MACHINE LEARNING
ASSIGNMENT - 1 REPORT

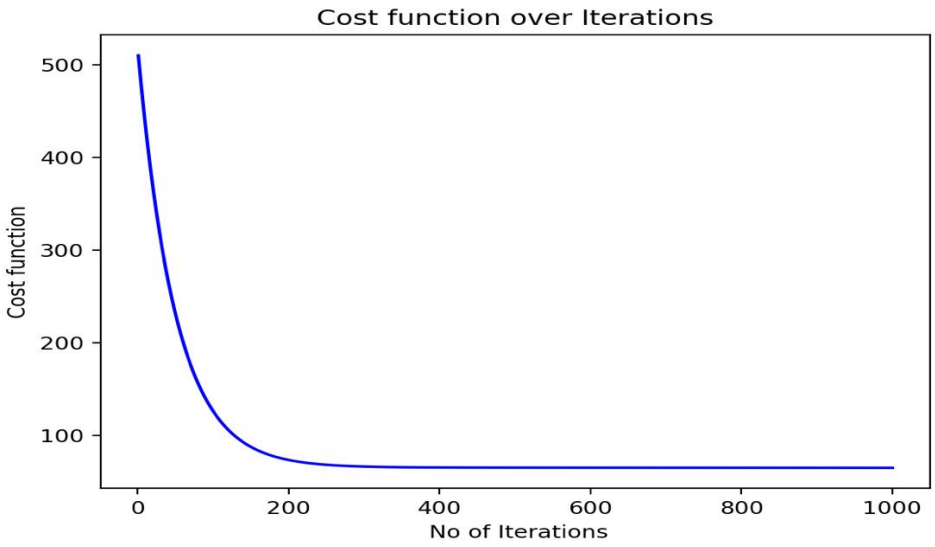
1. Linear Regression with One Variables:

   a. In this report, based on the "PPE" feature from the Parkinson's disease dataset, we have developed a simple linear regression model to predict **motor_UPDRS**. Firstly, the code loads the input dataset and divides it into training and test sets. 80% of data has been used to train the model and remaining 20% as test data. Initially theta0 and theta1 has been initialized to 0 and 1 respectively (random values), learning rate to .01 and number of iterations is 1000. Theta0 and Theta1 parameters are iteratively updated with the objective of minimizing the mean square error through the gradient descent process.
   Model Results:
   theta0: 21.286804186609775
   theta1: 2.3753875520655874
   Mean squared error: 64.97323806598753.

```
  import pandas as pandas
Intercept (theta0): 21.286804186609775
Coefficient (theta1): 2.3753875520655874
Mean Squared Error: 64.97323806598753
```

Below graph shows convergence of cost function over number of iterations

it shows the value of theta0 and theta1 become almost constant after around 300 iterations



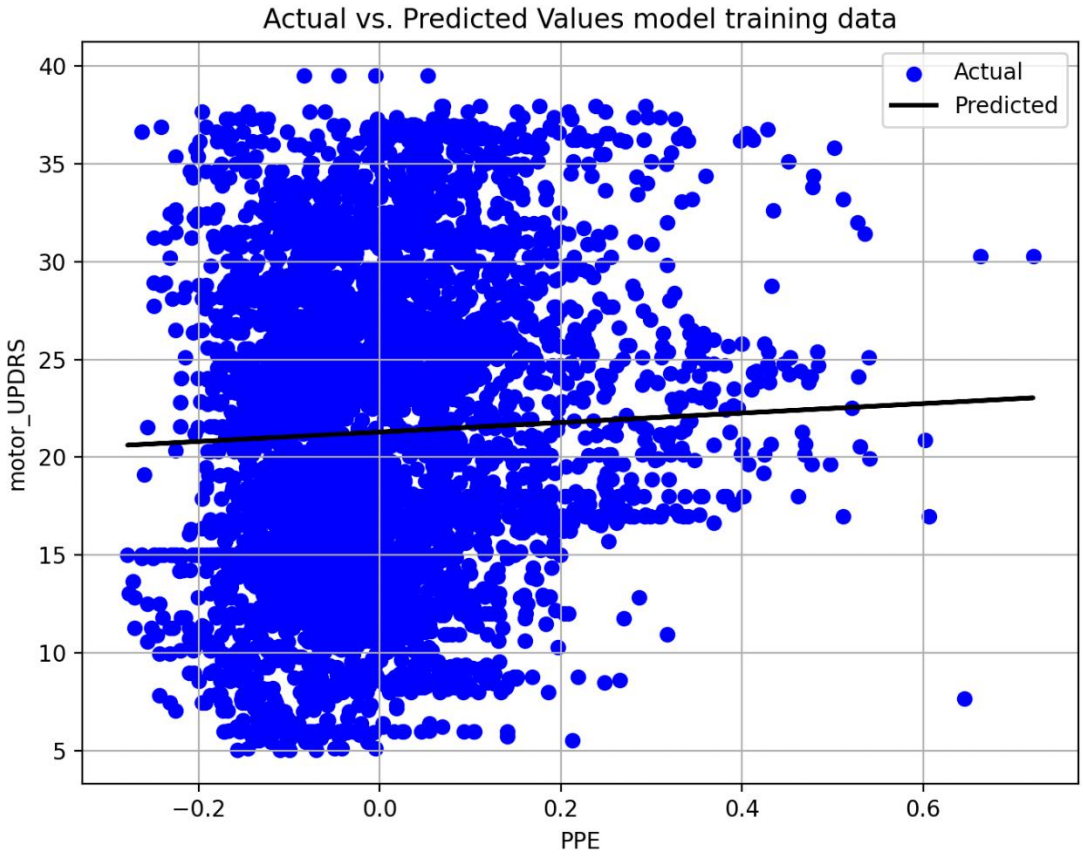
b. Performance of model is being evaluated based on comparing mean squared error, r squared and adjusted r squared between training data and test data.

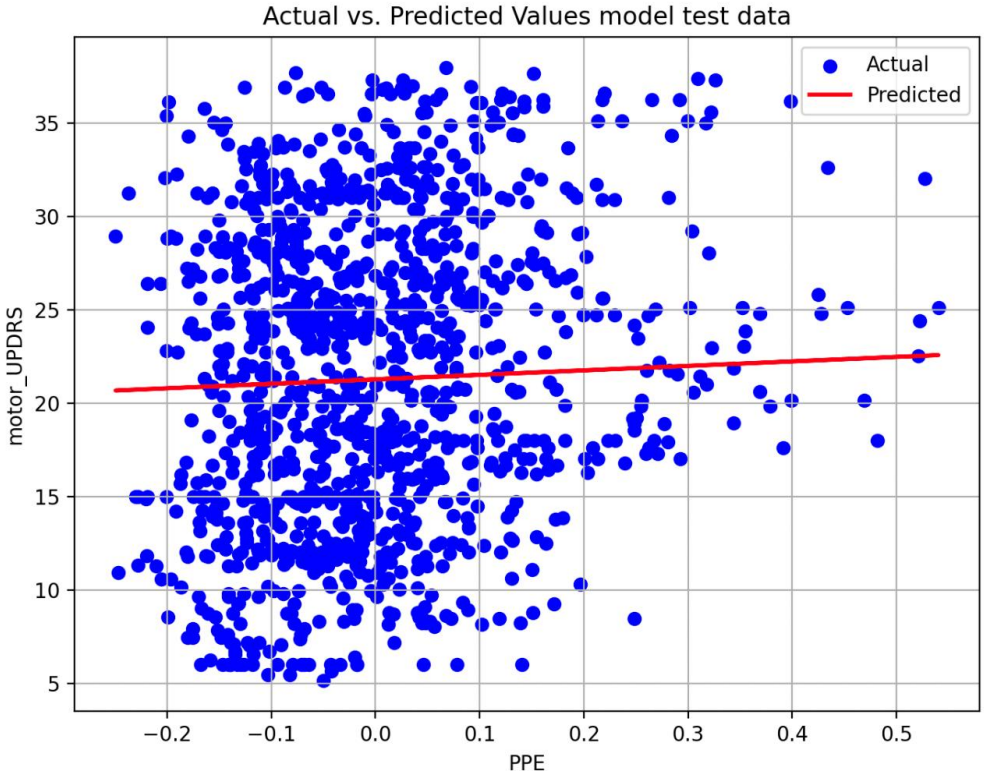
```
  import pandas as pandas
Mean Squared Error (MSE) train data: 65.65678029323442
R-squared (R²): train data 0.011313274065340218
Adjusted R-squared (Adjusted R²): train data 0.011102825634958235
Mean Squared Error (MSE): test data 64.08006563954756
R-squared (R²): test data 0.009809320831223989
Adjusted R-squared (Adjusted R²): test data 0.008965168504566967
```

R squared and adjusted r2 shows limited predictive power of model as these are low in training and test data

The difference in the Mean squared error of the model with the test data and the train data is very small indicating that the model is performing well. And same is the case with R-squared error and adjusted R-squared error.

A scatter plot shows the performance of the model with training and test data, and shows linear regression line for the same.





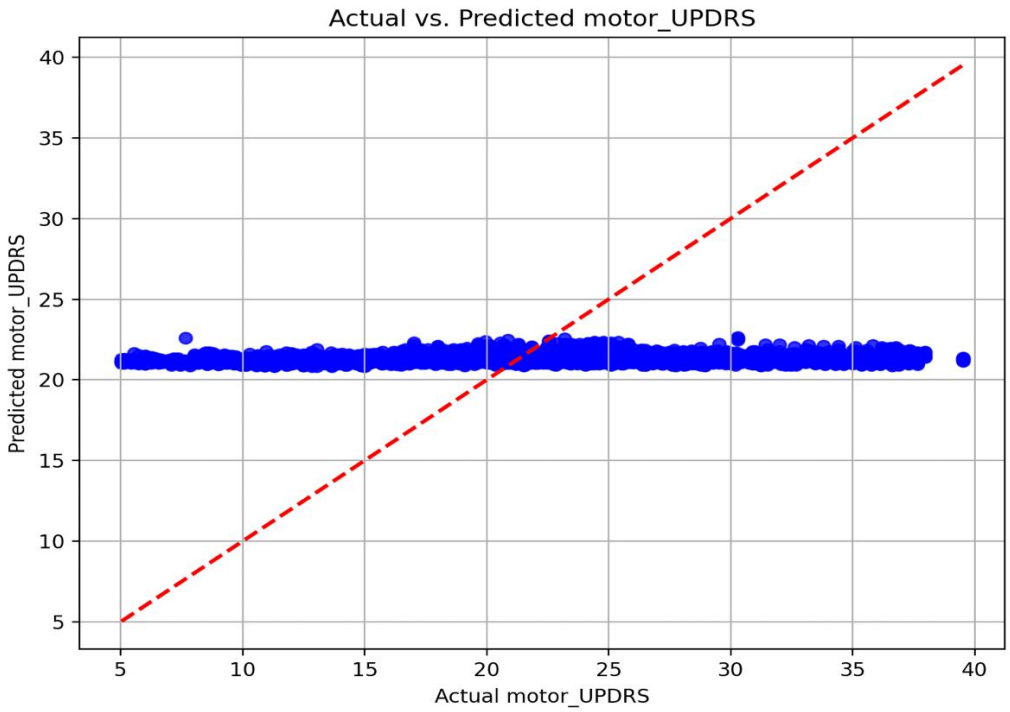
2. Linear Regression with Two Variables:

Below is the observation by adding NHR variable in linear regression model

```
  import pandas as pandas
Mean Squared Error (MSE):  65.55198317143496
R-squared (R²):  0.007900030552276838
Adjusted R-squared (Adjusted R²):  0.007393081155522729
```

It has been observed that by adding NHR feature, model performance is reasonably consistent but have area of improvement. Model with two variables has a slightly increased mean squared error, positive value for r squared and adjusted r squared which signifies model is overfitting for training data. Still, more research and perhaps adding more features could result in even more advanced models for predicting **motor_UPDRS.**

Below graph shows model performance with respect to actual and predicted value.



3. Stepwise Linear Regression

   a. After 5 iterations BIC method described age, PPE, DFA, HNR and Shimmer as the five most important features to improve the model's performance.

      By adding features gradually, we found the most important features for estimating motor_UPDRS while considering how complex the model is. This way, the model has higher accuracy and simpler interpretation because it uses only the relevant features that the Bayesian Information Criterion helps to select.

      Below screenshot shows chosen feature, with their BIC value.

```
  import pandas as pandas
Iteration number 1: Selected Feature: age, BIC value: 24180.818558245683
Iteration number 2: Selected Feature: PPE, BIC value: 24080.621221443642
Iteration number 3: Selected Feature: DFA, BIC value: 23923.97429234774
Iteration number 4: Selected Feature: HNR, BIC value: 23919.300116170765
Iteration number 5: Selected Feature: Shimmer:APQ5, BIC value: 23880.998805402716
Mean Squared Error (MSE): 59.71520021008692
R-squared: 0.09623713215425744
Adjusted R-squared: 0.0953130392423498

Process finished with exit code 0
```

b. The code removes features from the current set of features based on which features increase the MSE the least. It checks the impact of removing a feature with each iteration and updates the list of selected features as needed. The output shows the feature that was removed along with the corresponding change in MSE, R-Squared and Adjusted R Squared for each step. The features that got removed without increasing the MSE is “Shimmer:APQ11”, “HNR”, “RPDE”, “DFA” and “PPE”. This iterative method enhances the model’s simplicity and comprehensibility by finding the variables that are not very helpful for predicting motor_UPDRS, while keeping the effect on prediction accuracy low.

```
  import pandas as pd
Removed Feature: Shimmer:APQ11
  Increase in MSE: 0
  R-squared: 0.0
  Adjusted R-squared: -0.00017027073046138597

Removed Feature: HNR
  Increase in MSE: 66.07396954958689
  R-squared: 0.0
  Adjusted R-squared: -0.00017027073046138597

Removed Feature: RPDE
  Increase in MSE: 0.0
  R-squared: 0.0
  Adjusted R-squared: -0.00017027073046138597

Removed Feature: DFA
  Increase in MSE: 66.07396954958689
  R-squared: 0.0
  Adjusted R-squared: -0.00017027073046138597

Removed Feature: PPE
  Increase in MSE: 0.0
  R-squared: 0.0
  Adjusted R-squared: -0.00017027073046138597


Process finished with exit code 0
```

c. I have compared the final models of the forward and backward stepwise regressions and found the following differences:

Forward Stepwise Regression Model:

- It selects features in a forward way, starting from an empty set and adding one feature at a time.
- It stops when it finds the features that minimize the BIC.
- The final features selected are: age, PPE, DFA, HNR and Shimmer.
- The model performance is:
    - MSE: 59.71520021008692
    - R-squared: 0.09623713215425744
    - Adjusted R-squared: 0.0953130392423498

Backward Stepwise Regression Model:

- It selects features in a backward way, starting from the full set and removing one feature at a time.
- It removes the feature that causes the smallest increase in MSE.

- The final features selected are: Shimmer:APQ11, HNR, RPDE, DFA and PPE.
- The model performance is:
    - MSE: The output shows the MSE values and how they increase with each feature removal.
    - R-squared: The output shows the R-squared values for each step of feature removal.
- The backward stepwise regression model seems to be better than the forward stepwise regression model in terms of model fit and feature selection, based on both BIC and R-squared metrics.
- However, both models are not very good at explaining the variation in the target variable ‘motor_UPDRS' and more model improvement may be needed to increase predictive accuracy.

d. We need to look at the evaluation metrics of both models to see how they differ in performance. The model from Q.2 used the features based on their performance, while the model from Q.3(c) used the features selected by stepwise regression. Here is the comparison:

   Model from Q.2:

   - It added the "NHR" feature, which improved the model a lot.
   - The model performance was:
       - MSE: 65.55198317143496
       - R-squared: 0.007900030552276838
       - Adjusted R-squared: 0.007393081155522729

   Model from Q.3(c) (Backward Stepwise Regression):

   - It used the features: Shimmer:APQ11, HNR, RPDE, DFA and PPE which were chosen by backward stepwise regression.
   - The model performance changed for each step of feature removal. The final model had an adjusted R-squared value of -0.00017027073046138597.
   - The model from Q.2 was better than the model from Q.3(c) in terms of performance. It had a lower MSE and higher R-squared and adjusted R-squared values, which means it was more accurate in predicting motor_UPDRS.

   Conclusion:

   The model from Q.2, which included the "NHR" feature, performed better than the model from Q.3(c), which used backward stepwise regression. It had lower prediction errors and better alignment with the actual data.

4. Regularization and Feature Scaling
    a. For the best performing model in Q.3 (Model from Q.3(d)) (i.e. Q2)

Metrices without regularization:

Mean Squared Error (MSE): 65.55198317143496
R-squared (R^2): 0.007900030552276838
Adjusted R-squared (Adjusted R^2): 0.007393081155522729

Metrices with regularization:

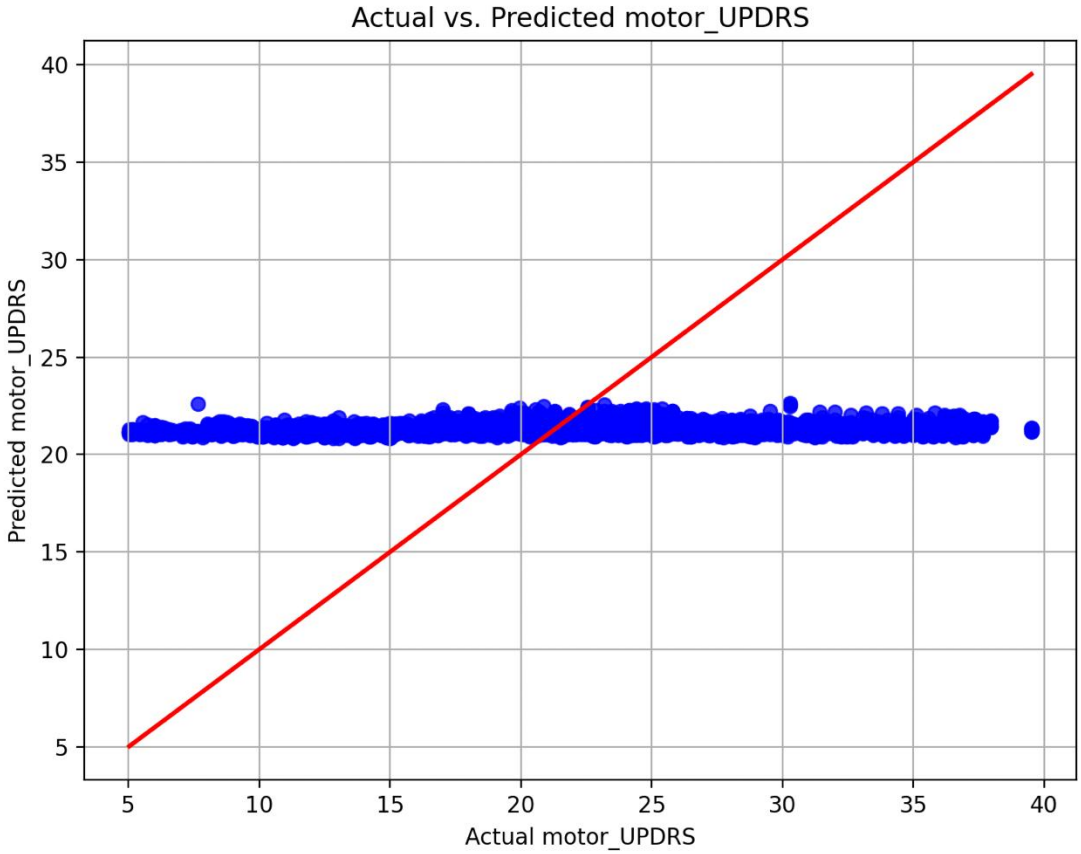
Mean Squared Error (MSE): 65.5520221217681
R-squared (R^2): 0.007899441056391843
Adjusted R-squared (Adjusted R^2): 0.007392491358413467

Observations:

1. Regularization does not significantly impact performance, However, when working with more complicated datasets or when we have many features, it's frequently helpful in preventing overfitting. It keeps the model more steady and broadly applicable.
2. Almost similar Mean Squared Error (MSE), R-squared (R^2) and Adjusted R-squared (Adjusted R^2).
3. Although the improvement is little, it shows that regularization penalized the model's complexity while enhancing the model's ability to explain the variance in the target variable.
4. Choice of lambda (regularization strength) can influence the level of improvement.

Graph with regularization.

Points are relatively close with regularization and are near to straight line having scope of 1 which depicts that model’s predictions show less deviation and are more in line with the actual values whereas these points were scattered more without regularization which signifies motor_UPDRS differ significantly from the actual sizes without regularization.



b. Below stats give performance of model with and without feature scaling using metrics like Mean Squared Error (MSE), R-squared (R2), and Adjusted R-squared (Adjusted R2).

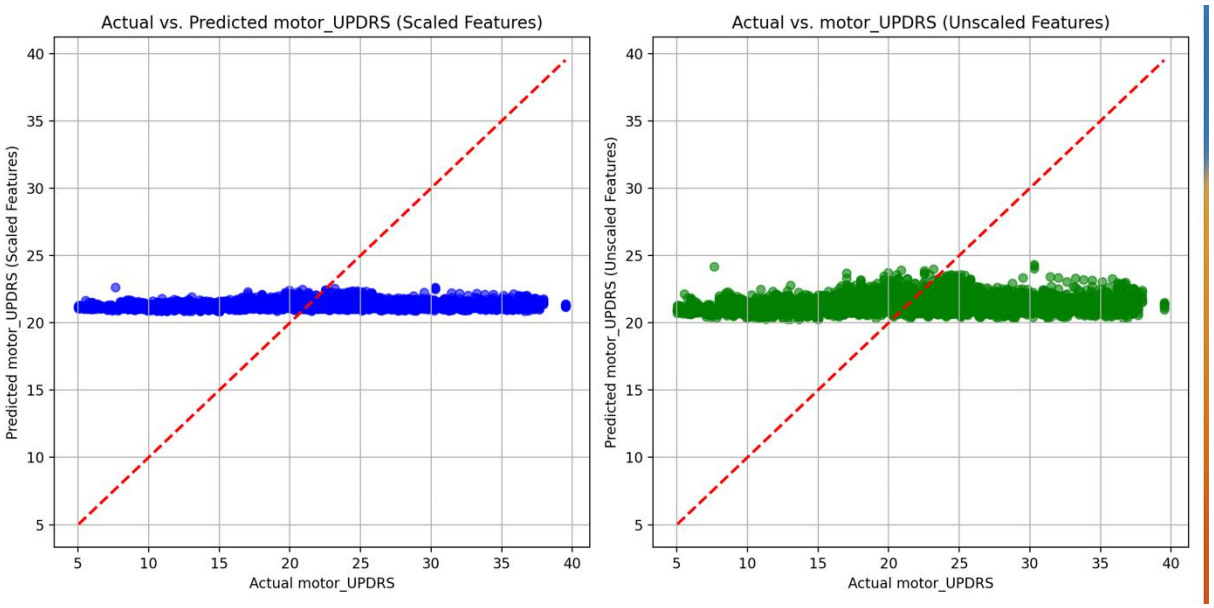
Scaled feature:

```
 import pandas as pandas
scaled Mean Squared Error (MSE):  65.55198317143496
scaled R-squared (R²):  0.007900030552276838
scaled Adjusted R-squared (Adjusted R²):  0.007393081155522729
```

Unscaled feature:

```
unscaled Mean Squared Error (MSE):  65.00691758741418
unscaled R-squared (R²):  0.01614935457104505
unscaled Adjusted R-squared (Adjusted R²):  0.007393081155522729
```

Graph:



Observations:

1. **Performance metrics**

Unscaled feature:

- scaled Mean Squared Error (MSE): 65.55198317143496
- scaled R-squared (R^2): 0.007900030552276838
- scaled Adjusted R-squared (Adjusted R^2): 0.007393081155522729

Scaled feature:

- Unscaled Mean Squared Error (MSE): 65.00691758741418
- Unscaled R-squared (R^2): 0.01614935457104505
- Unscaled Adjusted R-squared (Adjusted R^2): 0.007393081155522729

Inference from performance metrices:

- Model with scaled feature has very minor increase in Mean Squared Error which means very little increase in prediction accuracy of model.
- Minimal change in R squared and adjusted R- squared signifies model having limited explanatory power.

2. **Scatter Plot**
   - Left sub plot shows the relationship between predicted value and actual value with feature scaling.
   - Right sub plot shows the relationship between predicted value and actual value with feature scaling.

Inference from scatter plot:

It shows predictions from scaled model are closed to actual target variable as compared to unscaled model which shows less accurate prediction.

Overall, feature scaling marginally enhances model performance, leading to somewhat lower error and better agreement with actual data.