Optimization of Model

First attempt included 2537 entries:

1. YearStart == 2017 & 2019
2. LocationDesc != "National"
3. Stratification1 != "Total"
4. Percent of adults aged 18 years and older who have an overweight classification
5. Percent of adults aged 18 years and older who have obesity
6. Percent of adults who achieve at least 150 minutes a week of moderate-intensity aerobic physical activity or 75 minutes a week of vigorous-intensity aerobic activity (or an equivalent combination)
7. Percent of adults who achieve at least 150 minutes a week of moderate-intensity aerobic physical activity or 75 minutes a week of vigorous-intensity aerobic physical activity and engage in muscle-strengthening activities on 2 or more days a week
8. Percent of adults who achieve at least 300 minutes a week of moderate-intensity aerobic physical activity or 150 minutes a week of vigorous-intensity aerobic activity (or an equivalent combination)
9. Percent of adults who engage in muscle-strengthening activities on 2 or more days a weekPercent of adults who engage in no leisure-time physical activity
10. Percent of adults who report consuming fruit less than one time daily
11. Percent of adults who report consuming vegetables less than one time daily

Model with an initial approach described below:

```
optimized_model = tf.keras.models.Sequential()
optimized_model.add(tf.keras.layers.Dense(units=64, activation="relu", input_dim=89))
optimized_model.add(tf.keras.layers.Dense(units=4, activation="relu"))
optimized_model.add(tf.keras.layers.Dense(units=1))
opt = tf.keras.optimizers.Adam(learning_rate=0.0001)
optimized_model.compile(loss="mean_absolute_error", optimizer=opt)
```

```
 Layer (type)            Output Shape          Param #
=================================================================
 dense_21 (Dense)        (None, 64)            5760

 dense_22 (Dense)        (None, 4)             260

 dense_23 (Dense)        (None, 1)             5

=================================================================
Total params: 6,025
Trainable params: 6,025
Non-trainable params: 0
```

optimized_model_fit = optimized_model.fit(X_train_scaled, y_train, epochs=175,
validation_data=(X_test_scaled, y_test))

Performance:
Performance was measured as mean absolute error and may not have told us much about the performa nce. loss: 2.6320

Model with a second approach described below:

optimized_model = tf.keras.models.Sequential()
optimized_model.add(tf.keras.layers.Dense(units=64, activation="relu", input_dim=89))
optimized_model.add(tf.keras.layers.Dense(units=32, activation="relu"))
optimized_model.add(tf.keras.layers.Dense(units=8, activation="relu"))
optimized_model.add(tf.keras.layers.Dense(units=1))
optimized_model.compile(optimizer='adam', loss='mean_squared_error',
metrics=[coeff_determination])

Layer (type)	Output Shape	Param #
dense_23 (Dense)	(None, 64)	5760
dense_24 (Dense)	(None, 32)	2080
dense_25 (Dense)	(None, 8)	264
dense_26 (Dense)	(None, 1)	9

Total params: 8,113
Trainable params: 8,113
Non-trainable params: 0

optimized_model_fit = optimized_model.fit(X_train_scaled, y_train, epochs=100,
validation_data=(X_test_scaled, y_test))

Performance:
Performance was measured by mean squared error loss: 15.1706 and coeff_determination: 0.6875

Final attempt included 6289 entries:

1. YearStart == 2011, 2013, 2015, 2017, 2019
2. LocationDesc != "National"
3. Stratification1 != "Total"
4. Percent of adults aged 18 years and older who have an overweight classification
5. Percent of adults aged 18 years and older who have obesity
6. Percent of adults who achieve at least 150 minutes a week of moderate-intensity aerobic physical activity or 75 minutes a week of vigorous-intensity aerobic activity (or an equivalent combination)

7. Percent of adults who achieve at least 150 minutes a week of moderate-intensity aerobic physical activity or 75 minutes a week of vigorous-intensity aerobic physical activity and engage in muscle-strengthening activities on 2 or more days a week
8. Percent of adults who achieve at least 300 minutes a week of moderate-intensity aerobic physical activity or 150 minutes a week of vigorous-intensity aerobic activity (or an equivalent combination)
9. Percent of adults who engage in muscle-strengthening activities on 2 or more days a weekPercent of adults who engage in no leisure-time physical activity
10. ~~Percent of adults who report consuming fruit less than one time daily~~
11. ~~Percent of adults who report consuming vegetables less than one time daily~~

The model with a final approach described below:

```
optimized_model = tf.keras.models.Sequential()
optimized_model.add(tf.keras.layers.Dense(units=64, activation="leaky_relu", input_dim=87))
optimized_model.add(tf.keras.layers.Dense(units=32, activation="leaky_relu"))
optimized_model.add(tf.keras.layers.Dense(units=16, activation="leaky_relu"))
optimized_model.add(tf.keras.layers.Dense(units=8, activation="relu"))
optimized_model.add(tf.keras.layers.Dense(units=4, activation="relu"))
optimized_model.add(tf.keras.layers.Dense(units=1))
opt = tf.keras.optimizers.Adam(learning_rate=0.00001)
optimized_model.compile(loss="mse", optimizer=opt, metrics=["mean_absolute_percentage_error"])
```

Layer (type)	Output Shape	Param #
dense_12 (Dense)	(None, 64)	5632
dense_13 (Dense)	(None, 32)	2080
dense_14 (Dense)	(None, 16)	528
dense_15 (Dense)	(None, 8)	136
dense_16 (Dense)	(None, 4)	36
dense_17 (Dense)	(None, 1)	5

```
Total params: 8417 (32.88 KB)
Trainable params: 8417 (32.88 KB)
Non-trainable params: 0 (0.00 Byte)

optimized_model_fit = optimized_model.fit(X_train_scaled, y_train, epochs=1000,
validation_data=(X_test_scaled, y_test))
```

Performance:
Performance was measured by mean squared error loss: 9.0807 - mean_absolute_percentage_error: 8.2015, R-Squared Value = 0.8324604034423828