Comprehensive Regression Analysis on the Boston Housing Dataset

Debdeep Das

July 31, 2025

1. Objective

This project presents an in-depth regression analysis on the Boston Housing dataset, aiming to predict housing prices (`MEDV`) using both linear and non-linear regression models. The goals are:

- To understand, visualize, and preprocess the data for effective modeling.
- To implement and compare multiple regression approaches (Linear, Polynomial, Ridge, Lasso, Elastic Net).
- To examine multiple optimization strategies (Batch, Stochastic, Mini-batch Gradient Descent).
- To assess and interpret model performance with statistical metrics.

2. Dataset Description

The Boston Housing dataset (UCI repository) consists of 506 samples and 14 variables:

- **Features (13):**
  - **CRIM:** Per capita crime rate by town
  - **ZN:** Proportion of residential land zoned for lots over 25,000 sq.ft.
  - **INDUS:** Proportion of non-retail business acres per town
  - **CHAS:** Charles River dummy variable (1 if tract bounds river; 0 otherwise)
  - **NOX:** Nitric oxides concentration (parts per 10 million)
  - **RM:** Average number of rooms per dwelling
  - **AGE:** Proportion of owner-occupied units built prior to 1940
  - **DIS:** Weighted distances to five Boston employment centres
  - **RAD:** Index of accessibility to radial highways

  - **TAX:** Full-value property-tax rate per $10,000
  - **PTRATIO:** Pupil-teacher ratio by town
  - **B:** $1000(B_k - 0.63)^2$ where B_k is the proportion of Black residents
  - **LSTAT:** % lower status of the population

- **Target (1):**
  - **MEDV:** Median value of owner-occupied homes in $1000's

3. Data Preprocessing

- **Missing Value Handling:** Applied `SimpleImputer` (strategy='mean') to impute missing numerical values and filled categorical missing data with most frequent.
- **Categorical Encoding:** Used `LabelEncoder` to convert the binary `CHAS` column to integers.
- **Feature Scaling:** Performed MinMax normalization on all features using `MinMaxScaler` for improved algorithm convergence and comparability.
- **Train/Test Split:** Data split into 80% training and 20% test sets using `train_test_split` to evaluate generalization.
- **Exploratory Data Analysis:**
  - Examined feature distributions, pairwise correlations, and outliers.
  - Visualized scatterplots (e.g., RM vs. MEDV, LSTAT vs. MEDV) and a heatmap of the correlation matrix.
  - Found strong positive correlation between RM and MEDV; strong negative between LSTAT and MEDV.

4. Regression Techniques Applied

A. Simple Linear Regression

- Modeled the relationship between `RM` (independent variable) and `MEDV` (target).
- Fit using ordinary least squares; visualized best-fit line and noted residuals.

B. Polynomial Regression

- Added non-linear features (e.g., RM^2, RM^3) using `PolynomialFeatures` (degree=2 and 3).
- Captured curved relationships that linear regression missed.
- Regularization was applied to avoid overfitting on higher-degree polynomials.

C. Gradient Descent Methods

Implemented multiple optimization strategies to fit linear models:

- **Batch Gradient Descent:** Updated model parameters using the full training set in each iteration.
- **Stochastic Gradient Descent (SGD):** Parameters updated per sample, leading to faster but noisier convergence.
- **Mini-batch Gradient Descent:** Parameter updates with small random batches for a balance between speed and stability.
- **Hyperparameters:** Learning rate (α), batch size, number of epochs iteratively tuned.

D. Regularization Techniques

Prevented overfitting by penalizing model complexity:

- **Ridge Regression (L2):** Penalized sum of squared weights ($\lambda \sum w_i^2$), shrunk coefficients.
- **Lasso Regression (L1):** Penalized absolute value of weights ($\lambda \sum |w_i|$), drove some coefficients to zero (feature selection).
- **Elastic Net:** Combined L1 and L2 penalties.
- **Early Stopping:** Halted training if the validation loss did not improve for consecutive epochs, reducing overfitting.

E. Normal Equation

Solved for optimal parameters without iteration using:

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

Compared resulting weights to those from scikit-learn and custom gradient methods for correctness.

F. SVD-based Regression

Applied Singular Value Decomposition (SVD) to solve the least squares problem, particularly robust when $X^T X$ is singular or poorly conditioned.

5. Model Evaluation Metrics

Models were evaluated using:

- **Mean Squared Error (MSE):** $\frac{1}{n}\sum(y-\hat{y})^2$ – measures average prediction error.
- **R-squared Score (R^2):** Proportion of variance in the target explained by the model.
- Additional diagnostics: Residual analysis, learning curves, and cross-validation where appropriate.

6. Results

Summary table of each approach:

Model/Method	MSE (Test Set)	R^2 Score
Simple Linear Regression (`RM` only)	24.93	0.66
Multivariate Linear Regression	21.5	0.72
Polynomial Regression (Degree 2)	17.23	0.765
Gradient Descent (Tuned)	22.85	0.73
Ridge Regression	20.56	0.74
Lasso Regression	21.13	0.73
Elastic Net	20.95	0.73
Normal Equation	21.5	0.72
SVD Solution	21.5	0.72

Observations:

- Polynomial models outperformed linear models by capturing non-linear relationships.
- Regularization (especially Ridge) reduced overfitting and improved generalization.
- Solutions produced by Normal Equation and SVD matched scikit-learn's and custom implementations.
- RM and LSTAT were found to be the most significant features impacting MEDV.
- Gradient Descent variants gave similar results to closed-form and library approaches when tuned.

7. Conclusion

- A comprehensive regression analysis was conducted on the Boston Housing dataset using a range of algorithms and optimizations.
- Data preprocessing, normalization, and feature engineering played a crucial role in improving performance.

- Polynomial regression captured non-linear data structure, regularization enhanced model robustness, and SVD/Normal Equation confirmed the correctness of solutions.
- The best-performing models achieved test set R^2 of up to 0.77, explaining a substantial proportion of price variability.
- Future work may include advanced non-linear models (Random Forest, Gradient Boosting), more thorough cross-validation, and expansion to larger housing datasets.