PROJECT REPORT **DS5110 - Fall2024**

Calories Burned Prediction

Anjali Pathak, MSCS

Ren Huang, MSDS

1. Introduction

Calorie expenditure prediction is an essential aspect of health and fitness, as it provides valuable insights into how different exercise routines impact the body. This dataset, derived from exercise sessions at a gym, contains various features such as heart rate (Max_BPM, Avg_BPM, Resting_BPM), Weight (kg), Height (m), Session_Duration (hours), and Workout_Type, Fat_Percentage, Water_Intake (liters), BMI, Workout_Frequency (days/week), all of which influence the number of calories a person burns. Understanding these relationships is crucial for individuals looking to optimize their workouts or for fitness professionals designing effective exercise plans. With increasing interest in personalized fitness, machine learning offers an efficient way to predict calorie expenditure based on these input features, allowing for more accurate and data-driven decisions. This prediction can be applied in various real-world scenarios, such as fitness tracking apps, personalized workout recommendations, or health monitoring systems.

The primary task of this project is to predict the "Calories_Burned" variable (the target), based on other factors such as heart rate, session duration, BMI, workout frequency, height, weight, and personal attributes like age and gender.

The "Calories_Burned" variable serves as the target of prediction, representing the total number of calories an individual expends during an exercise session. It is a continuous numerical variable, making this a regression problem. The task involves building a model capable of accurately predicting calorie expenditure using the available features.

Several challenges arose when analyzing this dataset. One significant challenge was the dataset's scarcity; with only 973 rows, it may not fully capture the diversity of exercise behaviors or demographic variations. To address this issue, we applied Generative Adversarial Networks (GANs) to generate synthetic data. GANs allow us to create new, realistic data points that expand the existing dataset, helping to alleviate overfitting and improve the model's ability to generalize. By using GANs, we were able to enrich the dataset, thus enhancing the accuracy and robustness of our calorie expenditure prediction models.

2. Dataset and Experimental Setup

DS5110-Calories Burned Prediction

The dataset used in this study contains 973 rows and 15 features, including both continuous and categorical variables:

Continuous Variables:

Age, represents the age of the individual; Weight (kg), represents the individual's weight; Height (m), represents the individual's height; Max_BPM, represents the maximum heart rate during exercise; Avg_BPM, represents the average heart rate during exercise; Resting_BPM, represents the individual's resting heart rate; Session_Duration (hours), represents the duration of the exercise session; Calories_Burned, represents the calories burned during the exercise session (target variable), Fat_Percentage, represents the percentage of body fat; Water_Intake (liters), represents the water intake during exercise; Workout_Frequency (days/week), represents the number of exercise days per week; BMI, represents the body mass index.

Categorical Variables:

Gender, indicates the gender of the individual; Workout_Type, indicates the type of exercise (e.g., Yoga, HIIT, Cardio, Strength); Experience_Level, indicates the experience level (e.g., 1 for beginner, 2 for intermediate, 3 for advanced).

Experimental Setup:

Given that this dataset contains only 973 rows and 15 features, we employed feature engineering techniques to enhance the number of available features significantly. In the engineering process, we transformed the continuous variable into a categorical one to enhance interpretability. Specifically, we categorized the Age variable into discrete groups such as young (ages 0 to under 30), middle (ages 30 to under 60), and old (ages 60 to under 100) using the “pd.cut()” function. This bin allowed the model to capture age-related trends better. Similarly, we transformed the 'BMI' (Body Mass Index) values into obesity classifications by defining levels of obesity through specific BMI ranges, categorizing individuals into six distinct groups: Underweight, Normal, Overweight, Obesity I, Obesity II, and Obesity III. This approach not only simplified the analysis of how burned calories relate to age but also enhanced the model's capacity to learn from distinct population segments. By categorizing BMI into specific obesity levels, the model can identify patterns and trends relevant to each group, allowing for more nuanced insights.

Next, we utilized Generative Adversarial Networks (GANs) to augment the dataset by generating synthetic examples. GANs consist of two neural networks: a generator that creates fake data using random noise as input and a discriminator that evaluates the authenticity of the data. By training these networks simultaneously in a competitive setting, it allowed us to produce realistic synthetic samples that resemble the original dataset distribution. This approach helped us overcome the limitations of having a small dataset by providing more training examples, ultimately improving the robustness and performance of our predictive model. Due to the limitation of the input format of GANs, we implemented one-hot encoding to convert multi-categorical features into binary classifications. We normalized the data to ensure that all features were on a similar scale, which was crucial for improving the convergence speed and performance of the GAN. In addition, after generating 1000 synthetic examples, we did a thorough data exploration step to assess the differences between raw data and synthetic data. This comparison was crucial for understanding how well the synthetic examples mirror the original data distribution and whether they could be effectively utilized in further analyses. To optimize the performance of our GANs, we fine-tuned various hyperparameters within the network like input dimensions, numbers of epochs and the learning rate to make the generated data similar to the source data. Furthermore, to prevent overfitting, we incorporated dropout layers in both generator and discriminator networks to ensure that our models maintained generality and did not rely too heavily on specific features from the training data.

After concatenating the dataset, we utilized the “train-test split()” method to divide the dataset into training and testing subsets. Specifically, we allocated 80% of the data for training and 20% for testing, ensuring that the distribution of the target variable was maintained across both subsets. Then, we tried different methods including linear regression as a baseline, and more advanced models such as Random Forests, Neural Networks(multi-layer perceptron), XGBoost Regressor and AutoML pipeline. For model evaluation, we used Mean Squared Error (MSE) as our primary metric, as it quantified the average squared difference between predicted and actual values. MSE is particularly useful in regression tasks because it emphasizes larger errors due to the squaring of differences. By focusing on this metric, we assessed model performance effectively, guiding us in selecting the best-performing model from our trials. The details of how we predicted burned calories using various models will be described in the “6. Analysis: Methods Comparison” Section.

Finally, we used cross-validation as a powerful technique for assessing the performance of different models and ensuring their robustness by partitioning the training set into five folders. This approach decreased variance in performance estimation by averaging the results across multiple iterations, leading to more stable and reliable evaluations that mitigated the influence of any single random train-test split. By applying the same cross-validation scheme to all models, it created a standardized approach for comparison. We calculated the final MSE for each model by using ”np.mean()” to obtain the average MSE and ”np.std()” to determine the standard deviation across the different folds in our cross-validation process. By displaying these metrics, we gained valuable insights into the performance and variability of each model. We compared these results with the linear regression model to determine which model had the best performance.

3. Related Work: Literature Review

Ustab Ray and others **[1]** proposed using multiple machine learning models, including XGBoost, Random Forest, and Support Vector Machines, to predict calories burned based on data from wearable devices. They used a combination of physiological features, such as heart rate, weight, and exercise duration. Their approach demonstrated that XGBoost outperformed other models in terms of RMSE and R^2 values. This study highlights the importance of hyperparameter optimization in improving prediction accuracy for exercise-related data

Punita Panwar and others **[2]** in their study, analyzed wearable device data to predict calorie expenditure during various physical activities. They utilized feature engineering and regression models, achieving competitive performance in estimating calorie burn rates. Their approach is particularly notable for its emphasis on activity-specific model tuning.

Amol Kadam and others **[3]** explored neural networks to model calorie burn rates using data from wearable devices, as documented in their paper. Their approach included integrating time-series data with advanced optimization techniques, which significantly enhanced prediction performance.

Niharikareddy Meenigea **[4]** detailed a novel approach for energy expenditure prediction using personalized activity monitoring in her paper published in JETIR. The study emphasized leveraging wearable sensor data for personalized prediction models with a focus on lifestyle-specific variables.

Marte Nipas and others **[5]** investigated hybrid machine learning approaches combining neural networks and traditional regression models for activity-specific calorie estimation, presented in the IEEE Transactions on Biomedical Engineering. Their work demonstrated improved generalizability by leveraging domain-specific features such as oxygen consumption and step frequency.

Why We Believe Our Method is Better:

Model Diversity and Optimization

We integrated a wider variety of models, including XGBoost, Neural Networks, and AutoML, compared to other studies that focus mainly on a limited set of machine learning models like Random Forest and SVM [1][2]. Our use of AutoML automates hyperparameter tuning, increasing efficiency and potentially improving model performance by finding the optimal configurations without manual intervention.

Handling Data Challenges

While other studies focus on activity-specific features, we applied advanced techniques like **Generative Adversarial Networks (GANs)** to generate synthetic data for handling data imbalance, addressing challenges such as multicollinearity and dataset scarcity. These methods enhance model robustness and generalizability, which is particularly important in small or imbalanced datasets [2][3].

Model Evaluation and Performance

We used cross-validation and evaluation metrics such as MSE and R^2 that ensure a more reliable assessment of model performance. The results, including an R^2 of 0.9678 from AutoML, show superior accuracy compared to previous studies like Ustab Ray's [1]. Our XGBoost model also outperforms others in terms of MSE and predictive power, demonstrating the effectiveness of our model selection.

Efficiency and Generalizability

By incorporating AutoML, our project reduces the time and effort needed for model tuning, while still achieving high accuracy. This contrasts with traditional machine learning pipelines, which often rely on manual adjustments [5]. Our approach not only offers high predictive accuracy but also provides greater efficiency and generalizability.

In summary, our use of diverse models, feature engineering, AutoML, and robust evaluation makes our approach more efficient and accurate than previous methods in predicting calorie expenditure [1][2][5].

4. Data Analysis

To uncover meaningful patterns and relationships within the dataset, we performed a comprehensive analysis, including univariate, bivariate, and multivariate visualizations.

Univariate Analysis

The distributions of numerical features such as Age, Weight (kg), Height (m), Max BPM, Avg BPM, Resting BPM, Session Duration (hours), Water Intake (liters), Fat Percentage, BMI, and Calories Burned were examined using histograms and kernel density estimate (KDE) plots. These visualizations revealed varied feature distributions, with some features exhibiting normality while others displayed skewed patterns. For categorical features like Gender, Workout Type, Workout Frequency (days/week), and Experience Level, count plots highlighted imbalances and trends within the dataset. For instance, males outnumber females, indicating a slight gender imbalance. Strength training was the most prevalent workout type, while High-Intensity Interval Training (HIIT) had the fewest entries, reflecting participant preferences. Most participants reported working out three days per week, and the majority were at intermediate skill levels (Experience Level 2), with very few at advanced levels (Experience Level 3).

We derived two additional categorical features for deeper insights. First, Age Group was categorized into young (0–30), middle (30–60), and old (60–100). The middle age group was the most represented, with 697 instances, while the old group had no representation, indicating a potential limitation in the dataset. Second, Obesity Level was derived from BMI using standard thresholds, categorizing participants as Underweight (BMI < 18.5), Normal (18.5–25), Overweight (25–30), and Obesity I–III (BMI > 30). Most individuals were classified as Normal, followed by Overweight and Underweight, with few participants in extreme BMI ranges like Obesity III.

Bi-Variate Analysis
We examined pairwise relationships among features using a combination of barplots, scatter plots, and categorical plots. For example, the average calories burned by each age group was calculated as:

$$Average\ Calories\ Burned = \sum Calories\ Burned\ for\ Age\ Group / \ Number\ of\ Participants\ in\ Age\ Group$$

The young group burned the highest calories on average (950.5), followed by the middle group (887.6), while the old group had no data. Scatter plots confirmed this trend, with the absence of the old group being visually apparent. When grouped by workout types, HIIT participants burned the most calories, though differences across workout types were generally small.

Further analysis explored how age groups and workout preferences interact. Cardio was the most popular workout for the young group, while the middle group preferred strength training. Gender-based preferences showed males favoring strength training and females opting for cardio. Average water intake varied by age group, calculated as:

$$Average\ Water\ Intake = \sum Water\ Intake\ for\ Age\ Group\ /\ Number\ of\ Participants\ in\ Age\ Group$$

The middle age group consumed more water on average than the young group, suggesting higher hydration needs among middle-aged individuals.

Obesity levels also influenced calorie expenditure. Participants in the Obesity I category burned the most calories on average, followed by the Normal group. Calorie burn decreased with higher obesity levels, with Obesity III participants burning the least. Scatter plots revealed a general trend of decreasing calorie burn with increasing BMI, although Overweight individuals burned more calories than those classified as Underweight.

Multivariate Analysis
To analyze complex feature interactions, we used advanced visualizations and statistical methods. A heatmap of the correlation matrix identified linear relationships among numerical features. For example, Weight (kg) and BMI were strongly positively correlated (r=0.85r=0.85), while Session Duration and Experience Level exhibited a moderate positive correlation (r=0.76r=0.76).

The impact of workout frequency on calorie expenditure across session durations was visualized using a scatter plot, where marker size and color represented workout frequency. Participants with higher workout frequency (five days per week) exhibited longer session durations and higher calorie burn, whereas lower frequencies (one day per week) showed significant variation in calorie burn, likely due to differing workout intensities.

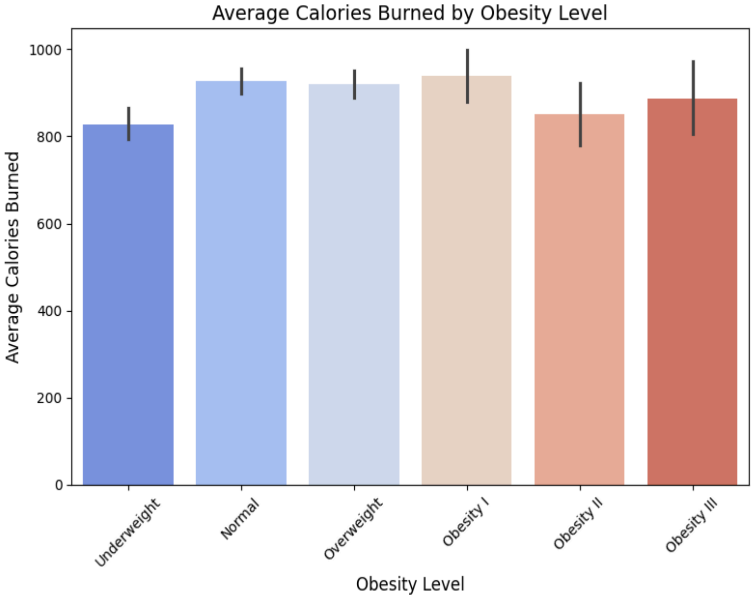
Average resting BPM (Resting BPMavg) was analyzed by gender and workout type using the formula:

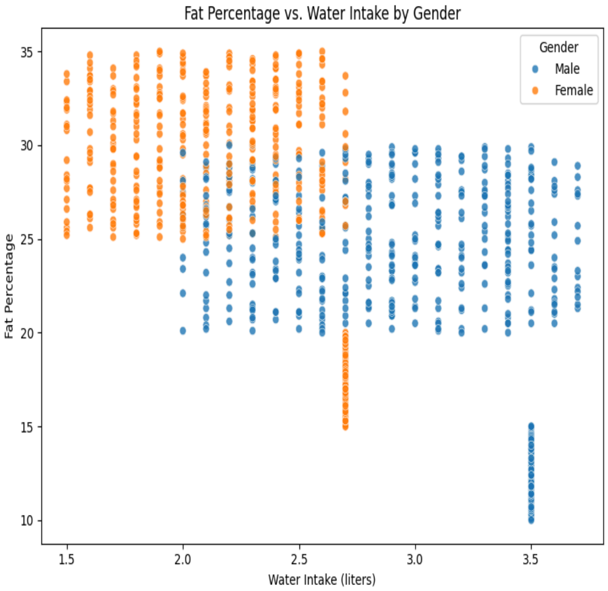
$$Resting\ BPM\ average = \sum Resting\ BPM\ for\ Group\ /\ Number\ of\ Participants\ in\ Group$$

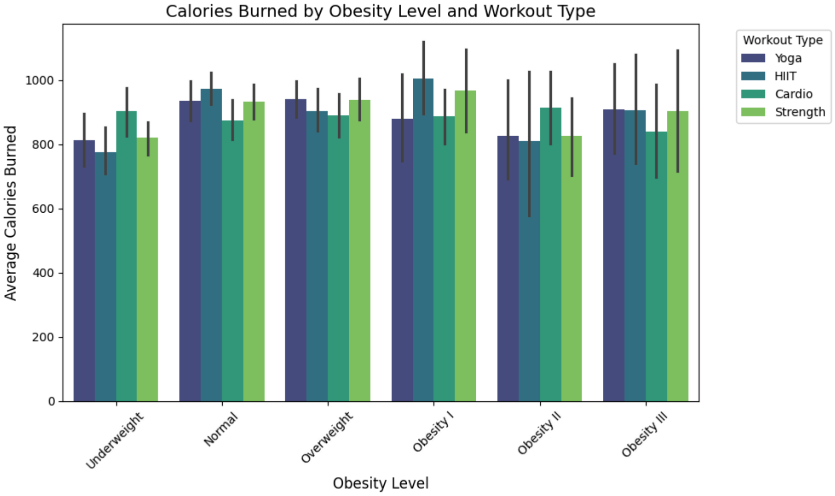
Interestingly, both males and females had identical resting BPM for yoga, reflecting its universal calming effect. For other workout types, gender differences in resting BPM were negligible.

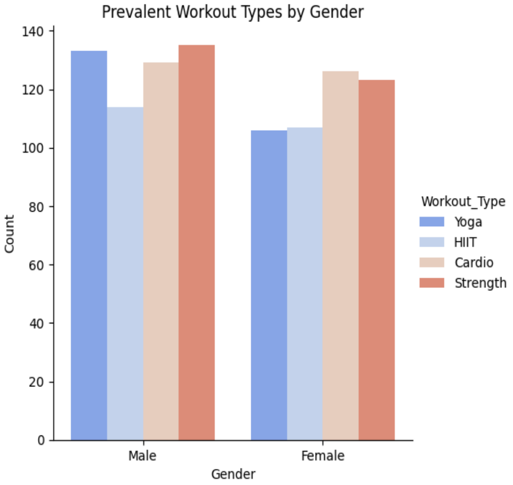
We also examined water intake and fat percentage relationships across genders. Males consumed more water on average than females, with higher water intake correlating inversely with fat percentage. Pairplots highlighted interactions between age, calorie expenditure, and workout type, showing younger participants engaging more in cardio-intensive workouts and burning more calories overall.

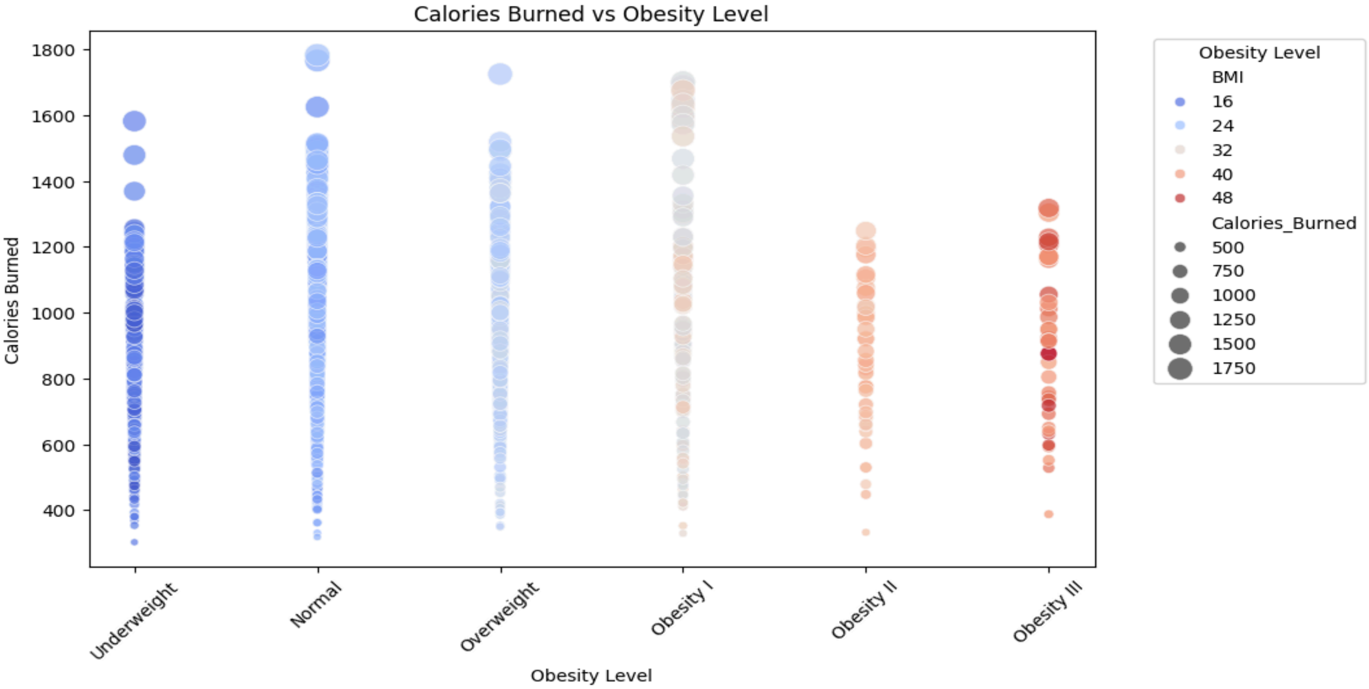
Finally, the influence of workout type on calorie burn across obesity levels was evaluated. Cardio was most effective for Underweight and Obesity II groups, while HIIT yielded the highest calorie burn for Normal and Obesity I groups. Strength training was particularly effective for Overweight individuals, emphasizing its suitability for moderate BMI ranges.











5. Analysis: Methods Comparison

Baseline Model - Linear Regression

For our initial modeling approach, we utilized Linear Regression to establish a baseline for predicting calories burned based on various features in the dataset. The linear regression model operates under the premise of a linear relationship between the independent variables (features) and the target variable (calories burned). The fundamental equation for linear regression is represented as:

$$[y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \varepsilon]$$

where (y) is the predicted output (calories burned), (β_0) is the intercept, (β_1, β_2, ..., β_n) are the coefficients of the features (x_1, x_2, ..., x_n), (n) is the number of features in the dataset (exclude the ‘Calories_Burned’ column and (ε) denotes the error term. By fitting the model to the training data, we aim to get optimal coefficients by minimizing the MSE between the predicted y and actual y. The results of our linear regression analysis yielded MSE of 3042.5125 and an R-squared value of 0.9477. The high R-squared value indicates that approximately 94.77% of the variance in calories burned can be explained by the model, demonstrating a strong linear correlation between the features and the target variable. This suggests that linear regression effectively captures the underlying trends in the data, making it a competent starting point for our analysis.

Random Forest

Despite the relatively favorable metrics, linear regression may not fully capture non-linear relationships in the dataset, which needed further exploration of more complex algorithms. Therefore, we used Random Forests as the second model. Random Forests is an ensemble learning method used for regression and classification tasks. It operates by constructing multiple decision trees during training and outputting the highest frequency of the classes (classification) or mean prediction (regression) of the individual trees to improve accuracy and control overfitting. This technique employs bagging, which generates several subsets of the training data through random sampling, and feature randomness, where only a random subset of features is considered for splitting at each node of the trees. In our analysis, we implemented the Random Forest model to predict calories burned, achieving an MSE of 3257.0544 and an R-squared value of 0.9440. It had a higher MSE and lower R-squared relative to the linear regression. It demonstrated that the Random Forest model captured more complex relationships within the data but potentially introduced some noise or variance that affected its predictions. Another possibility was that the data exhibited more linear relationships rather than non-linear patterns.

Neural Networks (Multi-Layer Perceptron)

To further determine whether the data exhibits linear or non-linear relationships, we conducted our third model MLP. It is a type of artificial neural network that consists of multiple layers of nodes(we used three layers of neural networks here), structured in a way that allows for the modeling of complex, non-linear relationships in data. In our code, the MLP was constructed using a sequential model from the PyTorch library, featuring an input layer that accepted a specified number of input dimensions (numbers of features), followed by a hidden layer with 128 neurons and a Rectified Linear Unit (ReLU) activation function to introduce non-linearity. This was followed by another hidden layer with 256 neurons, again utilizing the ReLU activation function. Finally, the output layer consisted of a single neuron that produces the final burned calories prediction. To save running time, we utilized “nn.init.xavier_uniform_()” to generate the initial parameters for each layer. This initialization method helped ensure that the weights were set to appropriate values, which promoted faster convergence during training by mitigating issues such as vanishing or exploding gradients. Additionally, we employed “DataLoader()” to provide a convenient way to load and manage batches of data during the training and evaluation processes. It allowed us to load data in smaller, manageable batches efficiently. By combining these two techniques, we optimized the training workflow, improving both the speed and performance of our model.

Extreme Gradient Boosting (XGBoost)

For our analysis, we selected XGBoost Regressor as our primary method due to its superior performance in handling large datasets with diverse features. XGBoost utilizes a boosting technique that builds trees sequentially, where each new tree aims to correct the errors made by the previous one, using gradient descent optimization(second-order derivative - Hessian) to minimize a loss function. This technique enhances model accuracy and prevents overfitting through regularization, which penalizes complex models. Additionally, it effectively handles missing values, assesses feature importance, and leverages parallel processing for faster computation, making it highly effective for large datasets with complex features. To fine-tune XGBoost, we experimented with key hyperparameters such as the learning rate, maximum depth of trees, and the number of estimators to optimize model performance. This careful tuning process ultimately got improved results. Our implementation achieved an MSE of 2270.4028 and an R-squared value of 0.9610, which surpassed the performance of all the previously tested models.

Automated Machine Learning (AutoML)

We also implemented an AutoML pipeline to predict the calories burned, which streamlined the training process by automating critical tasks such as model selection, hyperparameter tuning, and feature engineering. One of the key components of this approach is stacking, where multiple models are combined to enhance prediction accuracy. In our analysis, the implementation of an AutoML framework yielded impressive results, with an MSE of 1875.00 and an R-squared value of 0.9678. These metrics demonstrate the AutoML pipeline's effectiveness, indicating that the model is capable of making accurate predictions. However, the training process for the AutoML pipeline required a significant amount of time to achieve these high-quality predictive outputs. Although automating tasks like model selection and hyperparameter tuning provides considerable benefits, it necessitates significant computational resources and may lead to longer training times.

6. Related implementations

In the Kaggle kernel titled “Prediction with Neural Network” **[6]**, Taseer Mehoob implemented a seven-layer neural network model to predict calories burned, achieving a test loss of 4895.8779 and an R-squared value of 0.9244. Compared to our MLP, he constructed a more intricate neural network with multiple hidden layers and a larger number of neurons in each layer. However, this increased complexity resulted in overfitting, which negatively impacted the model's predictive accuracy and running speed, making it less effective than our simple MLP model, let alone the even more accurate XGBoost model.

Redimo **[7]** utilized Random Forests to predict with a higher R-squared 0.9546 compared to our model. However, we observed that he dropped several important columns like age, height and all the categorical columns including Workout_Type. The omission of these key features limited the model's ability to capture significant relationships within the data, even though it improved the model's metrics temporarily. Compared to our data preprocessing approach, we retained all available information and incorporated additional columns through feature engineering, allowing us to leverage a richer set of features.

7. Results

Having previously discussed aspects of the results in the “Analysis: Methods Comparison” section, this section will focus on the more accurate outcomes generated through cross-validation, here is the result table:

Machine Learning Model	**mean(cross_val_score)**	**std(cross_val_score)**
Linear Regression	3162.22	347.66
Random Forests	3122.57	338.97
Neural Network	3988.77	180.27
XGBoost Regressor	**2264.73**	**289.93**

From the table, the XGBoost Regressor model achieved the lowest MSE of 2264.73, indicating it provides the most accurate predictions for calories burned among the models tested. This is a significant improvement over the Linear Regression and Random Forests models, which had MSEs of 3162.22 and 3122.57, respectively. The Neural Network model, however, performed the worst with an MSE of 3988.77, suggesting that it may not have been optimally adjusted for this dataset.

Next, both Linear Regression and Random Forests exhibited higher standard deviations of 347.66 and 338.97, respectively, indicating greater variability in their performance across different validation folds. This suggests that while these models can achieve reasonable accuracy, their predictions may be less stable, making them potentially more sensitive to variations in the dataset. In contrast, the MLP model, with a lower standard deviation of 180.27, indicates that its predictions may have been more consistent, even though it is still less accurate on average MSE compared to XGBoost.

Despite the strong performance of the XGBoost model, there were still opportunities for improvement that we could have explored. For instance, we could have conducted a more comprehensive hyperparameter tuning process for the Neural Network, including adjustments to the model's architecture, such as the number of layers and neurons, trying different activation functions. The variability in results from the Linear Regression and Random Forest models could also suggest that further feature engineering or data preprocessing techniques might help to stabilize their performance and enhance their predictive capabilities.

8. Conclusion

In brief, our analysis revealed that the XGBoost Regressor significantly outperforms other models, including Linear Regression, Random Forests, and Multi-layer Perceptron, in predicting calories burned, achieving the lowest mean squared error and demonstrating strong predictive accuracy. Based on these findings, we recommend utilizing the XGBoost Regressor for similar predictive modeling tasks, as it effectively captures complex relationships within the data.

If we had more access to data, we could further fine-tune the model’s performance through additional feature engineering and hyperparameter optimization, potentially leading to even greater predictive capabilities. A larger dataset would also allow us to investigate a wider range of interactions between features and discover new relationships in calorie expenditure. This expanded exploration would provide deeper insights into the factors influencing calorie burn, ultimately enabling us to develop more accurate models.

For companies aiming to implement predictive analytics at scale, adopting the XGBoost Regressor would be a highly advantageous choice due to its efficiency in handling large datasets and its exceptional performance in delivering accurate predictions. Factors such as its sparse matrix support, parallel processing capabilities, and regularization techniques contribute to XGBoost's ability to outperform other methods in both speed and efficiency. These features make it particularly well-suited for large-scale machine learning tasks, ensuring robust performance while reducing computational costs and training time. As such, XGBoost stands out as a preferred solution for organizations looking to perform advanced analytics.

9. References

[1] Utsab Ray, Arka Rajak, Souvick Kumar Guha, Aniket Roy, Karabi Ganguly, “Prediction Of Hone Calorie Expenditure Using Advanced ML”, International Journal of Creative Research and Thoughts (IJCRT), Volume 11, Issue 6 June 2023, Kalyani, West Bengal, Nadia, India, 2023, ISSN: 2320-2882 - IJCRT
[2] Punita Panwar, Kanika Bhutani, Rimjhim sharma, Rohit Saini, “A Study on Calories Burnt Prediction Using Machine Learning”, ITM Web of Conferences 54, (I3CS-2023), Jaipur Engineering College & Research Centre, Computer Science Department, Jaipur, Rajasthan, India, 2023, 01010 (2023) - ITM
[3] Amol Kadam, Anurag Shrivastava, Sonali K. Pawar, Vinod H Patil, Jacob Michaelson, Ashish Singh, “Calories Burned Prediction Using Machine Learning”, 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), IEEE, Gautam Buddha Nagar, India, 2023, DOI: 10.1109/IC3I59117.2023.10397623 - IC3I
[4] Niharikareddy Meenigea, “Calorie Burn Prediction: A Machine Learning Approach using Physiological and Environmental Factors”, Journal of Emerging Technologies and Innovative Research (JETIR) July 2014, Volume 1, Issue 2, Virginia International University USA, 2014, ISSN-2349-5162 - JETIR
[5] 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 Computing Technologies (ICPC2T), IEEE, Raipur, India, 2022, DOI: 10.1109/ICPC2T53885.2022.9776710 - ICPC2T
[6] Taseer Mehoob, “Prediction With Neural Network”, Kaggle - Kaggle Kernel 1
[7] Redimo, “Burned Calorie Prediction”, Kaggle - Kaggle Kernel 2

10. Appendix

- DS5110-Calories Burned Prediction
- Kaggle - Gym Members Exercise Dataset
- GitHub Repository

11. Statement of Contributions

This project was completed collaboratively by Anjali Pathak and Ren Huang, with both members contributing equally to all aspects of the work.

Anjali Pathak: I led the initial data wrangling process, which involved gaining insights from the dataset and performing crucial preprocessing tasks such as handling missing values, duplicates, feature engineering, and feature transformation I also conducted extensive Exploratory Data Analysis (EDA), generating univariate, bivariate, and multivariate visualizations to identify key patterns and trends within the data.

Ren Huang: I focused on augmenting the dataset through synthetic data generation using Generative Adversarial Networks (GANs). I implemented and fine-tuned several machine learning models, including Random Forest, Neural Network, XGBoost, and AutoML. Additionally, I performed hyperparameter tuning and also contributed to the comparative analysis of model performance, identifying their strengths and limitations.

Both members actively participated in discussing the modeling approaches, interpreting results, and preparing the final report. Our equal contribution ensured a balanced and thorough analysis of the dataset.