Mini-Batch Gradient Descent Example

Mini-Batch Gradient Descent combines the advantages of **Batch Gradient Descent** (stability) and **Stochastic Gradient Descent** (speed). In Mini-Batch Gradient Descent, the dataset is divided into smaller subsets (mini-batches), and weight updates are performed on each mini-batch.

Step 1: Dataset and Model

Dataset

We use the same dataset as before, predicting house prices using two features:

- x_1 = Area (sq. ft.)
- x_2 = Bedrooms

x_1 **(Area)**	x_2 **(Bedrooms)**	y **(Price)**
2600	3	550000
3000	4	565000
3200	3	610000
3600	5	680000

Model

The model equation is:

$$\text{Predicted Price} = w_1 \cdot x_1 + w_2 \cdot x_2 + \text{bias}$$

Initial Parameters:

- $w_1 = 0.5$, $w_2 = 0.5$, $\text{bias} = 0$
- Learning rate (α) = 0.000000001
- **Mini-batch size** = 2 (Each batch contains 2 samples)

Step 2: Mini-Batch Gradient Descent Process

Batch 1: First Two Samples

For $x_1 = [2600, 3000]$, $x_2 = [3, 4]$, $y = [550000, 565000]$:

1. **Predicted Price for Each Sample**:
   - $\text{Predicted}_1 = w_1 \cdot 2600 + w_2 \cdot 3 + \text{bias} = 0.5 \cdot 2600 + 0.5 \cdot 3 + 0 = 1301.5$
   - $\text{Predicted}_2 = w_1 \cdot 3000 + w_2 \cdot 4 + \text{bias} = 0.5 \cdot 3000 + 0.5 \cdot 4 + 0 = 1502.0$
2. **Errors for Each Sample**:
   - $\text{Error}_1 = \text{Predicted}_1 - y_1 = 1301.5 - 550000 = -548698.5$
   - $\text{Error}_2 = \text{Predicted}_2 - y_2 = 1502.0 - 565000 = -563498.0$
3. **Average Gradients (Mini-Batch)**:
   - Gradient for w_1:

$$\frac{\partial MSE}{\partial w_1} = \frac{1}{2}\sum_{i=1}^{2} 2 \cdot \text{Error}_i \cdot x_{1i} = \frac{1}{2}[2 \cdot (-548698.5) \cdot 2600 + 2 \cdot (-563498.0) \cdot 3000] = -1693000082.5$$

- Gradient for w_2:

$$\frac{\partial MSE}{\partial w_2} = \frac{1}{2}\sum_{i=1}^{2} 2 \cdot \text{Error}_i \cdot x_{2i} = \frac{1}{2}[2 \cdot (-548698.5) \cdot 3 + 2 \cdot (-563498.0) \cdot 4] = -8542182.5$$

- Gradient for bias:

$$\frac{\partial MSE}{\partial \text{bias}} = \frac{1}{2}\sum_{i=1}^{2} 2 \cdot \text{Error}_i = \frac{1}{2}[2 \cdot (-548698.5) + 2 \cdot (-563498.0)] = -1112196.5$$

4. **Update Parameters**:
   - $w_1 = w_1 - \alpha \cdot \frac{\partial MSE}{\partial w_1} = 0.5 - 0.000000001 \cdot (-1693000082.5) = 0.501693$
   - $w_2 = w_2 - \alpha \cdot \frac{\partial MSE}{\partial w_2} = 0.5 - 0.000000001 \cdot (-8542182.5) = 0.50000854$
   - $\text{bias} = \text{bias} - \alpha \cdot \frac{\partial MSE}{\partial \text{bias}} = 0 - 0.000000001 \cdot (-1112196.5) = 0.001112$

Batch 2: Next Two Samples

For $x_1 = [3200, 3600]$, $x_2 = [3, 5]$, $y = [610000, 680000]$:

1. **Predicted Price for Each Sample**:
   - $\text{Predicted}_3 = w_1 \cdot 3200 + w_2 \cdot 3 + \text{bias} = 0.501693 \cdot 3200 + 0.50000854 \cdot 3 + 0.001112 = 1605.5$
   - $\text{Predicted}_4 = w_1 \cdot 3600 + w_2 \cdot 5 + \text{bias} = 0.501693 \cdot 3600 + 0.50000854 \cdot 5 + 0.001112 = 1808.7$
2. **Errors for Each Sample**:
   - $\text{Error}_3 = \text{Predicted}_3 - y_3 = 1605.5 - 610000 = -608394.5$
   - $\text{Error}_4 = \text{Predicted}_4 - y_4 = 1808.7 - 680000 = -678191.3$
3. **Average Gradients (Mini-Batch)**:
   - Similar calculations as before, averaging the gradients for w_1, w_2, and bias.
4. **Update Parameters**:
   - Update w_1, w_2, and bias using the new gradients.

Step 3: Repeat for All Mini-Batches

Continue the process for all mini-batches for multiple epochs (iterations over the entire dataset).

Advantages of Mini-Batch Gradient Descent

1. **Faster convergence** than batch gradient descent.
2. More stable updates compared to stochastic gradient descent.
3. Efficient use of vectorized operations in modern libraries like TensorFlow and PyTorch.