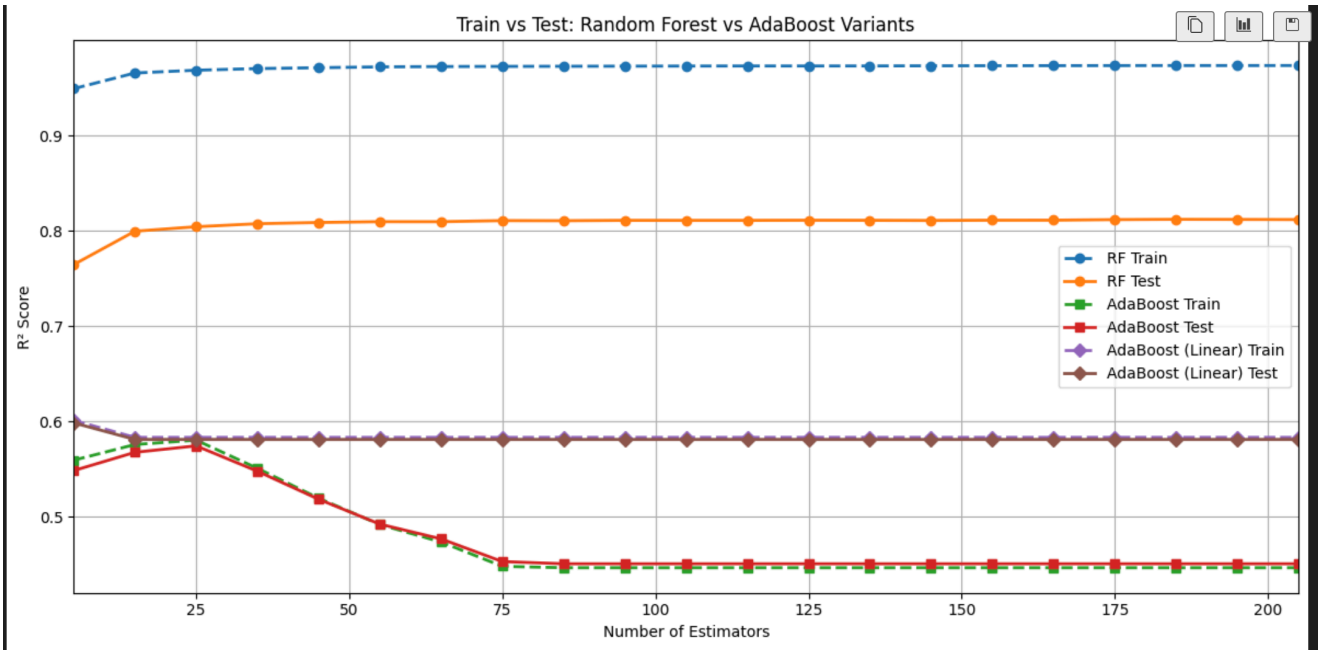
EXMPALE OF LEARNING FROM PAST YEAR

Note on Using AdaBoost in scikit-learn (Regression)

- **Issue Observed:**
  When using AdaBoostRegressor with default settings, both **train and test performance** may drop unexpectedly, even if other models are stable. This is **not typical overfitting behavior**.



- **Root Cause:**
  The default `base_estimator` for `AdaBoostRegressor` is a **DecisionTreeRegressor with** `max_depth=3` . This small tree depth can severely limit model capacity, especially if your other models (e.g., RandomForest) are much deeper.

- **Recommended Practice:**
  1. **Always specify your base estimator explicitly.**

```
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import AdaBoostRegressor

base_tree = DecisionTreeRegressor(max_depth=25)  # match depth with other models
model = AdaBoostRegressor(base_estimator=base_tree, n_estimators=100, random_state=42)
```

  2. **Ensure** `max_depth` **matches your other decision-tree-based models** if you want comparable performance.

3. **Check both train and test scores** to distinguish between underfitting (both scores low) and overfitting (train high, test low).

- **Key Takeaway:**
  - **Default parameters are rarely optimal.** Always inspect the `base_estimator` and its parameters.
  - For regression tasks with complex data, **increase tree depth** to allow AdaBoost to capture more patterns.