1.0 - Introduction

In this project, i have chosen a dataset that focuses on body fat measurements of individuals. By examining the relationship between various body measurements and body fat percentage, we can uncover patterns and trends that are crucial for health and fitness research. Here is a detailed overview of the dataset's content:

Dataset Link: Body Fat Dataset (https://www.kaggle.com/datasets/fedesoriano/body-fat-prediction-dataset)

Variables

Variable Description Index A unique identifier for every individual in the dataset. BodyFat The body fat percentage of the individual. Age The age of the individual in years. Height The height of the individual measured in inches. Weight The weight of the individual measured in pounds. Neck Neck circumference of the individual in inches. Chest Chest circumference of the individual in inches. Abdomen Abdomen circumference of the individual in inches. Hip Hip circumference of the individual in inches. Thigh Thigh circumference of the individual in inches. Knee Knee circumference of the individual in inches. Ankle Ankle circumference of the individual in inches. Biceps Biceps circumference of the individual in inches.

2.0 - Problem Statement

The primary objective of this analysis is to determine if there is a significant correlation between body measurements and body fat percentage among the individuals in the dataset. By examining the relationship between these variables, we aim to identify any patterns or trends that could inform health and fitness recommendations. Specifically, we seek to understand how various measurements (e.g., weight, height, and circumferences) influence body fat percentage and whether this relationship can be used to develop predictive models for health assessments. This analysis will provide valuable insights that can be applied to improve health and fitness strategies, ultimately contributing to better overall well-being.

Data Loading and Preprocessing

In [15]:
```
# %% [markdown]
# 1. Data Loading and Preprocessing:
#

# %%
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv("C:/Users/harik/OneDrive/Documents/NWU DOCS/ML/week7/archive/bodyf

# Display basic information about the dataset
print(df.info())
```

```python
# Summary statistics to understand data distribution
print(df.describe())

# Check for missing values
print(df.isnull().sum())

# Prepare the independent variable (X) and dependent variable (y)
# Assuming 'BodyFat' is the target variable
X = df.drop(columns=['BodyFat'])  # Features
y = df['BodyFat']  # Target

# Display the first few rows of X and y to verify the data
print(X.head())
print(y.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 252 entries, 0 to 251
Data columns (total 15 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   Density  252 non-null    float64
 1   BodyFat  252 non-null    float64
 2   Age      252 non-null    int64  
 3   Weight   252 non-null    float64
 4   Height   252 non-null    float64
 5   Neck     252 non-null    float64
 6   Chest    252 non-null    float64
 7   Abdomen  252 non-null    float64
 8   Hip      252 non-null    float64
 9   Thigh    252 non-null    float64
 10  Knee     252 non-null    float64
 11  Ankle    252 non-null    float64
 12  Biceps   252 non-null    float64
 13  Forearm  252 non-null    float64
 14  Wrist    252 non-null    float64
dtypes: float64(14), int64(1)
memory usage: 29.7 KB
None
          Density     BodyFat         Age      Weight      Height        Neck  \
count  252.000000  252.000000  252.000000  252.000000  252.000000  252.000000   
mean     1.055574   19.150794   44.884921  178.924405   70.148810   37.992063   
std      0.019031    8.368740   12.602040   29.389160    3.662856    2.430913   
min      0.995000    0.000000   22.000000  118.500000   29.500000   31.100000   
25%      1.041400   12.475000   35.750000  159.000000   68.250000   36.400000   
50%      1.054900   19.200000   43.000000  176.500000   70.000000   38.000000   
75%      1.070400   25.300000   54.000000  197.000000   72.250000   39.425000   
max      1.108900   47.500000   81.000000  363.150000   77.750000   51.200000   

            Chest     Abdomen         Hip       Thigh        Knee       Ankle  \
count  252.000000  252.000000  252.000000  252.000000  252.000000  252.000000   
mean   100.824206   92.555952   99.904762   59.405952   38.590476   23.102381   
std      8.430476   10.783077    7.164058    5.249952    2.411805    1.694893   
min     79.300000   69.400000   85.000000   47.200000   33.000000   19.100000   
25%     94.350000   84.575000   95.500000   56.000000   36.975000   22.000000   
50%     99.650000   90.950000   99.300000   59.000000   38.500000   22.800000   
75%    105.375000   99.325000  103.525000   62.350000   39.925000   24.000000   
max    136.200000  148.100000  147.700000   87.300000   49.100000   33.900000   

           Biceps     Forearm       Wrist  
count  252.000000  252.000000  252.000000  
mean    32.273413   28.663889   18.229762  
std      3.021274    2.020691    0.933585  
min     24.800000   21.000000   15.800000  
25%     30.200000   27.300000   17.600000  
50%     32.050000   28.700000   18.300000  
75%     34.325000   30.000000   18.800000  
max     45.000000   34.900000   21.400000  
Density    0
BodyFat    0
Age        0
Weight     0
```

```
Height     0
Neck       0
Chest      0
Abdomen    0
Hip        0
Thigh      0
Knee       0
Ankle      0
Biceps     0
Forearm    0
Wrist      0
dtype: int64
   Density  Age  Weight  Height  Neck  Chest  Abdomen    Hip  Thigh  Knee  \
0   1.0708   23  154.25   67.75  36.2   93.1     85.2   94.5   59.0  37.3
1   1.0853   22  173.25   72.25  38.5   93.6     83.0   98.7   58.7  37.3
2   1.0414   22  154.00   66.25  34.0   95.8     87.9   99.2   59.6  38.9
3   1.0751   26  184.75   72.25  37.4  101.8     86.4  101.2   60.1  37.3
4   1.0340   24  184.25   71.25  34.4   97.3    100.0  101.9   63.2  42.2

   Ankle  Biceps  Forearm  Wrist
0   21.9    32.0     27.4   17.1
1   23.4    30.5     28.9   18.2
2   24.0    28.8     25.2   16.6
3   22.8    32.4     29.4   18.2
4   24.0    32.2     27.7   17.7
0    12.3
1     6.1
2    25.3
3    10.4
4    28.7
Name: BodyFat, dtype: float64
```

Model Training

In [16]:
```python
# %% [markdown]
# 2. Model Training:
#

# %%
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

# Split the dataset into training (70%) and testing (30%) sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_sta

# Initialize the Linear Regression model
model = LinearRegression()

# Train the model using the training data
model.fit(X_train, y_train)

# Display the model's coefficients and intercept
print("Model Coefficients (Slope):", model.coef_)
print("Model Intercept:", model.intercept_)
```

```
Model Coefficients (Slope): [-3.98261574e+02  1.71417620e-02  2.18776999e-02 -1.5609
6890e-02
 -1.64625729e-02  1.44441367e-02  4.27485697e-02  1.21852935e-02
 -3.56783523e-02 -1.85760008e-02 -1.09764258e-01 -4.78884078e-02
  5.03318286e-03 -5.06636668e-02]
Model Intercept: 437.6663101932903
```

Evaluation using Mean Squared Error (MSE)

In [17]:
```python
# %% [markdown]
# 3. Evaluation using Mean Squared Error (MSE):
#

# %%
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import numpy as np

# Predict on the test set
y_pred = model.predict(X_test)

# Calculate evaluation metrics
mse = mean_squared_error(y_test, y_pred)  # Mean Squared Error
mae = mean_absolute_error(y_test, y_pred)  # Mean Absolute Error
rmse = np.sqrt(mse)  # Root Mean Squared Error
r2 = r2_score(y_test, y_pred)  # R-squared

# Display all metrics
print(f"Mean Squared Error (MSE): {mse}")
print(f"Mean Absolute Error (MAE): {mae}")
print(f"Root Mean Squared Error (RMSE): {rmse}")
print(f"R-squared: {r2}")
```

```
Mean Squared Error (MSE): 0.6257022006507393
Mean Absolute Error (MAE): 0.5402632667480103
Root Mean Squared Error (RMSE): 0.7910134010563533
R-squared: 0.9879118942880447
```

Reflection on the Problem and Solution

The evaluation metrics for the regression model provide insightful interpretations of the model's performance in predicting body fat percentage based on various features. Here’s how we can interpret the results based on the calculated metrics:

Mean Squared Error (MSE): A lower MSE indicates that the predictions are close to the actual values. If the MSE is acceptable based on the context of the problem, we can consider the model effective. However, high MSE values suggest that the model may require improvements, either by incorporating additional features or by exploring different algorithms.

Mean Absolute Error (MAE): The MAE provides a straightforward interpretation of the average prediction error in the same unit as the target variable (percentage). An acceptable MAE suggests that the model is reasonably accurate. A higher MAE might indicate that the model is consistently offtarget, requiring a reassessment of the features or model choice.

Root Mean Squared Error (RMSE): This metric, being in the same unit as the target variable, allows for intuitive understanding. A lower RMSE implies that the model's predictions closely follow the actual body fat percentages. RMSE is sensitive to outliers, so if the RMSE is disproportionately high, it may indicate that some extreme values are negatively impacting the model's performance.

Rsquared: An Rsquared value close to 1 implies that a substantial proportion of the variance in the body fat percentage can be explained by the features, indicating a good fit. However, if the Rsquared is low, it suggests that the model is not capturing the underlying relationship well, which may warrant further feature exploration or model adjustments.

In summary, while the regression model demonstrates a decent predictive capability, as indicated by the MSE, MAE, RMSE, and Rsquared values, there is potential for enhancement. Factors such as additional relevant features, data preprocessing, or even experimenting with more complex models could lead to improved performance. Continuous refinement based on these evaluations can help create a more robust and effective predictive model in healthrelated contexts.