Gradient Boosting Algorithm

Gradient Boosting is an ensemble learning algorithm that builds a strong predictive model by sequentially adding weak learners (typically shallow decision trees), where each new learner is trained to minimize the loss function by fitting the negative gradient (residual errors) of the current ensemble's predictions.

Why It's Called "Gradient" Boosting

Because each model:

- Learns the negative gradient of the loss
- Uses gradient descent in function space
- Optimizes any differentiable loss function

Step-by-Step Working of Gradient Boosting

🔹 Step 1: Choose a loss function

The loss function defines what "error" means.

- **Regression:** Mean Squared Error (MSE)

$$L(y, \hat{y}) = (y - \hat{y})^2$$

- **Classification:** Log loss

$$L(y, \hat{y}) = -[y\log(\hat{y}) + (1 - y)\log(1 - \hat{y})]$$

🔹 Step 2: Initialize the model

Start with a **constant prediction** that minimizes the loss.

For regression (MSE):

$$F_0(x) = \arg\min_c \sum (y_i - c)^2 = \text{mean}(y)$$

This is the **baseline model**.

🔹 Step 3: Compute residuals (negative gradients)

At iteration t, compute the residuals:

$$r_i^{(t)} = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}$$

For MSE loss:

$$r_i^{(t)} = y_i - \hat{y}_i$$

These residuals represent **what the model still fails to explain**.

◈ Step 4: Train a weak learner on residuals

Train a **shallow decision tree** $h_t(x)$to predict the residuals.

- Inputs: original features x
- Targets: residuals $r_i^{(t)}$

◈ Step 5: Compute optimal step size (optional but important)

Find the best multiplier $\gamma_t$for the new tree:

$$\gamma_t = \arg\min_{\gamma} \sum L(y_i, F_{t-1}(x_i) + \gamma h_t(x_i))$$

This ensures the update moves in the **best direction**.

◈ Step 6: Update the ensemble model

$$F_t(x) = F_{t-1}(x) + \eta \cdot \gamma_t \cdot h_t(x)$$

Where:

- η= learning rate (shrinkage)
- Controls overfitting

◈ Step 7: Repeat for T iterations

Repeat **Steps 3–6** until:

- Maximum trees reached
- Loss converges

- Validation error increases (early stopping)

5️⃣ Final Prediction

- **Regression:**

$$\hat{y} = F_T(x)$$

- **Classification:**
  Apply sigmoid or softmax to convert scores to probabilities.

Example –

Problem: Predict House Rent (Regression)

We want to predict **monthly rent** based on **house size**.

📊 Dataset (Small Real-World Dataset)

	House Size (sq ft)	Rent (₹ in thousands)
1	500	15
2	700	20
3	900	25
4	1100	28

🔧 Assumptions

- **Loss function:** Mean Squared Error (MSE)

$$L(y, \hat{y}) = (y - \hat{y})^2$$

- **Weak learners:** Decision stumps
- **Learning rate:** $\eta = 0.5$
- **Number of boosting rounds:** 2

🔷 Step 1: Initialize the Model

For MSE loss, the best constant prediction is the **mean of target values**:

$$F_0 = \frac{15 + 20 + 25 + 28}{4} = 22$$

Initial prediction for all houses:

$$\hat{y} = 22$$

◈ Step 2: Compute Residuals (Iteration 1)

$$r_i = y_i - \hat{y}_i$$

House	Actual Rent	Prediction	Residual
1	15	22	−7
2	20	22	−2
3	25	22	+3
4	28	22	+6

◈ Step 3: Train Weak Learner 1 on Residuals

Decision stump split:

Rule:

$$\text{If Size} \leq 800 \Rightarrow \text{predict } -4.5$$
$$\text{If Size} > 800 \Rightarrow \text{predict } +4.5$$

(These are mean residuals of each group.)

◈ Step 4: Update Model (Iteration 1)

$$F_1(x) = F_0(x) + \eta \cdot h_1(x)$$

House	$h_1(x)$	New Prediction
1	−4.5	$22 - 2.25 = 19.75$

House	$h_1(x)$	New Prediction
2	−4.5	19.75
3	+4.5	$22 + 2.25 = 24.25$
4	+4.5	24.25

🔷 Step 5: Compute Residuals (Iteration 2)

$$r_i = y_i - \hat{y}_i$$

House	Actual	New Prediction	Residual
1	15	19.75	−4.75
2	20	19.75	+0.25
3	25	24.25	+0.75
4	28	24.25	+3.75

🔷 Step 6: Train Weak Learner 2 on Residuals

Decision stump split:

Rule:

$$\text{If Size} \leq 900 \Rightarrow \text{predict } -1.25$$
$$\text{If Size} > 900 \Rightarrow \text{predict } +3.75$$

🔷 Step 7: Update Model (Iteration 2)

$$F_2(x) = F_1(x) + \eta \cdot h_2(x)$$

House	$h_2(x)$	Final Prediction
1	−1.25	$19.75 - 0.625 = 19.13$
2	−1.25	19.13
3	−1.25	$24.25 - 0.625 = 23.63$

House	$h_2(x)$	Final Prediction
4	+3.75	$24.25 + 1.875 = 26.13$

🏁 Final Gradient Boosting Model

$$F(x) = 22 + 0.5h_1(x) + 0.5h_2(x)$$

🔍 Example Prediction

New house: 1000 sq ft

- $h_1 = +4.5$
- $h_2 = +3.75$

$$\hat{y} = 22 + 0.5(4.5) + 0.5(3.75) = 26.125$$

🡒 Predicted rent ≈ ₹26,000