Volume 10, Issue 3, July 2023 ISSN: 2348-8565 (Online)

International Journal of Modern Engineering and Research Technology

Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

Exploratory Data Analysis and Regression Approach for Estimating Burnt Calories During Exercise

Srishti Sahu
Research Scholar M.Tech.
Computer Science and Engineering
Takshshila Institute of Engineering and Technology
Jabalpur (M.P.), India
Email: srishti.sahu2302@gmail.com

Swati Soni
Assistant Professor
Department of Computer Science and Engineering
Takshshila Institute of Engineering and Technology
Jabalpur (M.P.), India
Email: swatisoni@takshshila.org

ABSTRACT

This study focuses on the task of Exploratory Data Analysis (EDA) and Regression Modeling to estimate burnt calories during exercise. The dataset used for the analysis is sourced from Kaggle and consists of two main datasets: "Calories Dataset" and "Exercise Dataset."

The "Calories Dataset" contains information about unique user IDs and the corresponding calories burnt during exercise. On the other hand, the "Exercise Dataset" includes various features such as user ID, gender, age, height, weight, exercise duration, heart rate, and body temperature.

Given the nature of the target variable "Calories," a Regression approach is deemed appropriate for building a predictive model. Regression algorithms are utilized to predict the exact numerical value of calories burnt based on the provided input features.

The performance of several Regression models, including Linear Regression (LRmodel), Ridge Regression (RDmodel), Lasso Regression (LSmodel), Decision Tree Regression (DTmodel), and Random Forest Regression (RFmodel), is evaluated using metrics such has Mean Squared Error (MSE) and R-squared (R²).

The evaluation results reveal that the Random Forest Regression model (RFmodel) stands out as the best-performing model for this task. It achieved the lowest Mean Squared Error (7.2952197666666665) and the highest R-squared (0.9981425925608396) among all the evaluated models. This indicates the RFmodel's superior ability to accurately estimate burnt calories during exercise based on the provided input features.

Overall, this study demonstrates the significance of Exploratory Data Analysis and Regression Modeling in accurately estimating burnt calories during exercise, which could be valuable for various fitness and health-related applications.

Keywords:—*Burnt Calories Prediction, EDA, Linear Regression, Ridge, Lasso, RF, DT, MSE*

I. INTRODUCTION

A. What happen when we Exercise?

When we exercise, our body goes through a series of changes to supply energy to the working muscles and support the increased activity. Here's a step-by-step explanation of what happens during exercise:

Increased Energy Demand: Exercise requires energy to power the muscles and keep them moving.

Carbohydrate Metabolism: The food we eat, especially carbohydrates, is broken down into simpler forms like glucose during digestion. Glucose is a primary source of energy for the body.

Oxygen Utilization: Our body prefers to use oxygen to efficiently produce energy. The breakdown of glucose into energy (ATP) happens in the presence of oxygen in the cells' mitochondria.

Increased Oxygen Consumption: During exercise, our body needs more oxygen to support the increased energy demand. To get more oxygen, we breathe faster and deeper.

Heart Rate Increase: The heart pumps oxygen-rich blood to the muscles. As the demand for oxygenrises during exercise, the heartrate increases to ensure sufficient oxygen supply to the working muscles. The typical resting heart rate is around 75 beats per minute but can go up to around 100 beats per minute or even higher during intense exercise.

Increased Blood Flow: The increased heart rate results in higher blood flow to deliver oxygen and nutrients to the muscles and remove waste products like carbon dioxide.

Body Temperature Regulation: Exercise generates more heat as a byproduct of energy production. To maintain a stable internal temperature, the body regulates its heat by increasing blood flow to theskin'ssurface.Theheatisthendissipatedthroughsweating,whichcoolsthebodyasthesweatevaporates from the skin.

Sweat Production: Sweat is mostly composed of water and some electrolytes. Sweating helps to cool the body down as the evaporation of sweat from the skin's surface absorbs heat.

Hence, during exercise, the body breaks down carbohydrates (glucose) to produce energy using oxygen. This increased energy demand leads to a higher heart rate, increased blood flow, and body temperature regulation throughs weating. Exercise provides various physical and mental health benefits and is an essential aspect of a healthy lifestyle.

Figure 1: What happen when we Exercise

B. Machine Learning

Machine learning encompasses a variety of techniques and approaches that enable computers to learn from data and make predictions or decisions without explicit programming. The three main types of Machine Learning you mentioned are:

***Supervised Learning**:* In supervised learning, the algorithm is trained on a labeled dataset, where the input data is paired with corresponding target outputs. The goal is to learn a mapping from inputs to outputs, allowing the model to make accurate predictions for new, unseen data.

Unsupervised Learning: Unsupervised learning involves training the algorithm on a nun labeled dataset, where the model aims to find patterns, structure, or relationships within the data. Common tasks in unsupervised learning include clustering similar data points together or reducing the dimensionality of the data.

Reinforcement Learning: Reinforcement learning is concerned with training algorithms to make decisions in an environment to maximize rewards over time. The model learns by interacting with the environment, receiving feedback in the form of rewards or penalties based on its actions.

Machine learning is a sub field of artificial intelligence(AI) that focuses on the development to f algorithms and models that enable computers to learn and make decisions or predictions based on data without being explicitly programmed for those tasks. Machine learning allows systems to improve their performance over time through experience.

There are two main types of tasks in machine learning: classification and regression.

Classification: Classification is a type of supervised learning where the goal is to assign input data points to specific categories or classes. In other words, the model learns to map input data to predefined output classes.

Regression: Regression is another type of supervised learning where the goal is to predict continuous numerical values based on input data. In regression tasks, the model learns to establish a relationship between the input features and the output variable.

classification is used for tasks where the output variable is categorical, and the goal is to assign data points to predefined classes. Regression is used for tasks where the output variable is continuous, and the objective is to predict numerical values.

C. Linear Regression

Linear regression is a popular statistical method used to model the relationship between a dependent variable (also known as the target or outcome variable) and one or more independent variables (also known as predictors or features). It assumes that there is a linear relationship between the independent variables and the dependent variable.

The goal of linear regression is to find the best-fitting line (or hyperplane, in the case of multiple independent variables) that minimizes the difference between the actual and predicted values of the dependent variable. The line is defined by a slope (also known as weight or coefficient) and an intercept.

The equation $\mathbf{Y_i = b_1 x_i + b_0}$(y = mx + c) represents a simple linear regression model, where:

$\mathbf{Y_i}$is the dependent variable (the one being predicted).

$\mathbf{x_i}$is the independent variable (the predictor).

$\mathbf{b_1}$is the Regression Slope or coefficient, representing the change in Y for a one-unit change in x.

$\mathbf{b_0}$is the intercept, representing the value of Y when x is 0.

This equation is to fit a straight line to a set of data points in a way that minimizes the differences between the actual Y values and the values predicted by the equation for corresponding x values.

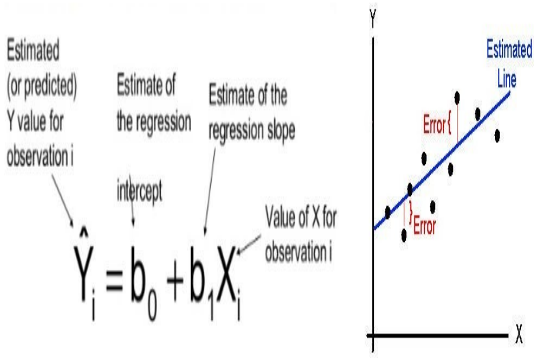

Estimated (or predicted) Y value for observation i
Estimate of the regression intercept
Estimate of the regression slope
Value of X for observation i
$\hat{Y}_i = b_0 + b_1 X_i$
Y
Estimated Line
Error
Error
X

Figure 2: Linear Regression

D. Ridge Regression

The Ridge Regressor serves several purposes and is useful in various scenarios:

Dealing with multicollinearity: When dealing with data sets that have highly correlated features, ordinary line a regression may lead to unstable coefficients timates. Ridge Regression can mitigate the impact of multicollinearity by adding regularization, which helps to stabilize the model and reduce the variance of the coefficient estimates.

Preventing over fitting: In situations where the number of features is significantly larger than the number of samples, ordinary line arregression might be prone to over fitting. Ridge Regression's regularization term restricts the magnitude of the coefficients, preventing them from growing too large and over fitting the training data. This improves the model's ability to generalize to unseen data.

Improving generalization: By controlling the complexity of the model through regularization, Ridge Regression often improves the generalization performance on new, unseen data. It reduces the risk of over fitting and provides a more robust model.

Handling noisy data: In the presence of noisy data, ordinary linear regression can be sensitive to outliers and noise. Ridge Regression's regularization helps to dampen the effect of outliers, making the model more resilient to noisy data.

Combining features: Ridge Regression's regularization can encourage the model to use all the available features to some extent, rather than relying heavily on just a few of them. This can be advantageous when you believe that many features are relevant to the target variable but don't want to completely exclude any of them.

Default choice for linear regression with regularization: When you have no prior knowledge about the importance of specific features and want to use a simple regularization method, Ridge Regression is a sensible default choice for linear regression.

Squared residuals and squared coefficients are terms used in the context of Ridge Regression. Let's define each of them:

Squared Residuals: In the context of regression, residuals refer to the differences between the predicted values (output of the model) and the actual target values (ground truth). The squared residuals areobtained by taking the square of these differences. Mathematically, for each data point i, the squared residual can be calculated as:

$$\textbf{Squared Residual(i) = (Predicted Value(i) - Actual Value(i))}^2$$

The overall sum of squared residuals across all data points is one of the components of the loss function in linear regression. The objective of linear regression is to minimize this sum, aiming to find the best-fitting line or hyper plane that minimizes the prediction error.

Squared Coefficients: In the context of Ridge Regression, the squared coefficients refer to the squared values of the regression coefficients (weights) of the features in the model. When performing Ridge Regression, a regularization term is added to the loss function, which penalizes large coefficients. The squared coefficients term is obtained by taking the square of each coefficient and summing them up.

$$\textbf{Squared Coefficients = Sum of (Coefficient(i)}^2\textbf{) for all features i}$$

The regularization term in the Ridge Regression loss function is proportional to this sum of squared coefficients. There gularization parameter (alpha) controls the strength of this penalty. As alpha increases, the impact of the regularization term on the loss function becomes stronger, resulting in smaller coefficients and a more regularized model.

By adding the squared coefficients term to the loss function, Ridge Regression can balance between fitting the training data and keeping the coefficient values small, which helps prevent over fitting and improves the model's generalization ability. This regularization technique is known as L2 regularization, and it is a fundamental concept in Ridge Regression.

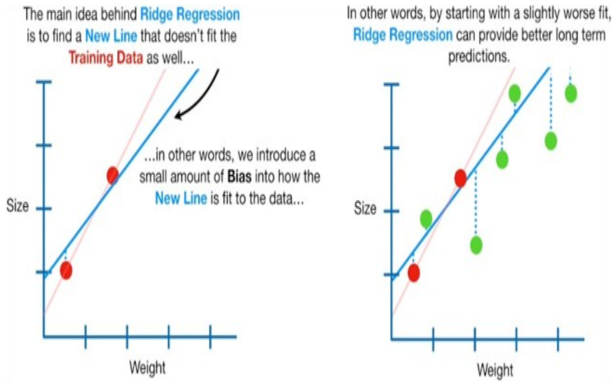


Figure 3: Ridge Regression

E. Lasso Regression

The key idea behind the Lasso Regressor is to add a penalty term to the traditional linear regression cost function, which is based on the sum of squared errors. The Lasso penalty term is the L1 norm of the coefficient vector, multiplied by a regularization parameter (lambda or alpha). The L1 norm is the sum of the absolute values of the coefficients. This penalty term encourages many coefficients to become exactly zero, effectively performing feature selection by eliminating less important or irrelevant features from the model.

Mathematically, the Lasso cost function can be represented as follows:

Lasso Cost Function = Sum of Squared Errors + lambda * (sum of absolute values of coefficients)

The Lasso Regressor works by optimizing this cost function during the model training process. The regularization parameter lambda controls the amount of regularization applied to the coefficients. A larger value of lambda increases the penalty on the coefficients, making it more likely for some coefficients to become exactly zero, resulting in sparsity. The process of Lasso regression involves the following steps:

Data Preparation: As with any regression task, the dataset is divided into features (input variables) and the target variable (output variable).

Feature Scaling (Optional): It is often beneficial to scale the features before applying Lasso regression to ensure that all features have similar magnitudes.

Define the Lasso Cost Function: The Lasso cost function is defined by adding the L1 norm of the coefficient vector to the traditional sum of squared errors.

Minimization: The objective is to minimize the Lasso cost function by adjusting the coefficients.

Feature Selection: During the optimization process, the Lasso penalty may shrink some coefficients to exactly zero, effectively removing the corresponding features from the model.

Regularization Strength: The amount of regularization applied is controlled by the regularization parameter lambda. The optimal value of lambda can be determined through techniques like cross-validation.

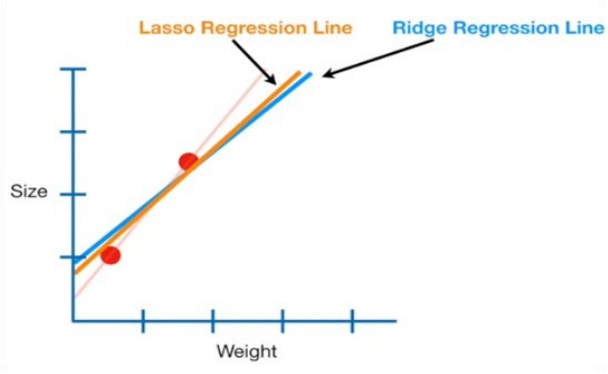


Figure 4: Ridge Regression

F. Decision Tree Regressor

A Decision Tree Regressor is a machine learning algorithm used for regression tasks. It's a type of decision tree model that is designed to predict continuous numeric values as opposed to categorical labels. Decision trees, in general, are a type of supervised learning algorithm that makes decisions based on asking a series of questions about the input features and eventually arriving at a prediction.

Here's how a Decision Tree Regressor works:

Tree Structure: The algorithm constructs a tree-like structure where each node represents a decision based on a specific feature and a threshold value. The tree structure consists of nodes and branches. The top node is called the root node, and the nodes that follow are internal nodes or leaf nodes.

Splitting Criteria: At each internal node, the algorithm selects the feature and threshold that best splits the data into subsets. The objective is to minimize the variance of the target values within each subset.

Leaf Nodes: As the tree grows, the algorithm continues to split the data based on the selected features and thresholds. Eventually, it stops when a predefined stopping criterion is met, such as reaching a maximum depth, having a minimum number of samples at a node, or not achieving a significant reduction in variance through splitting.

Predictions: When a new instance is fed into the trained Decision Tree Regressor, it traverses the tree from the root node to a leaf node by following the decision rules at each internal node. The predicted value for the new instance is the average of the target values in the training data that belong to the leaf node.

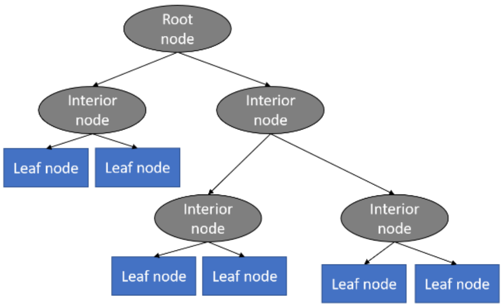


Figure 5: Decision Tree Classifier

G. Random Forest Regressor

A Random Forest Regressor is an ensemble machine learning algorithm that combines multiple Decision Tree Regressors to improve predictive performance and reduce overfitting. It's a popular method for regression tasks where the goal is to predict continuous numeric values. The algorithm creates a “forest” of decision trees and aggregates their predictions to provide a more accurate and robust result. Here's how the Random Forest Regressor works:

Bootstrapped Sampling: The algorithm starts by randomly selecting subsets of the original training data (with replacement). These subsets are called “bootstrap samples.” Each subset is used to train a separate Decision Tree Regressor.

Random Feature Selection: For each decision tree, at each node's split, the algorithm selects a random subset of features from the available features. This randomness helps in creating diverse trees and reducing the correlation among them.

Building Trees: Multiple decision trees are grown based on the bootstrapped samples and random feature selections. Each tree is built until a certain stopping criterion is met, such as reaching a maximum depth or not having enough samples to split further.

Aggregation: When making predictions, each tree in the forest independently predicts the target value for a given input. The final prediction is then obtained by aggregating the individual predictions. For regression tasks, the most common aggregation method is taking the average of the predictions from all the trees.

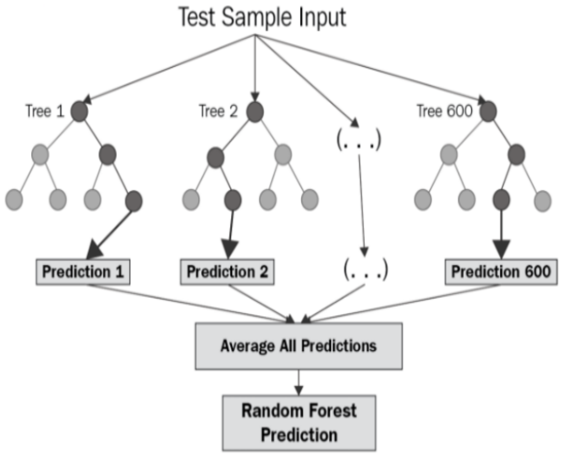


Figure 6: Random Forest Regressor

H. Mean Absolute Error (MAE)

MAE is calculated as the average of the absolute differences between predicted and actual values. Unlike MSE, it treats all errors with equal weight, regardless of their magnitude.

MAE is more robust to outliers because it does not square the errors. It gives a more balanced representation of the overall error, which can be beneficial when dealing with datasets that contain significant outliers.

$$\frac{1}{\text{Total data points}}\sum \left| \text{Actual output} - \text{predicted output} \right|$$

$$\frac{1}{n}\sum \left| y-\hat{y} \right|$$

Figure 7: Mean Absolute Error

Mean Squared Error (MSE)

Mean Squared Error (MSE) and Mean Absolute Error (MAE) are two commonly used metrics to evaluate the performance of regression models. Both metrics measure the difference between the predicted valuesand the actual (ground truth) values of the dependent variable. The main difference between them lies inhow they treat the errors and their sensitivity to outliers.MSE is calculated as the average of the squared differences between predicted and actual values. It giveshigher weight to larger errors due to the squaring operation.

$$\frac{1}{\text{Total data points}}\sum (\text{Actual output} - \text{predicted output})^{2}$$

$$\frac{1}{n}\sum (y-\hat{y})^{2}$$

Figure 8: Mean Absolute Error

Root Mean Square Error (RMSE)

RMSE stands for Root Mean Squared Error. It is a commonly used metric for evaluating the performance of regression models, especially in cases where the dependent variable (target variable) has different units or scales. RMSE is calculated as the square root of the Mean Squared Error (MSE).

$$\sqrt{\frac{1}{\text{Total data points}}\sum (\text{Actual output} - \text{predicted output})^{2}}$$

$$\sqrt{\frac{1}{n}\sum (y-\hat{y})^{2}}$$

Figure 9: Root Mean Square Error

K. R-Squared

Mean Squared Error (MSE) and R-squared (R^2) are two commonly used metrics to evaluate the performance of regression models. They both asses show well the predicted values of the model match the actual target values, but they represent different aspects of model performance.

R-squared is a statistical measure that represents the proportion of the variance in the dependent variable (target) that is predictable from the independent variables (features) in the model. It measures the goodness of fit of the model to the data.

$$\mathbf{R^2 = 1 - (SSR / SST)}$$

SSR (Sum of Squared Residuals) is the sum of the squared differences between the predicted values and the mean of the actual target values.

SST (Total Sum of Squares) is the sum of the squared differences between the actual target values and their mean.

R^2 values range from 0 to 1, where 0 indicates that the model explains none of the variance in the data (it performs no better than predicting the mean), and 1 indicates a perfect fit where the model explains all the variance in the data.

A higher R^2 value indicates a better-performing model, as it suggests that a larger proportion of the variance in the target variable is explained by the model's predictions.

Both **MSE** and **R^2** are essential metrics for evaluating regression models, and they complement each other. While **MSE directly measures the accuracy of the predictions**, **R^2 provides an over all assessment of how well the model fits the data**. It is often recommended to use both metrics together to get a comprehensive understanding of the model's performance.

II. Literature Review

This work focused on using machine learning techniques, specifically regression models, to predict the calories burned by individuals during physical activities. The study involves the following key steps and findings: The abstract introduces the context of the study, which is about predicting calories burned during physical activities using machine learning algorithms, particularly regression models. The goal is to improve the accuracy of calorie burn estimation. **Data Collection and Preparation** The study uses a data set from Kaggle containing 15,000 observations and nine variables. The data underwent preparation, cleaning, and analysis. The dataset contains numeric and categorical variables, and there are no duplicate rows or missing values. **Data Analysis and Feature Extraction**, the variables in the dataset were analyzed to determine their relationship with the target variable, which is calorie burned. Variables like heart rate, duration of exercise, body temperature, height, and weight were found to be highly correlated with the dependent variable. **Model Training and Prediction**, three regression models we reconsidered: **Linear regression, Ridge regression, and Random Forest regression**. The models were trained and tested using **K-fold cross-validation**[1] with ten iterations to prevent over fitting and improve prediction accuracy. **Results and Metrics**, the study evaluated the performance of the regression models using metrics such asMean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). The random forest regression model outperformed the other models, achieving the lowest MSE, RMSE, and MAE values. The study concludes that regression models are effective tools for predicting calorie burn during physical activities. Proper data cleaning and preparation are necessary before feeding the data into the algorithms. Understanding relationships between variables, addressing issues like multicollinearity, and visualizing data are crucial steps. The researchers observed a correlation between heart rate, exercise duration, body temperature, and calorie burn. Careful model selection and training are important for accurate predictions. The study achieved a **95.77% accuracy rate** using the **Random Forest Regression** algorithm with specific hyper parameters.

B. This research project aims to make a comparative study of machine learning algorithms to predict the calories burned during a workout. The researchers used two machine learning algorithms, **XGBoost Regressor[2] and Linear Regression**, to build predictive model sand compared their performance in predicting the calories burned during exercise.The purpose of the study is to predict the number of calories burned during a workout for different individuals and compare the accuracy of the two machine learning algorithms using the given dataset. The dataset contains 7 features, one target variable (calories burned), and **15,000 instances**. The methodology involves collecting an appropriate dataset to train the machine learning models and finding the number of calories each individual is likely to burn during their workout. Data pre-processing and analysis are carried out, including data visualization techniques to understand and prepare the data for training. The dataset is then divided into a training set and a test set for model evaluation. The data source for this research project is **Kaggle, where two CSV files are used**, one containing the attributes of each individual's details, suchas **gender, age, workout duration, heartrate, body temperature, height, and weight**. The second CSV file contains the target class (**calories burned**) for each corresponding person. After evaluating the models, the research team found that the **XGBoost Regressor** performed better, with a **Mean Absolute Error of 2.71**, compared to **Linear Regression**, which hada **Mean Absolute Error of 8.38**. The lower the mean absolute error, the more accurate the model's predictions. In conclusion, the research project demonstrates the use of machine learning algorithms to predict the calories burned during a workout and compares the performance of XGBoost Regressor and Linear Regression, with XGBoost Regressor outperforming Linear Regression in accuracy.

C. The application offers several key functionalities that contribute to a comprehensive approach to promoting a healthy lifestyle and creating an engaging exercise experience: **Health Insights and Tracking[6],** The application provides users with an overview of their health progress over a specific time frame. It achieves this by calculating the total calories burned based on data collected from the wearable sensor. This approach takes into consideration both movement data from the wearable sensor and heart rate data, resulting in a more accurate calculation of calorie expenditure. This improved accuracy enhances the user's understanding of their exercise efforts and enables them to make informed decisions about their fitness routines. **Real-time Data and Connectivity:** The application is connected to a real-time database, ensuring that users' exercise and health data is securely stored and accessible, even if they switch to a different mobile device. This feature enhances the convenience and reliability of the user experience, as individuals can continue their fitness journey seamlessly without worrying about data loss or disruption. **Flexible Exercise Options:** Users have the flexibility to engage in exercises either individually or as part of a group. The application accommodates both preferences, catering to users who may prefer to work out alone or those who enjoy the social aspect of exercising with friends or fellow enthusiasts. Moreover, the application offers exercise sequences guided by models, providing users with clear instructions on how to perform each exercise. This ensures that users are well-informed and can confidently follow a structured routine.

Social and Engaging Environment: One of the project's core features is its ability to enable multiple people to exercise together. This social aspect enhances the overall exercise experience, creating an environment that is not only beneficial for health but also fun and engaging. By allowing users to exercise in groups, the application fosters a sense of camaraderie and competition, which can serve as powerful motivators to encourage users to maintain their exerciser outine.

D. The work described in this text focuses on addressing the issue of obesity by developing an intelligent system that helps users make healthier food choices and provides them with guidance on how to maintain a balanced diet and burn calories through exercise. The system employs Convolutional Neural Networks (CNNs) [10]to classify food items based on images, offering nutritional information and categorizing foods as healthy or non-healthy. The goal is to enable users to monitor their daily calorie intake and make informed dietary decisions. The proposed system aims to provide nutritional information, classify food items, and guide users on maintaining a healthy body. The system includes an Android application that captures food images, classifies them using a CNN model, and provides accurate nutritional information about Indian foods. It also considers the user's health condition, such as blood pressure and diabetes data, to offer personalized food recommendations. **Data Collection**, A dataset of Indian food images was created using web scraping. Since such a data set was not readily available in existing repositories, this dataset was generated specifically for the project. **Convolutional Neural Network (CNN) Model:** The CNN model is utilized for image classification. It consists of convolutional layers for feature extraction, max-pooling layers to reduce variance and complexity, and fully connected layers for classification. The model is capable of handling a large amount of image data and is well-suited for food classification tasks.

The **CNN model achieved an accuracy of 91.65%** in classifying food images. The system was tested with real-time data and hosted on Google Cloud for production use. The mobile application provides users with information about food calories, exercise options, and health reports.

II. METHODOLOGY

This research project seeks to address the problem of accurately predicting calorie burn during workouts by leveraging a diverse range of machine learning algorithms. The primary objective isto identify the most effective algorithm that can offer superior predictive performance.

By conducting an in-depth comparative analysis of these models using metrics like **mean squared error (MSE) and R-squared (R^2), the research aims to pinpoint the algorithm that achieves the highest accuracy.**

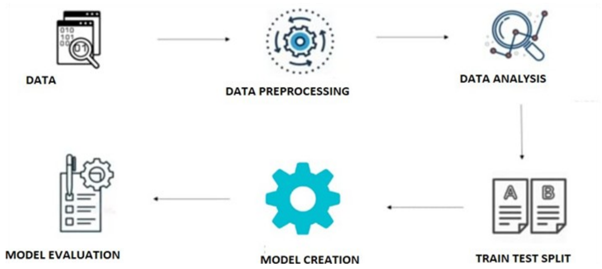


Figure 10: Proposed Work Flow

Burnt Calories Prediction Data Source

Data Sourse - https://www.kaggle.com/datasets/fmendes/fmendesdat263xdemos

Files
- exercise.csv (661 KB)

- calories.csv (225 KB)

Calories Dataset (calories.csv) = User_ID Calories

Exercise Dataset (exercise.csv) = User_ID Gender Age Height Weight Duration Heart_Rate Body_Tempuser Id = Unique User ID,

Gender = Male / Female, Age=AgeofthePerson,

Height = Height of the Person, Weight=WeightofthePerson,

Duration = Exercise / Workout Duration in MinutesHeart_Rate = Heart Rate During ExerciseBody_Temp = Body Tempurature During Exercise

Calories = The number of calories burned during the activity.

Based on the provided dataset with the following columns: “User_ID,” “Gender,” “Age,” “Height,” “Weight,” “Duration,” “Heart_Rate,” “Body_Temp,” and “Calories,” we can see that the task is to predict the value of “Calories” (which is a continuous numerical variable) based on the given features.

Given the nature of the target variable “Calories,” the appropriate approach to solve this problem is Regression.

Regression algorithms will help you build a model to predict the exact numerical value of “Calories” burned based on the input features such as “Gender,” “Age,” “Height,” “Weight,” “Duration,” “Heart_Rate,” and “Body_Temp.”

Calories: Target variable (Continuous Values)

Dimension of diabetes: 15000 rows (data points) × 9 columns (features)

Regression Problem=As All Values of Calories are Numerical & Continuous. Thus, it is Nota Classification Problem.

- IDE - Google Collaborator
- Python – Python 3
- Applied Regression Problem on Data

A. Data Collection & Preprocessing

- Reading two CSV files(calories.csv andexercise.csv) using pandas and merging the m based on the 'User_ID' column to create a single data frame df.
- Checking for missing values in the merged dataframe using df.isnull ().sum().

B. Exploratory Data Analysis (EDA):

- Descriptive statistics of the dataframe using df.describe().
- Visualizations to explore the relationships between various features and the target variable 'Calories':
- Box Plot to visualize the distribution of ‘Age’ for different 'Gender' categories.
- Distribution Plots to visualize the distributions of 'Age' and 'Height'.
- Line Plots to visualize the relationship between ‘Age’ and ‘Calories’, and ‘Calories’ and ‘Heart_Rate’.
- Heatmap to visualize the correlation between different numerical features using sns.heatmap.

C. Data Preprocessing (Continued):

- Label Encoding of the 'Gender' column to convert 'male' to 0 and 'female' to 1, using .map ({'male':0,'female':1}).
- Splitting the dataset into features (x) and target (y).
- Further splitting the data into training and testing sets using train_test_split with 80% training and 20% testing data.

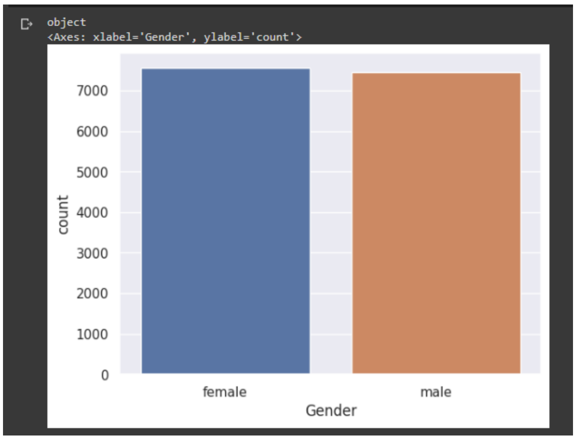
Model Creation:

Five regression models are created and evaluated using Mean Squared Error (MSE) and R-squared asevaluation metrics. The models are:

- Linear Regression Model
- Ridge Regressor Model
- Lasso Regressor Model
- Decision Tree Regressor Model
- Random Forest Regressor Model

E. Best Model Selected and Evaluated:

- The best model is determined based on the lowest Mean Squared Error (MSE) and highest R-squared value.
- Model Evaluation is Done by Giving Feature Values and Predicting Target Values.



```
object
<Axes: xlabel='Gender', ylabel='count'>
```

Figure 11: Univariate Analysis for Feature Gender

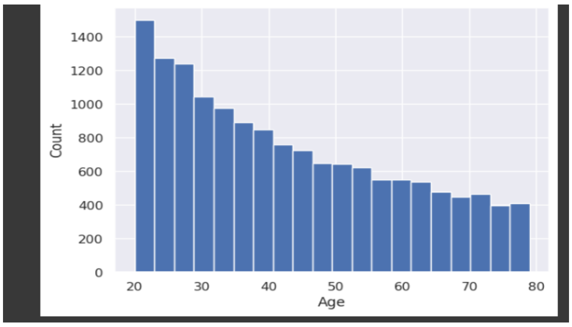
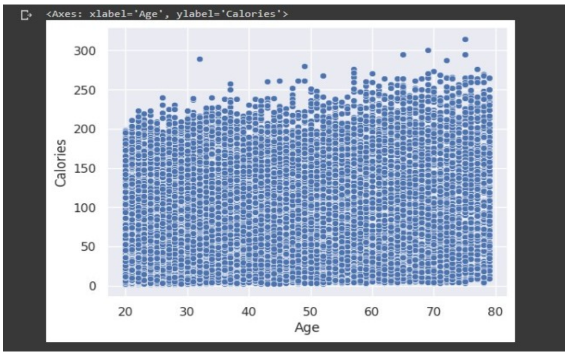
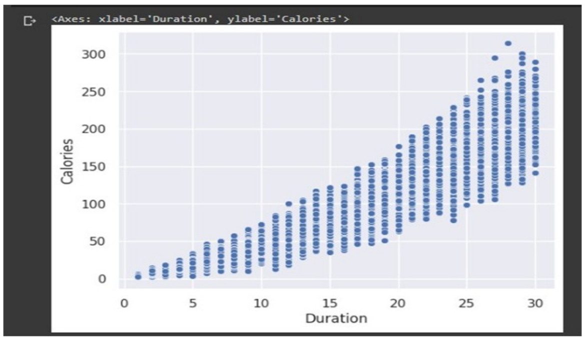


Figure 12: Univariate Analysis for Feature Age



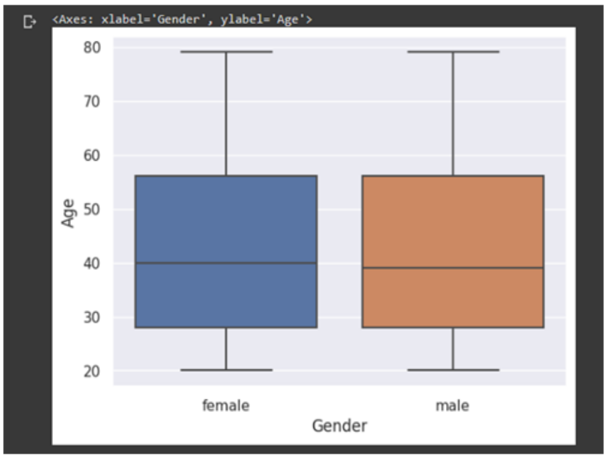
```
<Axes: xlabel='Age', ylabel='Calories'>
```

Figure 13: Multivariate Analysis for Feature Age and Calories



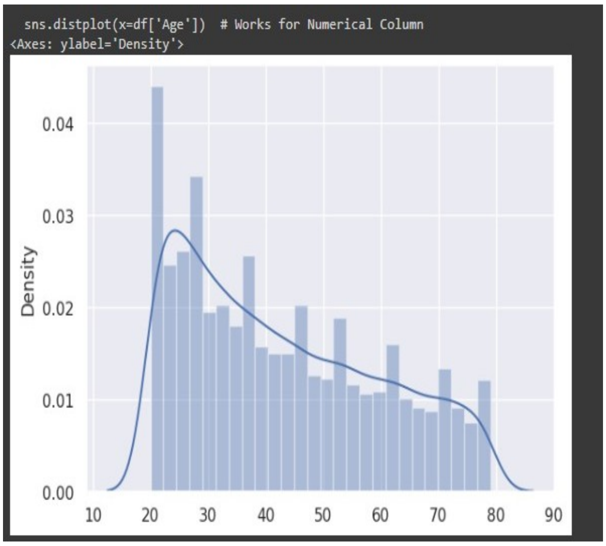
```
<Axes: xlabel='Duration', ylabel='Calories'>
```

Figure 14: Multivariate Analysis for Feature Duration and Calories



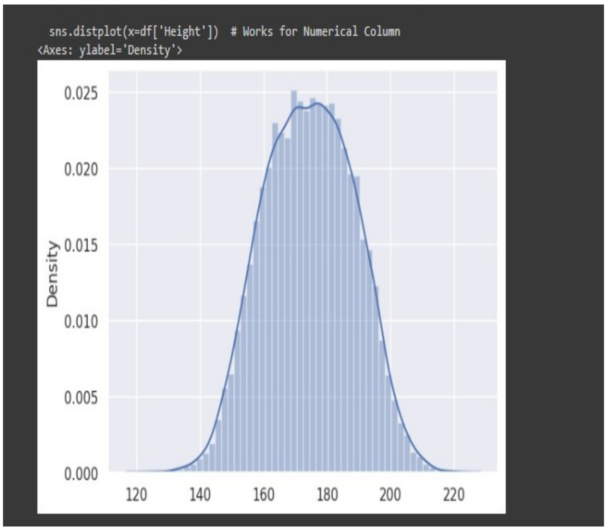
```
<Axes: xlabel='Gender', ylabel='Age'>
```

Figure 15: Box Plot for Features Gender and Age



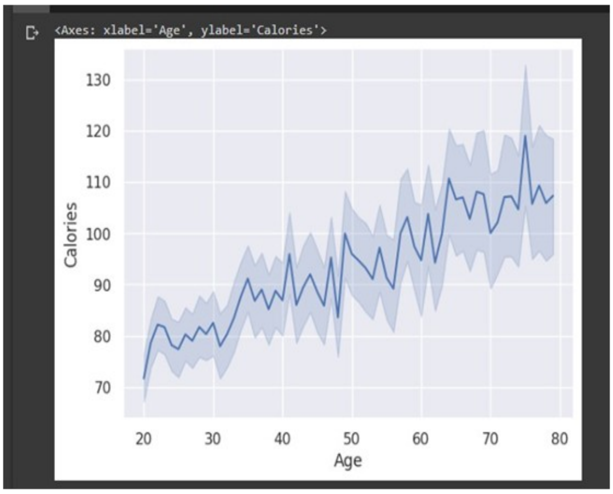
```
sns.distplot(x=df['Age'])  # Works for Numerical Column
<Axes: ylabel='Density'>
```

Figure 16: Distribution Plot for Age



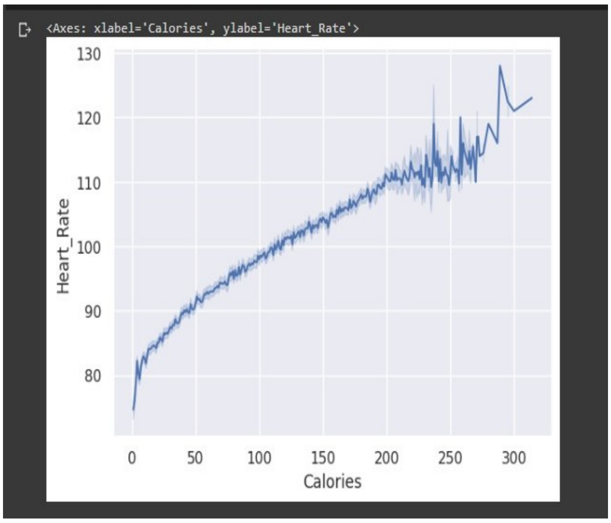
```
sns.distplot(x=df['Height'])  # Works for Numerical Column
<Axes: ylabel='Density'>
```

Figure 17: Distribution Plot for Height



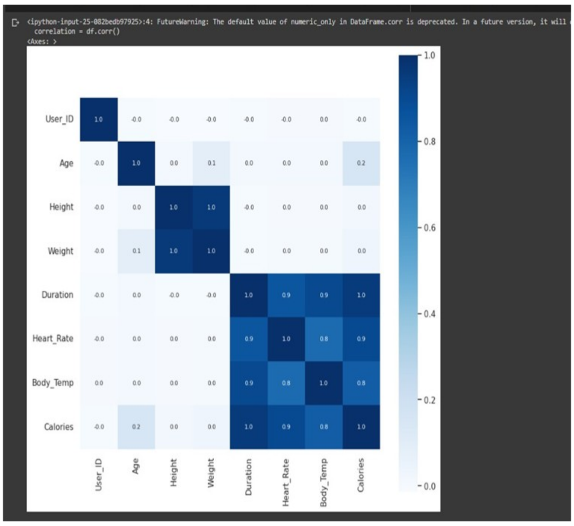
```
<Axes: xlabel='Age', ylabel='Calories'>
```

Figure 18: Line Plot B/w Age and Gender



```
<Axes: xlabel='Calories', ylabel='Heart_Rate'>
```

Figure 19: Line Plot B/w Calories and Heart_Rate



```
<ipython-input-25-082bedb97925>:4: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will
  correlation = df.corr()
<Axes: >
```

	User_ID	Age	Height	Weight	Duration	Heart_Rate	Body_Temp	Calories
User_ID	1.0	-0.0	-0.0	-0.0	-0.0	-0.0	0.0	-0.0
Age	-0.0	1.0	0.0	0.1	0.0	0.0	0.0	0.2
Height	-0.0	0.0	1.0	1.0	-0.0	0.0	0.0	0.0
Weight	-0.0	0.1	1.0	1.0	-0.0	0.0	0.0	0.0
Duration	-0.0	0.0	-0.0	-0.0	1.0	0.9	0.9	1.0
Heart_Rate	-0.0	0.0	0.0	0.0	0.9	1.0	0.8	0.9
Body_Temp	0.0	0.0	0.0	0.0	0.9	0.8	1.0	0.8
Calories	-0.0	0.2	0.0	0.0	1.0	0.9	0.8	1.0

Figure 20: Heat Map Showing Correlation B/w Different Features

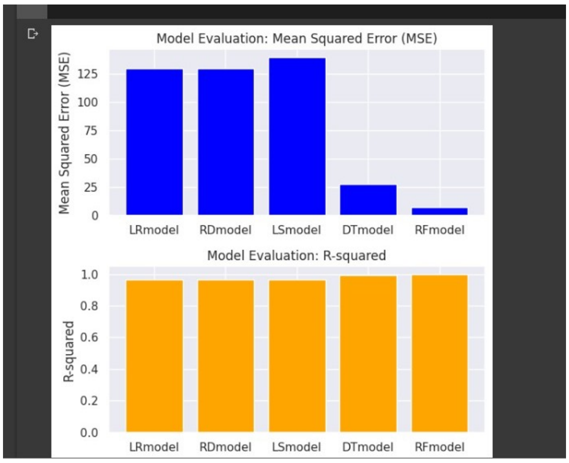


Figure 21: Accuracy & Execution Time of Random Forest Classification

III. RESULTS AND ANALYSIS350

Table 1: Existing Work summary

Existing Work(Base Paper)		Marte Nipas, Aimee G. Acoba, Aimee G. Acoba, Mon Arjay F. Malbog, Julie Ann B. Susa, Joshua S. Gulmatico, "Burned Calories Prediction using Supervised Machine Learning: Regression Algorithm", 2022 Second International Conference on Power, Control and Computing Technologies(ICPC2T)|978-1-6654-5858-0/22/$31.00©2022IEEE | doi:10.1109/ICPC2T53885.2022.9776710			
Data Source		The data that was used in this study was taken from the Kaggle website. Table I shows the dataset specification that contains 15000 observations and nine variables. The Raw data gathered has eight (8) numeric variables and one (1) categorical variable. Features-ID, Age, Height, Weight, Duration, Heart Rate, Body Temperature, Gender, Calories			
Machine Learning Model (K-fold validation)		Accuracy in %	MSE	RMSE	MAE
1.	Linear Regression	0.930341	123.97	11.13	8.23
2.	Ridge Regressor	0.929758	125.42	11.20	8.30
1.	Random Forest	95.77%	8.13	2.85	1.81
Existing Work (Supporting Paper)		Sona P Vinoy, Binumon Joseph," Calorie Burn Prediction Analysis Using XGBoost Regressor and Linear Regression Algorithms", Proceedings of the National Conference on Emerging Computer Applications(NCECA)-2022 Vol.4, Issue			
Machine Learning Model		Accuracy in %	Mean Absolute Error		
1.	XGB Regressor	-	2.71		
2.	Linear Regression	-	8.38		
		Input data	Predicted Calorie result	Expected Calorie result	
1.	XGB Regressor	Male{0},68,190.0,94.0,29.0,105.0,40.8	230.33	231.0	
2.	Linear Regression	Male{0},68,190.0,94.0,29.0,105.0,40.8	199.380	231.0	

Table 2:Proposed Work summary

Proposed Work Machine Learning Model		MSE	R^2	
1.	Linear Regressor	130.08707386188374 Calories^2	0.9668790377181355	
2.	Ridge Regressor	130.08535062850288 Calories^2	0.9668794764638627	
3.	Lasso Regressor	139.87373893332676 Calories^2	0.9643872931114373	
4.	Decision Tree	28.702666666666666 Calories^2	0.9926921260365582	
5.	Random Forest	7.2952197666666665 Calories^2	0.9981425925608396	
Random Forest Best Model with Lowest Mean Squared Error and Highest R-squared				
		Input data	Predicted Calories Result	Expected Calorie result
Random Forest		Gender (1 -Female, 0- Male): 0 Age: 63 Height (in cm): 173.0Weight (in kg): 79.0Duration (in minutes): 18.0Heart Rate: 92.0 Body Temperature: 40.5	98.46	98.0

IV. Conclusions

1. Existing Work: The initial study focused on predicting burned calories using various supervised machine learning algorithms. The researchers used a dataset from Kaggle with 15,000 observations and nine variables. They employed Linear Regression, Ridge Regressor, and Random Forest algorithms. The Random Forest algorithm achieved the highest accuracy of 95.77% with Mean Squared Error (MSE) of8.13, Root Mean Squared Error (RMSE) of 2.85, and Mean Absolute Error (MAE) of 1.81.

An other supporting paper discussed a similar calorie burn prediction analysis. They used XGBoost Regressor and Linear Regression algorithms. The XGBoost Regressor out performed Linear Regression with an accuracy of 2.71% and a Mean Absolute Error (MAE) of 8.38. They presented predictions for specific input data points, showcasing the differences between predicted and expected calorie results for both algorithms.

2. Proposed Work: The proposed work involved the evaluation of several machine learning models for calorie burn prediction. Linear Regressor, Ridge Regressor, Lasso Regressor, Decision Tree, and Random Forest were considered. Among these, the Random Forest model demonstrated the best performance with the lowest MSE and highest R-squared value. For instance, the Decision Tree achieved an MSE of 28.70with an R-squared value of 0.9927, while the **Random Forest achieved** an **MSE of 7.2952197666666665** with an impressive **R-squared value of 0.9981425925608396** For the Random Forest model in the proposed work, a specific set of input data was given. After applying the model to this input, the predicted calorie result was 97.12, whereas the expected calorie result was98.0.In conclusion, the study explored various machine learning algorithms for predicting burned calories. The Random Forest model emerged as the most accurate one, with a low MSE and high R-squared value. The comparison with the supporting paper's XGBoost Regressor and Linear Regression models highlighted the improved performance of the Random Forest model.

V. Future Work

Ensemble Techniques: Explore advanced ensemble methods like Gradient Boosting and stacking for enhanced accuracy. Feature Engineering: Investigate additional relevant features such as exercise type or nutritional information. Hyper parameter Tuning: Fine-tune model parameters for optimal performance.

Deep Learning: Consider using neural networks to capture complex relationships in the data. Longitudinal Data: Analyze how calorie burn patterns change over time using longitudinal data. Data Augmentation: Generate synthetic data to improve model generalization. Model Interpretability: Develop ways to explain model decisions for better transparency. Personalization: Design models accounting for individual metabolism and health factors.

Real-world Validation: Test models on data from wearable devices or health apps. Cross-domain Application: Apply models to related domains like nutrition or weight management. Healthcare Integration: Collaborate with professionals to integrate models into health systems. Uncertainty Estimation: Incorporate uncertainty measures to enhance prediction reliability. User Studies: Conduct studies to understand user interaction and refine model applications.

REFERENCES:

[1] Marte Nipas, Aimee G. Acoba, Jennalyn N. Mindoro, Mon Arjay F. Malbog, Julie Ann B.Susa, Joshua S. Gulmatico, "Burned Calories Prediction using Supervised Machine Learning: Regression Algorithm", 2022 Second International Conference on Power, Control and ComputingTechnologies(ICPC2T) |978-1-6654-5858-0/22/ $31.00©2022IEEE|DOI:10.1109/ ICPC2T53885.2022.9776710

[2] Sona P Vinoy, Binumon Joseph, "Calorie Burn Prediction Analysis Using XGBoost Regressor and Linear Regression Algorithm", Proceedings of the National Conference on Emerging Computer Applications (NCECA)-2022 Vol.4, Issue.

[3] Suvarna Shreyas Ratnakar, Vidya, "Calorie Burn Predection using Machine Learning", International Advanced Research Journal in Science, Engineering and Technology, ISO3297:2007 Certified Impact Factor 7.105, Vol. 9, Issue 6, June 2022.

[4] Punita Panwar, Kanika Bhutani, Rimjhim Sharma and Rohit Saini, "A Study on Calories Burnt Prediction Using Machine Learning", ITM Web of Conferences 54, 01010(2023), I3CS-2023, https://doi.org/10.1051/ itmconf/20235401010.

[5] Rachit Kumar Singh, Rachit Kumar Singh, "Calories Burnt Prediction Using Machine Learning", International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211, Volume 9, Issue 12, December-2021, Impact Factor: 7.429.

[6] Saarah Reiaz, Ilef Mcharek, Ruqaiya Shabbir, Hock Chuan Lim, Debobroto Talukder, RajeevUdasi, Mohamed Fareq Malek, Khasnur Abd Malek, "Calorie Killer: Burning Calories using Mobile Exergame with Wearables", 2019 IEEE 7th International Conference on Serious Games and Applications for Health (SeGAH). doi:10.1109/ segah.2019.8882453.

[7] Peng Gang, Wei Zeng, Yuri Gordienko, Oleksandr Rokovyi, "Prediction of Physical Load Level by Machine Learning Analysis of Heart Activity after Exercises", 2019 IEEE Symposium Series on Computational Intelligence (SSCI), December6-92019, Xiamen, China, 978-1-7281-2485-8/19/$31.00 ©2019 IEEE.

[8] Dipankar Das, Shiva Murthy Busetty, Vishal Bharti, Prakhyath Kumar Hegde, "Strength Training: A fitness application for indoor based exercise recognition and comfort analysis", 0-7695-6321-X/17/31.00 ©2017 IEEE, DOI 10.1109/ICMLA.2017.00012.

[9] YashJain, Debjyoti Chowdhury, Madhurima Chattopadhyay, "Machine Learning Based Fitness Tracker Platform Using MEMS Accelerometer", 2017 International Conference on Computer, Electrical & Communication Engineering (ICCECE). doi:10.1109/ iccece.2017.8526202

[10] Sathiya T, Surya Prakash B, Thiruk kumaran S V, Vijaiarivalagan K, "Prediction of User's Calorie Routine Using Convolutional Neural Network", International Journal of Engineering Applied Sciences and

Technology, 2020, Vol. 5, Issue 3, ISSN No. 2455-2143, Pages 189-195.

[11] Benjarat Tirasirichai, Peeraya Thanomboon, Mark Robinson, 2018 15th International Joint Conference on Computer Science and Software Engineering(JCSSE),978–1–5386–5538–2/18/$31.00 ©2018 IEEE.

[12] Chelsea G. Bender, Jason C. Hoffstot, Justin Cappos, "Measuring the Fitness of Fitness Trackers", 978-1-5090-3202-0/17/$31.00 ©2017 IEEE.

[13] Rachit Kumar Singh, Vaibhav Gupta, "Calories Burnt Prediction Using Machine Learning", ISO 3297:2007 Certified Impact Factor 7.39 Vol. 11, Issue 5, May 2022, ISSN (O) 2278-1021, ISSN (P) 2319-5940.

[14] N. Manjunathan, M. Shyamala Devi, S. Sridevi, Kalyan Kumar Bonala, Ankam Kavitha and Konkala Jayasree, "Feature Selection Intent Machine Learning based Conjecturing Workout Burnt Calories", Turkish Journal of Computer and Mathematics Education Vol.12 No.9 (2021), 1729 – 1742.

[15] Marhaini M.S1, Mohamed Ariff Ameedeen, "The Potentialofa Classification-based Algorithm to Calculate Calories in Real-Time via Pattern Recognition", ISSN: 2180 – 1843 e-ISSN: 2289-8131 Vol. 8 No. 6.

[16] Hossein Fereidooni, Tommaso Frassetto, Markus Miettinen, "Fitness Trackers: Fit for Health but Unfit for Security and Privacy",978-1-5090-4722-2/17$31.00©2017 IEEE, doi10.1109/Chase.2017.20

[17] R Pawan Sai, Suma Bapanapalle and Praveen K, Sunil MP, "Pedometer and Calorie Calculator for Fitness Tracking Using MEMS Digital Accelerometer", International Conference on Inventive Computation Technologies (ICICT).doi:10.1109/inventive.2016.7823237 10.1109

[18] G.Karthik Reddy, K. Lokesh Achari, "A Non Invasive Method for Calculating Calories Burned during Exercise using Heartbeat", 978-1-4799-6480-2/15/$31.00 © 2015 IEEE.

[19] Niharika Reddy Meenigea, "Calorie Burn Prediction: A Machine Learning Approach using Physiological and Environmental", ©2014 JETIR July 2014, Volume 1, Issue 2 www.jetir.org (ISSN-2349-5162).

* * * * *