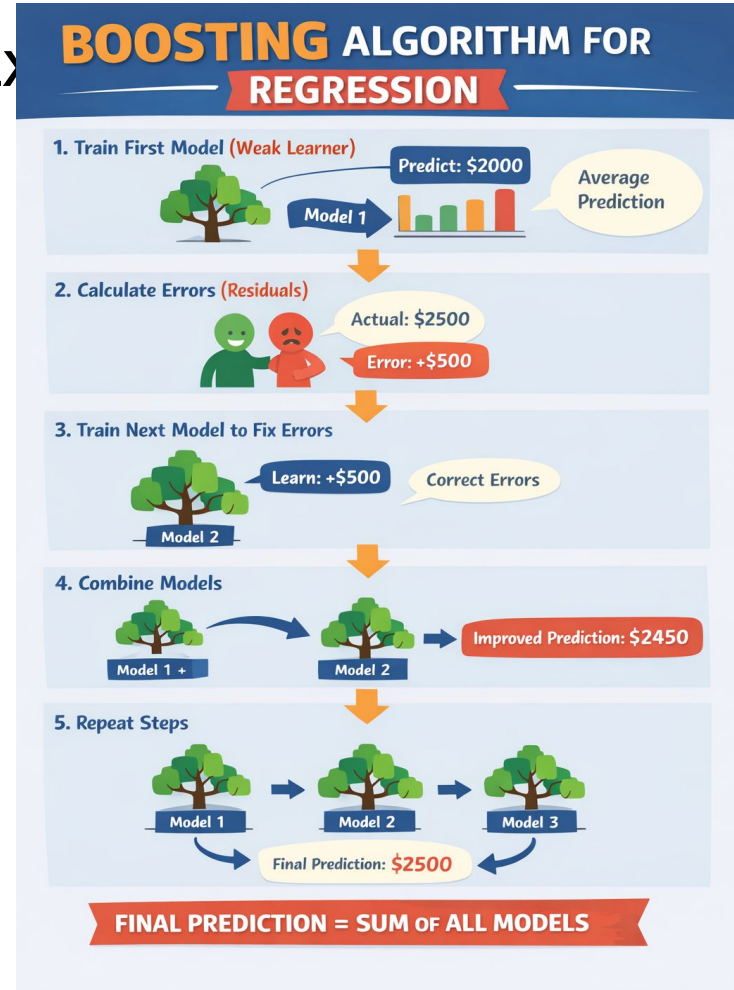# BOOSTING ALGORITHM

REGRESSION

Boosting Algorithm - Regressionzx

Build **many simple model** one after another.

Each model focus on the **error made by previous model**.
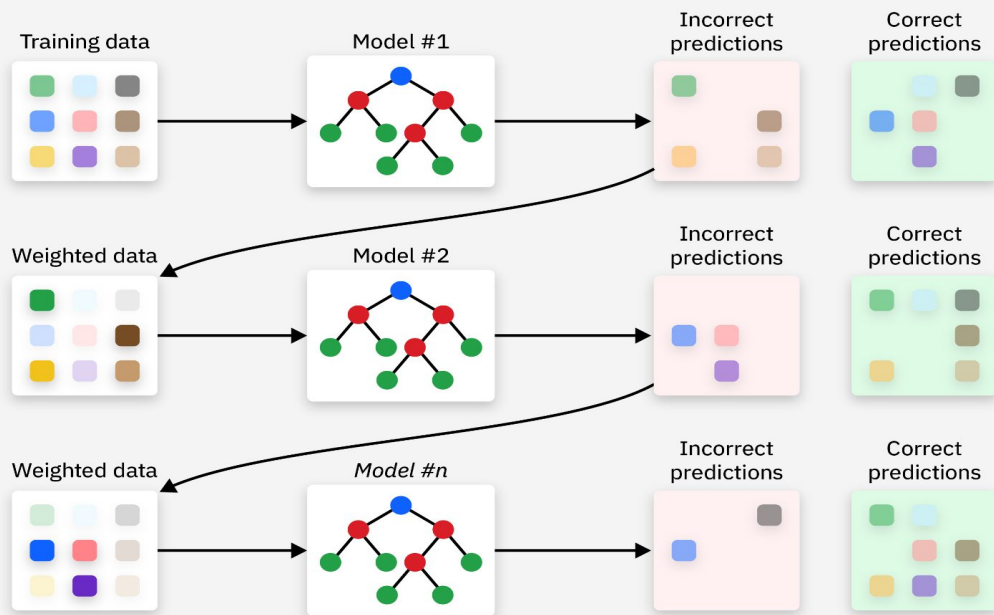
Algorithm implementing the boosting concept are:

- Gradient Boosting Machine or Gradient Boosting Decision Tree (GBM/GBDT)
- XGBoost (Extreme Gradient Boosting)
- LightGBM (Light Gradient Boosting Machine)
- AdaBoost.RT
- CatBoost (Categorical Boosting)

Gradient Boosting Machine or Gradient Boosting Decision Tree (GBM/GBDT)

Fix the previous model's mistakes step by step

Code:

```
from sklearn.ensemble import GradientBoostingRegressor


model = GradientBoostingRegressor()

model.fit(X_train, y_train)

y_pred = model.predict(X_test)
```
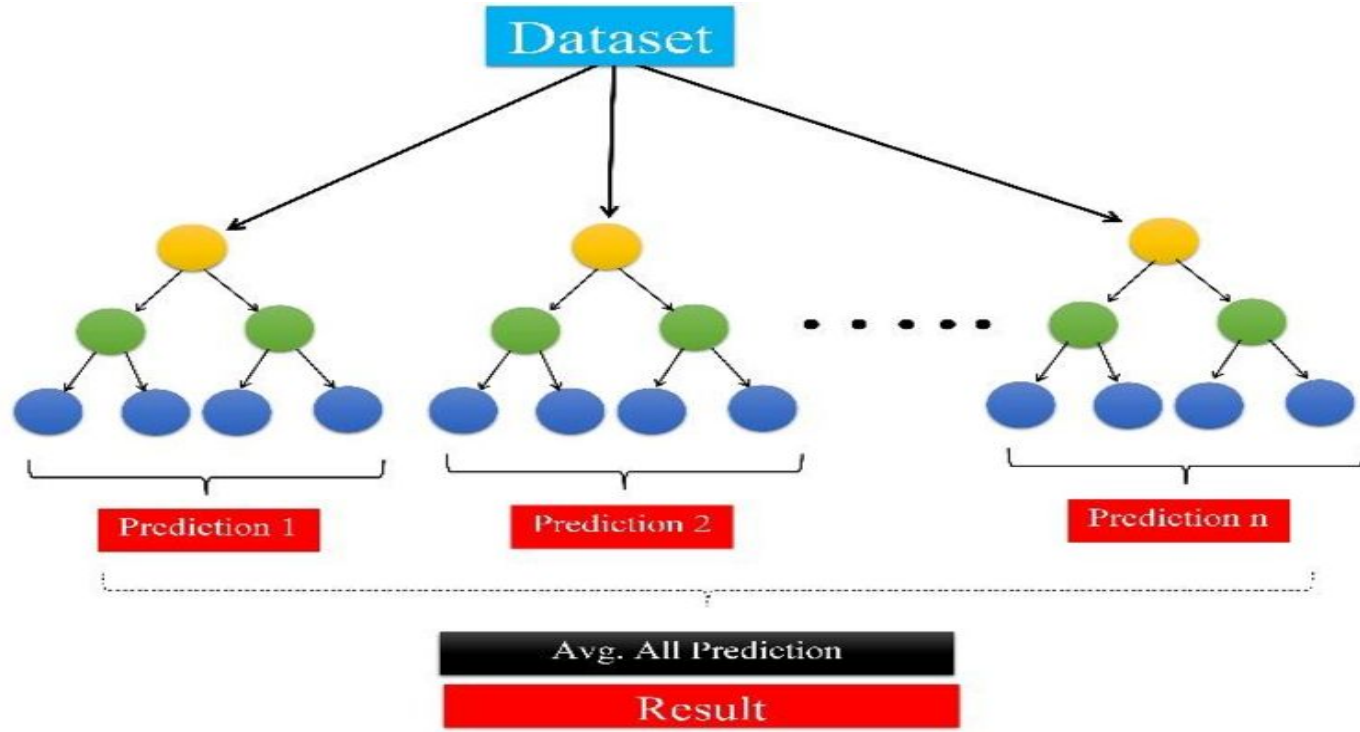
When to use:

- Learning fundamentals
- Very small dataset
- Simplicity > performance

XG BOOST:

Optimized, regularized and updated GBM. Level-wise tree growth.

CODE:

```python
from xgboost import XGBRegressor


xgb = XGBRegressor(
    n_estimators=200,
    learning_rate=0.05,
    max_depth=4,
    reg_alpha=0.1,
    reg_lambda=1.0
)
```
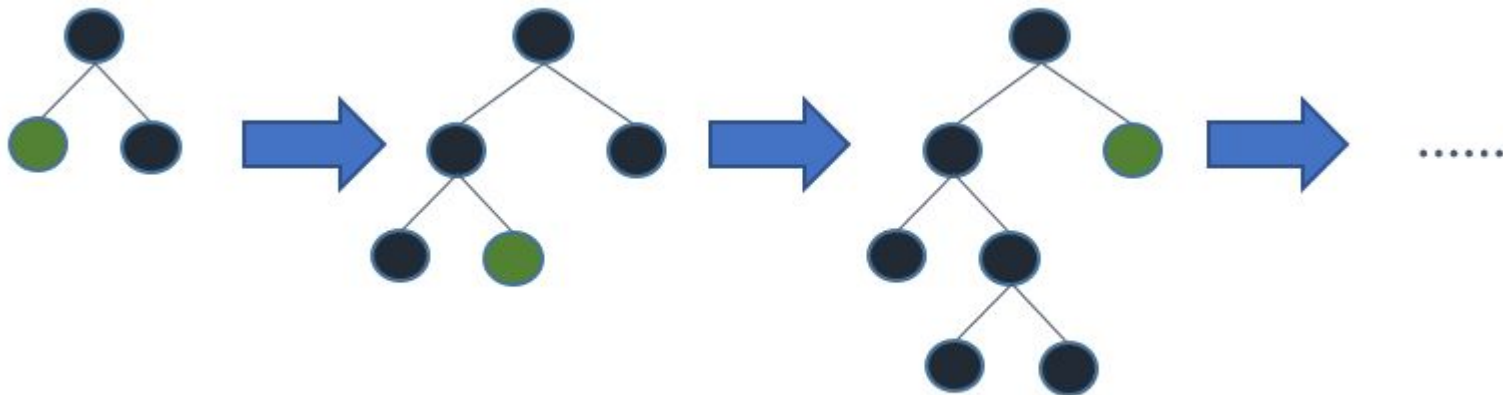
When to use:

- Medium datasets
- High accuracy needed
- Stable & reliable model

Light GBM:

Ultra fast. Leaf-wise tree growth.

Leaf-wise tree growth

CODE:
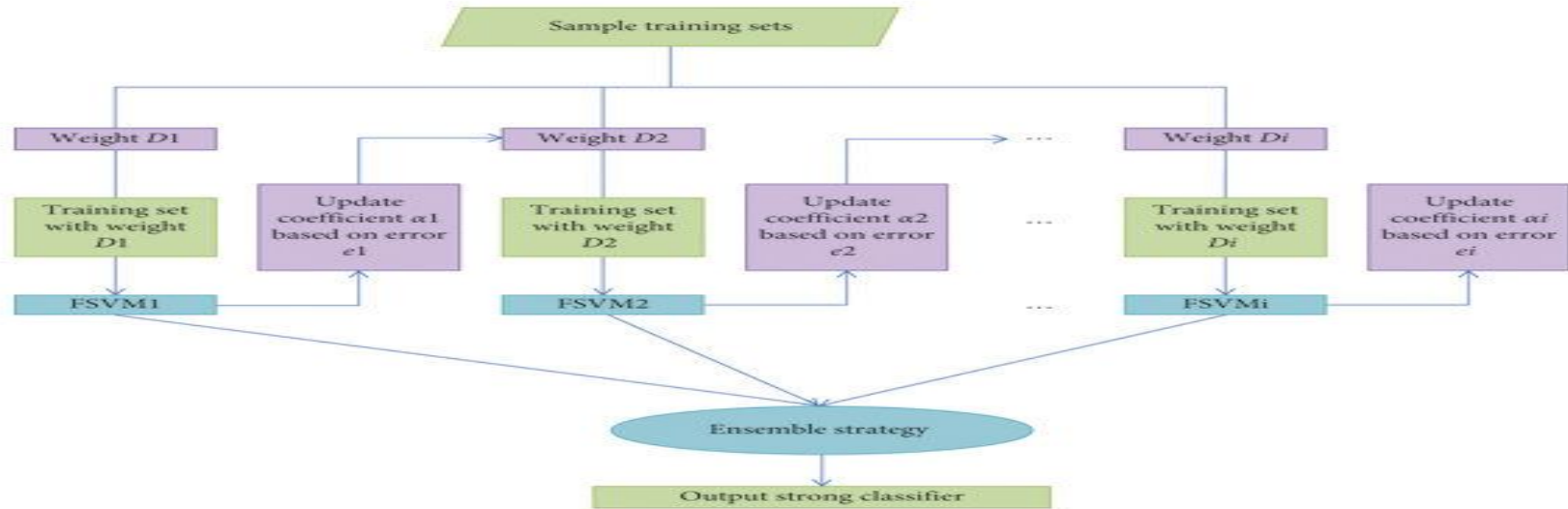
```
from lightgbm import LGBMRegressor


lgbm = LGBMRegressor(
    n_estimators=500,
    learning_rate=0.05,
    num_leaves=31,
    max_depth=7
)
```

When to use:

- Large datasets
- Time & memory matter
- Many categorical features

Ada Boost.RT:

It combines **many weak regression models**. Each new model focuses **more on the data points the previous models predicted badly.**

CODE:

from sklearn.ensemble import AdaBoostRegressor

```
regr = AdaBoostRegressor(random_state=0, n_estimators=100)
regr.fit(X, y)
AdaBoostRegressor(n_estimators=100, random_state=0)
```