 0.96 | 0.96 | 0.96 |

**[5] Y. Bai, Y. Guan, and W.-F. Ng, "Fatigue assessment using ECG and actigraphy sensors,"** ***Proceedings of the 2020 International Symposium on Wearable Computers*****, Sep. 2020. doi:10.1145/3410531.3414308**

The authors offer a comprehensive review on stress and fatigue detection using machine learning techniques, focusing on Fatigue assessment using ECG and actigraphy sensors. This paper highlights the integration of physiological signals and behavioral data for accurate assessment. The paper discusses the physiological responses to stress and fatigue, highlighting the role of biomarkers such as heart rate variability (HRV), electrodermal activity (EDA), and cortisol levels in quantifying stress levels. Through a systematic review of existing studies, the authors evaluate the efficacy of machine learning algorithms in detecting stress and fatigue from multimodal data sources. These ML models are Linear regression and deep learning model LSTM. The deep learning model performed better than the linear regression and the metrics used for model performance are MAE and RMSE. In this study, while deep learning approaches may yield superior outcomes, they offer less transparency compared to interpretable solutions, which allow for the identification of key human-understandable features. The review provides valuable insights into the challenges and opportunities associated with stress and fatigue detection, emphasizing the potential of machine learning-based approaches in personalized healthcare and wellness monitoring.

**[6] H. Luo, P.-A. Lee, I. Clay, M. Jaggi, and V. De Luca, "Assessment of fatigue using wearable sensors: A pilot study,"** ***Digital Biomarkers*****, vol. 4, no. Suppl. 1, pp. 59–72, Nov. 2020. doi:10.1159/000512166**

The study explores fatigue through the lens of multimodal sensor data and machine learning techniques. Fatigue, impacting various aspects of quality of life, lacks comprehensive assessment tools. The research involved 27 healthy subjects for the study and analyzed 405 recording days, utilizing wearable sensors for physical activity, vital signs, and daily fatigue questionnaires, which is similar to our actual main paper we are using for this project. Employing recurrent neural networks and supervised/unsupervised machine learning, the study identified significant relationships between self-reported fatigue and sensor data. Results indicated the effectiveness of causal convolutional neural networks for unsupervised representation learning and random forest for classification. Vital signs emerged as crucial factors for prediction of fatigue, especially for mental fatigue. Clustering analysis revealed a digital phenotype associated with fatigue, predominantly characterized by high physical activity intensity. The findings signify the potential of multimodal digital data to enhance understanding and measurement of non-pathological fatigue, suggesting avenues for future research in anomaly detection and prolonged monitoring.

**[7] H. Lu *et al.*, "Stresssense,"** ***Proceedings of the 2012 ACM Conference on Ubiquitous Computing*****, Sep. 2012. doi:10.1145/2370216.2370270**

The study addresses the impact of stress on individuals' well-being, focusing on changes in speech production as a physiological indicator. However, this study more focuses on a comparative analysis between different wearable measuring instruments, not just one instrument as most of the other analyzed papers here. It introduces StressSense, a system utilizing smartphone microphones for real-time stress detection through voice analysis. Unlike the previous study, which focused on fatigue, this research concentrates specifically on stress recognition. StressSense incorporates methods to adapt a general stress model to individual speakers and environmental conditions, achieving high accuracy rates across varied acoustic settings. In contrast to the previous study's emphasis on wearable sensors, StressSense utilizes ubiquitous mobile phones for unobtrusive stress monitoring. Notably, StressSense is the first system to integrate voice-based stress detection and model adaptation in real-life conversational contexts using smartphones. These findings highlight the system's potential for practical applications in stress management and wellness monitoring, distinguishing it from previous fatigue-focused research.

**[8] A. Leroux, R. Rzasa-Lynn, C. Crainiceanu, and T. Sharma, "Wearable devices: Current status and opportunities in pain assessment and management,"** ***Digital Biomarkers*****, vol. 5, no. 1, pp. 89–102, Apr. 2021. doi:10.1159/000515576**

The study explores the integration of wearable devices, ecological momentary assessment (EMA) data, and pain assessment and treatment. Unlike the previous studies on fatigue and stress, this research focuses on pain management more, which is still an element of fatigue and stress, which is why we thought it would be still a relevant study for our project to analyze. It examines various studies designs correlating pain scores with wearable device data and investigates the association between physical activity, physiological signals, and pain in patients. Results indicate limited research on incorporating wearable devices into pain assessment in natural settings, with most studies concentrating solely on physical activity. Promising correlations between pain scores and wearable device-derived signals are noted, suggesting potential for objective pain measurement. The conclusion highlights the opportunity to study the complex relationship between physiological signals, physical function, and pain in real-time environments. Integration of wearable devices and EMA could revolutionize pain management by establishing clinically meaningful endpoints.

**[9] M. J. Pinto-Bernal, C. A. Cifuentes, O. Perdomo, M. Rincón-Roncancio, and M. Múnera, "A data-driven approach to physical fatigue management using wearable sensors to classify four diagnostic fatigue states,"** ***Sensors*****, vol. 21, no. 19, p. 6401, Sep. 2021. doi:10.3390/s21196401**

The paper addresses the role of physical exercise in rehabilitation programs assisted by social robots, emphasizing the importance of understanding the optimal amount and intensity of exercise required for positive outcomes. It acknowledges challenges in monitoring patients' intensity to prevent extreme fatigue, which can lead to physical and physiological complications. While machine learning models have been applied in fatigue management, their practical utility is hindered by limited understanding of individual performance deterioration with fatigue, influenced by various factors such as exercise type, environment, and individual characteristics. The paper proposes a data analytic approach for managing fatigue in walking tasks, establishing feature and machine learning algorithm selection criteria. The ML models used are Logistic Regression (LR), K-Nearest Neighbor (KNN), Decision Tree (DT), Support Vector Machine (SVM), Artificial Neural Network (ANN), and Random Forest (RF) The model's performance was measured using Recall, which the true positive rate, accuracy, precision, and F1 score. The random forest model demonstrated superior performance in classifying four fatigue diagnosis states, achieving an average accuracy of ≥98% and F-score of ≥93% with ≤16 features. Additionally, the study evaluates prediction performance by reducing the number of sensors used, showing satisfactory results with as few as one or two inertial measurement units (IMUs).

**[10] D. Bustos *et al.*, "Machine learning approach to model physical fatigue during incremental exercise among firefighters," *Sensors*, vol. 23, no. 1, p. 194, Dec. 2022. doi:10.3390/s23010194**

The study addresses the significant threat physical fatigue poses to firefighters' health and safety, which can lead to decreased cognitive abilities and increased accident risk. While subjective scales and on-body sensors have been utilized to monitor physical fatigue, they have limitations such as task-specific assessments and model validation procedures. This study aims to develop a physical fatigue prediction model integrating cardiorespiratory and thermoregulatory measures with machine learning algorithms in a firefighter sample. Data from 24 participants during an incremental running protocol were collected, and various supervised machine learning algorithms were examined using physiological variables and participant characteristics to estimate four fatigue conditions. The tested algorithms included K-nearest neighbours, Boosted

Trees (Gradient-boosted Trees, XGBoosted Trees and RUSBoosted Trees), Bagged Trees, Random Forests, Support Vector Machines with different kernel functions (linear, quadratic, cubic and Gaussian) and Artificial Neural Networks.

The XGBoosted Trees algorithm demonstrated the best performance, achieving an average accuracy of 82%, with accuracies of 93% and 86% for low and severe fatigue levels, respectively. The study also evaluated different methods for assessing model performance, highlighting the practicality of group cross-validation. Overall, the study underscores the benefits of using multiple physiological measures to enhance physical fatigue modeling, offering a promising tool for health and safety management and paving the way for future research in field conditions.

## 5. Paper Categorization based on Utilized Models

| **Models** | **Papers** |
|---|---|
| Random Forest | 1, 3, 4,6, 9, 10 |
| K-nearest neighbours (KNN) | 1, 2, 3, 9, 10 |
| Support vector machine | 2, 3, 4,9, 10 |
| Artificial neural network (ANN) | 2, 6, 9 |
| Decision Tree | 1, 3,9 |
| Linear Regression | 2, 5 |
| XGBoost | 1, 10 |
| Gradient boosting | 3, 10 |
| Long Short- Term Memory  (LSTM ) | 1, 5 |
| Fully-Connected Neural Network (FCNN) | 1 |

| | |
|---|---|
| Sequential Neural Network | 4 |
| Logistic Regression | 9 |
| RUSBoosted Trees | 10 |
| Bagged Trees | 10 |

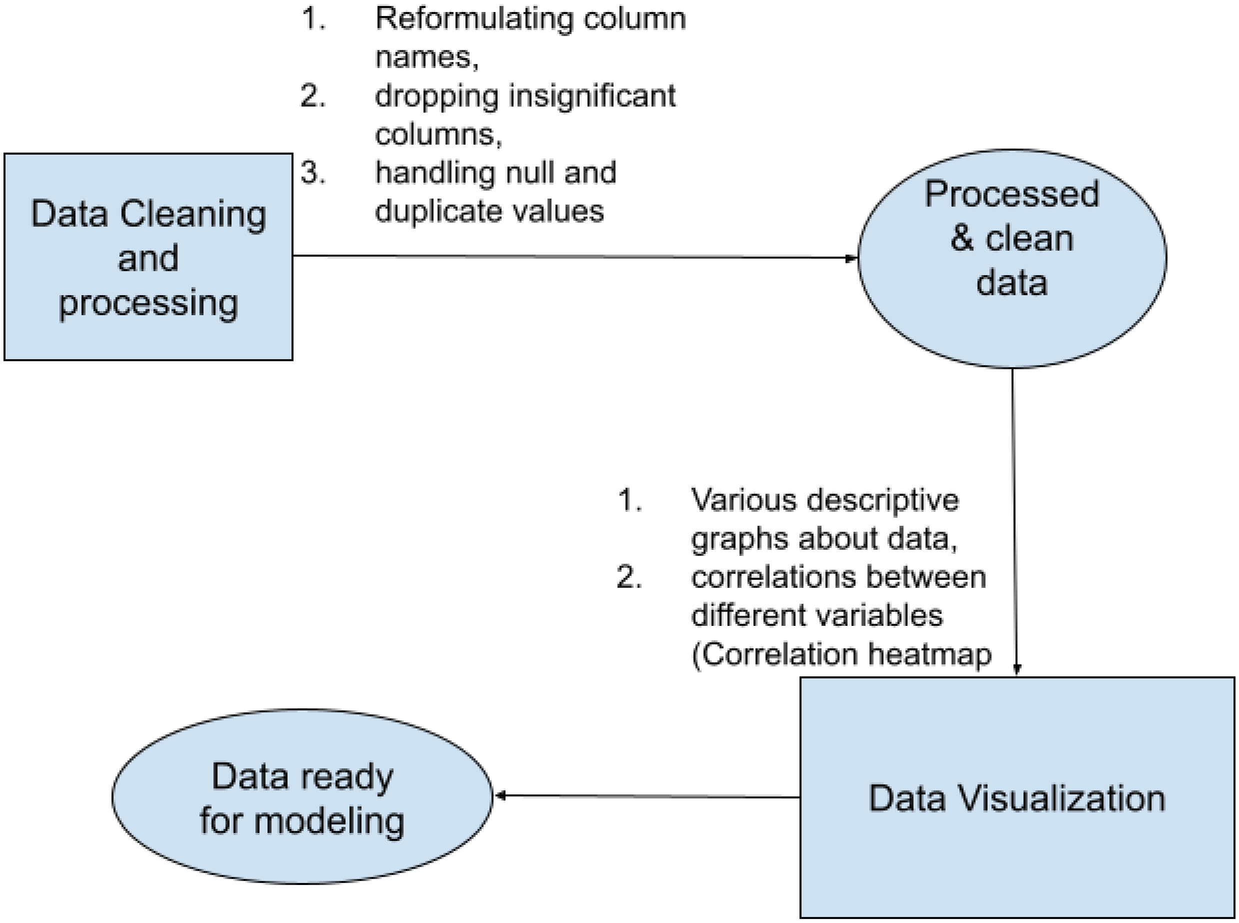
**6. Solution Implementation: Problem Resolution Approach Description**

We solved the problem of fatigueness through a comprehensive big data utilization. Since there is a lot of big data available concerning physical metrics in the human body, recorded by smart watches and similar popular instruments, we have a lot of potentially valuable data regarding physical metrics. Utilizing the data we chose earlier, we approached to solve this challenge by implementing various machine learning approaches to the data, mainly to perform supervised learning on the data to predict fatigueness and other objective variables.

First, an integral part of the solution was the preprocessing part. In the data preparation stage, we made sure the data is in the correct formats of data. The data was run through a data cleaning pipeline. The pipeline ensured that no faulty data was used in the modeling. Missing data were padded with 0 or with interpolation, whichever was more appropriate for the type of data and the model we are using. Moreover, visualization was an integral part of our solution. To better understand the data, and understand the outcomes of our models, we visualized the data before and after processing and modeling to have a visual representation of the data.

Our solution consists of multiple machine learning models being tested with the objective of finding the model with best accuracy fitting the type of data we are providing. The trials tested the same data and the same partitioning of the data. The only difference was the type of algorithm/model applied to the data. Evaluation of the data was also consistent across trials, and the model resulting in the best accuracy was considered the most optimum solution. The models used were chosen according to fitness to the type of data we have. For example, our use of KNN was due to the fact that our variables of interest were suitable for location based supervised learning.

## 7. High-Level Architecture Design


Data Cleaning and processing
1. Reformulating column names,
2. dropping insignificant columns,
3. handling null and duplicate values
Processed & clean data
1. Various descriptive graphs about data,
2. correlations between different variables (Correlation heatmap
Data Visualization
Data ready for modeling

A. Data Cleaning and Processing: Various reforms and editing took place. For example, some columns were added as aggregations of original column values from the dataset.

IDs of users were checked if all values were unique. Several columns were dropped from the original dataset as they were deemed insignificant to our objective analysis. Therefore, decreasing the number of columns analyzed, and consequently decreasing the dimensionality of the dataset, which is a good thing for machine learning modeling. Finally, the dataset was entirely checked and cleaned for null values, which was none found. Also various other evaluations and cleaning techniques were applied such as duplicate rows checking, and also changing the date column into a proper datetime format acceptable by machine learning algorithms.

B. Data Visualization: Descriptive figures such as bar charts, scatter plots and others were applied with the objective of better understanding the data, and driving insights off the data that could be utilized in the solution process. For example, through the use of data visualization, we can notice from the TotalSteps, VeryActiveMinutes, FairlyActiveMinutes, and LightlyActiveMinutes Columns that most people do not practice sports because the big difference between Total steps and active steps. Moreover, we notice that people are very active on Tuesday, Wednesday, and Thursday, so we can send motivational messages for people on other days. Also, a scatter plot was drawn of Steps taken and Calories burned during the day, which showed us the distribution of the calories/steps burned for users. This was a significant insightful visual for us to understand the variability of rates with which people burn calories with.

Finally, a correlation heatmap visualization was utilized as required. That was an important step in our research because this tells us insights on important columns or variables in the dataset we use. For example, two columns with very high correlation (~1) means we could drop one of the two columns without negatively affecting the predictive model. Rather, it would improve the model as less data is entered with the same quality of prediction is preserved. In our case, the heatmap helped us notice that the TotalSteps and VeryActiveMinutes Columns have the highest influence on the Calories column.

## 8. Proof, Evaluation, and Discussion

### 1. Data Preparation and Cleaning

Data preparation and cleaning are essential steps to ensure the quality and integrity of the dataset. Initially, we obtained the raw dataset and conducted a thorough examination to identify any inconsistencies or missing values. Fortunately, we found that there were no missing values in the data. We also checked the "Id" column and discovered that there are 33 unique IDs with no duplicates. This step confirms that each record corresponds to a unique individual, which is crucial for accurate analysis.

The original dataset consisted of 15 features: Id, ActivityDate, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDistance, VeryActiveDistance, ModeratelyActiveDistance, LightActiveDistance, SedentaryActiveDistance, VeryActiveMinutes, FairlyActiveMinutes, LightlyActiveMinutes, SedentaryMinutes, and Calories.

After cleaning, we reduced the dataset to 8 significant features by dropping the less important ones. The final dataset includes the following features: Id, ActivityDate, TotalSteps, VeryActiveMinutes, FairlyActiveMinutes, LightlyActiveMinutes, SedentaryMinutes, and Calories. This streamlined dataset focuses on the most relevant attributes for our analysis.

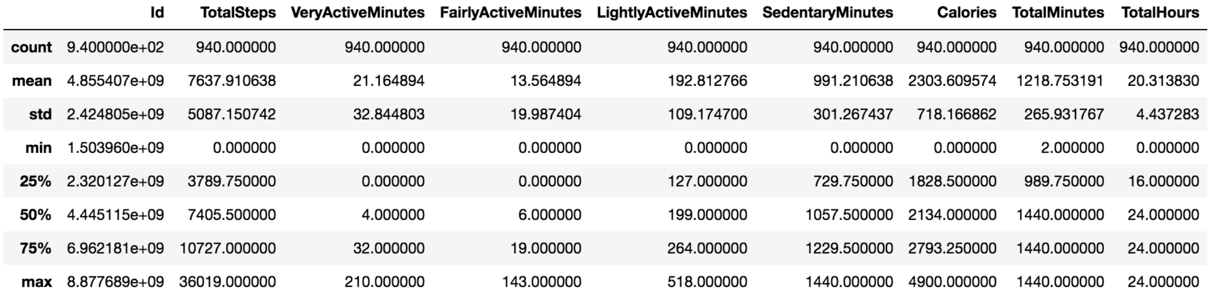
## 2. Data Transformation and Manipulation

Data transformation and manipulation are crucial steps in preparing a dataset for analysis. In this process, the column "ActivityDate" is renamed to "Date" to enhance clarity and consistency. Renaming columns makes the dataset more understandable and aligns it better with the context of the analysis. Next, new columns are created to calculate the total minutes and hours spent on various activity levels, such as Fairly Active, Lightly Active, Sedentary, and Very Active. This aggregation provides a comprehensive view of the total activity duration for each record, facilitating a more detailed analysis of activity patterns.

The dataset is then inspected for any missing values and to ensure that the data types of each column are appropriate for analysis. This step is essential for maintaining data quality and reliability in the subsequent analysis. The "Date" column is converted to a DateTime data type, which allows for easier manipulation and analysis of temporal data. This conversion enables the extraction of the day of the week and other time-based calculations. Additionally, a new column is created to extract the day of the week from the "Date" column. This provides further insights into the distribution of activities across different days, enriching the analysis.

Finally, the dataset is scanned for duplicate rows. Identifying and handling duplicate records is crucial for ensuring data integrity and avoiding biases in the analysis due to redundant information.

## 3. Data Visualization



| | Id | TotalSteps | VeryActiveMinutes | FairlyActiveMinutes | LightlyActiveMinutes | SedentaryMinutes | Calories | TotalMinutes | TotalHours |
|---|---|---|---|---|---|---|---|---|---|
| **count** | 9.400000e+02 | 940.000000 | 940.000000 | 940.000000 | 940.000000 | 940.000000 | 940.000000 | 940.000000 | 940.000000 |
| **mean** | 4.855407e+09 | 7637.910638 | 21.164894 | 13.564894 | 192.812766 | 991.210638 | 2303.609574 | 1218.753191 | 20.313830 |
| **std** | 2.424805e+09 | 5087.150742 | 32.844803 | 19.987404 | 109.174700 | 301.267437 | 718.166862 | 265.931767 | 4.437283 |
| **min** | 1.503960e+09 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 2.000000 | 0.000000 |
| **25%** | 2.320127e+09 | 3789.750000 | 0.000000 | 0.000000 | 127.000000 | 729.750000 | 1828.500000 | 989.750000 | 16.000000 |
| **50%** | 4.445115e+09 | 7405.500000 | 4.000000 | 6.000000 | 199.000000 | 1057.500000 | 2134.000000 | 1440.000000 | 24.000000 |
| **75%** | 6.962181e+09 | 10727.000000 | 32.000000 | 19.000000 | 264.000000 | 1229.500000 | 2793.250000 | 1440.000000 | 24.000000 |
| **max** | 8.877689e+09 | 36019.000000 | 210.000000 | 143.000000 | 518.000000 | 1440.000000 | 4900.000000 | 1440.000000 | 24.000000 |

***Figure 1: Summary of Descriptive Statistics Table***

As shown aove, the summary of descriptive statistics table, which includes the following:

- Count: The number of non-null values in the column.
- Mean: The average value of the column.
- Minimum: The smallest value in the column.
- 25th Percentile (Q1): The value below which 25% of the data fall.
- Median (50th Percentile or Q2): The middle value of the column when the data is sorted.
- 75th Percentile (Q3): The value below which 75% of the data fall.
- Maximum: The largest value in the column.
- Standard Deviation: The measure of the amount of variation or dispersion in the values.

We can notice from the TotalSteps, VeryActiveMinutes, FairlyActiveMinutes, and LightlyActiveMinutes Columns that most people do not practice sports because the big difference between Total steps and active steps

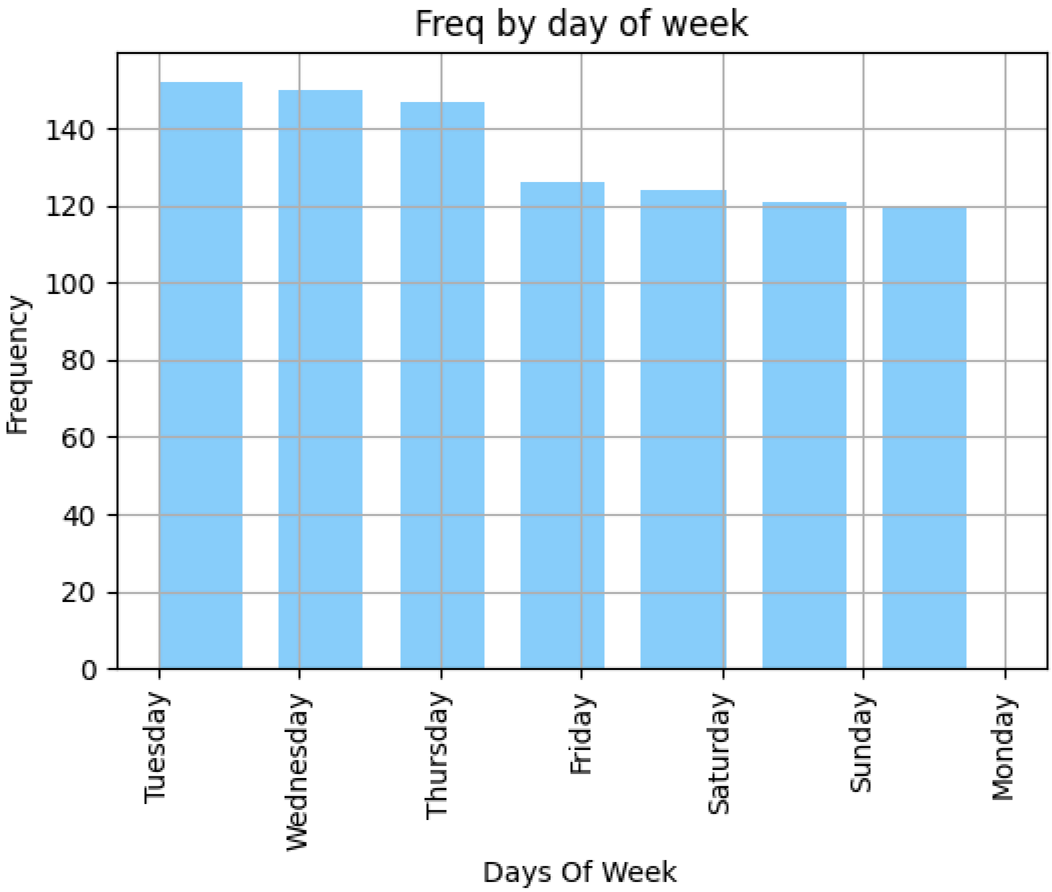


***Figure 2: Frequency by day of week graph***

As shown in figure 2, we can see that people are very active in tuesday, wednesday, and thursday, so it is recommended to send motivation message for people in the other days. As we can see that people are less active on monday and sunday.

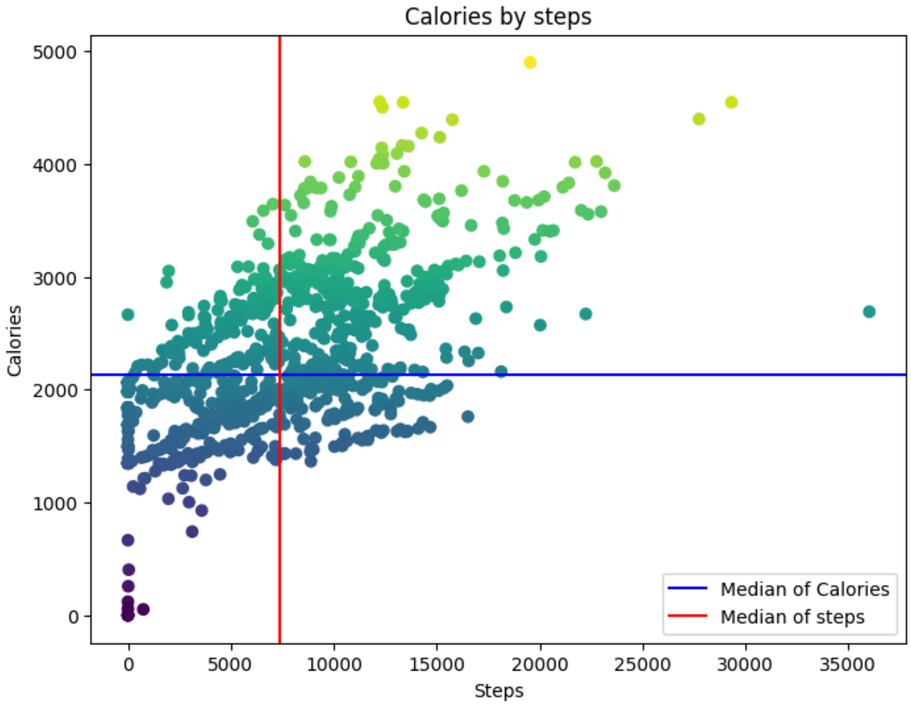


***Figure 3: Calories by steps graph***

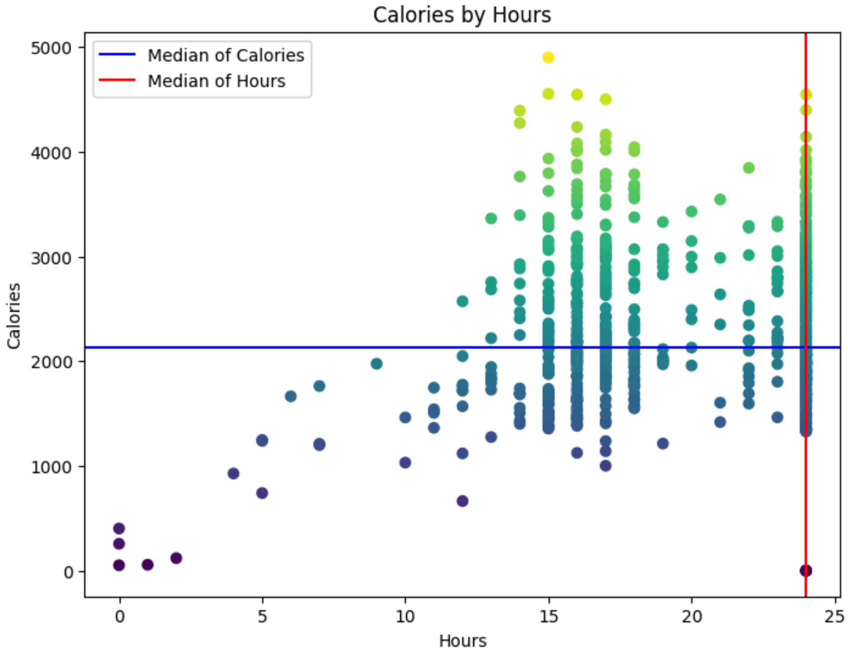


***Figure 4: Calories by hours graph***

To visualize the relationship between the TotalSteps column and the Calories column, we plotted figures 3 and 4. As shown in figure 3 calories by steps, the red line represents median of steps while blue represents median of calories. While in figure 4 calories by hours, the red line represents median of hours while blue represents median of calories. We notice that there is a weak relationship between them, and this happened because the few number of active minutes.

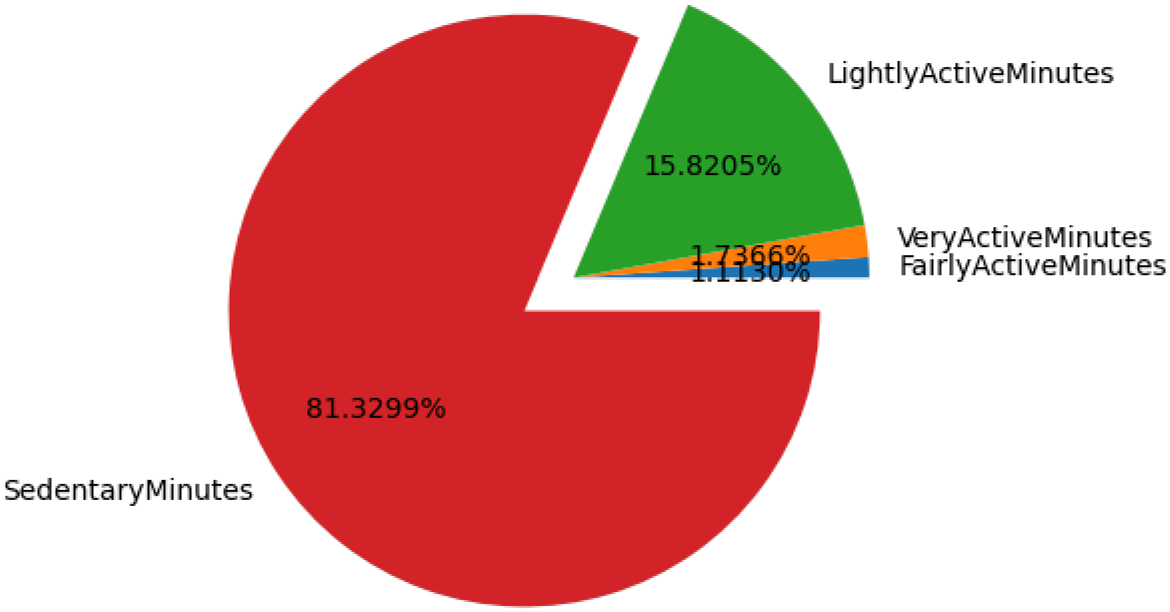


***Figure 5: The Percentae of each Variable Pie Chart***

To visualize the percentage of each of the following variables: VeryActiveMinutes, FairlyActiveMinutes, LightlyActiveMinutes, SedentaryMinutes, we plotted the pie chart shown above. We can say that 81 percent of users use the program to calculate calories burned in normal daily activities, and they are also very active in the middle and end of the week.

## 4. Correlation Heatmap

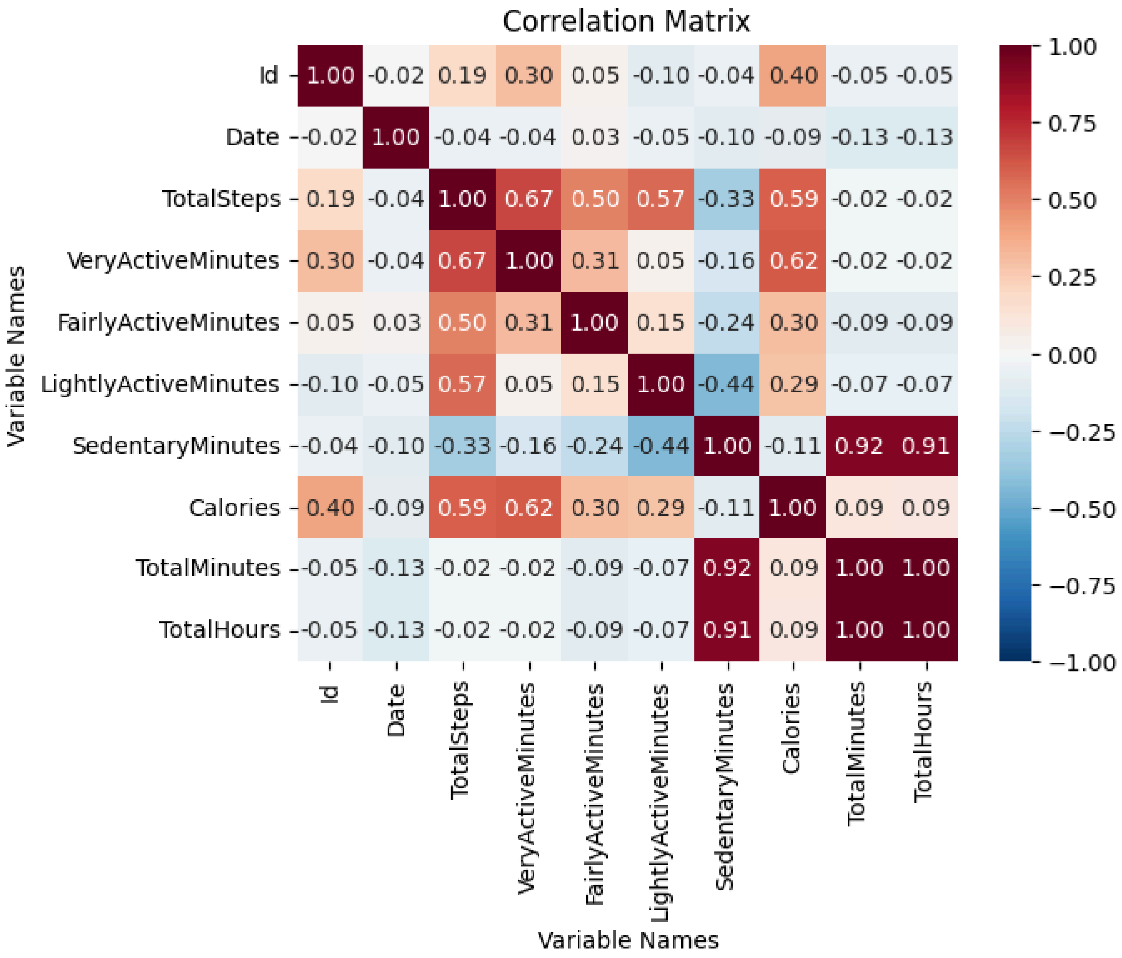


***Figure 6: Correlation Heatmap***

As shown in fiure 6 the correlation heatmap is displayed. The heatmap provides a visual representation of the correlation between pairs of variables in a dataset. There are various correlations as we can seee for example we  have strong correlation when the correlation coefficient is close to 1 or -1, indicating a strong linear relationship between variables. A correlation coefficient of 1 implies a perfect positive linear relationship, while -1 implies a perfect negative linear relationship. For example, there a positive strong correlation between SedentaryMinutes and TotalMinutes. There are also weak correlation  when the correlation coefficient is close to 0, indicating a weak or no linear relationship between variables. A correlation coefficient of 0 suggests no linear relationship between variables. For example, there is a weak  correlation between date and Id and FairyActiveMintues and Id. From the correlation heatmap we can notice that the TotalSteps and VeryActiveMintues Columns have the highest influence on the Calories column.

5. **Modeling with H2O.**

Because data was unavailable for Fatigue Assessment Score, the data column VeryActiveMinutes was utilized as a replacement for fatigue score data for this paper. Here are the steps followed to model with H2O:

1. Importing Libraries:
- The code starts by importing the necessary libraries from H2O, including different types of estimators (models) such as Gradient Boosting, Deep Learning, Random Forest, XGBoost, Naive Bayes, and Generalized Linear models.
2. Initializing H2O:
- The h2o.init() function initializes the H2O cluster, allowing us to use H2O functionalities for machine learning tasks.
3. Loading Dataset:
- The dataset is loaded using the h2o.import_file() function, which reads the data from a CSV file ("dailyActivity_merged.csv") into an H2O frame. This frame is the data structure used by H2O for handling datasets.
4. Splitting Data:
- The dataset is split into training and test sets using the split_frame() function. 80% of the data is allocated for training (train), and 20% for testing (test). The seed parameter ensures reproducibility by setting the random seed for the split.
5. Defining Response Column:
- The response column is specified as "VeryActiveMinutes", indicating the target variable we aim to predict. This column represents the number of very active minutes, which could be a target for prediction in our machine learning models.
6. Defining Algorithms:
- A list of classification algorithms, along with their names and corresponding H2O estimator classes, is defined. These algorithms include Gradient Boosting, Deep Learning, Random Forest, XGBoost, and Generalized Linear Models (GLM).
7. Model Training:
- For each algorithm in the list, a model is trained using the train() function. The input features (x) are specified as all columns in the dataset (data.columns), and the response variable (y) is set to the response column ("VeryActiveMinutes"). The model is trained on the training data (train).
8. Storing Models:
- The trained models are stored in a dictionary (models) using the algorithm names as keys.
9. Model Evaluation:
- After training, each model is evaluated using Mean Squared Error (MSE) as the evaluation metric on the test set. The performance metrics for each model are stored in a dictionary (evaluations).

10. Selecting the Best Model:

- The best-performing model is identified based on the lowest MSE value. The name of the best model along with its MSE is printed.

11. Shutdown H2O:

- Finally, the H2O cluster is shut down using h2o.shutdown(), releasing the resources used by H2O.

## 6. Comparing the Results of this phase with the referenced paper

**Phase Results**

GBM model MSE: 96.73375184333615

DeepLearning model MSE: 111.18498668411308

RandomForest model MSE: 103.84657880434781

**XGBoost model MSE: 73.26421453191911**

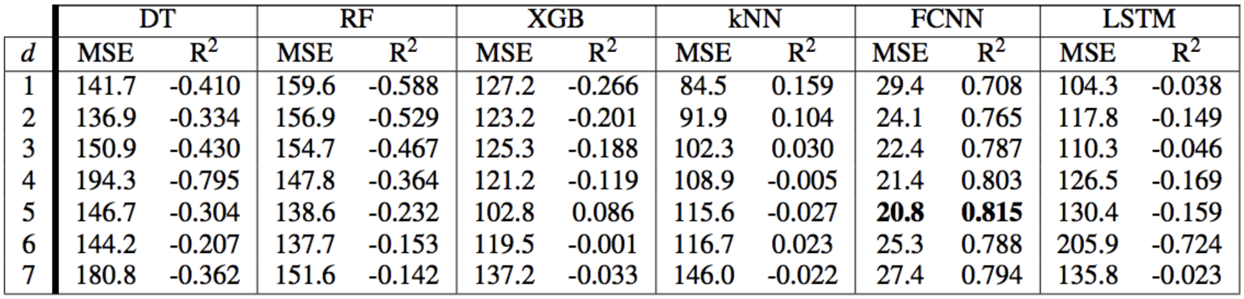
GLM model MSE: 186.09144360812093

**The best model is XGBoost with MSE: 73.26421453191911**

The best-performing model in this phase is  gradient boosting machine, achieved an MSE of 73.26, demonstrating the efficacy of the solution in addressing the problem at hand. XGBoost (Extreme Gradient Boosting) is an optimized implementation of gradient boosting machines designed for speed and performance. It improves upon traditional gradient boosting by introducing regularization techniques and parallelization, leading to faster training and better accuracy.

**Paper Results**

Table 5: Model performance of the six different ML algorithms for estimating the Fatigue Assessment Score (FAS). *d* represents the number of time steps (days) of the input features that were used as input to the model to estimate the FAS-score.



| | DT | | RF | | XGB | | kNN | | FCNN | | LSTM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *d* | MSE | $R^2$ | MSE | $R^2$ | MSE | $R^2$ | MSE | $R^2$ | MSE | $R^2$ | MSE | $R^2$ |
| 1 | 141.7 | -0.410 | 159.6 | -0.588 | 127.2 | -0.266 | 84.5 | 0.159 | 29.4 | 0.708 | 104.3 | -0.038 |
| 2 | 136.9 | -0.334 | 156.9 | -0.529 | 123.2 | -0.201 | 91.9 | 0.104 | 24.1 | 0.765 | 117.8 | -0.149 |
| 3 | 150.9 | -0.430 | 154.7 | -0.467 | 125.3 | -0.188 | 102.3 | 0.030 | 22.4 | 0.787 | 110.3 | -0.046 |
| 4 | 194.3 | -0.795 | 147.8 | -0.364 | 121.2 | -0.119 | 108.9 | -0.005 | 21.4 | 0.803 | 126.5 | -0.169 |
| 5 | 146.7 | -0.304 | 138.6 | -0.232 | 102.8 | 0.086 | 115.6 | -0.027 | **20.8** | **0.815** | 130.4 | -0.159 |
| 6 | 144.2 | -0.207 | 137.7 | -0.153 | 119.5 | -0.001 | 116.7 | 0.023 | 25.3 | 0.788 | 205.9 | -0.724 |
| 7 | 180.8 | -0.362 | 151.6 | -0.142 | 137.2 | -0.033 | 146.0 | -0.022 | 27.4 | 0.794 | 135.8 | -0.023 |

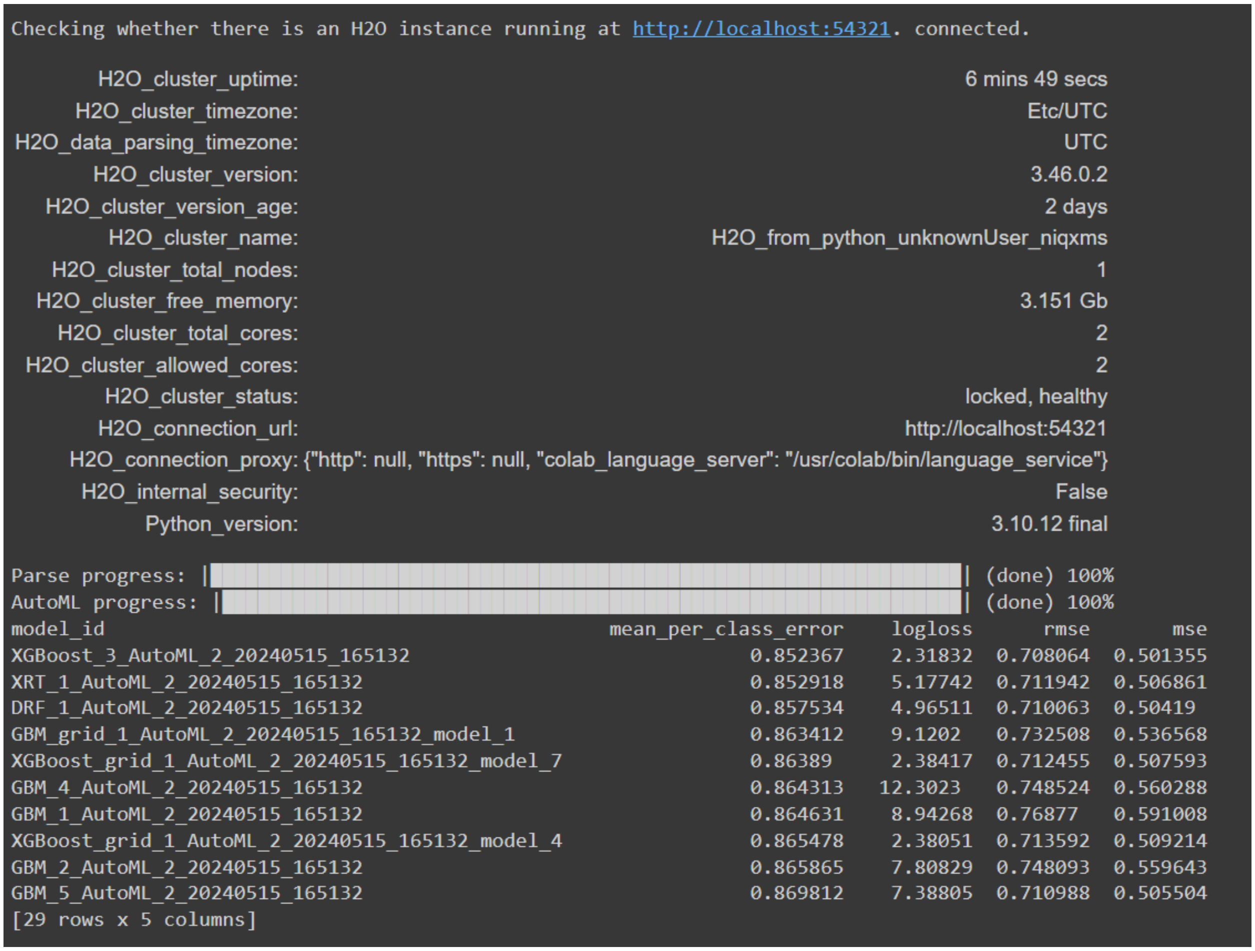
The best performing model, made with a FCNN  (Fully Convolutional Neural Network) with d = 5, is highlighted in bold typeface, with MSE = 20.8.

## Comparing Phase and Paper Results

There are some differences between this phase and the paper which are:

1. In the paper they used Fatigue Assessment Score, but because this data was not available so we decided to use the data column VeryActiveMinutes as a replacement for fatigue score data for the sake of this paper.
2. In this phase, we implemented the following models: Gradient Boosting, Deep Learning, Random Forest, XGBoost, and Generalized Linear Models (GLM). In thw paper, they implemented the following models: Decision Tree (DT), Random Forest (RF), XGBoost (XGB), k-Nearest Neighbor (kNN), Fully-Connected Neural Network (FCNN) and LSTM (Long Short- Term Memory) neural network.
3. To assess model performance, we both used Mean Squared Error (MSE).
4. In this phase, the best performing model is XGBoost with MSE: 73.264 while in the paper FCNN  (Fully Convolutional Neural Network) with d = 5, is highlighted in bold typeface, with MSE = 20.8.

## AutoML leaderboard, ROC,  and AUC

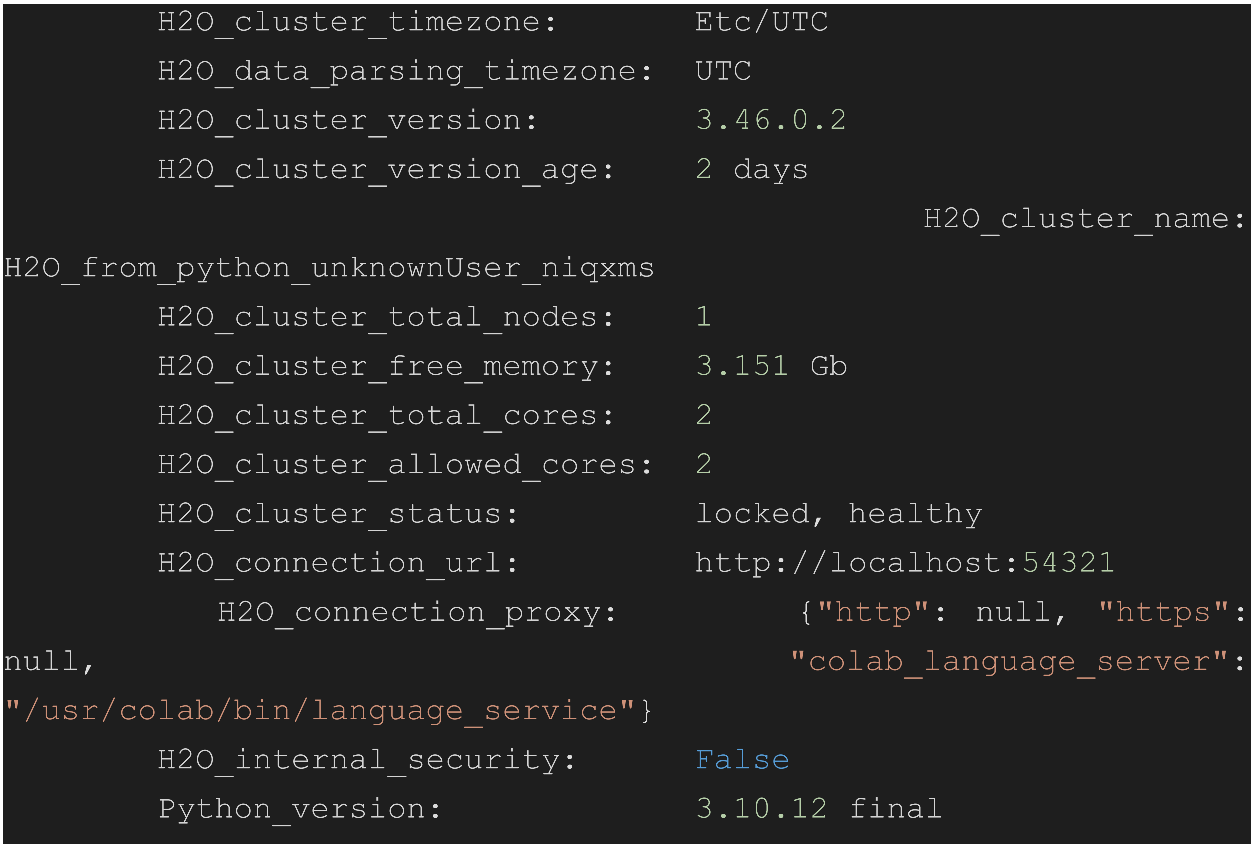


```
Checking whether there is an H2O instance running at http://localhost:54321. connected.

         H2O_cluster_uptime:                                            6 mins 49 secs
       H2O_cluster_timezone:                                                   Etc/UTC
 H2O_data_parsing_timezone:                                                       UTC
        H2O_cluster_version:                                                  3.46.0.2
    H2O_cluster_version_age:                                                    2 days
           H2O_cluster_name:                        H2O_from_python_unknownUser_niqxms
    H2O_cluster_total_nodes:                                                         1
   H2O_cluster_free_memory:                                                  3.151 Gb
     H2O_cluster_total_cores:                                                        2
   H2O_cluster_allowed_cores:                                                        2
          H2O_cluster_status:                                           locked, healthy
        H2O_connection_url:                                      http://localhost:54321
      H2O_connection_proxy: {"http": null, "https": null, "colab_language_server": "/usr/colab/bin/language_service"}
       H2O_internal_security:                                                    False
             Python_version:                                               3.10.12 final

Parse progress: |████████████████████████████████████████████████████████████████| (done) 100%
AutoML progress: |███████████████████████████████████████████████████████████████| (done) 100%
model_id                                                  mean_per_class_error    logloss      rmse       mse
XGBoost_3_AutoML_2_20240515_165132                                    0.852367    2.31832  0.708064  0.501355
XRT_1_AutoML_2_20240515_165132                                        0.852918    5.17742  0.711942  0.506861
DRF_1_AutoML_2_20240515_165132                                        0.857534    4.96511  0.710063  0.50419
GBM_grid_1_AutoML_2_20240515_165132_model_1                           0.863412    9.1202   0.732508  0.536568
XGBoost_grid_1_AutoML_2_20240515_165132_model_7                       0.86389     2.38417  0.712455  0.507593
GBM_4_AutoML_2_20240515_165132                                        0.864313   12.3023   0.748524  0.560288
GBM_1_AutoML_2_20240515_165132                                        0.864631    8.94268  0.76877   0.591008
XGBoost_grid_1_AutoML_2_20240515_165132_model_4                       0.865478    2.38051  0.713592  0.509214
GBM_2_AutoML_2_20240515_165132                                        0.865865    7.80829  0.748093  0.559643
GBM_5_AutoML_2_20240515_165132                                        0.869812    7.38805  0.710988  0.505504
[29 rows x 5 columns]
```

```
H2O_cluster_uptime:          6 mins 49 secs
```



```
        H2O_cluster_timezone:       Etc/UTC
        H2O_data_parsing_timezone:  UTC
        H2O_cluster_version:        3.46.0.2
        H2O_cluster_version_age:    2 days
                                                        H2O_cluster_name:
H2O_from_python_unknownUser_niqxms
        H2O_cluster_total_nodes:    1
        H2O_cluster_free_memory:    3.151 Gb
        H2O_cluster_total_cores:    2
        H2O_cluster_allowed_cores:  2
        H2O_cluster_status:         locked, healthy
        H2O_connection_url:         http://localhost:54321
           H2O_connection_proxy:         {"http": null, "https":
null,                                         "colab_language_server":
"/usr/colab/bin/language_service"}
        H2O_internal_security:      False
        Python_version:             3.10.12 final
```

**Conclusion:**

Given the AutoML leaderboard, XGBoost was the most accurate model according to the evaluation tools used. We concluded that this model, which was not used in the main paper we referenced in this paper, would have been a significant addition to the paper and to the results of the main objective. The fatigue detection problem is a challenging yet important problem, and the constant exploring of different models concerning this problem is vital for the realization of these objectives. The use of advanced big data technologies added significant support to the research process, since the amount of data available can not be processed with conventional technologies. Rather, big data technologies used, such as H2O, made the process more feasible. Through H2O AutoML technology, we were able to compare many machine learning models together in record time and efficiency on cluster. The use of machine learning and big data technologies, especially H2O AutoML, was an integral part of this paper.

## References

[1] Bustos, D., et al. "Machine learning approach to model physical fatigue during incremental exercise among firefighters." Sensors 23.1 (2022): 194.

[2] Pinto-Bernal, M. J., et al. "A data-driven approach to physical fatigue management using wearable sensors to classify four diagnostic fatigue states." Sensors 21.19 (2021): 6401.

[3] Bai, Y., Guan, Y., & Ng, W.-F. "Fatigue assessment using ECG and actigraphy sensors." Proceedings of the 2020 International Symposium on Wearable Computers (2020).

[4] Albadawi, Y., AlRedhaei, A., & Takruri, M. "Real-time machine learning-based driver drowsiness detection using visual features." Journal of Imaging 9.5 (2023): 91.

[5] Cos, C.-A., et al. "Enhancing mental fatigue detection through physiological signals and machine learning using contextual insights and efficient modelling." Journal of Sensor and Actuator Networks 12.6 (2023): 77.

[6] Hooda, R., Joshi, V., & Shah, M. "A comprehensive review of approaches to detect fatigue using machine learning techniques." Chronic Diseases and Translational Medicine 8.1 (2022): 26–35.

[7] Husom, E., et al. "Machine learning for fatigue detection using Fitbit Fitness Trackers." Proceedings of the 10th International Conference on Sport Sciences Research and Technology Support (2022).

[8] Leroux, A., et al. "Wearable devices: Current status and opportunities in pain assessment and management." Digital Biomarkers 5.1 (2021): 89–102.

[9] Lu, H., et al. "Stresssense." Proceedings of the 2012 ACM Conference on Ubiquitous Computing (2012).

[10] Luo, H., et al. "Assessment of fatigue using wearable sensors: A pilot study." Digital Biomarkers 4.Suppl. 1 (2020): 59–72.