Calories Burnt Prediction

April 19, 2025

[1]:
```
import pandas as pd
```

[2]:
```
import numpy as np
```

[3]:
```
import matplotlib.pyplot as plt
```

[4]:
```
import seaborn as sns
```

[5]:
```
from sklearn.model_selection import train_test_split
```

[6]:
```
from xgboost import XGBRegressor
```

[7]:
```
from sklearn import metrics
```

[8]:
```
calories =pd.read_csv('calories.csv')
```

[9]:
```
calories.head()
```

[9]:
```
    User_ID  Calories
0  14733363     231.0
1  14861698      66.0
2  11179863      26.0
3  16180408      71.0
4  17771927      35.0
```

[11]:
```
exercise_data=pd.read_csv('exercise.csv')
```

[12]:
```
exercise_data.head()
```

[12]:
```
    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
```

Combining the Two DataFrames

[13]:
```
calories_data=pd.concat([exercise_data,calories['Calories']],axis=1)
```

[14]:
```
calories_data.head()
```

[14]:
```
    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

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

[15]:
```
calories_data.shape
```

[15]:
```
(15000, 9)
```

[16]:
```
calories_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15000 entries, 0 to 14999
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   User_ID     15000 non-null  int64
 1   Gender      15000 non-null  object
 2   Age         15000 non-null  int64
 3   Height      15000 non-null  float64
 4   Weight      15000 non-null  float64
 5   Duration    15000 non-null  float64
 6   Heart_Rate  15000 non-null  float64
 7   Body_Temp   15000 non-null  float64
 8   Calories    15000 non-null  float64
dtypes: float64(6), int64(2), object(1)
memory usage: 1.0+ MB
```

[17]:
```
calories_data.isnull().sum()
```

[17]:
```
User_ID       0
Gender        0
Age           0
Height        0
Weight        0
Duration      0
Heart_Rate    0
```

```
Body_Temp     0
Calories      0
dtype: int64
```

```
[18]: calories_data.describe()
```

```
[18]:             User_ID           Age        Height        Weight      Duration  \
count  1.500000e+04  15000.000000  15000.000000  15000.000000  15000.000000
mean   1.497736e+07     42.789800    174.465133     74.966867     15.530600
std    2.872851e+06     16.980264     14.258114     15.035657      8.319203
min    1.000116e+07     20.000000    123.000000     36.000000      1.000000
25%    1.247419e+07     28.000000    164.000000     63.000000      8.000000
50%    1.499728e+07     39.000000    175.000000     74.000000     16.000000
75%    1.744928e+07     56.000000    185.000000     87.000000     23.000000
max    1.999965e+07     79.000000    222.000000    132.000000     30.000000

         Heart_Rate     Body_Temp      Calories
count  15000.000000  15000.000000  15000.000000
mean      95.518533     40.025453     89.539533
std        9.583328      0.779230     62.456978
min       67.000000     37.100000      1.000000
25%       88.000000     39.600000     35.000000
50%       96.000000     40.200000     79.000000
75%      103.000000     40.600000    138.000000
max      128.000000     41.500000    314.000000
```

```
[26]: data=calories_data.drop(['Gender','User_ID'],axis=1).copy()
```

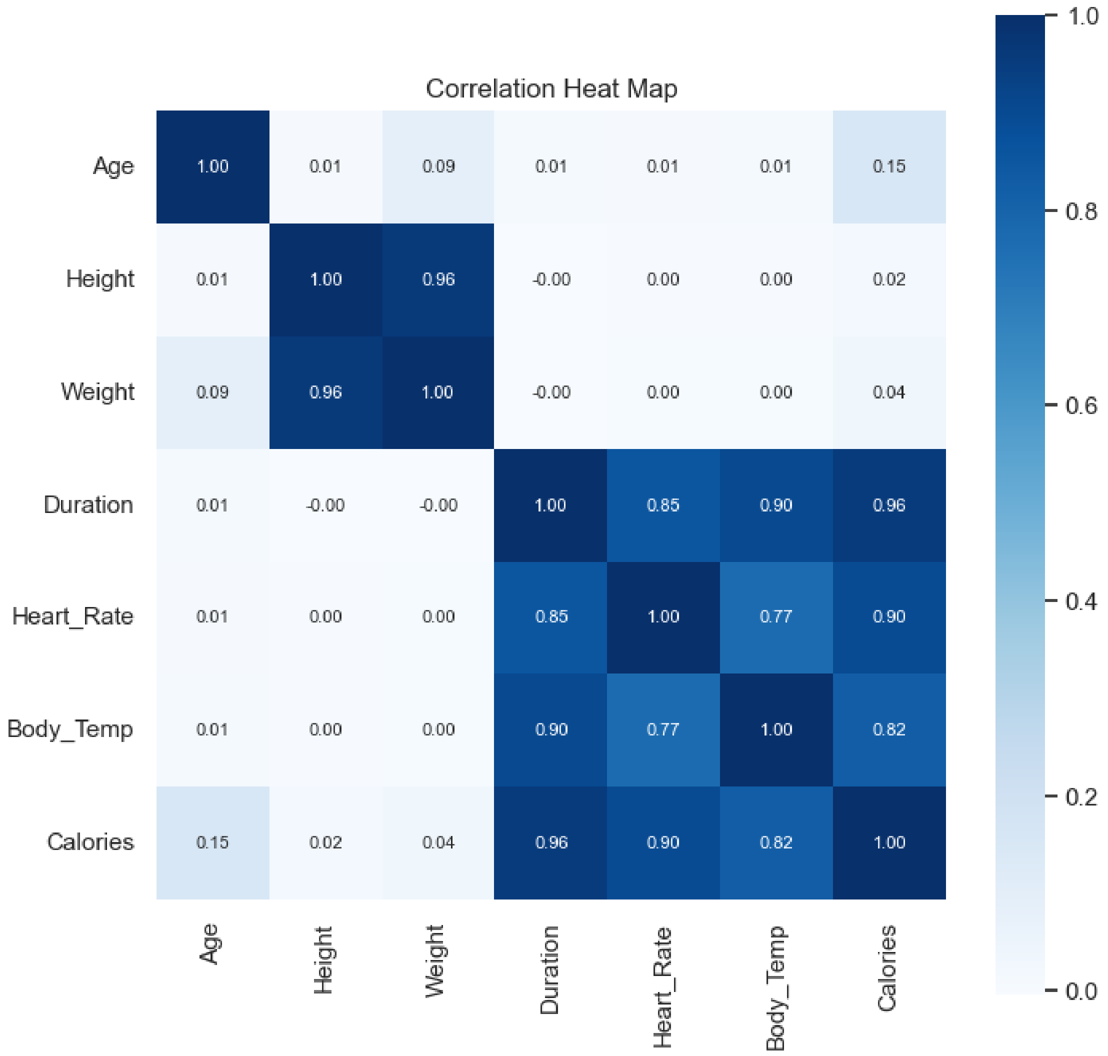
```
[27]: data.head()
```

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

```
[28]: correlation=data.corr()
```

```
[52]: plt.figure(figsize=(8,8))
sns.heatmap(correlation,cbar=True,annot=True,annot_kws={'size':8},fmt='.
 ↪2f',cmap='Blues',square=True)
plt.title("Correlation Heat Map")
```

```
[52]: Text(0.5, 1.0, 'Correlation Heat Map')
```

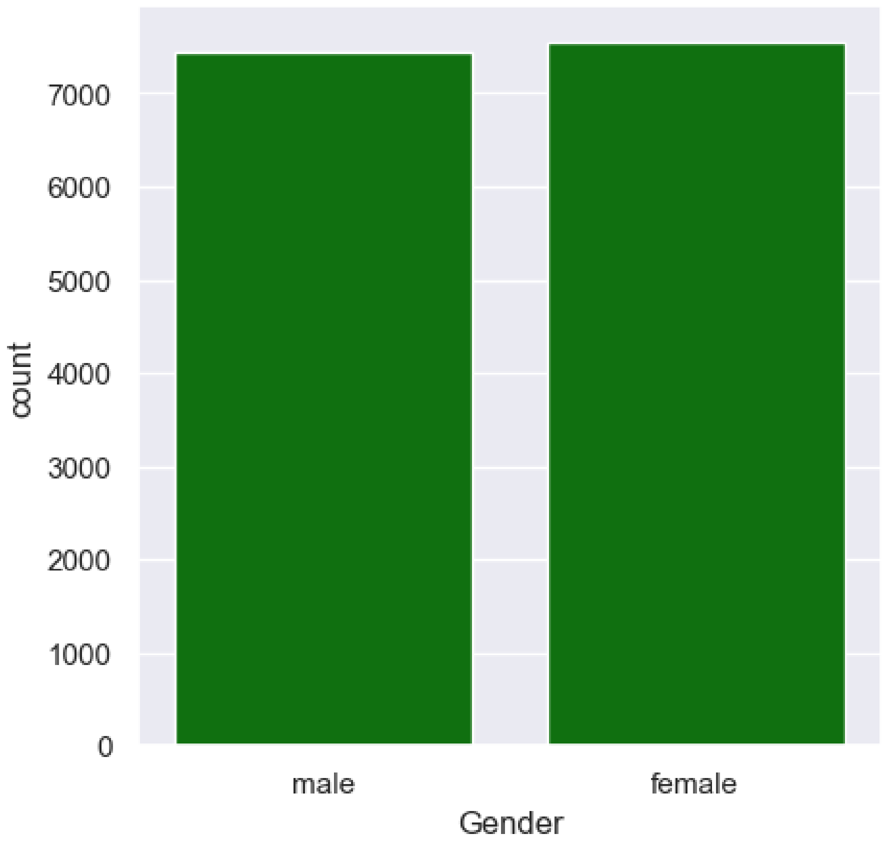


Data Visualization

```
[31]: sns.set(style="darkgrid")
```

```
[37]: plt.figure(figsize=(5,5))
      sns.countplot(x="Gender",data=calories_data,color='green')
```

```
[37]: <Axes: xlabel='Gender', ylabel='count'>
```



[33]:

```
calories_data['Gender'].value_counts()
```

[33]:

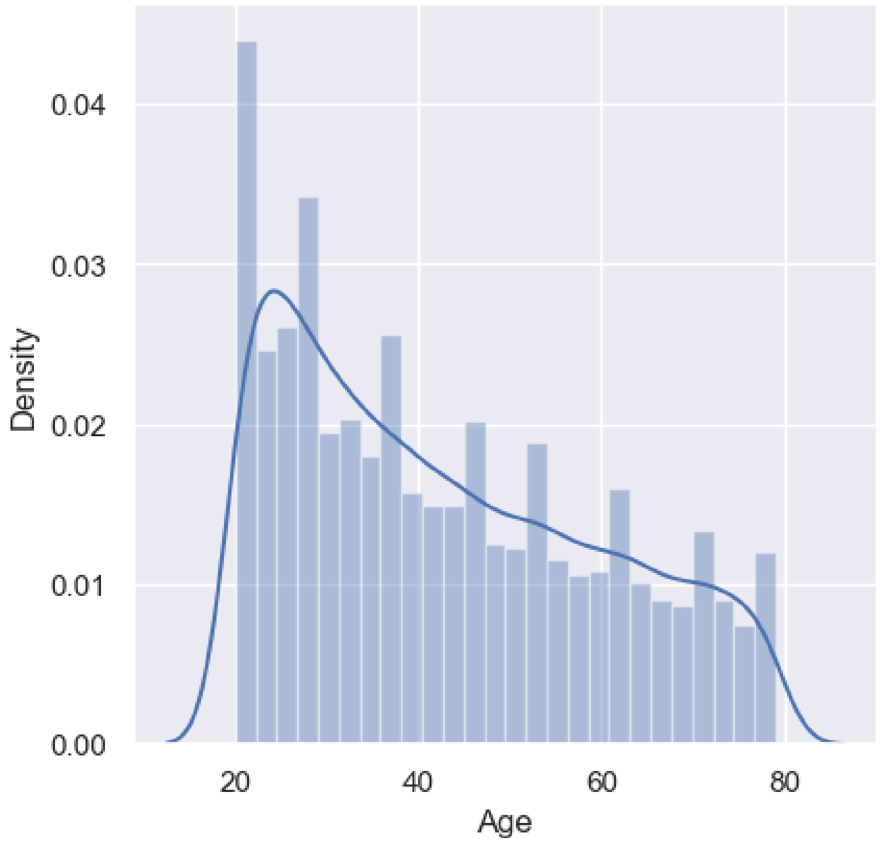
```
Gender
female    7553
male      7447
Name: count, dtype: int64
```

[42]:

```
plt.figure(figsize=(5,5))
sns.distplot(calories_data['Age'])
```

```
C:\Users\indhu\AppData\Local\Temp\ipykernel_6152\3801789279.py:2: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

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

```
[42]: <Axes: xlabel='Age', ylabel='Density'>
```



```
[48]: calories_data['Age'].value_counts().reset_index()
```

```
[48]:     Age  count
      0    20    512
      1    21    497
      2    22    489
      3    26    452
      4    25    435
      5    24    426
      6    23    411
      7    27    396
      8    28    390
      9    29    379
      10   32    370
      11   31    339
      12   35    327
      13   30    325
      14   33    320
```

15	38	311
16	34	285
17	37	283
18	36	279
19	39	276
20	40	260
21	44	259
22	42	257
23	41	252
24	43	250
25	46	233
26	45	233
27	52	225
28	47	223
29	50	220
30	48	219
31	54	217
32	49	208
33	55	204
34	53	201
35	59	198
36	51	197
37	56	190
38	63	182
39	58	182
40	61	181
41	62	180
42	57	177
43	64	176
44	60	172
45	65	169
46	73	167
47	71	163
48	67	162
49	70	154
50	68	150
51	79	147
52	66	147
53	69	145
54	74	142
55	76	139
56	72	137
57	78	133
58	77	130
59	75	117

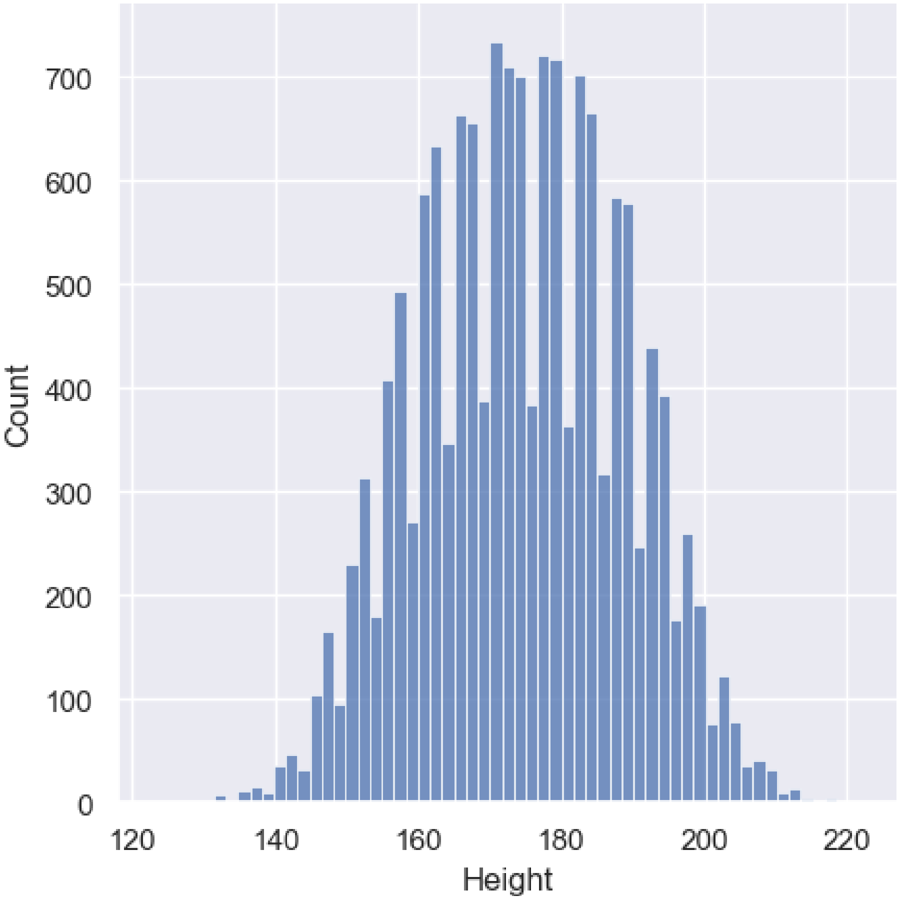
[43]:

```
plt.figure(figsize=(5,5))
sns.displot(calories_data['Height'])
```

[43]:

```
<seaborn.axisgrid.FacetGrid at 0x1c72cb1a900>
```

```
<Figure size 500x500 with 0 Axes>
```



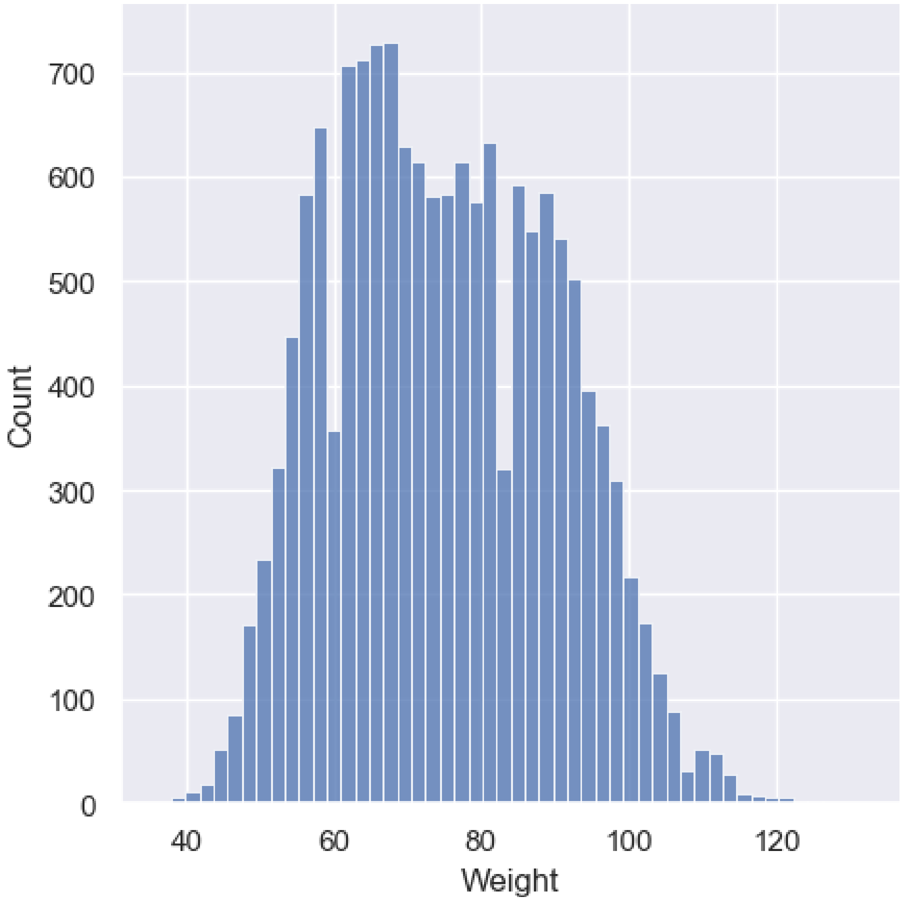
[44]:

```
plt.figure(figsize=(5,5))
sns.displot(calories_data['Weight'])
```

[44]:

```
<seaborn.axisgrid.FacetGrid at 0x1c72dce1190>
```

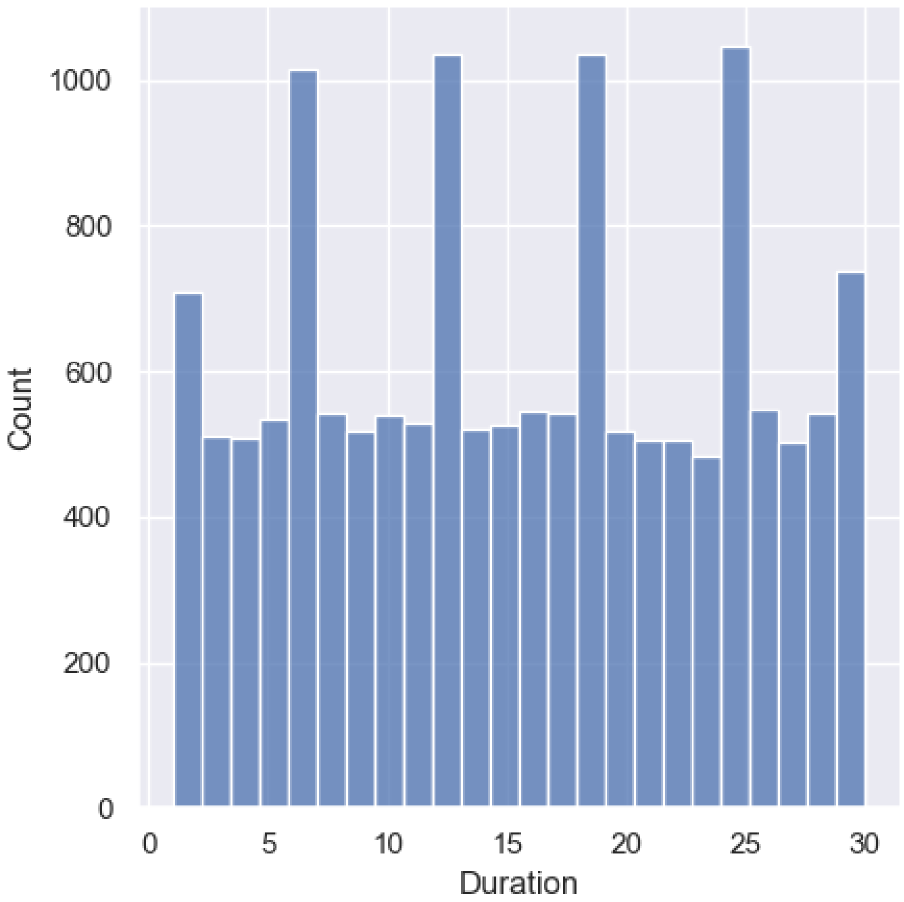
```
<Figure size 500x500 with 0 Axes>
```



[45]:
```
plt.figure(figsize=(5,5))
sns.displot(calories_data['Duration'])
```

[45]:
```
<seaborn.axisgrid.FacetGrid at 0x1c72dcf37d0>
```

```
<Figure size 500x500 with 0 Axes>
```



```
[47]: calories_data['Duration'].value_counts().reset_index()
```

```
[47]:     Duration  count
      0       26.0    548
      1       16.0    546
      2       17.0    543
      3       28.0    541
      4        8.0    541
      5       10.0    539
      6        5.0    533
      7        6.0    533
      8       11.0    528
      9       15.0    527
      10      25.0    526
      11      13.0    523
      12      14.0    522
      13      24.0    521
      14      19.0    521
```

```
15       20.0    518
16        9.0    517
17       18.0    514
18       12.0    512
19        3.0    511
20        4.0    509
21       21.0    506
22       22.0    505
23       27.0    502
24       23.0    485
25       29.0    483
26        7.0    482
27        2.0    479
28       30.0    255
29        1.0    230
```

[49]:

```
calories_data.columns
```

[49]:

```
Index(['User_ID', 'Gender', 'Age', 'Height', 'Weight', 'Duration',
       'Heart_Rate', 'Body_Temp', 'Calories'],
      dtype='object')
```

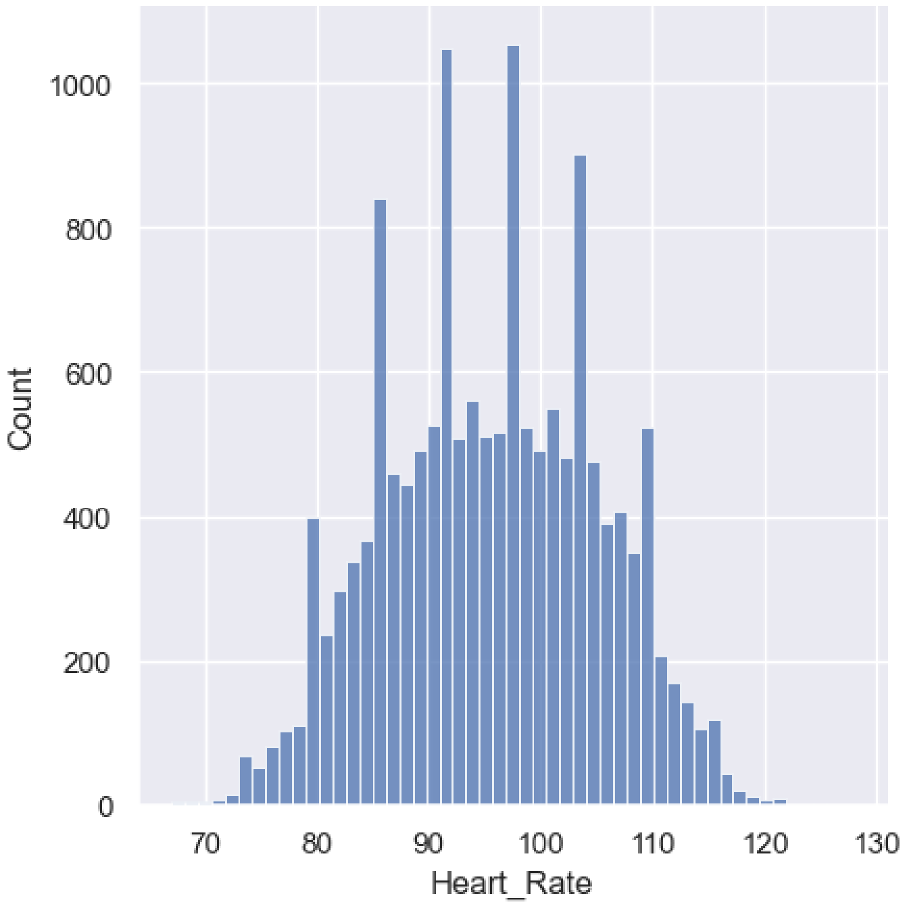
[50]:

```
plt.figure(figsize=(5,5))
sns.displot(calories_data['Heart_Rate'])
```

[50]:

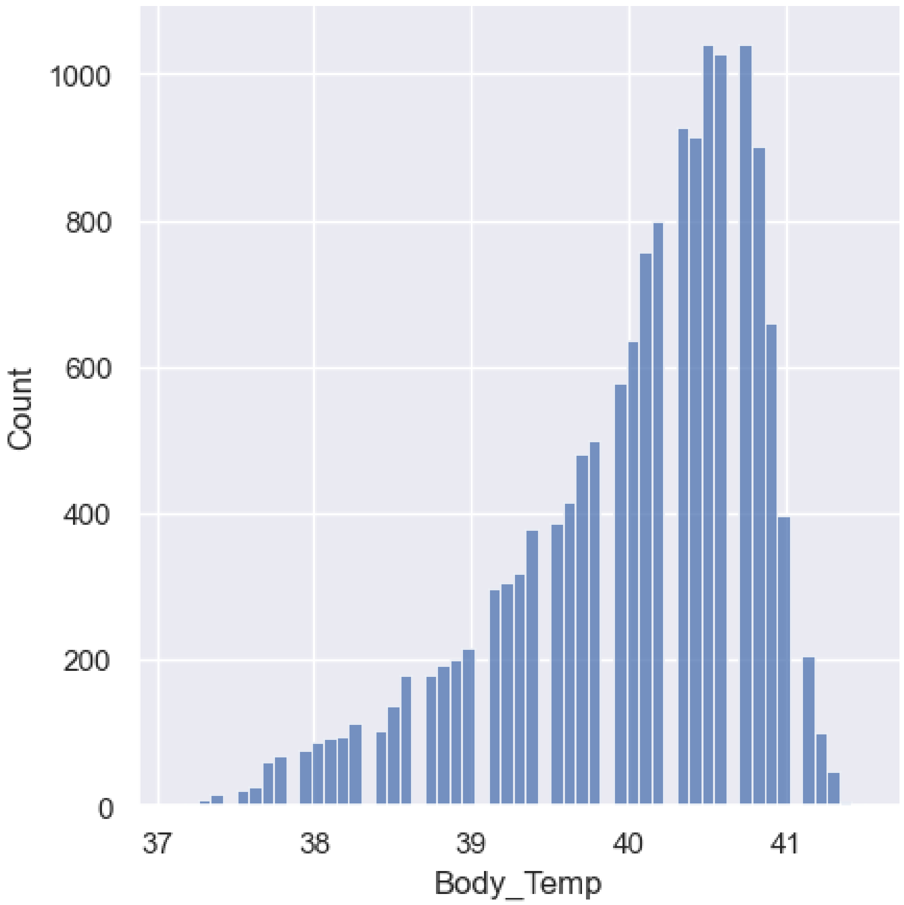
```
<seaborn.axisgrid.FacetGrid at 0x1c72cb5c5c0>
```

```
<Figure size 500x500 with 0 Axes>
```



```
[51]: plt.figure(figsize=(5,5))
      sns.displot(calories_data['Body_Temp'])
```

```
[51]: <seaborn.axisgrid.FacetGrid at 0x1c72e7a9f10>

      <Figure size 500x500 with 0 Axes>
```



[53]:

```
calories_data['Gender']=calories_data['Gender'].map({'male':0,'female':1})
```

[54]:

```
calories_data.head()
```

[54]:

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

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

[55]:
```
calories_data1=calories_data.copy()
```

[56]:
```
calories_data2=calories_data.copy()
```

[57]:
```
calories_data1.head()
```

[57]:
```
   User_ID  Gender  Age  Height  Weight  Duration  Heart_Rate  Body_Temp  \
0  14733363       0   68   190.0    94.0      29.0       105.0       40.8
1  14861698       1   20   166.0    60.0      14.0        94.0       40.3
2  11179863       0   69   179.0    79.0       5.0        88.0       38.7
3  16180408       1   34   179.0    71.0      13.0       100.0       40.5
4  17771927       1   27   154.0    58.0      10.0        81.0       39.8

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

[58]:
```
calories_data2.head()
```

[58]:
```
   User_ID  Gender  Age  Height  Weight  Duration  Heart_Rate  Body_Temp  \
0  14733363       0   68   190.0    94.0      29.0       105.0       40.8
1  14861698       1   20   166.0    60.0      14.0        94.0       40.3
2  11179863       0   69   179.0    79.0       5.0        88.0       38.7
3  16180408       1   34   179.0    71.0      13.0       100.0       40.5
4  17771927       1   27   154.0    58.0      10.0        81.0       39.8

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

[59]:
```
X=calories_data.drop(['User_ID','Calories'],axis=1)
```

[60]:
```
Y=calories_data['Calories']
```

[61]:
```
X.head()
```

[61]:
```
   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
```

[62]:
```
Y.head()
```

[62]:
```
0    231.0
1     66.0
2     26.0
3     71.0
4     35.0
Name: Calories, dtype: float64
```

[63]:
```
train_x,test_x,train_y,test_y=train_test_split(X,Y,test_size=0.2,random_state=2)
```

[64]:
```
print(X.shape,train_x.shape,test_x.shape)
```

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

[67]:
```
regressor=XGBRegressor()
```

[68]:
```
regressor.fit(train_x,train_y)
```

[68]:
```
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=None, device=None, early_stopping_rounds=None,
             enable_categorical=False, eval_metric=None, feature_types=None,
             gamma=None, grow_policy=None, importance_type=None,
             interaction_constraints=None, learning_rate=None, max_bin=None,
             max_cat_threshold=None, max_cat_to_onehot=None,
             max_delta_step=None, max_depth=None, max_leaves=None,
             min_child_weight=None, missing=nan, monotone_constraints=None,
             multi_strategy=None, n_estimators=None, n_jobs=None,
             num_parallel_tree=None, random_state=None, …)
```

[69]:
```
train_x_prediction=regressor.predict(train_x)
```

[70]:
```
score=metrics.r2_score(train_y,train_x_prediction)
```

[71]:
```
print(score)
```

```
0.9995691477017405
```

[72]:
```
test_x_prediction=regressor.predict(test_x)
```

[73]:
```
score=metrics.r2_score(test_y,test_x_prediction)
```

[74]:
```
print(score)
```

```
0.998800624504713
```

[83]:
```
error_score=metrics.mean_absolute_error(test_y,test_x_prediction)
```

[84]:
```
print(error_score)
```

```
1.4833678883314132
```

[175]:
```
X_new=test_x.iloc[59]
```

[176]:
```
nparray=np.asarray(X_new)
```

[177]:
```
reshaped=nparray.reshape(1,-1)
```

[178]:
```
X_new_df=pd.DataFrame(reshaped,columns=train_x.columns)
```

[179]:
```
prediction=regressor.predict(X_new_df)
```

[180]:
```
print(prediction)
```

```
[79.10804]
```

[181]:
```
print(test_y.iloc[59])
```

```
81.0
```

[99]:
```
calories_data1.head()
```

[99]:
```
    User_ID  Gender  Age  Height  Weight  Duration  Heart_Rate  Body_Temp  \
0  14733363       0   68   190.0    94.0      29.0       105.0       40.8
1  14861698       1   20   166.0    60.0      14.0        94.0       40.3
2  11179863       0   69   179.0    79.0       5.0        88.0       38.7
3  16180408       1   34   179.0    71.0      13.0       100.0       40.5
4  17771927       1   27   154.0    58.0      10.0        81.0       39.8

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

[100]:
```
from sklearn.linear_model import LinearRegression
```

[184]:
```
M=calories_data1.drop(['User_ID','Calories','Height'],axis=1)
```

[185]:
```
M.head()
```

[185]:
```
   Gender  Age  Weight  Duration  Heart_Rate  Body_Temp
0       0   68    94.0      29.0       105.0       40.8
1       1   20    60.0      14.0        94.0       40.3
2       0   69    79.0       5.0        88.0       38.7
3       1   34    71.0      13.0       100.0       40.5
```

```
4       1   27    58.0      10.0        81.0       39.8
```

[186]:

```
N=calories_data1['Calories']
```

[187]:

```
N.head()
```

[187]:

```
0    231.0
1     66.0
2     26.0
3     71.0
4     35.0
Name: Calories, dtype: float64
```

[188]:

```
model=LinearRegression()
```

[189]:

```
train_m,test_m,train_n,test_n=train_test_split(M,N,test_size=0.2,random_state=2)
```

[190]:

```
model.fit(train_m,train_n)
```

[190]:

```
LinearRegression()
```

[191]:

```
testing_prediction=model.predict(test_m)
```

[192]:

```
score3=metrics.r2_score(test_n,testing_prediction)
```

[193]:

```
print(score3)
```

```
0.9668198408924703
```

[202]:

```
M_new=test_m.iloc[58]
nparray=np.asarray(M_new)
reshaped=nparray.reshape(1,-1)
M_new_df=pd.DataFrame(reshaped,columns=train_m.columns)
```

[203]:

```
prediction3=model.predict(M_new_df)
```

[204]:

```
print(prediction3)
```

```
[197.27624246]
```

[205]:

```
print(test_n.iloc[58])
```

```
211.0
```

[131]:

```
print(test_m.iloc[89])
```

```
Gender        0.0
Age          75.0
Weight       83.0
Duration     23.0
```

```
Heart_Rate    104.0
Body_Temp      40.8
Name: 14342, dtype: float64
```

```
[132]: calories_data2.head()
```

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

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

```
[133]: E=calories_data2.drop(['User_ID','Height','Body_Temp','Calories'],axis=1)
```

```
[134]: E.head()
```

```
[134]:    Gender  Age  Weight  Duration  Heart_Rate
       0       0   68    94.0      29.0       105.0
       1       1   20    60.0      14.0        94.0
       2       0   69    79.0       5.0        88.0
       3       1   34    71.0      13.0       100.0
       4       1   27    58.0      10.0        81.0
```

```
[135]: F=calories_data2['Calories']
```

```
[136]: F.head()
```

```
[136]: 0    231.0
       1     66.0
       2     26.0
       3     71.0
       4     35.0
       Name: Calories, dtype: float64
```

```
[137]: train_e,test_e,train_f,test_f=train_test_split(E,F,test_size=0.2,random_state=2)
```

```
[138]: model.fit(train_e,train_f)
```

```
[138]: LinearRegression()
```

[139]:

```
test_e_prediction=model.predict(test_e)
```

[140]:

```
score4=metrics.r2_score(test_f,test_e_prediction)
```

[141]:

```
print(score4)
```

```
0.958947413916142
```

[150]:

```
E_new=test_e.iloc[45]
nparray=np.asarray(E_new)
reshaped=nparray.reshape(1,-1)
E_new_df=pd.DataFrame(reshaped,columns=train_e.columns)
```

[151]:

```
prediction33=model.predict(E_new_df)
```

[152]:

```
print(prediction33)
```

```
[75.01715258]
```

[153]:

```
print(test_f.iloc[45])
```

```
67.0
```

[ ]:

```
```