Equation $y = mx + b$

Initial setup

Variable	Value
m_0	-1
b_0	1
Learning rate α	0.1
Data point	$(1,3)$, $(3,6)$
Number of samples (n)	2

Initial predictions (step 1) Iteration (0)

Using $y = mx + b$

for $x = 1$

$\hat{y}_1 = (-1)(1) + 1 = 0$

for $x = 3$

$\hat{y}_2 = (-1)(3) + 1 = -2$

Calculate MSE_0

$$MSE_0 = \frac{1}{n}\sum (y_i - \hat{y}_i)^2 = \frac{1}{2}\left[(3-0)^2 + (6-(-2))^2\right]$$

$$MSE_0 = \frac{1}{2}\left[9+64\right] = \frac{73}{2} = \boxed{36.5}$$

Calculation of gradient

$$\frac{\partial MSE}{\partial m} = \frac{2}{n}\sum x_i(\hat{y}_i - y_i)$$

$$= \frac{2}{2}\left[1(0-3) + 3(-2-6)\right] = -3 - 24 = \boxed{-27}$$

$$\frac{\partial MSE}{\partial b} = \frac{2}{n} \sum (\hat{y}_i - y_i) = \frac{2}{2}[(0-3)+(-2-6)]$$

$$= -3-8 = \boxed{-11}$$

Update parameters

$$m_1 = m_0 - \alpha . \frac{\partial MSE}{\partial m} = -1-0.1(-27) = -1+2.7$$

$$= \boxed{1.7}$$

$$b_1 = b_0 - \alpha \frac{\partial MSE}{\partial b} = 1-0.1(-11) = 1+1.1$$

$$= \boxed{2.1}$$

Calcutions of MSE1

$$\hat{y}_1 = 1,7(1)+2.1 = 3.8$$

$$\hat{y}_2 = 1.7(2)+2.1 = 7.2$$

$$MSE_1 = \frac{1}{2}\left[(3-3.8)^2 + (6-7.2)^2\right]$$

$$= \frac{1}{2}[0.64+1,44] = \frac{2.08}{2} = \boxed{1.04}$$

Improvement $MSE_0 = 36.5 \rightarrow MSE_1 = 1.04$

ACE Scanner

$m_1 = 1.7$, $b_1 = 2.1$

$\hat{y}_1 = 1.7(1) + 2.1 = 3.8$

$\hat{y}_2 = 1.7(3) + 2.1 = 7.2$

Gradients

$$\frac{\partial MSE}{\partial m} = \frac{2}{2}\left[(1)(3.8 - 3) + 3(7.2 - 6)\right]$$

$$= (0.8 + 3.6)$$

$$= \boxed{4.4}$$

$$\frac{\partial MSE}{\partial b} = \frac{2}{2}\left[(3.8 - 3) + (7.2 - 6)\right]$$

$$= 0.8 + 1.2$$

$$= \boxed{2.0}$$

Updated Parameters

$m_2 = 1.7 - 0.1(4.4) = \boxed{1.26}$

$b_2 = 2.1 - 0.1(2.0) = \boxed{1.9}$

Calculations of MSE_2

$\hat{y}_1 = 1.26(1) + 1.9 = 3.16$

$\hat{y}_2 = 1.26(3) + 1.9 = 5.58$

$$MSE_2 = \frac{1}{2}\left[(3 - 3.16)^2 + (6 - 5.58)^2\right]$$

$$= \frac{1}{2}\left[0.0256 + 0.1764\right] = \boxed{0.101}$$

The final MSE decreased

$MSE_1 = 1.04 \longrightarrow MSE_2 = 0.101$

Given $m_2 = 1.26$, $b_2 = 1.9$

$\hat{y}_1 = 1.26 + 1.9 = 3.16$

$\hat{y}_2 = 3.78 + 1.9 = 5.58$

Gradients

$$\frac{\partial MSE}{\partial m} = (1)(0.16) + 3(-0.42) = 0.16 - 1.26$$
$$= \boxed{-1.1}$$

$$\frac{\partial MSE}{\partial b} = 0.16 - 0.42 = \boxed{-0.26}$$

Updated variables

$$m_3 = 1.26 - 0.1(-1.1) = \boxed{1.37}$$
$$b_3 = 1.9 - 0.1(-0.26) = \boxed{1.926}$$

Calculations of MSE_3

$\hat{y}_1 = 1.37 + 1.926 = 3.296$

$\hat{y}_2 = 4.11 + 1.926 = 6.036$

$$MSE_3 = \frac{1}{2}\left[(3 - 3.296)^2 + (6 - 6.036)^2\right]$$
$$= \frac{1}{2}\left[0.0876 + 0.0013\right] = \boxed{0.044}$$

The final results shows that $MSE_3 = 0.044$ as the value continue to decrease.

Final Progression Table

Iteration	m	b	MSE
0	-1	1	36.5
1	1.7	2.1	1.04
2	1.26	1.9	0.101
3	1.37	1.926	0.044

Final Trend Analysis

→ MSE decreased significantly in each step from 36.5 → 1.04 → 0.101 → 0.044

→ Parameters are converging towards optimal values

→ m is changing more rapidly than b, which makes sense given the data.

→ Model is improving steadily.

Comprehensive Analysis

- **Convergence Assessment**:
  Each iteration shows smaller gradient steps and a decreasing MSE, proving the model is **converging steadily**.

- **Error Reduction**:
  MSE reduced from **36.5 to 0.044**, a total reduction of **36.456** — over **99% improvement**.

- **Learning Rate Effectiveness**:
  The learning rate α=0.1\alpha = 0.1α=0.1 was effective — it enabled **fast improvement** without overshooting or divergence.

- **Prediction Improvement**:
  Final predictions:
  - For x=1x = 1x=1, prediction = 3.296 (actual = 3)
  - For x=3x = 3x=3, prediction = 6.036 (actual = 6)
    These small errors show that the model learned the pattern well.

Conclusions

- Gradient descent successfully optimized the linear model.
- The parameters mmm and bbb are converging smoothly.
- MSE decreased in every iteration, confirming consistent learning.
- Final predictions are **very close** to the true values, proving good model performance.

Further Insights

- The pattern of MSE decline can be used to predict when to **stop training**.
- Parameter update rates can inform **adaptive learning rate strategies** in larger models.
