Machine Learning Project 1: Overfitting, Underfitting, and Hyperparameters

1. Introduction

This project explores regression models, specifically focusing on overfitting, underfitting, and hyperparameter tuning. The primary task is to build and compare different linear models to predict diabetes progression using the well-known **Diabetes dataset**. We first implement a linear regression model using an analytical solution (without regularization), followed by a Lasso regression model, which adds L1 regularization. Finally, we experiment with different levels of regularization and analyze the performance of the models in terms of prediction accuracy, overfitting, and underfitting.

2. Data

We use the **Diabetes dataset** from scikit-learn, which contains 442 samples and 10 features representing different baseline variables like BMI, blood pressure, etc., for predicting the progression of diabetes one year after baseline.

The dataset is split into training and testing sets in an 80-20 ratio using `train_test_split()` to allow for model evaluation on unseen data:

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

3. Linear Regression Model with a Subset of Features

In this section, we experiment with Linear Regression using a subset of features from the Diabetes dataset. This approach aims to assess how excluding certain features impacts the model's performance. By training the model on a limited set of features, we can observe the change in Mean Squared Error (MSE) and R^2 score, providing insights into the influence of individual features on the model's accuracy and predictive power.

Methodology:

1. **Feature Selection:** We selected a subset of 8 features from the original 10 in the Diabetes dataset. The selected features were `[8, 1, 5, 0, 7, 2, 9, 4]` (using zero-based indexing). This subset was chosen at random to test the model's performance with fewer features.

2. **Model Training:** We trained the Linear Regression model using Scikit-Learn's `LinearRegression` class on this subset of features, allowing us to examine how feature reduction affects model complexity and predictive capability.
3. **Model Evaluation:** The model's performance was evaluated using Mean Squared Error (MSE) and R^2 score on the test data. These metrics reflect the accuracy of the model's predictions compared to the baseline model that used all features.

   Results:

1. **Selected Features**: The model was trained on 8 features: `[8, 1, 5, 0, 7, 2, 9, 4]`.
2. **Mean Squared Error (MSE)**: The MSE for this model was calculated at 2901.77, similar to the baseline model with all features. This indicates that reducing the features had a minimal effect on the prediction error, suggesting that the excluded features contributed little to the model's accuracy.
3. **R^2 Score**: The R^2 score was 0.452, which is nearly the same as the full-feature model's R^2 score of 0.453. This score indicates that about 45.2% of the variance in diabetes progression is explained by this subset-based model.
   a. **Coefficients**: The coefficients for each selected feature were as follows:

      ```
      [ 712.63, -185.85, 325.46, 94.81, 109.38, 640.54, 107.61,
      -625.74]
      ```

4. Each coefficient represents the weight assigned to the respective feature. A higher absolute value of a coefficient indicates a stronger impact of that feature on the target variable.

Using a subset of features yielded results that closely resembled the full-feature model, as evidenced by the negligible change in MSE and R^2 score. This finding suggests that the excluded features may not be critical to the model's predictive power, allowing us to maintain similar accuracy with a simpler model. Reducing the number of features can improve interpretability and may prevent overfitting, especially when some features are redundant.

This subset approach highlights the importance of identifying key predictors. While training models with all available features is standard, this experiment demonstrates that feature selection can lead to simpler models with nearly identical performance, offering benefits in interpretability and generalization.

4. Linear Regression Model (Analytical Solution) with optimizing MSE

In the first step, we implemented a basic linear regression model using the **normal equation** to minimize the Mean Squared Error (MSE). The normal equation is represented as:

$$w = (X^T X)^{-1} X^T y$$

This equation allows us to compute the weights directly, avoiding iterative processes like gradient descent.

Results:

- **Mean Squared Error (MSE)**: 2900.19
- **R² Score**: 0.453

The R² score of 0.453 means that about **45.3% of the variance** in the progression of diabetes is explained by this linear model. The MSE of 2900.19 reflects the average squared difference between predicted and actual values, which is a direct measure of the prediction error.

The weights (coefficients) for the features were as follows:

```
[ -132.57, -239.80, 519.07, 324.39, -792.18, 476.75, 98.97, 177.04,
751.28, 67.62]
```

Each weight represents the contribution of the corresponding feature to the predicted diabetes progression. However, the model uses all features, which could potentially lead to overfitting, especially if some features are irrelevant.

5. Lasso Regression Model

Next, we implemented a Lasso regression model, which includes **L1 regularization** to control the model's complexity by shrinking the coefficients of some features to zero. This reduces the number of non-zero coefficients, making the model more interpretable.

We trained the Lasso model using an **alpha value of 0.5**, which balances the trade-off between fitting the data and minimizing the number of non-zero weights.

Results:

- **Mean Squared Error (MSE)**: 2945.15
- **R² Score**: 0.444
- **Lasso Weights**:

```
[  0.          -0.         513.5874669  165.4975634   -0.
  -0.         -72.68339144   0.         354.65010136   0.        ]
```

Unlike the linear regression model, the Lasso model assigns zero coefficients to several features, reducing the complexity of the model. Specifically, only **three features** (out of ten) are retained, with non-zero weights. The R^2 score is slightly lower at 0.444, indicating that the Lasso model explains about **44.4%** of the variance in diabetes progression, slightly less than the linear regression model. However, the model is simpler and easier to interpret because it focuses on fewer features.

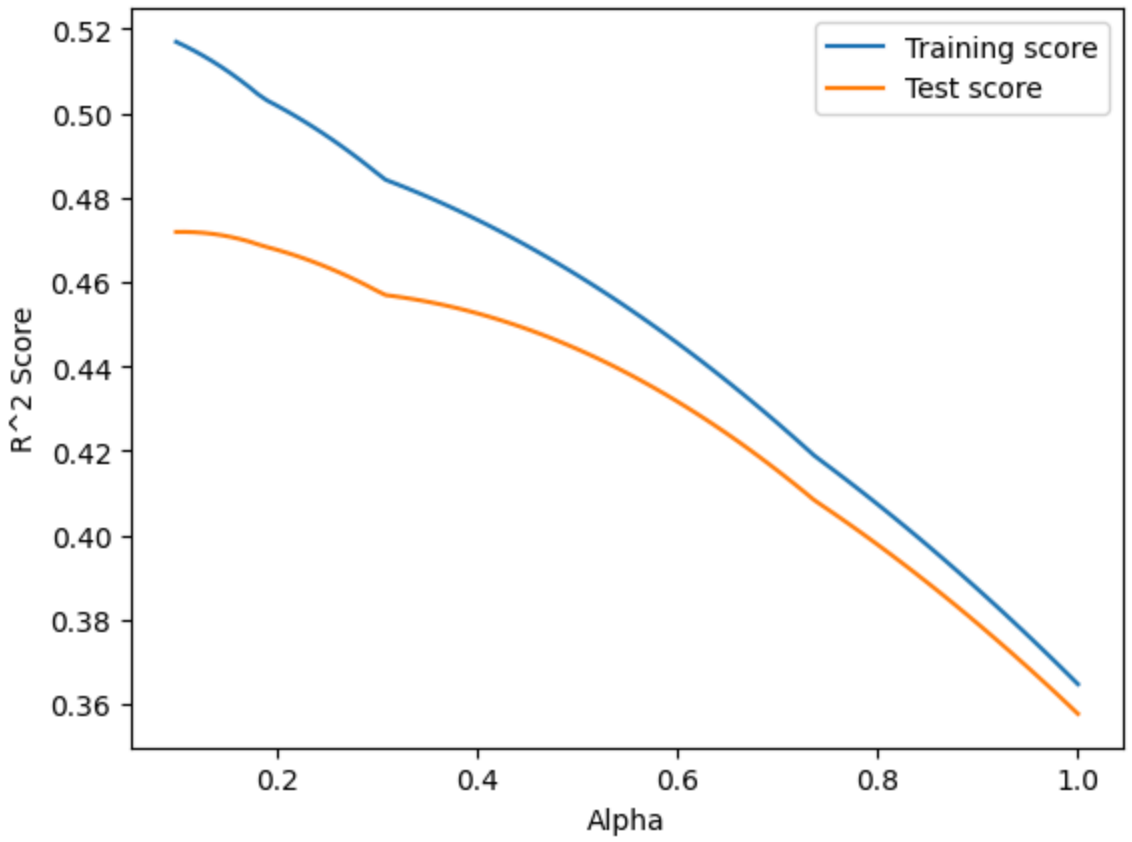
Comparison with Linear Regression:

- **Linear Regression**: Uses all features, resulting in a higher R^2 score (0.453) but a more complex model that might overfit the training data.
- **Lasso Regression**: Reduces the number of non-zero weights, leading to a simpler and more interpretable model but with a slightly lower R^2 score (0.444). This suggests Lasso may perform better in terms of **generalization**, especially in cases where many features are irrelevant.

6. Model Complexity and Hyperparameter Tuning

In this part, we explored how varying the regularization parameter alpha affects the model's performance. We trained **100 Lasso models** with alpha values ranging from **0.1 to 1.0** and plotted their **training** and **testing** R^2 scores.

The plot of training and test scores as a function of alpha is shown below:



Results:

The plot reveals a typical pattern:

- **For lower alpha values (around 0.1)**, the model performs well on the training data but shows signs of **overfitting**, as the gap between the training and testing scores is larger.
- **As alpha increases**, both the training and testing scores decrease. This is because the regularization becomes stronger, shrinking the coefficients, which leads to underfitting. The training score drops because the model is less flexible and fails to capture the data's complexity, while the test score also declines as the model becomes too simple.

This behavior is consistent with the theory that increasing regularization can prevent overfitting by reducing model complexity, but excessive regularization can lead to underfitting, where the model is too simple to capture meaningful patterns in the data.

7. Final Model Selection

In selecting the final model, we considered three approaches: the unoptimized Linear Regression model using a subset of features, the optimized Linear Regression model using all features, and Lasso Regression with tuned regularization.

- **Linear Regression with a Subset of Features**: This model used 8 out of the 10 available features, with results indicating minimal performance difference from the full-feature model. Specifically, it achieved a Mean Squared Error (MSE) of 2901.77 and an R^2 score of 0.452, explaining about 45.2% of the variance in diabetes progression. The similar R^2 score suggests that the excluded features were likely not essential, as this simpler model performed comparably to the full model. This approach has the advantage of reducing complexity, which can help interpretability without significant loss in accuracy.
- **Optimized Linear Regression with All Features**: Using all features with the normal equation, this model minimized Mean Squared Error (MSE) effectively, yielding an MSE of 2900.19 and an R^2 score of 0.453. This score indicates that around 45.3% of the variance in diabetes progression is explained by the model, showing only a slight improvement over the subset-feature model. However, including all features might increase the risk of overfitting, particularly if some features are irrelevant to the prediction task.
- **Lasso Regression with Regularization**: Through tuning the regularization parameter α\alphaα, we found that an α\alphaα value of 0.2 yielded the best balance between training and testing performance, achieving an R^2 score of 0.46 on the test data. This model demonstrated slight improvement in generalizability by using fewer features (only retaining those with non-zero coefficients), making it more interpretable and less prone to overfitting compared to the non-regularized Linear Regression models.

Final Model Selection and Recommendation:

The **Lasso Regression model with $\alpha=0.2$\alpha = 0.2α=0.2** is recommended as the final model due to its balance between simplicity, interpretability, and generalizability. While it achieves a similar R^2 score to the other models, it uses fewer features, making it less likely to overfit and more resilient to noise in the data. This model captures essential patterns without overcomplicating the structure, providing an optimal solution for predicting diabetes progression in a way that maintains accuracy and reduces unnecessary complexity.

8. Conclusion

In this project, we implemented and compared a basic linear regression model and a Lasso regression model with varying levels of regularization. The linear regression model provided the best performance in terms of the R^2 score but used all features, which can lead to overfitting. In contrast, the Lasso model reduced the number of non-zero coefficients, making the model more interpretable but slightly less accurate.

Through hyperparameter tuning, we found that an **alpha value of 0.2** provided the best balance between training and test performance, with an R^2 score of **0.46** on the test data. This suggests that Lasso regression is a powerful tool for controlling model complexity, allowing us to build more generalizable models that avoid overfitting.