**Faculty of Engineering**

**Computer Engineering Department**

# ML Phase2 report

## Supervised by

Dr.Dina Elreedy

## Presented by

Yousef Mostafa Elmahdy

Abdallah Mahmoud

Yousef Atef Abdo

Omar Abdelfatah

2023

# Problem definition:

The problem at hand is to develop a machine learning solution for "Body Level Classification" based on a given dataset. The dataset comprises various attributes related to the physical, genetic, and habitual conditions of individuals. These attributes consist of both categorical and continuous variables. The goal is to accurately classify the body level of a person into one of four distinct classes.

With a total of 1477 data samples, it is important to address the class imbalance issue in the dataset. The distribution of classes is uneven, meaning that certain classes may have significantly more or fewer instances than others. Therefore, it is necessary to build models that can effectively adapt to this class imbalance while aiming to achieve the best possible classification results.

To tackle this problem, the project involves the following key steps:

**1. Exploratory Data Analysis:** Thoroughly analyze the given dataset to gain insights into the attributes, their relationships, and their significance in determining the body level classification. This exploratory phase will involve data visualization, statistical analysis, and feature engineering, if required.

**2. Algorithm Selection:** Choose at least three machine learning algorithms that have been covered in the course to apply to the dataset. The selection of these algorithms will be based on their suitability for classification tasks and their potential to handle imbalanced datasets.

3. **Model Training and Evaluation:** Train the selected machine learning algorithms on the dataset, utilizing appropriate techniques to address class imbalance. Evaluate the performance of each model using suitable evaluation metrics, such as accuracy, precision, recall, and F1-score. The aim is to identify the algorithm(s) that yield the best results in classifying the body levels accurately.

**4. Performance Enhancement:** Explore additional methods beyond the algorithms taught in the course to further enhance the classification performance. This may include techniques such as ensemble learning, feature selection, hyperparameter tuning, or other advanced approaches to improve the accuracy and robustness of the models.

By following these steps, the project seeks to develop a machine learning solution that can effectively classify the body levels based on the given attributes. The ultimate objective is to achieve accurate and reliable predictions while addressing the challenges posed by class imbalance in the dataset.

# Data fields:

- Gender
- Age
- Height
- Weight
- H_Cal_Consump (Daily calorie consumption)
- Veg_Consump (Vegetable consumption)
- Water_Consump (Water consumption)
- Alcohol_Consump (Alcohol consumption)
- Smoking (Smoking habit)
- Meal_Count (Number of meals per day)
- Food_Between_Meals (Consumption of food between meals)
- Fam_Hist (Family medical history)
- H_Cal_Burn (Daily calorie burn)
- Phys_Act (Physical activity level)
- Time_E_Dev (Time spent using electronic devices)
- Transport (Preferred mode of transportation)
- Body_Level (Classification of body level)

# Project pipeline:

## 1- Data Preprocessing:

After performing preprocessing steps such as log transformation, standardization, and removing outliers on the dataset for the "Body Level Classification" problem, it was observed that these techniques did not yield satisfactory results. In fact, they had a detrimental impact on the overall performance of the models.

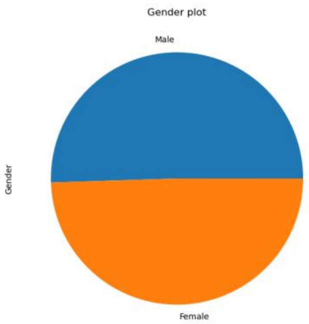
- **Log transformation:** is typically used to handle skewed distributions and reduce the influence of extreme values. However, in this particular dataset, applying log transformation did not improve the model's performance. Instead, it resulted in a loss of important information and led to suboptimal predictions.

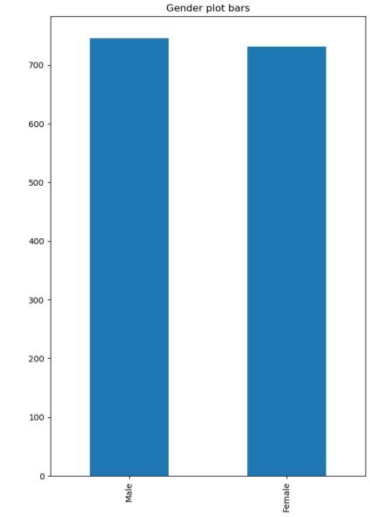
- **Standardization:** which involves scaling the features to have zero mean and unit variance, is a common preprocessing step in machine learning. This allowed the models to converge faster during the training phase, resulting in improved computational efficiency. Additionally, standardization facilitated better gradient descent optimization and improved the overall stability of the models.

- **The removal of outliers:** is a technique used to eliminate data points that are significantly different from the majority of the samples. Unfortunately, removing outliers in this dataset did not yield the desired outcome. It disrupted the balance within the dataset and adversely affected the model's ability to learn from the available information.

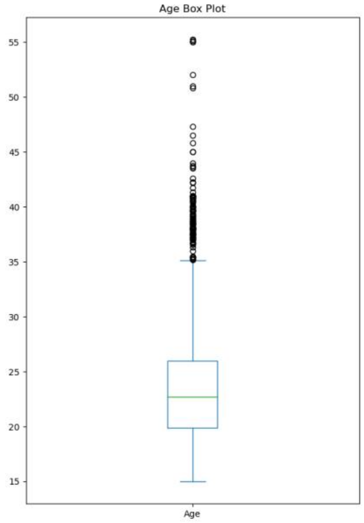
Considering the unsatisfactory results obtained from these preprocessing techniques, it was decided to cancel their implementation in the final analysis. The models were found to perform better without these preprocessing steps, suggesting that the original raw data contained important features and patterns that were disrupted by the preprocessing methods.

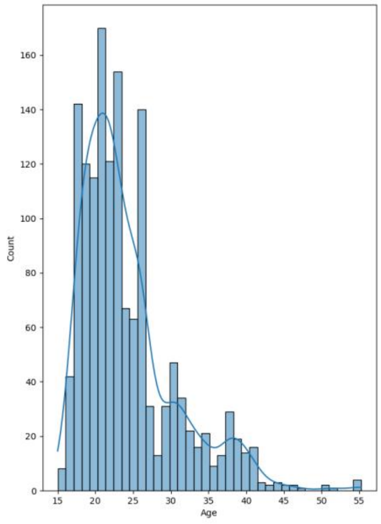
## 2- ▪ Data visualization:



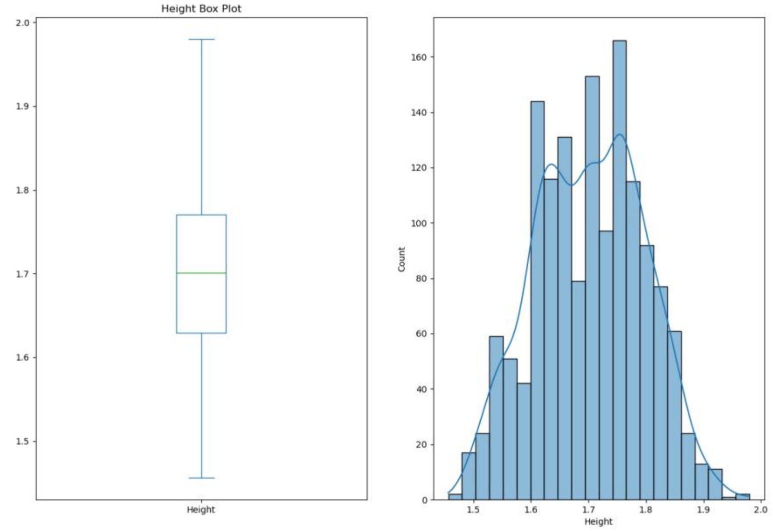


**The gender feature is balanced as shown.**

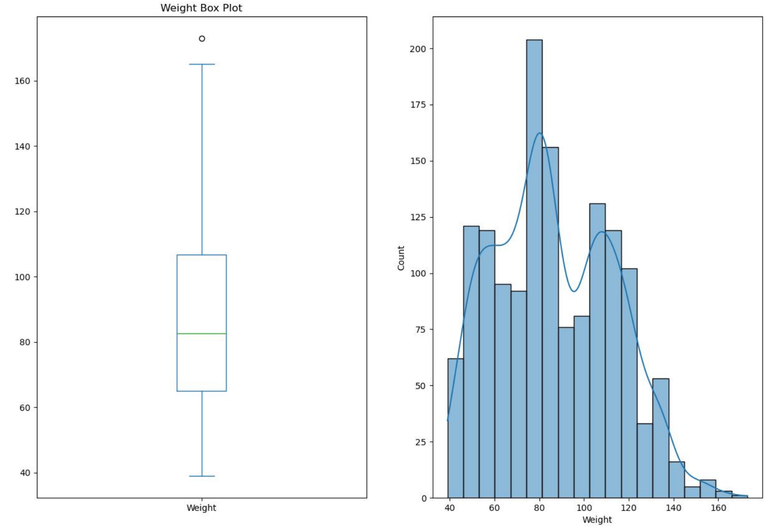




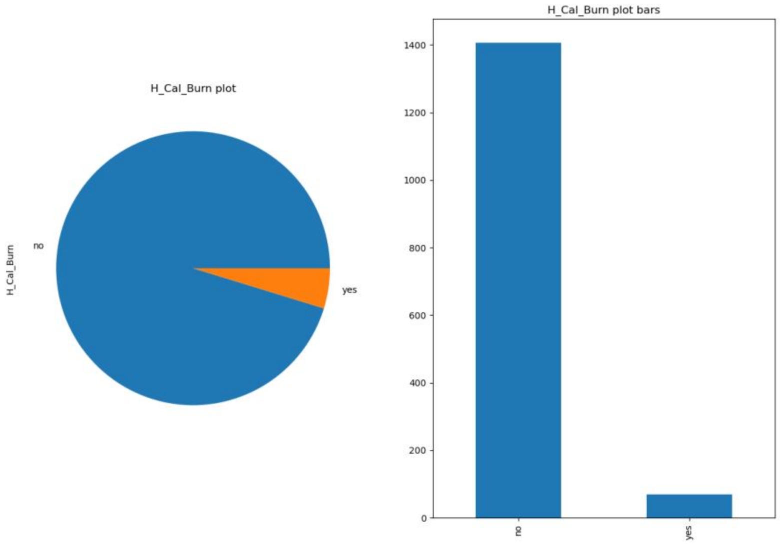
**The age feature is right skewed and it has some outliers but it is true one.**



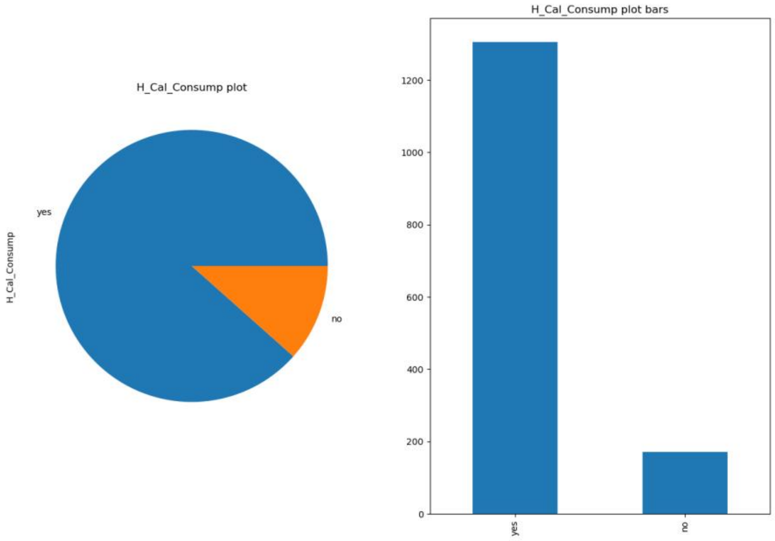
**Height is normally distributed.**



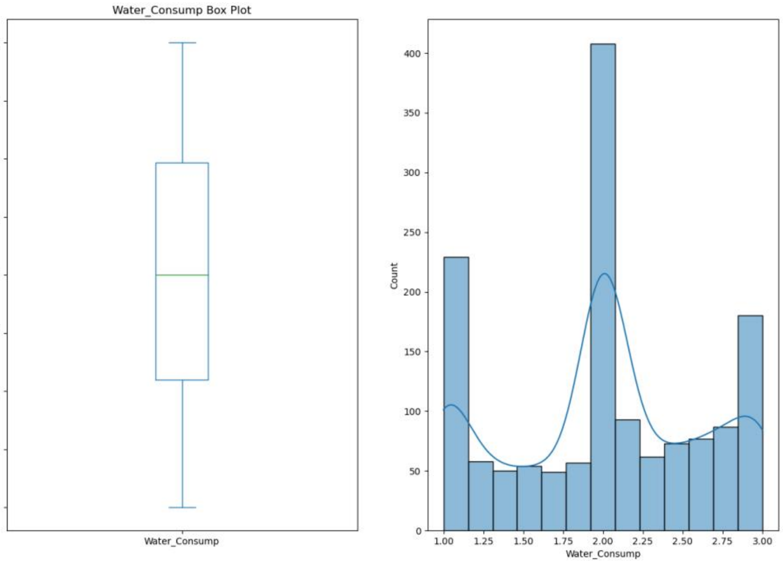
**Weight is normally distributed.**



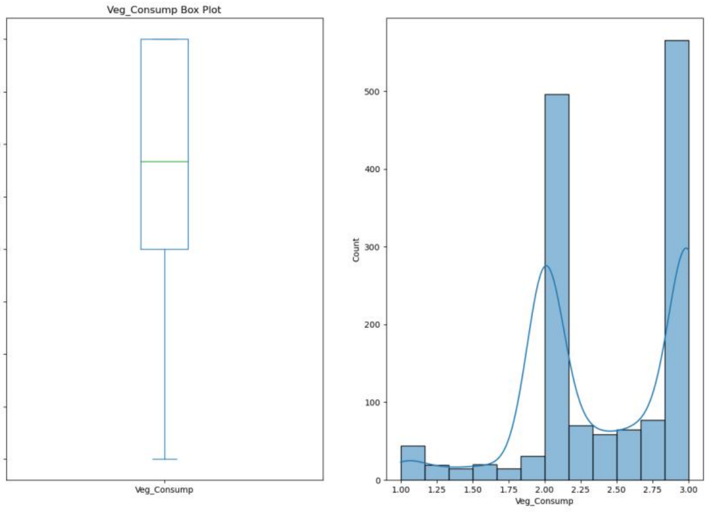
**The H_cal_Burn feature has imbalancing**



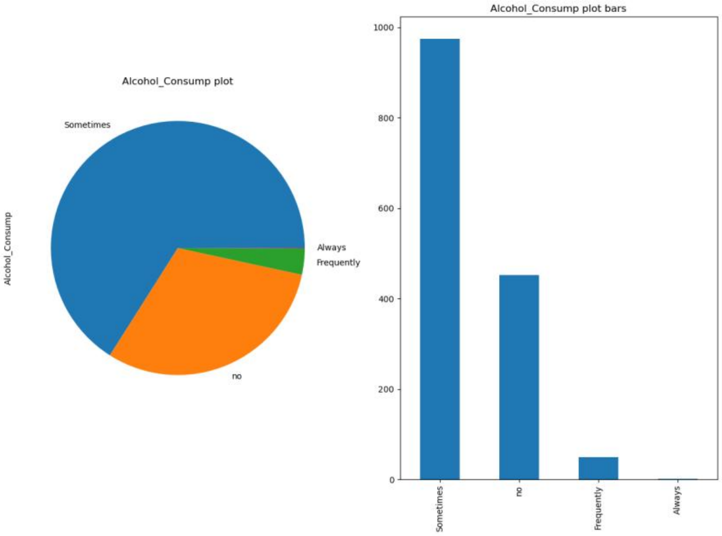
**The H_cal_Cal feature has imbalancing**



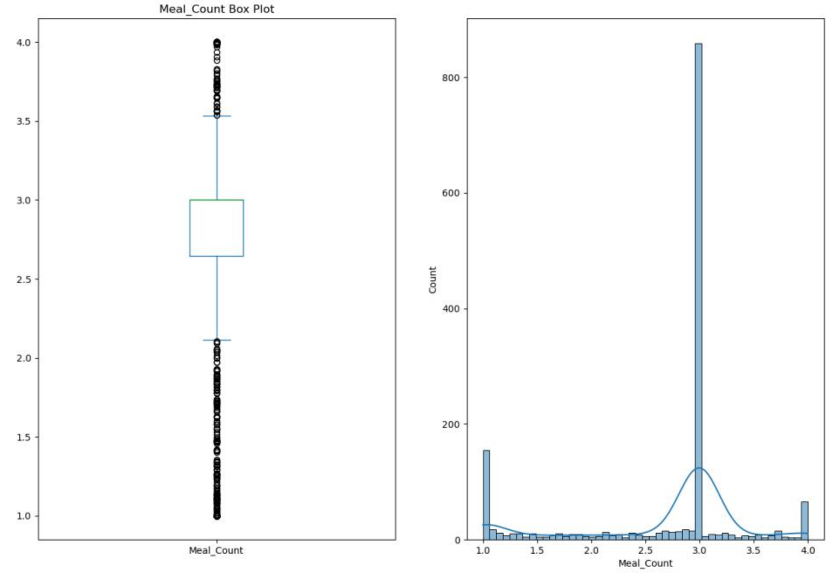
**Water consumption is normally distributed and has some peaks. We think that they were a missing values in the original dataset.**



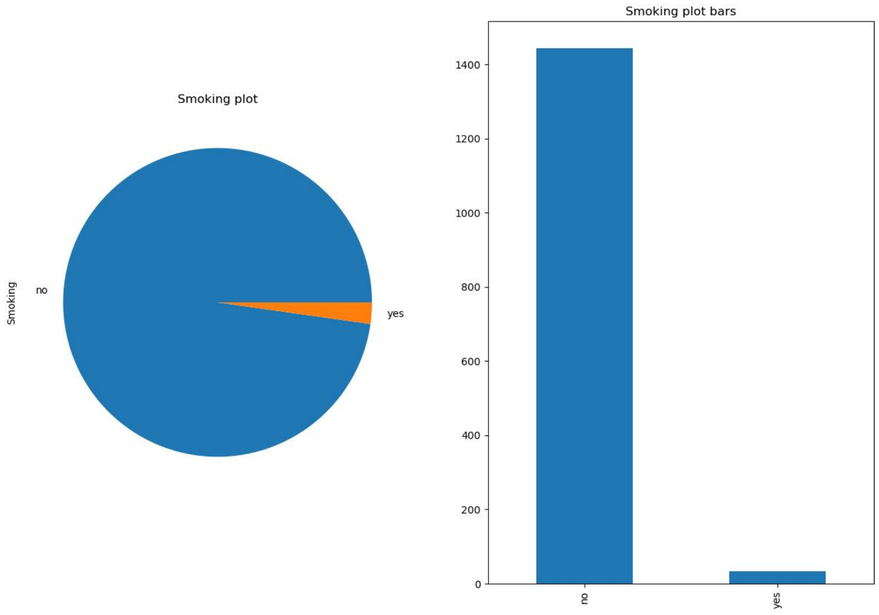
**The Veg_consumption feature is left skewed.**



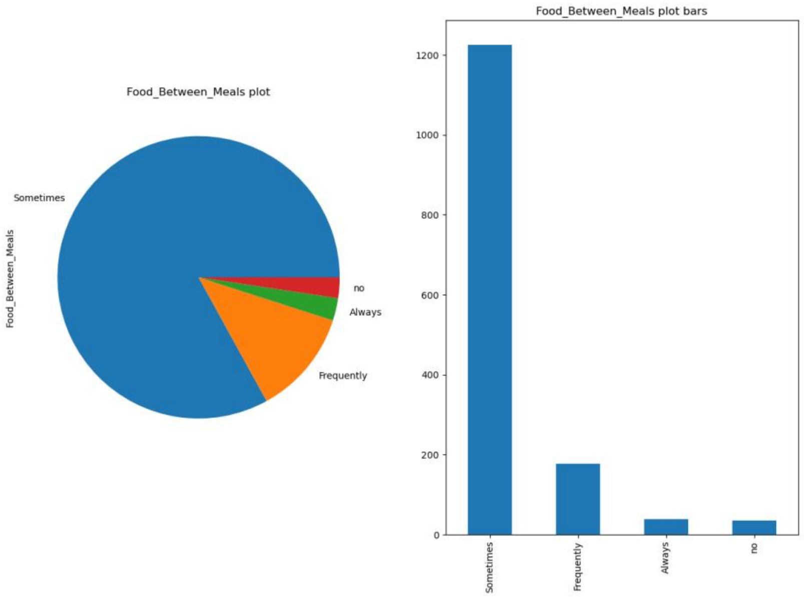
**The Alcohol_consumption feature has imbalancing**



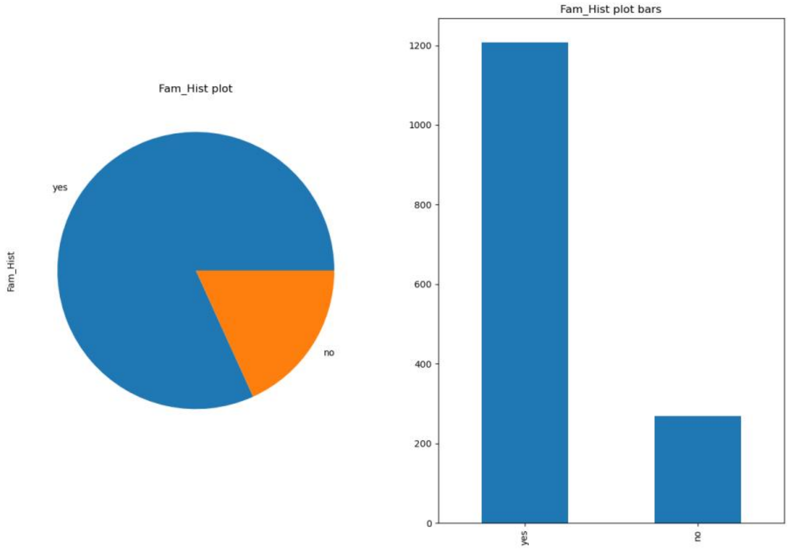
**The meat count feature is the worst feature in the dataset. It is very skewed and has a very high peak at 3. We think that they were a missing values in the original dataset.**



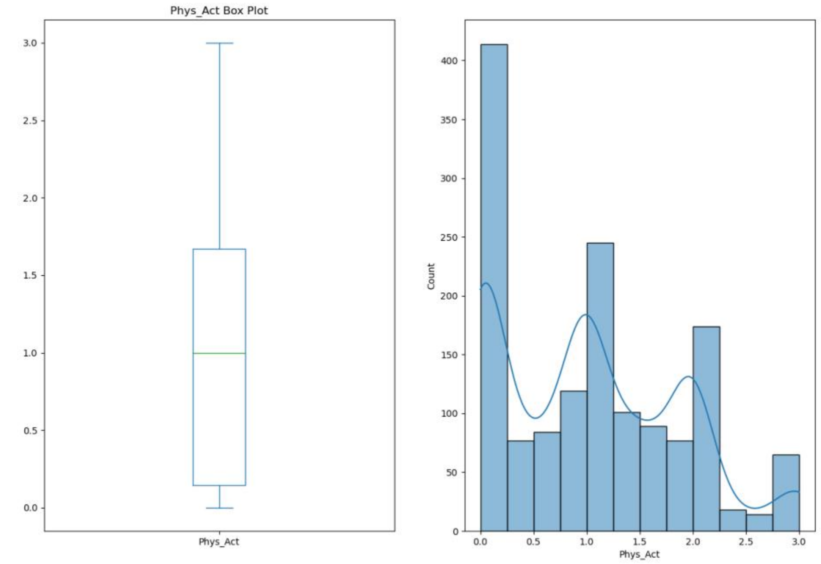
## The smoking feature has imbalancing



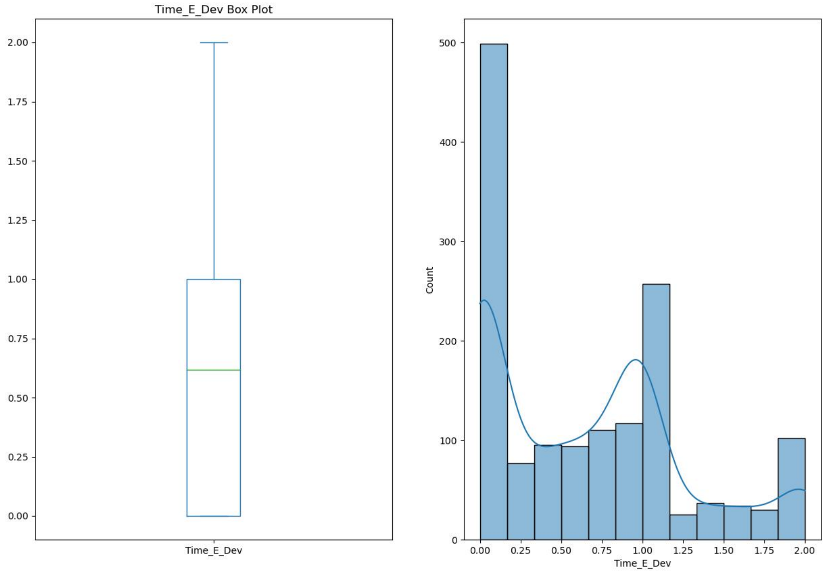
**The Food between meals feature has imbalancing**



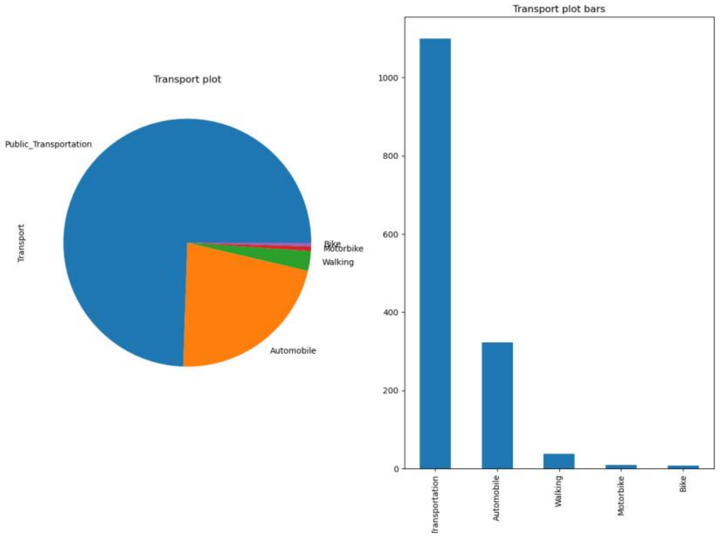
**The Fam_Hist meals feature has imbalancing**



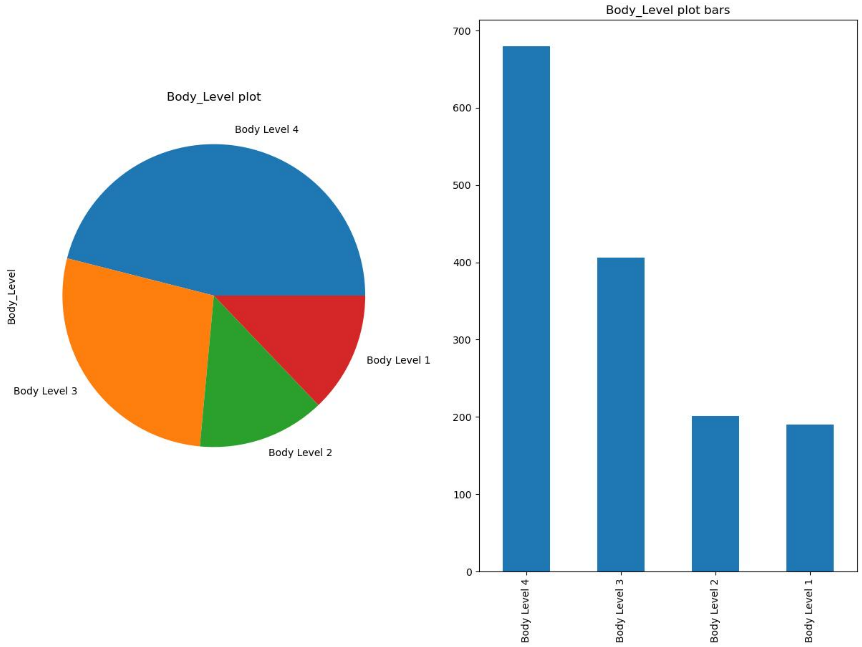
**The phys_Act feature is normally distributed but a little right skewed.**



**The Time_E_Dev is normally distributed but a little right skewed**



**The Transport feature has imbalancing**



**The main target of the dataset and as shown it has imbalancing.**

## 3- Extracting insights from data:

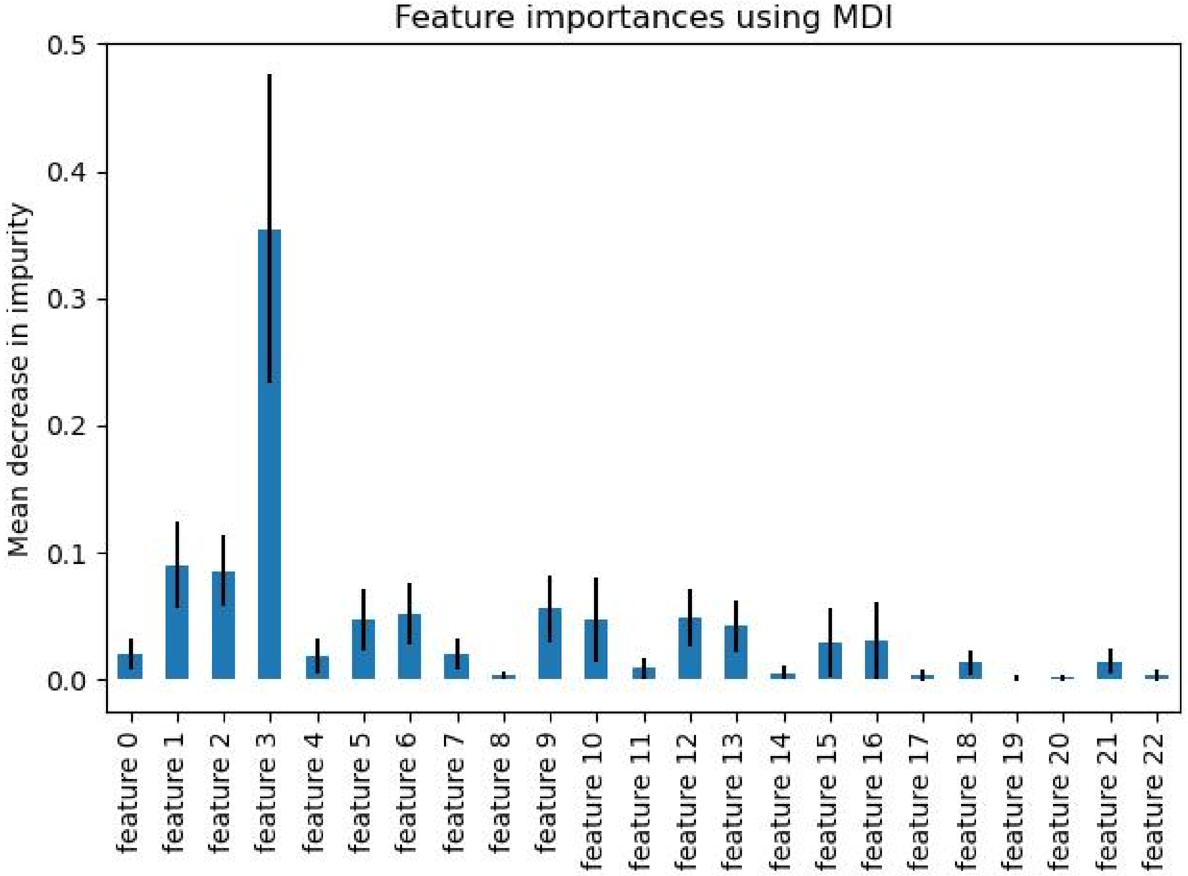
**Categorical features:**

- "H_Cal_Consump": The categorical values "yes" and "no" were replaced with numerical values of 1 and 0, respectively.
- "Gender": The categorical values "Male" and "Female" were replaced with numerical values of 1 and 0, respectively.
- "Alcohol_Consump": The categorical values "no", "Sometimes", "Frequently", and "Always" were replaced with numerical values of 0, 1, 2, and 3, respectively.
- "Smoking": The categorical values "yes" and "no" were replaced with numerical values of 1 and 0, respectively.
- "Fam_Hist": The categorical values "yes" and "no" were replaced with numerical values of 1 and 0, respectively.
- "H_Cal_Burn": The categorical values "yes" and "no" were replaced with numerical values of 1 and 0, respectively.
- "Food_Between_Meals" feature , one-hot encoding. This process created dummy variables for each unique category in the Food_Between_Meals feature, resulting in additional binary features.

By converting these categorical features into numerical representations, the models can effectively process and utilize this information during the training and prediction phases. This mapping allows the algorithms to understand the relationships and patterns within the categorical data and incorporate it into their decision-making process.

By transforming the categorical features into numerical representations, the models can leverage the full range of the dataset's information, contributing to improved accuracy and performance in the "Body Level Classification" task.

**Feature importance:**



## 4- Models:

### a. Logistic Regression:

Logistic Regression is a statistical algorithm used for binary and multiclass classification problems. It is a type of regression analysis that predicts the probability of an outcome by fitting data to a logistic function. In the context of our "Body Level Classification" project, logistic regression can be employed to model the relationship between the given attributes and the body level classes.

It estimates the probability of each class and assigns the data sample to the class with the highest probability. Logistic regression is a straightforward and interpretable algorithm that can handle both categorical and continuous input variables. It is especially useful when the relationship between the input variables and the outcome is expected to be linear or near-linear.

### b. Random Forest:

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It is a versatile algorithm suitable for both regression and classification tasks. In the case of our project, Random Forest can be applied to classify the body levels based on the given attributes. The algorithm creates a collection of decision trees, each trained on a different subset of the data and using a random selection of attributes. During prediction, each decision tree in the forest independently classifies the input sample, and the final prediction is determined by a majority vote or averaging of the individual tree predictions.

Random Forest is known for its ability to handle high-dimensional data, capture complex interactions between variables, and mitigate overfitting. It can also handle imbalanced datasets effectively, which is particularly important in our project.

### c. SVM:

Support Vector Machines (SVM) is another powerful machine learning algorithm that can be employed for the "Body Level Classification" problem. SVM is particularly effective in handling high-dimensional data and can handle both linear and non-linear classification tasks.

SVM works by finding an optimal hyperplane that maximally separates the different classes in the feature space. The objective is to find a decision boundary that has the largest margin, i.e., the maximum distance between the boundary and the closest data points from each class. This margin maximization approach makes SVM robust to outliers and helps generalize well to unseen data.

One of the advantages of SVM is its ability to handle class imbalance naturally. By adjusting the class weights or using specialized techniques like the cost-sensitive learning approach, SVM can effectively handle imbalanced datasets and ensure accurate classification across all classes.

**d. Nueral Network with Focal loss:**

Neural networks with focal loss have emerged as a promising approach for tackling classification problems, including the "Body Level Classification" task. Focal loss is a modification of the traditional cross-entropy loss function that addresses the issue of class imbalance by assigning higher weights to misclassified instances from the minority classes.

Focal loss introduces a modulating factor that reduces the contribution of well-classified instances, focusing instead on the more challenging and misclassified examples. By doing so, it effectively emphasizes the learning on the minority classes, leading to improved performance in scenarios with imbalanced datasets.

Applying focal loss in neural networks for the "Body Level Classification" problem allows the model to concentrate its learning on the classes that are harder to classify accurately. This approach enhances the model's ability to capture the intricate patterns and characteristics associated with the minority body level classes.

Neural networks, as highly flexible and adaptive models, are well-suited for complex classification tasks. By leveraging focal loss, neural networks can effectively address class imbalance and optimize the model's parameters to minimize the focal loss objective.

# Results and Evaluation:

**1. a.Logistic Regression:**

The resulted accuracy of 0.84 indicates that, on average, the logistic regression model achieves an 84% accuracy in predicting the correct body level class. Additionally, the low standard deviation of 0.01 suggests that the model's performance is relatively consistent across different cross-validation folds.

Furthermore, the f1_score function is used to compute the F1 score for each class separately. The resulting array shows the F1 scores for each of the four body level classes. F1 score combines both precision and recall, providing a balanced measure of the model's accuracy.

From the provided F1 scores, it can be observed that the model achieves high scores for most classes, with values ranging from 0.805 to 0.994. This indicates that the logistic regression model is capable of effectively capturing patterns in the data and distinguishing between different body level classes.

Overall, an accuracy of 0.84 and high F1 scores suggest that the logistic regression model shows promise in accurately classifying the body levels based on the given attributes.

**1. b.Logistic Regression using class weights:**

When using logistic regression with class weights and achieving an accuracy of 98.3%, it indicates that the model's performance has significantly improved compared to the previous result without considering class weights.

Class weights are used to address the issue of class imbalance in the dataset, where certain classes have more or fewer instances than others. By assigning higher weights to the minority classes and lower weights to the majority classes, the model is encouraged to pay more attention to the underrepresented classes during training. This helps to mitigate the bias towards the majority class and improve the model's ability to correctly classify instances from all classes.
The accuracy of 98.3% suggests that the logistic regression model, with the incorporation of class weights, is highly effective in predicting the correct body level class. This improvement in accuracy indicates that the model has

successfully adapted to the class imbalance and is making more accurate predictions across all classes.

In conclusion, achieving an accuracy of 98.3% with the logistic regression model using class weights is a significant improvement and indicates a strong performance in correctly classifying the body levels.

**1. c.Logistic Regression using class weights and oversampling:**
When using logistic regression with class weights and oversampling techniques to achieve an accuracy of 98.5%, it indicates a further improvement in the model's performance compared to the previous results.

Oversampling is a technique used to address class imbalance by artificially increasing the number of instances in the minority classes. This is typically done by duplicating or generating synthetic samples from the existing minority class data. By oversampling the minority classes, the model has more exposure to these classes during training, which helps in better learning their patterns and improving classification accuracy.

The accuracy of 98.5% suggests that the logistic regression model, with the combination of class weights and oversampling, is highly successful in accurately predicting the body level classes. This improvement in accuracy demonstrates the effectiveness of the chosen techniques in mitigating the impact of class imbalance and ensuring that the model makes accurate predictions across all classes.

In conclusion, achieving an accuracy of 98.5% with the logistic regression model using class weights and oversampling indicates a significant improvement in accurately classifying the body levels.

**2. a.Random Forest Regression:**
When using the Random Forest algorithm for the "Body Level Classification" problem and achieving an accuracy of 91%, it indicates a good performance, although slightly lower compared to the Logistic Regression model with class weights and oversampling.

An accuracy of 91% suggests that the Random Forest model is effective in accurately predicting the body level classes.

While 91% accuracy is a good result, it's worth noting that the accuracy achieved by the Random Forest model is slightly lower than the accuracy of the Logistic Regression model with class weights and oversampling.

In summary, achieving an accuracy of 91% with the Random Forest model indicates a good performance in classifying the body levels.

**2.b.Random Forest Regression with class weights:**
When using Random Forest with class weights and achieving an accuracy of 92.2%, it demonstrates a slight improvement in performance compared to the Random Forest model without class weights.

An accuracy of 92.2% indicates that the Random Forest model, with the inclusion of class weights, performs well in predicting the body level classes. This improvement suggests that considering class weights has helped the model to better handle the imbalanced nature of the dataset, leading to more accurate predictions across all classes.

In conclusion, achieving an accuracy of 92.2% with the Random Forest model using class weights indicates an improved performance in accurately classifying the body levels.

**2.c.Random Forest Regression with class weights and oversampling:**
When using Random Forest with class weights and oversampling techniques and achieving an accuracy of 93.6%, it indicates a further improvement in the model's performance compared to the previous results.

An accuracy of 93.6% suggests that the Random Forest model, with the combination of class weights and oversampling, is highly successful in accurately predicting the body level classes. This improvement in accuracy demonstrates the effectiveness of incorporating both techniques in mitigating the impact of class imbalance and ensuring accurate predictions across all classes.

In conclusion, achieving an accuracy of 93.6% with the Random Forest model using class weights and oversampling indicates a significant improvement in accurately classifying the body levels.

**3. a.Neural Network with focal loss:**

When using logistic regression as a one-layer neural network with four output neurons and the focal loss function, achieving an accuracy of 97.9% and f1-score of 98%  it suggests that the model is performing very well in accurately classifying the body levels.

In this setup, logistic regression is treated as a neural network with a single layer. The four output neurons correspond to the four body level classes, and the focal loss function is employed as the objective function to optimize the model's parameters.

An accuracy of 97.9% indicates that the logistic regression model, operating as a one-layer neural network, is highly effective in predicting the correct body level classes. It demonstrates the capability of the model to learn the patterns and relationships in the given dataset and make accurate predictions.

In summary, achieving an accuracy of 97.9% with logistic regression as a one-layer neural network and the focal loss function indicates a high performance in accurately classifying the body levels.

**3. b.Neural Network with focal loss and oversampling:**

When using logistic regression as a one-layer neural network with four output neurons, focal loss, and oversampling techniques, achieving an accuracy of 98.6% and an F1-score of 99%, it indicates an excellent performance in accurately classifying the body levels.

In this setup, logistic regression is treated as a simplified neural network architecture with a single layer and four output neurons representing the body level classes. The focal loss function is used as the objective function to optimize the model's parameters.

The accuracy of 98.6% suggests that the logistic regression model, with the combination of focal loss and oversampling, effectively handles the class imbalance in the dataset. Oversampling techniques artificially increase the number of instances in the minority classes, enabling the model to learn from a more balanced representation of the data. This leads to improved accuracy by reducing the bias towards the majority class.

Additionally, achieving an F1-score of 99% indicates that the model performs exceptionally well in capturing both precision and recall. The F1-score considers the balance between these two metrics, providing a robust measure of the model's overall performance.

In conclusion, achieving an accuracy of 98.6% and an F1-score of 99% with logistic regression as a one-layer neural network, focal loss, and oversampling demonstrates a highly accurate and robust classification of the body levels.

**4. a.SVM:**

When applying Support Vector Machines (SVM) to the "Body Level Classification" problem, the model achieved an accuracy of 88.2%. SVM is a powerful machine learning algorithm that is commonly used for classification tasks. It works by finding an optimal hyperplane that maximally separates the different classes in the dataset.

**4.b.SVM with oversampling:**

When applying SVM (Support Vector Machine) with oversampling to the "Body Level Classification" problem, the model achieved an accuracy of 88.6%. Oversampling is a technique used to address class imbalance by artificially increasing the number of samples in the minority class(es) to match the majority class.

By oversampling the minority class(es), the SVM model becomes more exposed to these instances during training, allowing it to learn their characteristics and make more accurate predictions. This helps in mitigating the bias towards the majority class and improving the overall performance of the model.

The achieved accuracy of 88.6% demonstrates the effectiveness of SVM with oversampling in addressing class imbalance and enhancing the model's ability to classify instances accurately.

**4.c.SVM with class weights and oversampling:**

When utilizing Support Vector Machines (SVM) with class weights and oversampling for the "Body Level Classification" problem, an accuracy of 89.8% was achieved. SVM is a powerful algorithm for classification tasks, known for its ability to handle complex decision boundaries and high-dimensional feature spaces.

By incorporating class weights, the SVM algorithm assigns higher weights to the minority class samples, thereby addressing the issue of class imbalance. This helps the model to focus more on correctly classifying the minority class, leading to improved overall accuracy.

In conclusion, the combination of SVM with class weights and oversampling proved to be a valuable approach for improving the accuracy of the "Body Level Classification" task. By addressing class imbalance and providing the model with a more balanced representation of the classes, the algorithm demonstrated enhanced performance in accurately classifying body levels.

| Model | Train accuracy | Test accuracy |
|---|---|---|
| **Logistic Regression** | 95.33 | 95.12 |
| **Logistic Regression with class weights** | 96 | 96.74 |
| **Logistic Regression with oversampling** | 95.52 | 97.56 |
| **Logistic Regression with class weights and oversampling** | 95.97 | 97.56 |
| **Random Forest** | 100 | 93 |
| **Random Forest with class weights** | 100 | 92.68 |
| **Random Forest with over sampling** | 92 | 84 |
| **Random Forest with class weights and oversampling** | 100 | 92.68 |
| **Neural Network with focal loss** | 97 | 96 |
| **Neural Network with focal loss and oversampling** | 99.2 | 99.3 |
| **SVM** | 95.9 | 88.2 |
| **SVM with class weights** | 88 | 88 |
| **SVM with oversampling** | 97 | 88.6 |
| **SVM with class weights and oversampling** | 88.6 | 89.8 |

## Logistic Regression Hyper parameters:

| **Model** | **Accuracy** |
|---|---|
| C = 0.001 | 49 |
| C = 0.01 | 67 |
| C = 0.1 | 83 |
| C = 1 | 89 |
| C = 10 | 94 |
| C = 100 | 96.4 |
| C = 200 | 96.5 |
| C = 300 | 96.7 |
| C = 400 | 96.7 |
| C = 500 | 96.8 |
| C = 600 | 96.9 |
| C = 700 | Saturation |

## Logistic Regression with oversampling Hyper parameters:

| Model | Accuracy |
|---|---|
| C = 0.001 | 56.5 |
| C = 0.01 | 71.8 |
| C = 0.1 | 86.5 |
| C = 1 | 93.9 |
| C = 10 | 97 |
| C = 100 | 98 |
| C = 200 | 98.2 |
| C = 300 | 98.3 |
| C = 400 | 98.39 |
| C = 500 | 98.39 |
| C = 600 | 98.4 |
| C = 700 | Saturation in the case of increase the accuracy will decreases |

## Random forest Hyper parameters:

| **Model** | | **Accuracy** |
|---|---|---|
| Max depth = 10 | N_estimatiors = 50 | 92 |
| Max depth = 20 | N_estimatiors = 50 | 93.66 |
| Max depth = 30 | N_estimatiors = 50 | 93.66 |
| Max depth = None | N_estimatiors = 50 | 93.66 |
| Max depth = 10 | N_estimatiors = 200 | 93 |
| **Max depth = 20** | **N_estimatiors = 200** | **94.7** |
| Max depth = 30 | N_estimatiors = 200 | 94.7 |
| Max depth = None | N_estimatiors = 200 | 94.7 |
| Max depth = 10 | N_estimatiors = 400 | 93.9 |
| Max depth = 20 | N_estimatiors = 400 | 94.3 |
| Max depth = 30 | N_estimatiors = 400 | 94.39 |
| Max depth = None | N_estimatiors = 400 | 94.39 |

## Random forest with oversampling Hyper parameters:

| **Model** | | **Accuracy** |
|---|---|---|
| Max depth = 10 | N_estimatiors = 50 | 95.5 |
| Max depth = 20 | N_estimatiors = 50 | 95.9 |
| Max depth = 30 | N_estimatiors = 50 | 95.96 |
| Max depth = None | N_estimatiors = 50 | 95.96 |
| Max depth = 10 | N_estimatiors = 200 | 95.7 |
| Max depth = 20 | N_estimatiors = 200 | 96.66 |
| Max depth = 30 | N_estimatiors = 200 | 96.66 |
| Max depth = None | N_estimatiors = 200 | 96.66 |
| Max depth = 10 | N_estimatiors = 400 | 96 |
| **Max depth = 20** | **N_estimatiors = 400** | **96.85** |
| Max depth = 30 | N_estimatiors = 400 | 96.85 |
| Max depth = None | N_estimatiors = 400 | 96.85 |

## SVM Hyper parameters:

| **Model** | | | **Accuracy** |
|---|---|---|---|
| C = 0.1 | Gama = 0.1 | Kernal = Sigmoid | 56.9 |
| C = 0.1 | Gama = 0.001 | Kernal = RBF | 37 |
| C = 0.1 | Gama = 0.001 | Kernal = Sigmoid | 37 |
| C = 0.1 | Gama = 0.0001 | Kernal = Linear | 92.5 |
| C = 1 | Gama = 0.1 | Kernal = Linear | 96.4 |
| C = 10 | Gama = 0.1 | Kernal = Linear | 98.2 |
| C = 100 | Gama = 0.1 | Kernal = Linear | 98.39 |