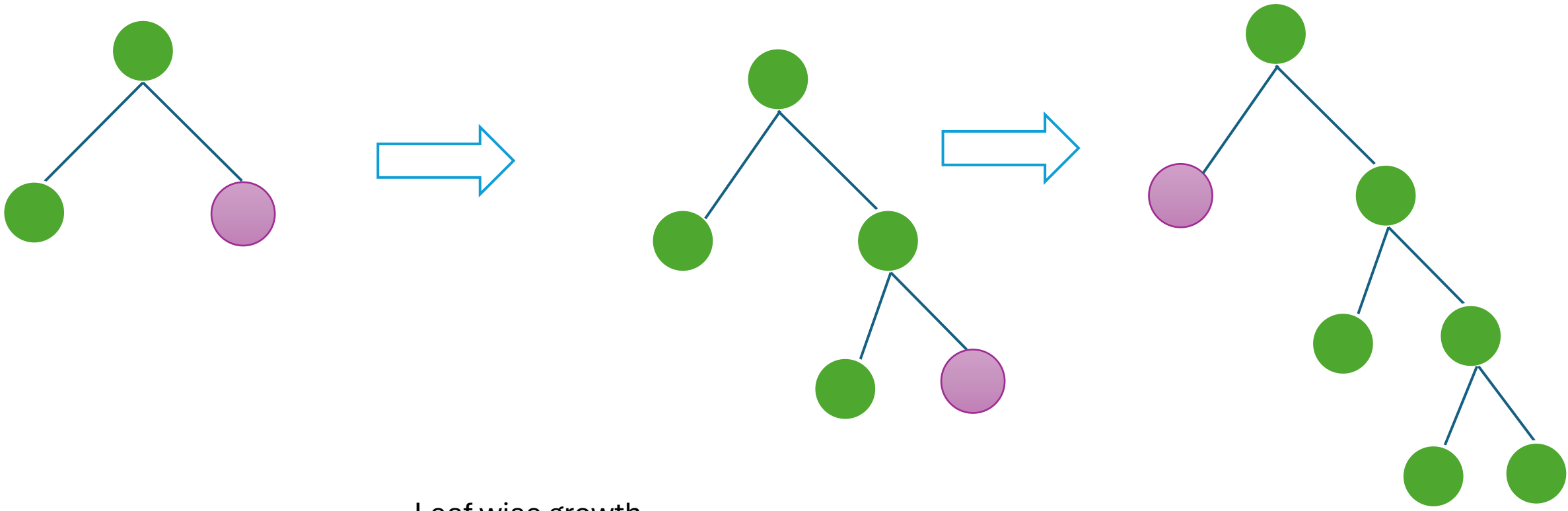
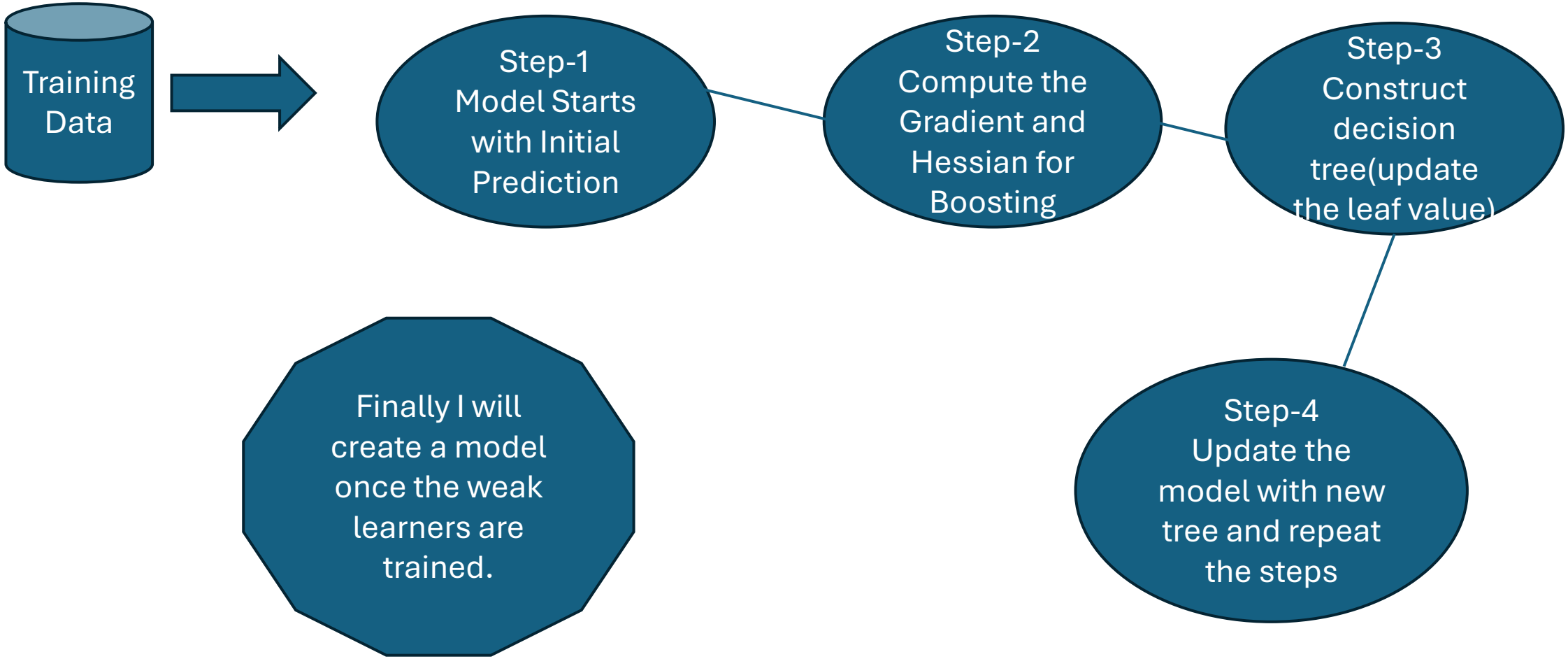
LGBM REGRESSION

- LGBM regressor is an Gradient boosting algorithm. It optimized for speed and efficiency .
- It outperforms the other boosting algorithm.
- It uses the gradient boosting decision tree.
- It built trees leaf wise instead level wise.
- It will train the weak leaners and combine them to make as a strong learner to built the predictive model.



Leaf wise growth

LGBM Regression work process:



Formulas:

Mean Squared Error (MSE) loss function:

$$L(y, F(x)) = \frac{1}{n}\sum_{i=1}^{n}(y_i - F(x_i))^2$$

Gradient (First derivative of loss function):

$$g_i = \frac{\partial L}{\partial F(x_i)} = 2(F(x_i) - y_i)$$

Hessian (Second derivative of loss function):

$$h_i = \frac{\partial^2 L}{\partial F(x_i)^2} = 2$$

Formula for leaf value update:

$$w_l = -\frac{\sum g_i}{\sum h_i + \lambda}$$

where λ is the regularization term.

Formula for model update:

$$F_{m+1}(x) = F_m(x) + \eta h_m(x)$$

where η is the learning rate.

Formula for R-squared:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

Pros:

- Faster than traditional boosting algorithms.
- Handles large datasets efficiently.
- Supports categorical variables natively.
- Provides high accuracy with fine-tuned parameters.

Cons:

- Sensitive to hyperparameter tuning.
- May overfit if **num_leaves** is set too high.
- Not ideal for small datasets due to leaf-wise growth.
- Requires GPU for the best performance in large-scale problems.