Linear Regression

Linear Regression is a supervised learning algorithm used to model the relationship between input variables and a continuous output variable. It assumes that the output is a linear combination of the input features. The goal is to estimate parameters that best explain the relationship between inputs and output by minimizing prediction error.

1. Problem Formulation

This section defines what problem linear regression is solving.

Input Space $\mathcal{X} \subseteq \mathbb{R}^p$ where Each input vector: $x \in \mathbb{R}^p$

Output Space $\mathcal{Y} \subseteq \mathbb{R}$ The output is continuous and real-valued.

Data Distribution

We assume training samples:

$$(x_i, y_i) \sim P_{X,Y}$$

are independently and identically distributed (i.i.d).

Learning Objective (Expected Risk Minimization)

The true objective is to minimize expected prediction error:

$$\min_{\beta} \mathbb{E}_{(x,y)}[(y - x^T\beta)^2]$$

Since the true distribution is unknown, we approximate using empirical risk.

2. Model Specification

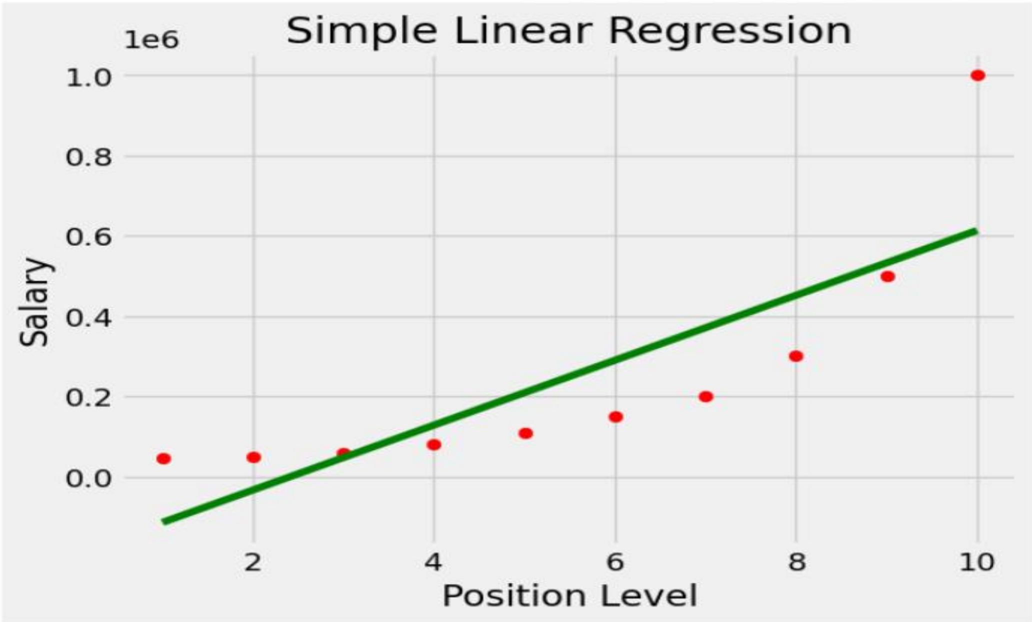
This section defines the mathematical structure of the model.

Hypothesis Function

Linear model:

$$h_\beta(x) = x^T\beta$$

The prediction is a linear combination of features.



Parameter Space $\beta \in \mathbb{R}^p$

The model learns these parameters from data.

Structural Assumptions

1. Linearity in parameters
2. Additive noise model: $y = x^T \beta + \varepsilon$
3. Common assumptions:
   - $E[\varepsilon] = 0$
   - $Var(\varepsilon) = \sigma^2$
   - Errors are independent

3. Loss Function

This section defines how prediction error is measured.

Explicit Mathematical Form

Squared error loss:

$$\ell(y, x^T \beta) = (y - x^T \beta)^2$$

Why This Loss?

1. Penalizes large errors heavily.
2. Smooth and differentiable.
3. Leads to a closed-form solution.
4. Corresponds to Gaussian noise assumption.

Convexity Properties

Loss: $L(\beta) = \| y - X\beta \|^2$

Hessian: $\nabla^2 L(\beta) = 2X^T X$

Since $X^T X$is positive semi-definite, squared loss is convex.
If Xhas full rank, it is strictly convex.

4. Objective Function

This section defines what is minimized during training.

Empirical Risk

$$R(\beta) = \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2$$

Matrix form: $R(\beta) = \frac{1}{n} \| y - X\beta \|^2$

Regularized Formulation (Ridge Example)

To reduce overfitting:

$$R(\beta) = \frac{1}{n} \| y - X\beta \|^2 + \lambda \| \beta \|^2$$

Regularization penalizes large coefficients.

5. Optimization Method

This section explains how parameters are estimated.

Closed-form Solution

Setting gradient to zero:

$$X^T X\beta = X^T y$$

Solution:

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

Exists only if $X^T X$is invertible.

Gradient Expression

$$\nabla L(\beta) =- 2X^T(y - X\beta)$$

Convergence Guarantees

Since the objective is convex:

- Any local minimum is global minimum.
- Gradient descent converges with proper learning rate.

Computational Complexity

- Forming $X^T X$: $O(np^2)$
- Matrix inversion: $O(p^3)$

Total complexity: $O(np^2 + p^3)$

6. Statistical Interpretation

<u>MLE Connection</u>

Assume: $\varepsilon \sim N(0, \sigma^2)$

Then: $y|x \sim N(x^T\beta, \sigma^2)$

Maximizing likelihood equals minimizing squared loss.

Thus: OLS = Maximum Likelihood Estimator under Gaussian noise

Noise Model Assumption

- Additive
- Gaussian
- Homoscedastic

Probabilistic Meaning

The model estimates conditional expectation:

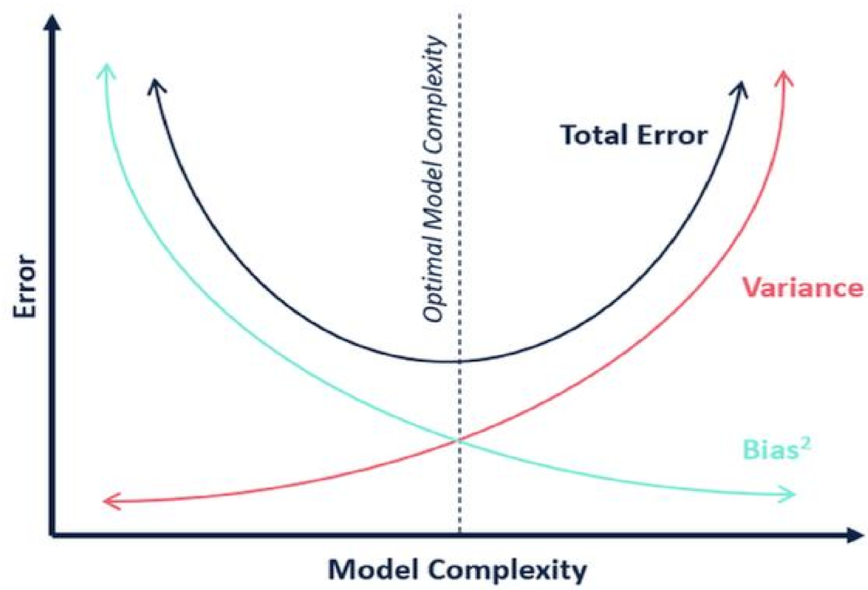
$$E[y|x] = x^T\beta$$

7. Regularization & Generalization

This section explains model complexity control.

Bias–Variance Tradeoff

- OLS: Low bias, potentially high variance.
- Ridge: Slightly higher bias, lower variance.



Overfitting Behavior

Overfitting occurs when:

- Too many features

- High multicollinearity
- Small dataset

Capacity Control

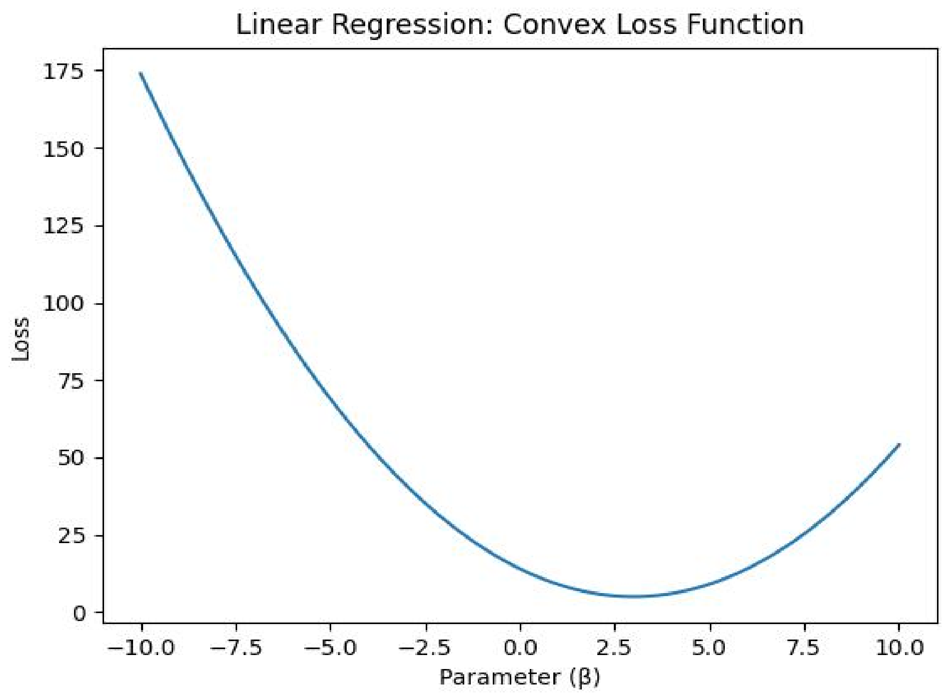
Controlled by:

- Number of parameters
- Regularization strength λ

8. Theoretical Properties

Convexity / Global Optimality

Squared loss is convex → unique global minimum if full rank.



Consistency

Under standard assumptions: $\widehat{\beta} \to \beta \quad$ as $n \to \infty$.

Stability

If smallest eigenvalue of $X^T X$ is small:

- Variance increases
- Estimates become unstable

Variance formula:

$$Var(\widehat{\beta}) = \sigma^2 (X^T X)^{-1}$$

9. Computational Complexity

Training Time

Closed-form: $O(np^2 + p^3)$

Gradient descent: $O(np)$ per iteration

Inference Time

Prediction per sample: $O(p)$

Memory Complexity

- Storing data: $O(np)$
- Storing parameters: $O(p)$

10. Limitations

This section highlights weaknesses.

When It Fails

1. Nonlinear relationships
2. p >> n
3. Severe multicollinearity
4. Presence of strong outliers

Assumption Violations

- Non-Gaussian noise
- Heteroscedasticity
- Correlated errors

Sensitivity Issues

- Sensitive to outliers

- Sensitive to scaling
- Numerically unstable if ill-conditioned