# Supervised Algorithms In Action

## Module 4: Regression vs. Classification Foundations

*Implementing Linear Regression & Random Forest Models*

# MODULE AGENDA

## Math Foundations

Decoding the divergence between predicting quantities and categorical labels.

## Model Execution

Scikit-Learn implementation for Linear Regression and Random Forests.

## Final Deliverables

Operational supervised models trained on high-fidelity custom datasets.

PART

ONE

# Mathematical Foundations

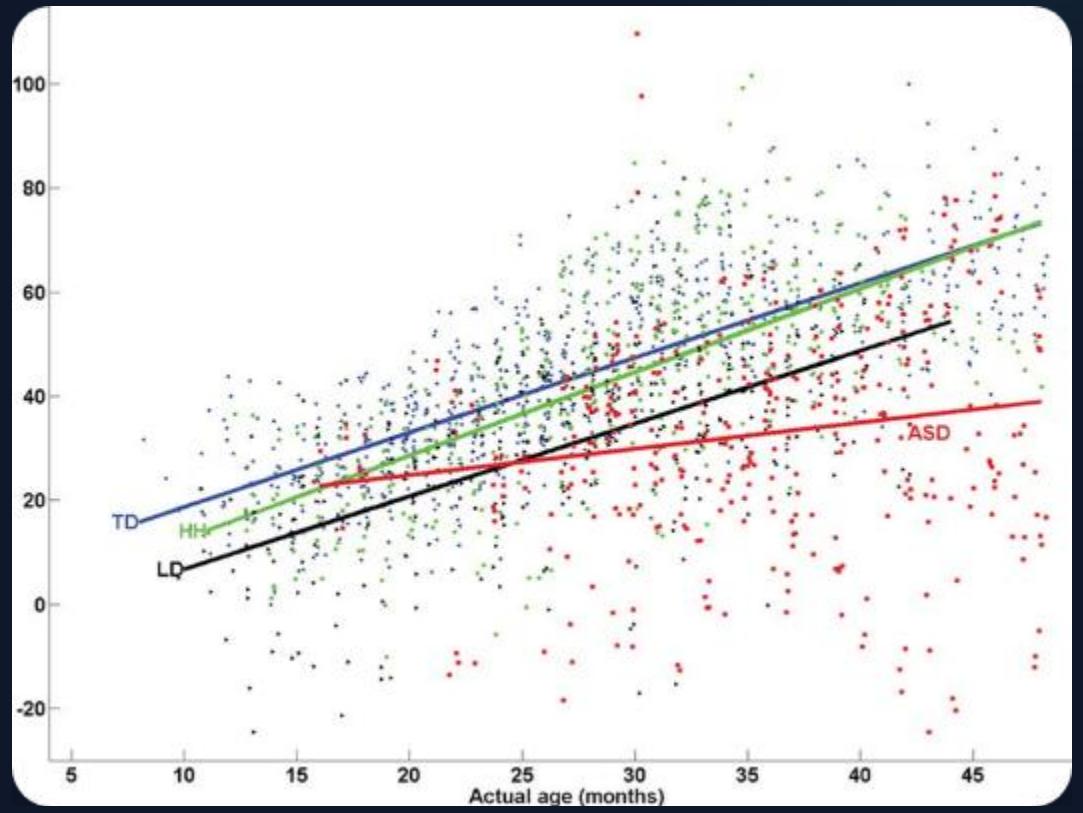
# THE CORE OF REGRESSION

## Numerical Prediction

Regression models approximate a mapping function from input variables (X) to continuous output variables (y).

$$MSE = \frac{1}{n} \sum_{i=1}^{n} y_i - \hat{y}_i^2$$

The objective is to minimize the Mean Squared Error (MSE) to find the line of best fit.



# CLASSIFICATION FRAMEWORK

## Discrete Labels

Predicting distinct groups. Unlike regression, the output is a class identifier (e.g., Cat vs. Dog, Spam vs. Safe).

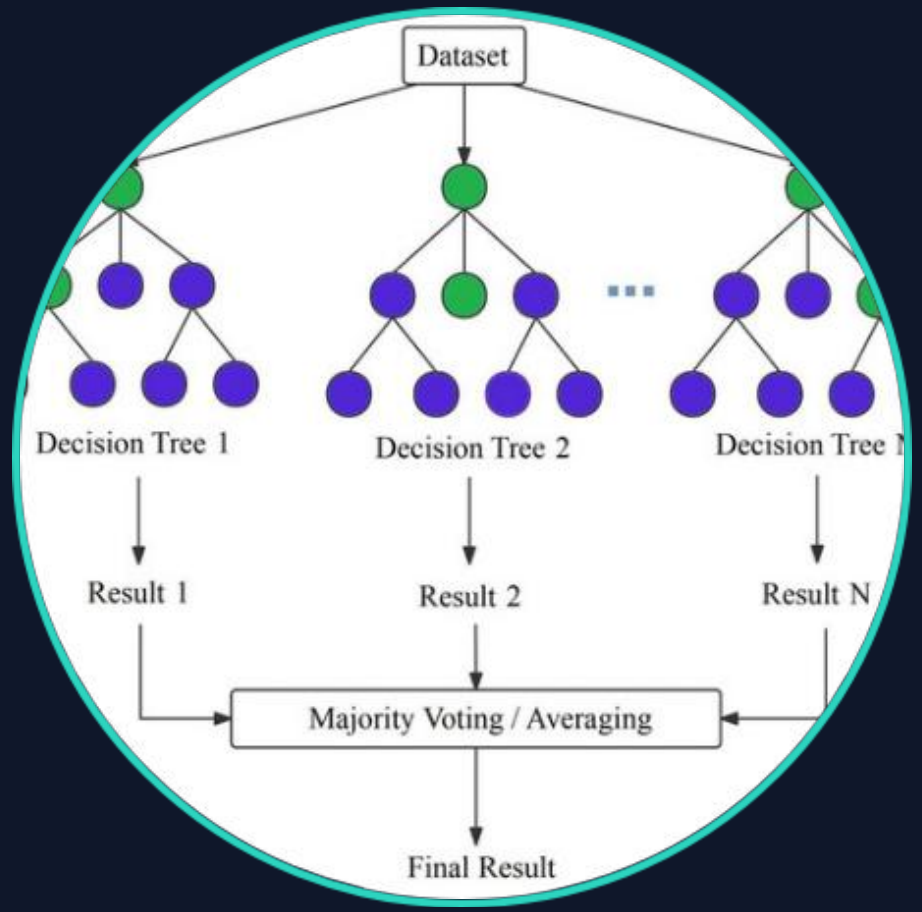
## Decision Boundaries

Algorithms learn to construct geometric boundaries that maximize the separation between labeled clusters.

# LINEAR REGRESSION WORKFLOW

- **Model Setup:** Instantiate the algorithm using LinearRegression() from sklearn.
- **Feature Selection:** Prepare a 2D matrix (X) for features and a 1D vector (y) for the target.
- **Optimization:** The model computes optimal coefficients (weights) using Ordinary Least Squares.

# ENSEMBLE LEARNING: RANDOM FOREST




## Strength in Numbers

The RandomForestClassifier(n_estimators=100) builds a forest of diverse decision trees.

By aggregating "votes" from 100 trees, it drastically reduces overfitting and handles non-linear relationships with high robustness.

PART

TWO

# Code & Deliverables

# THE LEARNING ENGINE


## Pattern Recognition

The .fit(X_train, y_train) method is where the mathematical magic happens. It exposes the algorithm to historical data to minimize loss and optimize parameters.

This single line of code initiates the training phase for both Linear Regression and Random Forests.

# METRIC EVALUATION

| Task Type | Standard Metric | Evaluation Target |
| --- | --- | --- |
| Regression | Mean Absolute Error (MAE) | Minimize distance to real values |
| Classification | Accuracy Score | Maximize correct label assignments |
| Robustness | R-Squared | Explain the variance in data |

# DELIVERABLE 1: PRICE PREDICTOR

## Linear Regression Model

Using custom real-estate datasets, we implement a model to predict housing prices based on square footage and location metadata.

**Status:** Operational. Captures linear trends to provide accurate numerical estimates for market valuation.

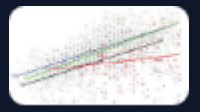
# Q&A

## Module 4: Supervised Algorithms in Action

Your operational models are the first step toward advanced AI proficiency.

```
foundation.fit(student_knowledge, success)
```

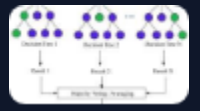
# IMAGE SOURCES



https://cdn.corporatefinanceinstitute.com/assets/multiple-linear-regression.png

Source: corporatefinanceinstitute.com

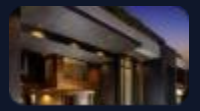
---



https://www.mdpi.com/forests/forests-15-02110/article_deploy/html/images/forests-15-02110-g001-550.jpg

Source: www.mdpi.com

---



https://www.buildofy.com/blog/content/images/2025/06/21.jpeg

Source: www.buildofy.com