Case 1

02582 Computational Data Analysis

s260170, s260219, s257659

March 28, 2026

Introduction/Data description

We are given a regression problem where the goal is to predict a continuous response y from a $p = 100$-dimensional feature matrix $\mathbf{X}$, with only $n = 100$ training observations. This setting places us squarely in the *curse of dimensionality*: with $p = n$, a naive ordinary least squares model will overfit severely, finding spurious patterns that do not generalise to new data.

The training set (`case1Data.csv`) consists of 100 observations of y and 100 features. The prediction set (`case1Data_Xnew.csv`) contains 1,000 new observations for which we must produce predictions $\hat{y}_{\text{new}}$.

The feature matrix is composed of 95 continuous numerical variables (`x_01`, ..., `x_95`) and 5 categorical variables (`C_01`, ..., `C_05`). The data is not clean: approximately 14.9% of entries in $\mathbf{X}$ are missing at random, with individual features missing between 20 and 27 values out of 100. One categorical variable (`C_02`) takes a single constant value across all observations and carries no predictive information. The response y has no missing values, with a mean of 211.5 and a standard deviation of 72.5, ranging from 39.7 to 375.8.

The central challenge is therefore threefold: to handle missing data appropriately, to encode categorical variables meaningfully, and to select a model that controls variance without introducing excessive bias.

Model and method

We faced a severe curse of dimensionality where the number of features equals the number of observations ($p = n$). Standard ordinary least squares regression is strictly inadequate here because it will severely overfit the training data. We initially explored tree-based ensembles (XGBoost, Random Forest) and margin-based methods (Support Vector Regression) to capture potential non-linear relationships. However, given the linear structure of the problem and the need for interpretable sparsity, we ultimately chose Elastic Net as our primary approach.

Elastic Net is a regularised regression method that combines L_1 and L_2 penalties. The combined objective is:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \alpha \left[\frac{1-\rho}{2} \|\boldsymbol{\beta}\|_2^2 + \rho \|\boldsymbol{\beta}\|_1 \right] \right\} \tag{1}$$

where $\alpha > 0$ controls the overall regularisation strength and $\rho \in [0, 1]$ is the L_1 ratio. When $\rho = 1$ the model reduces to LASSO; when $\rho = 0$ it reduces to Ridge regression. The L_1 penalty enforces sparsity by driving irrelevant coefficients exactly to zero, performing automatic feature selection. The L_2 penalty handles multicollinearity by shrinking correlated predictors proportionally rather than arbitrarily selecting one.

Model selection

We performed hyperparameter tuning over a grid of 60 α values (logarithmically spaced from 10^{-3} to 10^{2}) and seven L_1 ratio values ($\rho \in \{0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 1.0\}$). This search was embedded within the inner loop of our nested cross-validation scheme, selecting the parameter combination that minimised mean squared error across the inner validation folds. The final model trained on all 100 observations converged to $\rho = 1.0$, effectively becoming a pure LASSO, retaining 26 non-zero coefficients out of 115 total features after one-hot encoding.

Missing values

Approximately 14.9% of the entries in the continuous features were missing. We addressed this using a K-Nearest Neighbors (KNN) imputer with $k = 5$. This exploits the multivariate structure of the data to estimate missing values more accurately than mean or median substitution, which would ignore correlations between features.

After imputation, all numerical features were standardised using a `RobustScaler`, which scales based on the interquartile range rather than the mean and standard deviation. This ensures that outliers, which can be introduced in sparse regions of the feature space during KNN imputation, do not disproportionately distort the regression coefficients.

Factor handling

Exploratory analysis revealed that `C_02` takes the constant value 72 across all observations. A zero-variance feature carries no predictive information, so it was removed entirely. For the remaining four categorical variables (`C_01`, `C_03`, `C_04`, `C_05`), missing values were imputed using the mode. We then applied One-Hot Encoding, expanding the four variables (each with 5 levels) into 20 binary columns, bringing the total feature space to 95 + 20 = 115 columns.

Model validation

Given the severely limited sample size of $n = 100$, maximising the data used for training is paramount. A single static train-test split would yield a highly volatile performance estimate. We therefore implemented a nested cross-validation strategy.

The outer loop uses 5-fold cross-validation to estimate generalisation error, ensuring every observation serves as a test point exactly once. Within each outer fold, a fresh pipeline is constructed and an inner 5-fold cross-validation tunes the Elastic Net hyperparameters exclusively on the 80 training observations of that fold. This nested architecture prevents data leakage during hyperparameter optimisation: the test fold is never seen by the imputer, scaler, or model during training or tuning.

To generate predictions for the 1,000 unseen targets, the full pipeline was retrained on all 100 training observations using a 10-fold inner cross-validation to maximise the information extracted from the limited sample.

Results

Our final model is a nested cross-validated Elastic Net pipeline combining KNN imputation, robust scaling, one-hot encoding of categorical variables, and automatic sparsity-inducing regularisation.

The 5-fold nested cross-validation yielded a final estimated $\widehat{\text{RMSE}} = 27.55$, which represents our most statistically rigorous assessment of generalisation error. This estimate avoids the high variance of a single small test set and prevents optimistic bias from data leakage during preprocessing or hyperparameter tuning.

The final model trained on the full dataset converged to an L_1 ratio of 1.0 (pure LASSO) with 26 non-zero coefficients out of 115, confirming that the true signal in this 100-dimensional problem is sparse and that the regularisation successfully suppressed noise features.

Use of artificial intelligence

We used Claude (Anthropic) as an AI assistant during this project. The AI was used to support our understanding of certain concepts and to assist with the drafting and editing of this report. All code, modelling decisions, interpretations, and conclusions are our own.