A PROJECT REPORT

ON

CALBURNT

For the partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted

By

Siddharth Singh

(2001921520060)

Diwanshi Sharma

(2001921520024)

Under the Supervision of

Ms. Anju Chanda

G.L. BAJAJ INSTITUTE OF TECHNOLOGY & MANAGEMENT, GREATER NOIDA

Affiliated to

DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY LUCKNOW

2022-23

Declaration

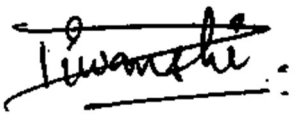
We hereby declare that the project work presented in this report entitled **"CalBurnt",** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science & Engineering, submitted to A.P.J. Abdul Kalam Technical University, Lucknow, is based on my own work carried out at Department of Computer Science & Engineering, G.L. Bajaj Institute of Technology & Management, Greater Noida. The work contained in the report is original and project work reported in this report has not been submitted by me/us for award of any other degree or diploma.

Signature:

Name: Siddharth Singh

Roll No: 2001921520060

Signature:



Name: Diwanshi Sharma

Roll No: 2001921520024

Date:

Place: Greater Noida

Certificate

This is to certify that the Project report entitled **"CalBurnt" done by Diwanshi Sharma (2001921520024)** and **Siddharth Singh (2001921520060)** is an original work carried out by them in Department of Computer Science & Engineering, G.L Bajaj Institute of Technology & Management, Greater Noida under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Date:

Ms. Anju Chanda
Signature of the Supervisor

Dr. Sansar Singh Chauhan
Head of the Department

Acknowledgement

The merciful guidance bestowed to us by the almighty made us stick out this project to a successful end. We humbly pray with sincere heart for his guidance to continue forever.

We pay thanks to our project guide Ms**. Anju Chanda** who has given guidance and light to us during this project. Her versatile knowledge has caused us in the critical times during the span of this project.

We pay special thanks to our Head of Department Dr. Sansar Singh Chauhan who has been always present as a support and help us in all possible way during this project.

We also take this opportunity to express our gratitude to all those people who have been directly and indirectly with us during the completion of the project.

We want to thanks our friends who have always encouraged us during this project.

At the last but not least thanks to all the faculty of CSE department who provided valuable suggestions during the period of project.

Abstract

Regular physical activities are essential to staying healthy and fit. The estimation of calories burned by individuals is based on a formula and MET charts. This study aims to predict the calories burned using a regression model as one of the machine learning algorithms to give more accurate results. Data preparation, cleaning, and analysis are the primary steps before they can be fed to the regression models. Model training and testing using K-fold validation were done to determine the best model for the study. The performance and prediction accuracy of regression models were evaluated based on the result of model testing after ten (10) iterations. The average accuracy was computed and the result shows that XGB-Regressor is the best model for the study with an accuracy of 99.63%. It is very important to visualize and study the relationships of the variables in the data because it may affect the performance of the algorithm in predicting the value of the target variable. The XGB-Regressor regression model was able to predict the calories burned with a high accuracy rate.

TABLE OF CONTENT

LIST OF FIGURES

Figures

LIST OF TABLES

Tables

Chapter 1

Introduction

The body temperature and the heartbeat will rise when we exercise or workout. The variables that we take here such as time scale for which the individual carrying out the workout training and what is the average beats per minute and then the temperature. Then we additionally take the height, weight, gender and age of the person to predict how tons energy the person may be burning.

A machine learning XGB regressor algorithm and linear regression algorithms are used to predict calories burned depends on the workout duration, body temperature, height, weight and age of the person.

1.1 Description

The variety of energy burned each day is immediately connected to weight loss, weight gain, or weight maintenance. To shed pounds, a person ought to burn greater calories than they take in, developing a calorie deficit. but, to do that, they want to recognize what number of calories they burn each day. Most people think about calories as most effective having to do with food and weight reduction. Calorie, a unit of energy or heat variously defined. Calorie may be defined as the amount of energy that is vital to increase 1 gram(g)of water by means of 1 ° C.

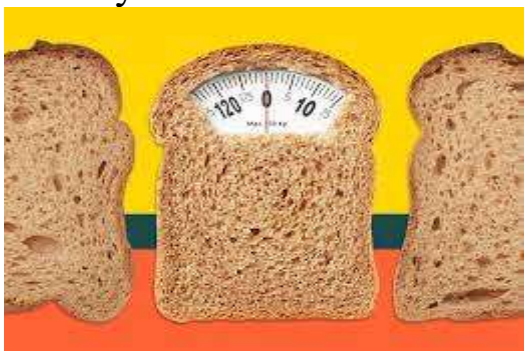


Fig1.1

This measurement can be carried out to lots of different strength releasing

mechanisms outdoor of the human body. In the case of human body, calories are measure of how much energy the body requires to function.

In order to be able to exercise how a whole lot of calories are burned every day is important to any man or woman trying to preserve, lose, or maintain weight. Understanding what elements contribute to calorie burning can help a person regulate their diet or workout program to deal with the aim. There are many factors that affect how much calories a person burns each day. Some of the elements that effect day by day calorie burn aren't in a person's manage at the same time as others may be changed. These factors include:

In case of age, the older a person is, the fewer calories burned per day. Gender: men burn greater energy than women. Quantity of daily activity: Body composition: those with more muscular tissues, burn more calories than people who've much less muscle. Body size: larger people burn greater calories than smaller human beings, even at relaxation. Thermogenesis: that is the amount of strength our body uses to break down meals.

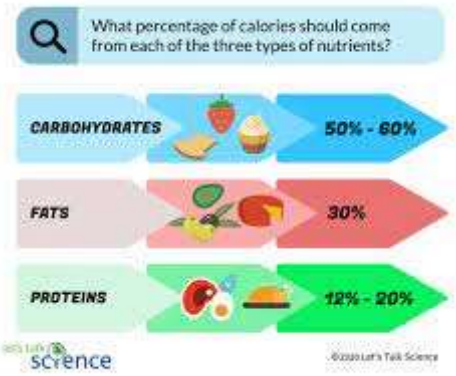


Fig 1.2

1.2 Key Facts

• Now it is usually diagnosed that a excessive-calorie food regimen with inadequate bodily pastime consequences in overweight or obesity, which in return should result in illnesses together with noninsulin-established diabetes, hypertension, cardiovascular sicknesses, endometrial cancer, and gallstones.

• Raised BMI is a major fear element for noncommunicable sick consisting

of cardiovascular diseases (especially coronary heart sickness and stroke), that have been the principle cause of dying in 2012-diabetes, musculoskeletal disorders (in particular osteoarthritis – a as a substitute disabling degenerative joint problems.

1.3 Machine learning

Machine learning is a dimension of artificial intelligence that's in certain mark out as the software applications turn out to be more correct at predicting outcomes without being explicitly programmed to accomplish that. To predict the new output values historical data is used by these algorithms.

Fig 1.3

Machine learning let the user to feed a computer algorithm, massive amount of information and then the computer will examine and make data-driven suggestions and decisions focusing on only the input data

1.4 Purpose of Study

This document is to predict the calorie burned during the workout of different people and compare the two algorithms in machine learning by looking through the data sets. The dataset used in this study has 7 features, one target variable, and 15000 instances.

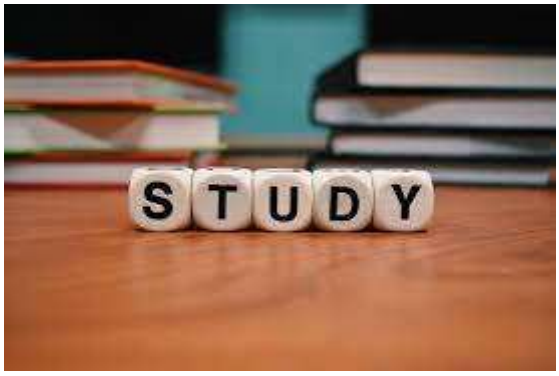


Fig. 1.4

We are using this data sets to train a dataset and find out the accurate algorithms and its mean absolute error and find the best model.

Chapter 2

Existing System

2.1 Introduction

In this project, we will create a Calorie burnt prediction model with the help of machine learning in python. In machine learning, we use XGB Regressor.

2.2 XGB Regressor

XGBoost is a powerful approach for building supervised regression models. The validity of this statement can be inferred by knowing about its (XGBoost) objective function and base learners. The objective function contains loss function and a regularization term. It tells about the difference between actual values and predicted values, i.e. how far the model results are from the real values.

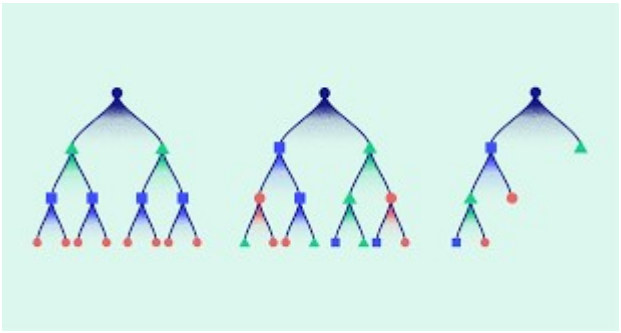


Fig. 2.1

The most common loss functions in XGBoost for regression problems is reg: linear, and that for binary classification is reg: logistics. Ensemble learning involves training and combining individual models (known as base learners) to get a single prediction, and XGBoost is one of the ensembles learning methods. XGBoost expects to have the base learners which are uniformly bad

at the remainder so that when all the predictions are combined, bad predictions cancel out and better one sums up to form final good predictions.

2.3 Python

Python was the language of choice for this project. This was an easy decision for the multiple reasons.

Fig. 2.2

• Python as a language has an enormous community behind it. Any problems that might be encountered can be easily solved with a trip to Stack Overflow. Python is among the most popular languages on the site which makes it very likely there will be a direct answer to any query.

• Python has an abundance of powerful tools ready for scientific computing. Packages such as NumPy, Pandas, and SciPy are freely available and well documented. Packages such as these can dramatically reduce, and simplify the code needed to write a given program. This makes iteration quick.

• Python as a language is forgiving and allows for programs that 7 look like pseudo code. This is useful when pseudocode given in academic papers needs to be implemented and tested. Using Python, this step is usually reasonably trivial. However, Python is not without its flaws. The language is dynamically typed and packages are notorious for Duck Typing. This can be frustrating when a package method returns something that, for example, looks like an array rather than being an actual array. Coupled with the fact that standard Python documentation does not explicitly state the return type of a method, this can lead to a lot of trials and error testing that would not

otherwise happen in a strongly typed language. This is an issue that makes learning to use a new Python package or library more difficult than it otherwise could be.

2.4 Python Libraries

Fig. 2.3

1.Pandas Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name “Pandas” has a reference to both “Panel Data”, and “Python Data Analysis”.

2.Numpy NumPy stands for ‘Numerical Python’ or ‘Numeric Python’. It is 8 an open source module of Python which provides fast mathematical computation on arrays and matrices. Since, arrays and matrices are an essential part of the Machine Learning ecosystem. NumPy’s main object is the homogeneous multidimensional array. It is a table with same type elements, i.e, integers or string or characters (homogeneous), usually integers. In NumPy, dimensions are called axes. The number of axes is called the rank.

3.Matplotlib Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

4.Sklearn(scikit learning) Scikit-learn is an open-source Python library that implements a range of machine learning, pre-processing, cross-validation,

and 9 visualization algorithms using a unified interface. Important features of scikit-learn:

- Simple and efficient tools for data mining and data analysis. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means, etc.

- Accessible to everybody and reusable in various contexts.

- Built on the top of NumPy, SciPy, and matplotlib.

- Open source, commercially usable – BSD license.

Chapter 3

Problem Formulation

3.1. Introduction to Problem Statement

In today's fast paced lifestyle people are just busy with their daily schedules i.e., their work, business and other errands. And even if they spare some time for themselves, they tend to spend it doing some indolent activities like, scrolling their social media, watching television or just laying on bed without any motives and wasting their time.

So, it becomes a necessity, spending our time wisely. We need to ditch that sedentary lifestyle and lead a healthy one. And there is no other better way to lead a healthy and fruitful lifestyle other than exercising.

Fig 3.1

Fig 3.2

So, when we exercise it also becomes very important to keep a track of the calories burnt during our workout session. No matter what exercises we do we can always track our calories based on various parameters.

Here our model comes into play, it helps us to determine the number of

calories burnt on the basis of certain information provided to it e.g., Heart-rate, duration, weight etc. It uses the XGB-Regressor to predict the number of calories burnt with high accuracy.

Chapter 4

System Analysis & Design

4.1. Data Collection

Fig 4.1

4.1.1 Choosing the dataset

In order to develop a good ML model, we need to choose the dataset wisely. As the entire prediction capabilities of ourmodel will be based on that dataset.

Generally, it is recommended for more accurate results we Shall collect first-hand data i.e., data collected on our own. But sometimes it becomes difficult to collect a large amount of data so one should go with certain trusted sites.

Here, we obtained our dataset from Kaggle.com as it is a trusted site for datasets. The datasets on Kaggle come with high accuracy.

4.2. Data Analysis

Fig 4.2

4.2.1 Structuring the dataset

Our dataset was divided into two separate files so we merged them into a single file for further operations. It consisted data of total 1500 individuals.

4.3. System Design

Fig 4.3

4.3.1. Splitting Data

In order to develop the model, we split the data into two parts. One for training other for testing.

4.3.2. XGB-Regressor

This regressor is used for the prediction of values. Among Various regressors the XGB is the most accurate one. There are other Regressors too like the linear Regressor. Now we load the training part of the dataset into the XGBoost Regressor for predicting the value with high accuracy.

Chapter 5

Implementation

5. Implementation using XGB-Regressor

1.Import the libraries

Various libraries are imported and each library has certain modules Which help us in performing various actions on our data.

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import sklearn.model_selection
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
```

Fig 5.1

2.Load the training dataset

Once we import the libraries, after that we imported a dataset from Kaggle for training our ML model.

The data set is divided into two files one named as ‘exercise’ other as ‘calories’.

The ‘exercise’ file is divided into 8 parts namely:

User_ID, **Gender,** **Age,** **Height,** **Weight,** **Duration,** **Heart_Rate** & **Body_Temp**

And ‘calories’ file is divided into two columns namely:

User_ID & **Calories**

```
calories.head(10)
```

	User_ID	Calories
0	14733363	231.0
1	14861698	66.0
2	11179863	26.0
3	16180408	71.0
4	17771927	35.0
5	15130815	123.0
6	19602372	112.0
7	11117088	143.0
8	12132339	134.0
9	17964668	72.0

Table 5.1

```
exercise.head(10)
```

	User_ID	Gender	Age	Height	Weight	Duration	Heart_Rate	Body_Temp
0	14733363	male	68	190.0	94.0	29.0	105.0	40.8
1	14861698	female	20	166.0	60.0	14.0	94.0	40.3
2	11179863	male	69	179.0	79.0	5.0	88.0	38.7
3	16180408	female	34	179.0	71.0	13.0	100.0	40.5
4	17771927	female	27	154.0	58.0	10.0	81.0	39.8
5	15130815	female	36	151.0	50.0	23.0	96.0	40.7
6	19602372	female	33	158.0	56.0	22.0	95.0	40.5
7	11117088	male	41	175.0	85.0	25.0	100.0	40.7
8	12132339	male	60	186.0	94.0	21.0	97.0	40.4
9	17964668	female	26	146.0	51.0	16.0	90.0	40.2

Table 5.2

3.Merging the files

After obtaining the two files we merge them through a function Called ‘concat’.

We merge ‘Calories’ column of file ‘calories’ with the file

'exercise'. Both the files had 'User_ID' as the common column. After concatenation saving the resultant in the 'calories' file.

```
calories=pd.concat([exercise,calories['Calories']],axis=1)
```

Fig 5.2

```
calories.head()
```

	User_ID	Gender	Age	Height	Weight	Duration	Heart_Rate	Body_Temp	Calories
0	14733363	male	68	190.0	94.0	29.0	105.0	40.8	231.0
1	14861698	female	20	166.0	60.0	14.0	94.0	40.3	66.0
2	11179863	male	69	179.0	79.0	5.0	88.0	38.7	26.0
3	16180408	female	34	179.0	71.0	13.0	100.0	40.5	71.0
4	17771927	female	27	154.0	58.0	10.0	81.0	39.8	35.0

Table 5.3

4.Shape

The 'shape' function determines the number of rows and Columns in our dataset.

```
calories.shape
```

```
(15000, 9)
```

Fig 5.3

5.Categorising data

Analyzing data to determine the number of males and females in the Dataset and also the age distribution among various age groups.

Gender count:

```
sns.countplot(calories['Gender'])
```

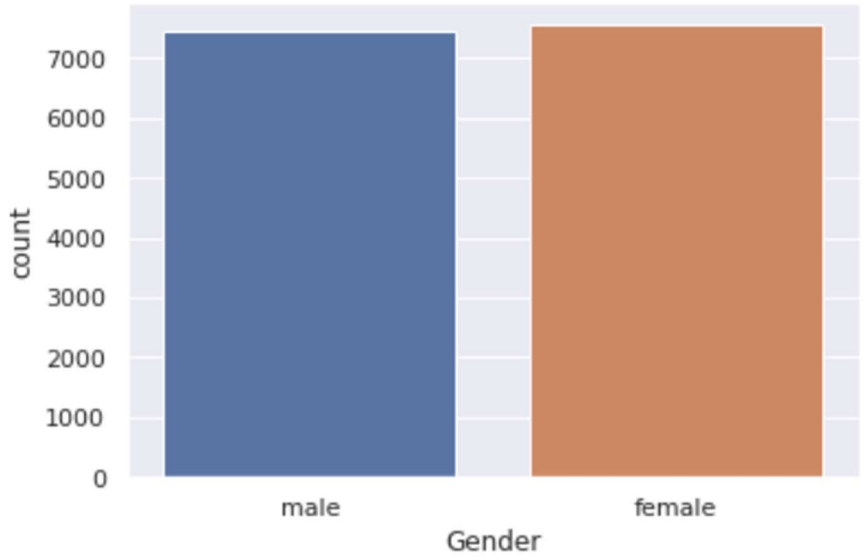


Fig 5.4

Age Distribution:

```
sns.distplot(calories['Age'])
```

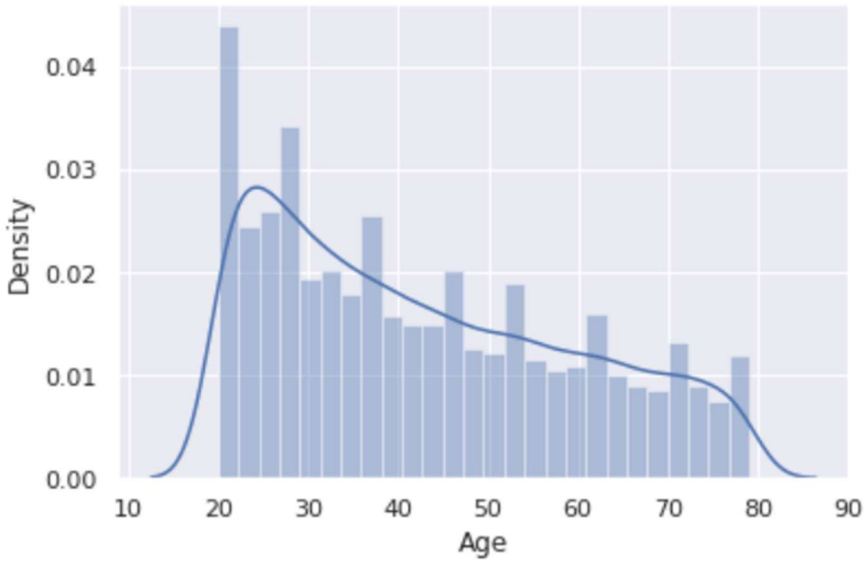


Fig 5.5

6.Generating a HeatMap

Heatmap is defined as a graphical representation of data using colors to visualize the value of the matrix. In this, to represent more common values or higher activities brighter colors basically reddish colors are used and to represent less common or activity values, darker colors are preferred. Heatmap is also defined by the name of

the shading matrix. Heatmaps in Seaborn can be plotted by using the seaborn.heatmap() function.

Parameters used in .heatmap()

1. data->name of the variable which stores the correlation
2. cbar->to determine whether colorbar will be present or not
3. annot->if true write data value in each cell
4. cmap->base colour of the shades
5. fmt->String formatting code to use when adding annotations
6. square->If True, set the Axes aspect to “equal” so each cell will be square-shaped
7. annot_kws->is used to change the text properties, typically the font size

.fig() is used to for a figure of size 12*12 of the heatmap.

```
plt.figure(figsize=(12,12))
sns.heatmap(correlation, cbar=True, square=True, fmt='1f', annot=True, annot_kws={'size':8}, cmap='Reds')
```

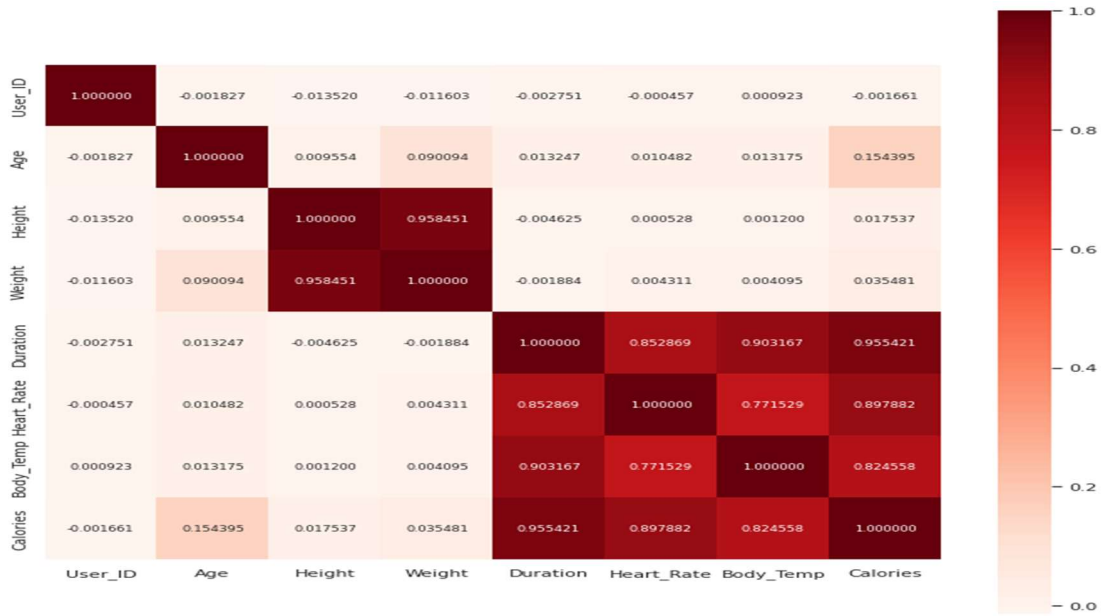


Fig 5.6

7.Replace non numerical attribute with numerical

In our dataset the column ‘Gender’ has non numerical values, so replacing them with numerical values.

```
calories.replace({"Gender":{'male':0,'female':1}},inplace=True)
```

	User_ID	Gender	Age	Height	Weight	Duration	Heart_Rate	Body_Temp	Calories
0	14733363	0	68	190.0	94.0	29.0	105.0	40.8	231.0
1	14861698	1	20	166.0	60.0	14.0	94.0	40.3	66.0
2	11179863	0	69	179.0	79.0	5.0	88.0	38.7	26.0
3	16180408	1	34	179.0	71.0	13.0	100.0	40.5	71.0
4	17771927	1	27	154.0	58.0	10.0	81.0	39.8	35.0

Table 5.4

8.Structuring the dataset for training

Dropping 'User_ID' from calories (not useful for ML). Dropping 'Calories' from calories(Calories is the target Column). After dropping storing 'calories' in X And "Calories' column in Y.

```
X=calories.drop(columns=['User_ID','Calories'],axis=1)
Y=calories['Calories']

print(X)
```

```
      Gender  Age  Height  Weight  Duration  Heart_Rate  Body_Temp
0          0   68   190.0    94.0      29.0       105.0       40.8
1          1   20   166.0    60.0      14.0        94.0       40.3
2          0   69   179.0    79.0       5.0        88.0       38.7
3          1   34   179.0    71.0      13.0       100.0       40.5
4          1   27   154.0    58.0      10.0        81.0       39.8
```

```
print(Y)
```

```
0      231.0
1       66.0
2       26.0
3       71.0
4       35.0
```

Table 5.5

9. Splitting the Dataset for Training and Testing

Dividing the dataset into two parts where one part is used for training our model and the second part for the testing purpose.

```
[ ]  X_train, X_test, Y_train, Y_test = train_test_split(X, Y,  test_size=0.2 ,random_state=2)
```

```
[ ]  print(X.shape, X_train.shape, X_test.shape)

     (15000, 7) (12000, 7) (3000, 7)
```

Fig 5.7

10. Training the Model

Now we train the dataset using the XGB-Regressor. We load the X_train & Y_train in the model for training purpose.

```
# loading the model
model =  XGBRegressor ()
```

```
#training the model with X_train
model.fit(X_train, Y_train)
```

Fig 5.8

Chapter 6

Result

6.1. Model Evaluation

1. Prediction on training data and test data

Predicting the calories on the basis of training and test data i.e. X_train and X_test and comparing with the original values of calories.

```
training_data_prediction= model.predict(X_train)
```

```
test_data_prediction= model.predict(X_test)
```

```
print(test_data_prediction)
```

```
[129.06204  223.79721   39.181965 ... 145.59767   22.53474   92.29064 ]
```

Fig 6.1

2.Comparing the score

Comparing the score of the test-data and training-data with Y_train and Y_test. The closer the values of r^2_train and r^2_test is, the more accurate model.

```
r2_train= metrics.r2_score( Y_train, training_data_prediction)
print(r2_train)
```

```
0.9966777021480265
```

```
r2_test= metrics.r2_score( Y_test, test_data_prediction)
print(r2_test)
```

```
0.9963065655529431
```

Fig 6.2

3. Mean Absolute Error

Finding the MAE in our prediction model.

```
mae = metrics.mean_absolute_error( Y_test, test_data_prediction)
```

```
print("Mean Absolute Error=", mae)
```

```
Mean Absolute Error= 2.7159012502233186
```

Fig 6.3

6.2. Result

1. Taking user-input

We take the input from the user about various details regarding his/her exercise sessions and based on that we predict the approximate number of calories burnt by the individual.

```
data= { 'Gender':1,
       'Age':39,
        'Height':175,
        'Weight':74,
        'Duration':16,
        'Heart_Rate':96,
        'Body_Temp':40.2}

df=pd.DataFrame(data,index=[0])
df
```

	Gender	Age	Height	Weight	Duration	Heart_Rate	Body_Temp
0	1	39	175	74	16	96	40.2

```
new_prediction= model.predict(df)
print(new_prediction)
```

```
[79.45823]
```

Fig 6.4

2. Model Score

Calculating the score of our model in percentage value.

Accuracy of the model is:

```
model.score( X_test,Y_test) *100
```

```
99.63065655529431
```

Fig 6.5

Chapter 7

Conclusion & Future Scope

7.1. Conclusion

We all know the importance of leading a healthy lifestyle nowadays. If we are able to track our calories along with exercising that's like icing on the cake. In this tutorial we've learned how crucial our our health is and how to build a ML model for predicting calories burnt.

We were successful in predicting the calories burnt using the XGB-Regressor in Machine Learning.

7.2. Improvements and Future Scope

Besides XGB-Regressor another regressor like the linear regressor can be used for the prediction. However, the XGB is more accurate than the linear regressor but we can compare the results of both the regressors to calculate the difference in percentage between the two and compare the accuracies of both of them.

We can also work on making our model even more accurate by reducing the value of the MAE of our model so that its score increases even further.

References

1. *"What is Machine Learning?". www.ibm.com. Retrieved 2021-08-15.*
2. *Zhou, Victor (2019-12-20). "Machine Learning for Beginners: An Introduction to Neural Networks". Medium. Retrieved 2021-08-15.*
3. Jump up to:[a] [b] Domingos 2015, Chapter 6, Chapter 7.
4. *Alpaydin (2020). Introduction to Machine Learning (Fourth ed.). MIT. pp. xix, 1–3, 13–18. ISBN 978-0262043793.*
5. *Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. **3** (3): 210–229. CiteSeerX 10.1.1.368.2254. doi:10.1147/rd.33.0210.*
6. R. Kohavi and F. Provost, "Glossary of terms," Machine Learning, vol. 30, no. 2–3, pp. 271–274, 1998.
7. *Gerovitch, Slava (9 April 2015). "How the Computer Got Its Revenge on the Soviet Union". Nautilus. Retrieved 19 September 2021.*
8. *Lindsay, Richard P. (1 September 1964). "The Impact of Automation On Public Administration". Western Political Quarterly. **17** (3): 78–81. doi:10.1177/106591296401700364. ISSN 0043-4078. S2CID 154021253. Retrieved 6 October 2021.*