Conducting and Presenting Multiple Linear Regression Analysis Using R

Jishnu Sai Matra

Fresenius University of Applied Science

Data Analysis for Decision-Making (WS 2025/26)

Prof. Dr. Stephan Huber

2025-12-10

Author Note

Correspondence concerning this article should be addressed to Jishnu Sai Matra, Email: matra.jishnu@stud.hs-fresenius.de

Abstract

This paper compares ordinary least squares (OLS) regression with Lasso regression using the Boston Housing dataset. While both models demonstrated similar predictive performance ($R^2 \approx 0.74$), they differ in their treatment of predictor variables. Lasso regression incorporates L1 regularization, which shrinks coefficient estimates and can perform automatic feature selection. The analysis illustrates practical implementation of both methods and discusses their respective advantages and limitations. Results suggest that while regularization had minimal impact on model fit for this dataset, Lasso regression remains valuable for feature selection and improving model interpretability.

Conducting and Presenting Multiple Linear Regression Analysis Using R

Word count: 1876

1 Introduction to Linear Regression and Lasso Regression

Linear regression is one of the most fundamental tools in statistics and data analysis. According to (Hubchev, n.d.), in its simplest form — simple linear regression — we model the relationship between a single continuous outcome variable (Y) and one predictor (X) using the equation:

$$Y = \alpha + \beta X + \varepsilon$$

where β represents the change in Y for a one-unit increase in X, α is the intercept, and ε is the random error term. The goal is to find the best-fitting line by minimizing the sum of squared residuals (ordinary least squares, OLS). In practice, linear regression is widely used in economics, social sciences, medicine, marketing, and almost any field where we want to understand or predict how one variable responds to changes in others — think predicting house prices from square footage, exam scores from study hours, or sales from advertising spend.

When we have more than one predictor, the model naturally extends to multiple linear regression:

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta X + \varepsilon$$

This allows us to control for several factors at once and isolate the effect of each predictor while holding the others constant (ceteris paribus). In RStudio, the most common way to fit these models is using the built-in lm() function from the base stats package. A typical command looks like lm(y ~ x1 + x2 + x3, data = mydata), which is incredibly straightforward and returns coefficients, p-values, R-squared, and other diagnostics almost instantly.

One can explore more about multiple linear regression from my colleague's report where she explains, step by step, how to conduct multiple linear regression using RStudio's built-in lm() model on the classic mtcars dataset. Her work is a great hands-on introduction if you're just getting started with lm().

However, when the number of predictors becomes large or when predictors are highly correlated (multicollinearity), ordinary least squares can produce unstable estimates and overfitted models. This is where regularized methods like Lasso regression come in.

Lasso (Least Absolute Shrinkage and Selection Operator), introduced by Tibshirani (1996), is a modification of linear regression that adds a penalty term to the loss function. Instead of just minimizing the sum of squared errors, Lasso minimizes:

$$\text{Sum of squared errors} + \lambda \times \Sigma|\beta|$$

This L1 penalty forces some coefficients to shrink all the way to exactly zero, automatically performing variable selection and giving us a simpler, more interpretable model. The tuning parameter λ (lambda) controls the strength of the penalty: $\lambda = 0$ gives the usual OLS solution, while a very large λ can shrink all coefficients to zero.

In this report, we will walk through how to perform multiple linear regression using both the standard lm() function and the glmnet package for Lasso regression (Tibshirani, 1996). We will fit both models on the same dataset, compare coefficients, R-squared values, selected variables, and cross-validated prediction error, so you can clearly see the differences in practice.

1.1 Key Terms Explained

To understand the mechanics of these models, it is helpful to define the key terms used in our analysis:

- **Coefficients**: These numbers tell us the expected change in the outcome for a one-unit change in the predictor, holding everything else constant. In Lasso, some may be exactly zero, meaning that predictor is completely dropped from the model.

- **R-squared (R^2)**: Shows what proportion of the variation in Y is explained by the model. In regular lm() output it's directly reported, but with Lasso we often look at adjusted R^2 or cross-validated R^2 because the penalty slightly reduces apparent fit on training data.

- **Penalty (L1)**: The core innovation of Lasso — the sum of the absolute values of the coefficients. Because of the sharp "corner" of the absolute value function, it tends to produce sparse solutions (many zeros).

- **Regularization**: The general idea of adding a penalty to prevent the model from fitting noise (overfitting). Lasso uses L1 regularization; Ridge uses L2 (sum of squares); Elastic Net combines both.

- **Lambda (λ)**: The knob we turn to control how much regularization we want. We usually try many λ values using cross-validation and pick the one that gives the best predictive performance (often denoted $\lambda_1$se or λmin in software).

1.2 Suitable Data for Multiple Linear Regression and Lasso Regression

Multiple linear regression (both standard `lm()` and Lasso) works well when the outcome is continuous or nearly continuous (e.g., house prices, fuel efficiency, exam scores, wages) and predictors are numeric or categorical (converted to dummy variables). Popular datasets include `mtcars`, Boston Housing, `diamonds`, and `penguins`. However, it cannot handle binary outcomes (use logistic regression), ordered categories (use ordinal regression), count data (use Poisson/negative binomial), survival times (use Cox models), or clustered data (use mixed-effects models). In short: if your dependent variable is a meaningful numeric quantity you can average and appears reasonably linear (possibly after transformation), you're good to go.

For this project, we use the Boston Housing dataset (Harrison & Rubinfeld, 1978; available via `MASS::Boston`). It contains housing values in 506 Boston suburbs from the 1970s, with 14 variables. The outcome `medv` is median home value (in $1000s)—a classic continuous target. The 13 predictors include per capita crime rate (`crim`), proportion of residential land zoned for large lots (`zn`), average rooms per dwelling (`rm`), proportion of non-retail business acres (`indus`), nitric oxide concentration (`nox`), pupil–teacher ratio (`ptratio`), and other neighborhood socio-economic and environmental characteristics.

This dataset is well-suited for demonstrating Lasso regression because several predictors are moderately to highly correlated
(e.g., `rad` and `tax`, `nox` and `indus`, `dis` and `age`), which makes ordinary least-squares coefficients unstable—exactly where Lasso shines through automatic variable selection and shrinkage. Note: This dataset has been deprecated in some R packages due to ethical concerns (one variable, `b`, relates to proportion of Black residents, raising bias issues). Despite this, it remains widely used educationally when the focus is purely methodological and the sensitive variable is handled carefully. For comparing `lm()` versus Lasso, it remains an ideal pedagogical choice.

1.3 Statistical Approach

Two regression models were fitted to the data. First, an ordinary least squares regression model was estimated using the `lm()` function in R, which provides a baseline without regularization. Second, a Lasso regression model was fitted using the `glmnet` package, which adds an L1 penalty term to the loss function. The optimal regularization parameter (λ) was selected through 10-fold cross-validation (Harris (2023)).

The Lasso regression objective function minimizes:

$$\text{RSS} + \lambda \sum_{j=1}^{p} |\beta_j|$$

where RSS is the residual sum of squares, λ is the regularization parameter, and β_j represents the regression coefficients. This penalty encourages sparsity by shrinking some coefficients to exactly zero. Model performance was evaluated using R^2, which measures the proportion of variance in the outcome variable explained by the predictors.

1.4 Variable Selection: Traditional Ways vs. Lasso

With many potential variables, you first try **theory** (pick based on research), **correlation checks**(drop highly correlated ones), or **stepwise regression** (add/remove variables one-by-one using p-values or AIC in R's `step()`). But these are manual, unstable, and struggle with tons of correlated predictors. **Lasso fixes this** by throwing *all* variables into the model and using an L1 penalty that automatically shrinks weak coefficients to exactly zero—leaving only the important ones. Tune lambda with cross-validation in `glmnet()`, and you're done. Simple, stable, and perfect for high-dimensional data.

1.5 Implementation

The analysis was conducted in R (version 4.5.1). First, the necessary packages were loaded and the data prepared:

```
# Load required packages
library(glmnet)
library(MASS)


# Load and prepare data
data(Boston)
X <- as.matrix(Boston[, -14])  # Predictor matrix
y <- Boston$medv               # Outcome variable
```

2 Linear Regression Model

The baseline OLS model was fitted using all available predictors:

```r
# Fit linear model
lm_model <- lm(medv ~ ., data = Boston)
lm_coef <- coef(lm_model)
lm_r_squared <- summary(lm_model)$r.squared
```

The linear model yielded an R^2 of 0.7406, indicating that approximately 74.1% of the variance in housing values is explained by the predictors.

3 Lasso Regression Model

For the Lasso model, cross-validation was used to determine the optimal λ value:

```r
# Fit Lasso model with cross-validation
cv_model <- cv.glmnet(X, y, alpha = 1)
best_lambda <- cv_model$lambda.min

# Fit final model with optimal lambda
lasso_model <- glmnet(X, y, alpha = 1, lambda = best_lambda)
lasso_coef <- coef(lasso_model)

# Calculate predictions and R-squared
lasso_predictions <- predict(lasso_model, newx = X)
lasso_r_squared <- 1 - (sum((y - lasso_predictions)^2) /
                        sum((y - mean(y))^2))
```

The cross-validation procedure selected $\lambda = 0.016$. The resulting Lasso model achieved an R^2 of 0.7404.

4 Results

4.1 Model Performance

Both models demonstrated comparable predictive performance on the training data. The OLS regression achieved an R^2 of 0.7406, while the Lasso regression achieved an R^2 of 0.7404. The difference in R^2 values was minimal (2×10^{-4}), indicating that the regularization penalty had little impact on in-sample fit. This similarity in performance suggests that while Lasso provides coefficient shrinkage, it does not substantially sacrifice explanatory power in this dataset.

5 Discussion

This analysis highlights Lasso regression as a practical alternative to traditional OLS regression. Both methods produced similar R^2 values, but differ in handling predictors: OLS retains all predictors with unconstrained coefficients, whereas Lasso applies regularization, shrinking coefficients and enabling automatic feature selection.

The similar performance indicates that the Boston Housing dataset does not suffer from severe multicollinearity or overfitting. In datasets with more predictors or stronger correlations, Lasso's benefits would be more evident. Note that the reported R^2 values are in-sample estimates; testing on held-out data would better assess generalization.

Lasso offers advantages in interpretability by identifying influential predictors and controlling large coefficients, improving model stability. However, it assumes sparsity in true coefficients, and highly correlated predictors may lead Lasso to arbitrarily select variables. In such cases, methods like elastic net, which combines L1 and L2 penalties, may be preferable.

6 Conclusion

This handout illustrated the implementation and comparison of linear and Lasso regression using the Boston Housing dataset. Both models achieved similar predictive performance, with R^2 values above 0.73. The Lasso model successfully applied regularization to shrink coefficient estimates while maintaining explanatory power. These results highlight that Lasso regression can be a valuable tool in the statistical modeling toolkit, particularly when feature selection or improved generalization is desired. Future analyses could extend this comparison by evaluating model performance on independent test data and exploring alternative regularization approaches.

Github Repo:
https://github.com/randomoranges/Multiple_Linear_Regression_Analysis_Using_R

7 Affidavit

I hereby affirm that this submitted paper was authored unaided and solely by me. Additionally, no other sources than those in the reference list were used. Parts of this paper, including tables and figures, that have been taken either verbatim or analogously from other works have in each case been properly cited with regard to their origin and authorship. This paper either in parts or in its entirety, be it in the same or similar form, has not been submitted to any other examination board and has not been published.

I acknowledge that the university may use plagiarism detection software to check my

thesis. I agree to cooperate with any investigation of suspected plagiarism and to provide any additional information or evidence requested by the university.

7.1 Checklist

- ☒ The handout contains 7 pages of text.
- ☒ The submission contains the Quarto file of the handout.
- ☒ The submission contains the Quarto file of the presentation.
- ☒ The submission contains the HTML file of the handout.
- ☒ The submission contains the HTML file of the presentation.
- ☒ The submission contains the PDF file of the handout.
- ☒ The submission contains the PDF file of the presentation.
- ☒ The title page of the presentation and the handout contain personal details (name, email, matriculation number).
- ☒ The handout contains a bibliography, created using BibTeX with an APA citation style.
- ☒ Either the handout or the presentation contains R code that demonstrates coding expertise.
- ☒ The filled out Affidavit.
- ☒ The link to the presentation and the handout published on GitHub.

[Jishnu Sai Matra,] [10 Dec 2025,] [Leverkusen]

8 References

Harris, M. (2023). *Regression with cross-validation in r*. https://www.statswithr.com/tutorials/regression-with-cross-validation-in-r

Hubchev, V. (n.d., n.d.). *8 regression analysis – quantitative methods*. https://hubchev.github.io/qm/regression.html

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x