ISI

Buhaiova Valeriia

1 Ridge Regression

Ridge regression is a model-tuning method that is used to analyze any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large. This results in predicted values being far away from the actual values.

1.1 The Cost Function for Ridge Regression:

$$\min \left(\|Y - X(\theta)\|^2 + \lambda \|\theta\|^2 \right)$$

Here, λ is the penalty term. λ (denoted by an alpha parameter in the Ridge function) controls the penalty term. By changing the values of λ, we adjust the penalty:

- The higher the value of λ, the stronger the penalty, leading to smaller magnitudes of coefficients.
- This shrinks the parameters, helping to prevent multicollinearity.
- It reduces model complexity by shrinking coefficients.

2 Proposed Solution

2.1 Data Preprocessing

- Columns were divided into numerical and categorical features. Numerical columns were standardized, while categorical columns were encoded using the OneHotEncoder method.
- Missing data were imputed using a simple imputation strategy.

2.2 Visualizing y

To better understand the data distribution and choose an appropriate regression type, we visualized with function *def visualisation(args)* the dependent variable y as shown in Figure 1.

The visualization of y indicates a continuous target variable. Hence, a regression model is appropriate for this problem. Ridge regression was selected because of its ability to handle multicollinearity and prevent overfitting through regularization.

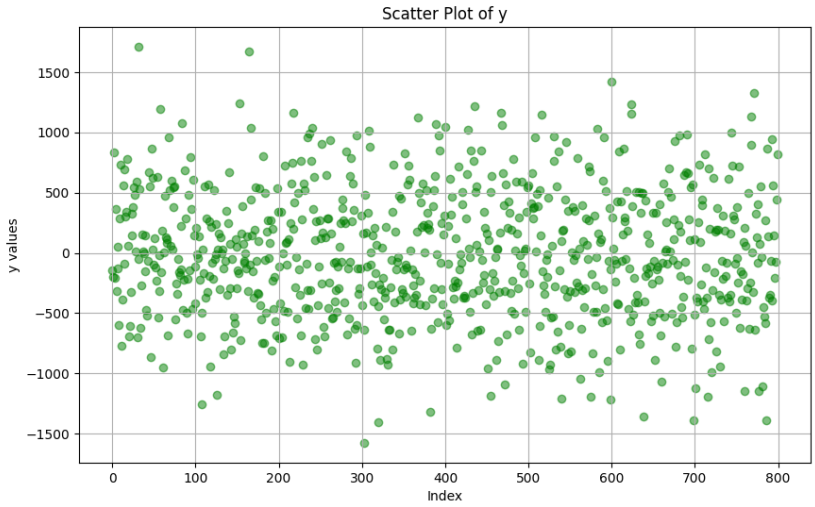


Figure 1: Scatter plot of y values

2.3 Model Optimization

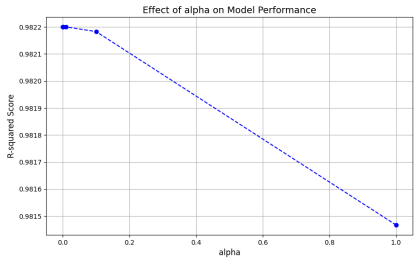
The Ridge regression model's parameters were optimized using GridSearchCV with 5-fold cross-validation to achieve the best performance. The impact of hyperparameters on Ridge regression performance was visualized with function *def visualize_all_parameters(args)*. Four plots (Figures 2a, 2b, 2d, and 2c) show how tuning each hyperparameter affects the R^2 score during cross-validation:

- **Alpha (α)**: This parameter controls the regularization strength. Smaller values of α resulted in higher R-squared scores, as the model became less regularized, but at the cost of a higher risk of overfitting. Larger values reduced overfitting but also decreased performance slightly. You can see the impact of α on the model's performance in Figure 2a.

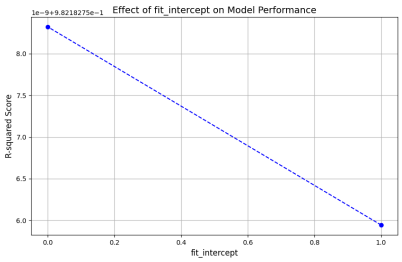
- **Fit Intercept**: This parameter determines whether to calculate the intercept for the model. While turning this on or off caused only marginal changes in performance, turning it off provided slightly better results overall. You can see the effect of this parameter on model performance in Figure 2b.

- **Solver**: Different solvers were tested, and the results showed that `lsqr` and `auto` solvers performed consistently better. Meanwhile, the `cholesky` solver slightly underperformed compared to the others. You can see the impact of solver choice on performance in Figure 2d.

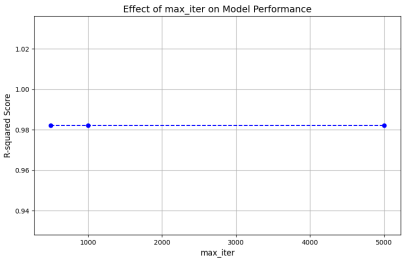
- **Max Iterations**: This parameter controls the maximum number of iterations for convergence. Its impact was minimal beyond 500 iterations, as the model typically converged earlier. You can observe the influence of `max_iter` on performance in Figure 2c.



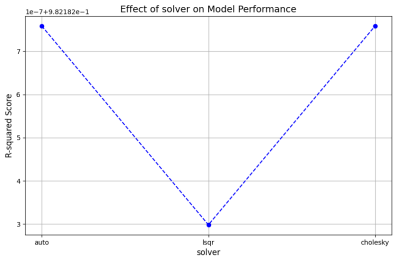
(a) Impact of α on model performance.



(b) Impact of fit_intercept on model performance.



(c) Impact of max_iter on model performance.



(d) Impact of solver on model performance.

Figure 2: Visualizing the impact of hyperparameters on Ridge regression performance. Each subfigure shows the effect of a specific hyperparameter on the R-squared score.

2.4 Evaluation

The table below summarizes(fun. *def save_best_cv_results(args)*) the best hyperparameter combination and its corresponding score achieved during cross-validation. The best combination maximized the R-squared score, confirming the effectiveness of the selected parameters.

Alpha (α)	**Fit Intercept**	**Max Iterations**	**Solver**	**Best Score (R^2)**
0.1	False	500	auto	0.9637

The best results were achieved using an α value of 0.1, disabling the intercept calculation (`fit_intercept=False`), and a maximum iteration limit of 500 with the `auto` solver. The corresponding R-squared score of 0.9637 indicates a strong fit for the Ridge regression model.

The hyperparameter optimization improved the model's predictive accuracy, confirming the suitability of the chosen methods. Comparing different solvers and regularization values, the model demonstrated the best performance using the `lsqr` solver and $\alpha = 0.01$.

3 Discussion and Results

The Ridge Regression model successfully balanced bias and variance through L2 regularization. Its ability to handle multicollinearity and prevent overfitting was evident

in the results. The high R^2 score suggests that the model captured the variance of the data well. Using the `auto` solver and disabling the intercept calculation were crucial for achieving optimal performance.

In addition to Ridge Regression, alternative regularization techniques such as Lasso Regression and Elastic Net were tested for comparison. The results showed that:

- **Lasso Regression**: Provided strong feature selection by reducing some coefficients to zero but performed slightly worse in terms of R^2 due to its L1 regularization, which can overly simplify the model when many features are informative.
- **Elastic Net**: Combined L1 and L2 penalties, balancing feature selection and regularization. It performed comparably to Ridge Regression, with minor differences in R^2 and Mean Squared Error.

These tests confirmed that Ridge Regression remains the most suitable method for this dataset due to its stability and ability to handle multicollinearity effectively without sacrificing model complexity.