Supervised Learning: The Big Idea

Data Science with Python – BSc Course

25–30 Minutes

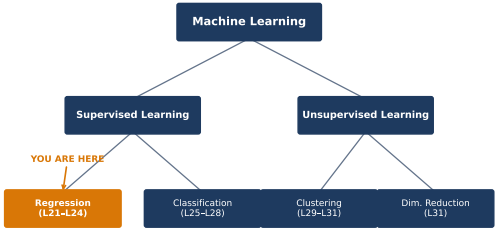

“If only past data could teach me...”
?
Stock Returns
Fraud?
Transactions

The Two Questions of Supervised Learning

Given labeled historical data, we can ask:

- **Labeled** = each example comes with the correct answer
- **How much?** Predict a number → **Regression**
- **Which one?** Assign a category → **Classification**

Read the chart: The taxonomy shows where regression and classification sit within ML.



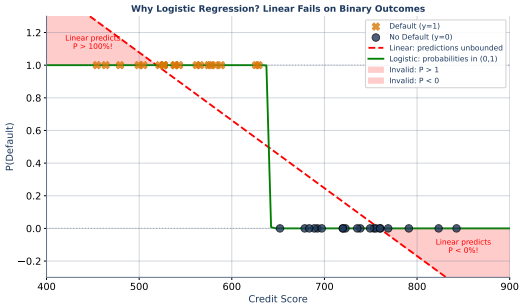
Supervised learning = learning from examples with known answers. Example: past stock prices paired with next-day returns.

Regression vs. Classification

Two flavors of prediction:

- **Regression:** Output is a number (price, return, temperature)
- **Classification:** Output is a label (fraud/legit, buy/sell, sector)

Read the chart: Left: regression fits a line through data. Right: classification draws a boundary between classes.



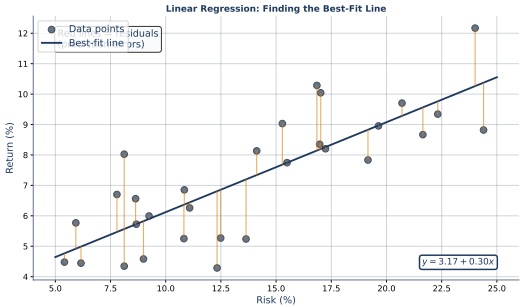
The output type determines which model family to use. Stock return forecasting = regression; default prediction = classification.

Linear Regression – Fit a Line Through Data

The simplest regression model:

- Find the straight line that best fits the data
- Minimize the distance between predictions and reality
- Workhorse of finance: factor models, risk, forecasting

Read the chart: Each dot is an observation. The line minimizes total squared distance to all points.



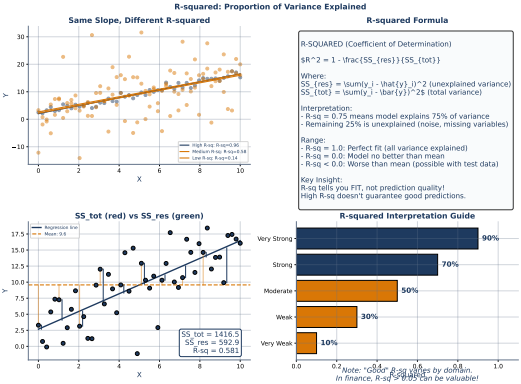
Linear regression is the foundation for nearly all regression methods. Example: predicting bond yields from macroeconomic indicators.

How Good Is Our Line? – R-Squared

R-squared measures explanatory power:

- Close to 1: model explains most variation
- Close to 0: barely better than guessing the mean
- In finance, even small R-squared can be profitable

Read the chart: High R-squared means points cluster tightly around the line; low R-squared means wide scatter.



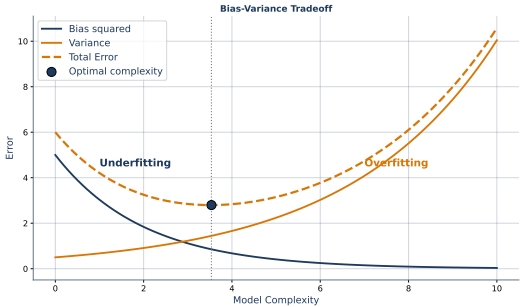
Finance insight: $R^2 = 0.05$ on daily stock returns is considered excellent – 5% explained is enough for a profitable strategy.

Overfitting – Memorizing vs. Learning

The central challenge of machine learning:

- **Underfitting:** Too simple, misses the pattern
- **Overfitting:** Memorizes noise, fails on new data
- **Bias** = oversimplifying; **Variance** = chasing noise

Read the chart: Left: too simple. Middle: just right. Right: memorized noise.



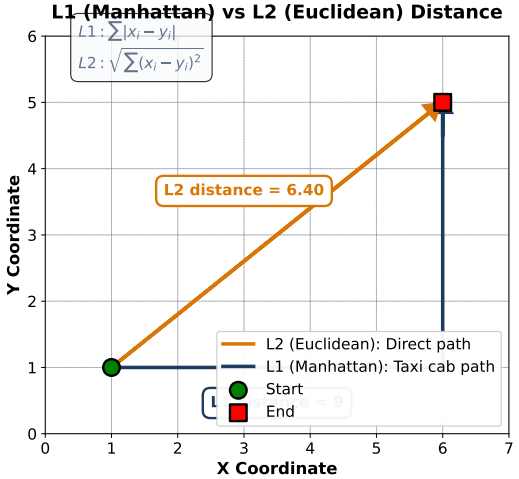
A model trained on 2020 stock data that fails in 2021 likely overfit to 2020-specific noise.

Regularization – Keeping Models Honest

Penalize complexity to prevent overfitting:

- **L1 (Lasso):** Pushes unimportant weights to exactly zero
- **L2 (Ridge):** Shrinks all weights toward zero evenly

Read the chart: The diamond (L1) forces weights to hit zero at corners. The circle (L2) shrinks evenly.



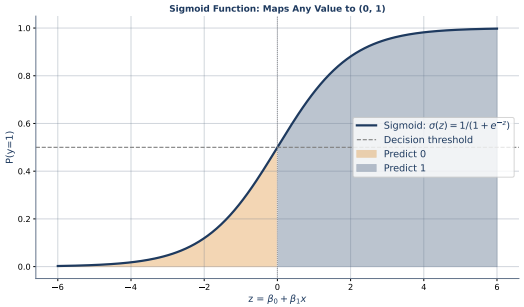
Example: Lasso drops 8 of 10 financial features to zero, keeping only the 2 that matter for predicting returns.

Logistic Regression – The S-Curve for Yes/No

Classification with a smooth probability curve:

- Maps any input to a probability between 0 and 1
- The sigmoid creates the characteristic S-shape
- Predict "yes" when probability exceeds a threshold

Read the chart: The S-curve maps any input to a probability. Flat at extremes, steep in the middle.



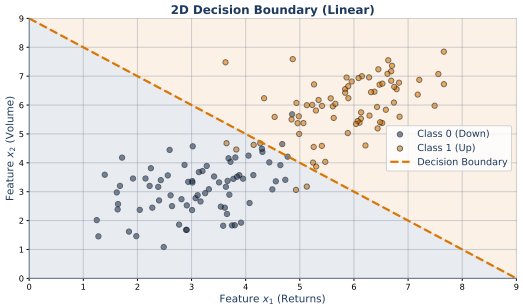
Despite its name, logistic regression is a classification method. Example: if P(default) ¿ 0.5, flag the loan.

Decision Boundary – Where the Model Draws the Line

The boundary separating predicted classes:

- Logistic regression creates a linear boundary
- Points on one side are predicted positive, other negative
- The boundary position depends on learned weights

Read the chart: The diagonal line separates predicted classes. Colors show which side wins.



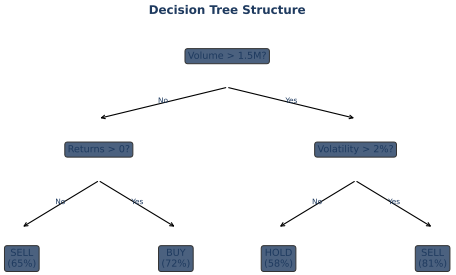
Example: income ¿ $50k AND debt ratio ¡ 0.4 = “approve loan”; otherwise = “deny.”

Decision Trees – If-Then Rules

Split data with a sequence of questions:

- Each node asks a yes/no question about a **feature**
- Data flows left or right based on the answer
- Leaves contain the final prediction

Read the chart: Follow the tree from top to bottom. Each branch is a yes/no question about a feature.



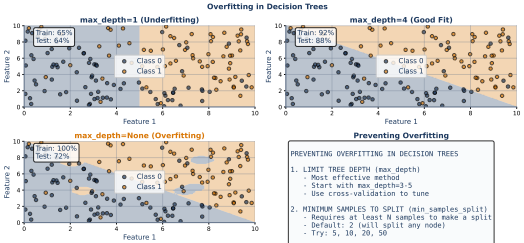
Example: "Is P/E ratio ¡ 15?" Yes → value stock. No → "Is revenue growing?" Yes → growth stock.

Tree Depth – Shallow vs. Deep

Tree complexity controls the bias-variance tradeoff:

- **Shallow trees:** Simple rules, may underfit
- **Deep trees:** Complex rules, risk overfitting

Read the chart: Training accuracy rises with depth. Validation accuracy peaks then drops – the signature of overfitting.



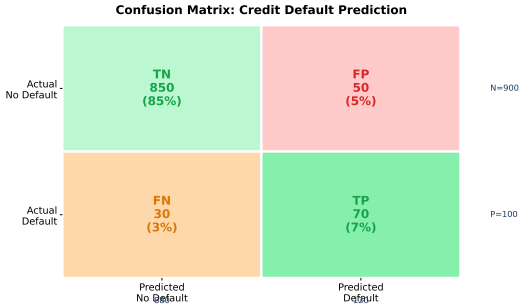
Depth = 2 uses 3 questions (simple); depth = 20 memorizes every training case. Typical sweet spot: depth 4–8.

Confusion Matrix – TP, FP, TN, FN

The four outcomes of binary classification:

- **TP / TN:** Correct predictions
- **FP** (false alarm): Predicted yes, actually no
- **FN** (missed): Predicted no, actually yes

Read the chart: The 2×2 grid shows all four outcomes. Diagonal = correct, off-diagonal = errors.



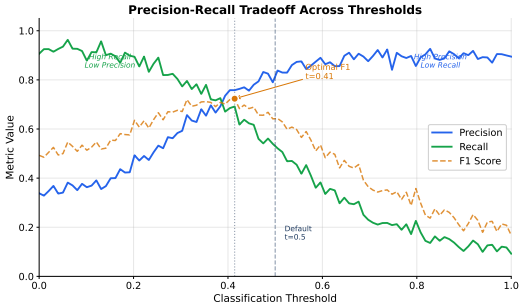
100 transactions: 90 TN + 5 TP + 3 FP + 2 FN = 95% accuracy but 2 missed frauds.

Precision vs. Recall – The Tradeoff

Two complementary views of classifier quality:

- **Precision:** Of predicted positives, how many are correct?
- **Recall:** Of actual positives, how many did we find?
- Improving one typically hurts the other

Read the chart: As the threshold moves right, precision rises but recall falls – the fundamental tradeoff.



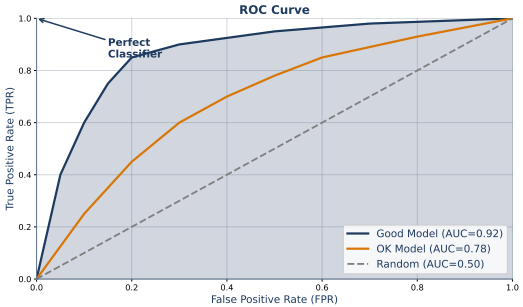
Fraud detection needs high recall (catch every fraud); spam filters need high precision (don't block real email).

ROC Curve – Classifier Performance at a Glance

Visualize performance across all thresholds:

- X-axis: false positive rate; Y-axis: true positive rate
- AUC summarizes overall quality in one number
- AUC = 0.5 means random guessing; 1.0 is perfect

Read the chart: The curve bows toward the top-left corner. More bow = better model.



Model A (AUC = 0.92) outranks Model B (AUC = 0.78) at every threshold choice.

When to Use What

Choosing the right supervised learning method:

Method	Best For	Watch Out For
Linear Regression	Continuous targets, interpretability	Assumes linear relationship
Ridge / Lasso	Many features, multicollinearity	Requires tuning penalty strength
Logistic Regression	Binary outcomes, probability estimates	Linear decision boundary only
Decision Trees	Non-linear patterns, explainability	Overfits without depth limits
Random Forests	Robust predictions, feature importance	Slower, less interpretable

Start simple, add complexity only when the data demands it.

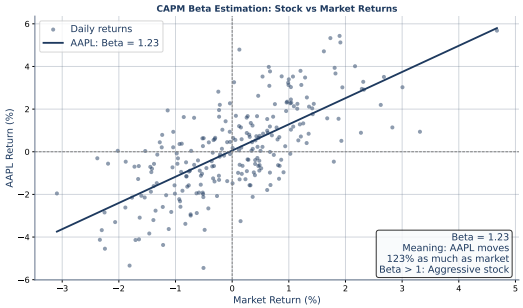
Random Forest = many decision trees voting together. Multicollinearity = features duplicating information (e.g. revenue and sales).

Finance Application – CAPM Beta

Linear regression in action:

- Regress stock returns on market returns
- The slope is beta – the stock's market sensitivity
- One of the most widely used models in finance

Read the chart: Each dot is one day's returns. The slope of the line IS the CAPM beta.



Beta ¿ 1 means the stock amplifies market moves; beta ¡ 1 means it dampens them. AAPL beta $\approx$ 1.2.

Finance Application – Fraud Detection and Class Imbalance

Real-world classification is rarely balanced:

- Fraud is rare: 99.9% legit, 0.1% fraudulent
- Predicting “not fraud” always gets 99.9% accuracy
- **Resampling** and **cost-sensitive learning** fix it

Read the chart: Different resampling strategies compared. SMOTE and cost-sensitive learning improve recall.

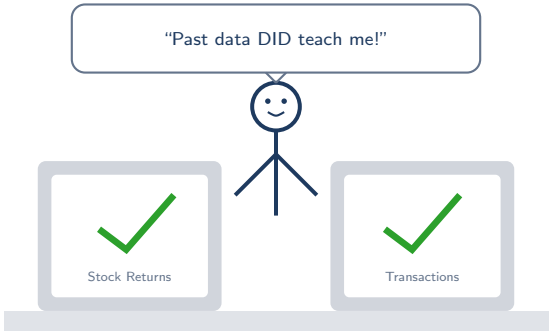
Accuracy is misleading with imbalanced classes – use precision, recall, and AUC instead.

Key Takeaways

Six things to remember about supervised learning:

1. Supervised learning trains on labeled examples with known outcomes
2. **Regression** predicts numbers; **Classification** predicts categories
3. Overfitting is the central enemy – regularization is the cure
4. Decision trees split data with interpretable if-then rules
5. Always evaluate with the right metric: check precision AND recall
6. Finance uses supervised learning everywhere – from CAPM to fraud detection

Supervised learning is the most widely used branch of machine learning in industry and finance.


“Past data DID teach me!”
Stock Returns
Transactions