# ML Lab — Assignment:2

August 20, 2025

## Objective

Implement linear regression from scratch using `numpy` by (a) solving the normal equation, and (b) using gradient descent. Secondly also implement it using `scikit-learn`'s `LinearRegression`. Produce plots that show fit quality, (loss vs iterations), and gradient/ loss-surface visualizations.

## 1 Dataset

california house prediction test.csv
train.csv

## 2 Tasks

### A. Data preprocessing

1) Training- train.csv
2) Validation- test.csv
3) Standardize features (zero mean, unit variance)
4) Add an intercept (bias) column of ones to the design matrix $X$ before computing parameters.

### B. Implement Normal Equation (closed-form)

- Derive and implement the solution:

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

where $X \in \mathbb{R}^{n \times (d+1)}$ has the bias column.

### C. Implement Batch Gradient Descent (iterative)

1) Implement the mean squared error (MSE) loss:

$$J(\theta) = \frac{1}{2n} \sum_{i=1}^{n} (h_\theta(x^{(i)}) - y^{(i)})^2$$

where $h_\theta(x) = x^\top \theta$.

2) Implement batch gradient descent updates:

$$\theta \leftarrow \theta - \alpha \nabla J(\theta), \quad \nabla J(\theta) = \frac{1}{n} X^\top (X\theta - y)$$

3) Support the following hyperparameters: learning rate $\alpha$, number of iterations, and an optional early stopping tolerance on change in loss.

4) Track and store training loss at each iteration for plotting.

### E. Comparisons with scikit-learn

- Fit `sklearn.linear_model.LinearRegression()` on the same training data and compare both the results(the one you did with numpy and the other— scikit-library)

### F. Visualizations

Produce clear, labeled figures. Save as PNG/PDF and include in the report.

1) **Loss vs iterations:** Plot training loss (MSE) against iteration number for gradient descent. Include curves for different learning rates and for unscaled vs scaled features to show the effect of scaling.

2) Plot validation loss

3) Gradient/loss surface visualization

### G. Evaluation metrics

Compute and report these metrics on the test set:

- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- Coefficient of Determination $R^2$
- Optionally: Mean Absolute Error (MAE)

Compare metrics across the three implementations (normal equation, gradient descent, sklearn).