Linear Regression with Gradient Descent and Regularization

Kashyap Sureshchandra Patel

September 6, 2025

Algorithm 1: Linear Regression with Gradient Descent and Regularization

Input: Learning rate η, number of epochs T, batch size b, regularization flag r, regularization method $m \in \{\text{L1, L2, ElasticNet}\}$, regularization strength λ, mixing parameter α

Output: Optimal parameter vector θ

Initialize: $\theta \leftarrow \mathbf{0}$

Add bias column of ones to X: $X_b = [\mathbf{1}, X]$

$n \leftarrow$ number of samples, $d \leftarrow$ number of features (including bias)

for *epoch* $= 1$ *to* T **do**
- Shuffle dataset (X_b, y) randomly
- **for** *each mini-batch* (X_{batch}, y_{batch}) *of size* b **do**
  - $y_{\text{pred}} \leftarrow X_{\text{batch}} \cdot \theta$
  - Gradient: $\nabla J(\theta) \leftarrow \frac{2}{b} X_{\text{batch}}^{\top}(y_{\text{pred}} - y_{\text{batch}})$
  - **if** $r =$ *True* **then**
    - $\theta_{\text{reg}} \leftarrow \theta$, set $\theta_{\text{reg},0} \leftarrow 0$
    - **if** $m =$ *L1* **then**
      - $\nabla J(\theta) \leftarrow \nabla J(\theta) + \lambda \cdot \text{sign}(\theta_{\text{reg}})$
    - **else if** $m =$ *L2* **then**
      - $\nabla J(\theta) \leftarrow \nabla J(\theta) + 2\lambda \cdot \theta_{\text{reg}}$
    - **else if** $m =$ *ElasticNet* **then**
      - $\nabla J(\theta) \leftarrow \nabla J(\theta) + \lambda\Big(\alpha \cdot \text{sign}(\theta_{\text{reg}}) + 2(1-\alpha) \cdot \theta_{\text{reg}}\Big)$
  - Update step: $\theta \leftarrow \theta - \eta \cdot \nabla J(\theta)$
- **if** *epoch* $\bmod 100 = 0$ **then**
  - Compute training MSE: $\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - X_b^{(i)}\theta)^2$
  - Print epoch and MSE