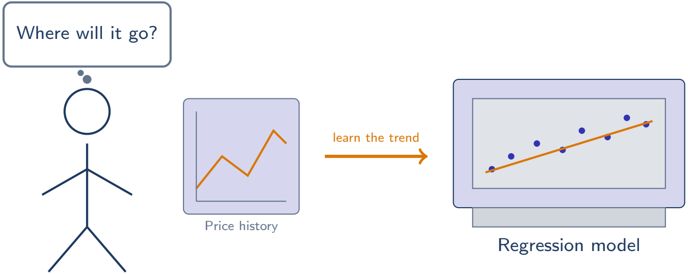
Regression in Supervised Learning

Data Science with Python – BSc Course

90 Minutes

The Prediction Challenge



"What if we could learn the trend?"

Every price forecast, risk estimate, and return prediction is a regression problem.

Learning Objectives

After this lecture you will be able to:

1. **Explain** how OLS finds the best-fit line by minimising squared errors
2. **Compare** OLS, Ridge, and Lasso regression and when each applies
3. **Evaluate** models using MSE, RMSE, R^2, and adjusted R^2
4. **Apply** walk-forward cross-validation for time-series data
5. **Justify** factor model selection for return attribution

Bloom's taxonomy: Remember → Understand → Apply → Analyze → Evaluate

These five objectives map to Bloom's levels 2–5.

Bridge: From Statistics to Prediction

What you already know

- Mean, variance, standard deviation
- Correlation between two variables
- Scatter plots and trend lines

What this lecture adds

- Prediction with confidence bounds
- Regularization to prevent overfitting
- Factor models for return attribution

Seven stops from fitting a line to understanding its limits.

Given past data, can we predict the future—and know when we cannot?

Three sub-questions guide this lecture:

1. How does OLS find the "best" line, and when does it fail?
2. How do Ridge and Lasso protect against overfitting?
3. How do we measure whether our predictions are useful?

Prediction without understanding uncertainty is speculation, not science.

Three cautionary tales from finance:

- **LTCM (1998):** Over-fitted bond convergence models lost $4.6 billion in weeks
- **JPMorgan VaR (2012):** Flawed risk regression underestimated losses by $6 billion ("London Whale")
- **Quant crisis (2007):** Crowded factor models unwound simultaneously

Definition: *Regression* estimates a continuous target y from features $\mathbf{x}$ by fitting $\hat{y} = f(\mathbf{x})$.

Every model is wrong; the question is whether it is useful—George Box.

The Cost of Getting It Wrong

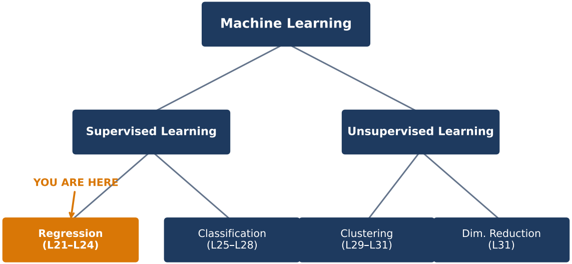
Prediction errors are asymmetric in finance:

Error type	**Example**	**Business cost**
Overestimate return	Allocate to losing asset	Direct capital loss
Underestimate return	Miss profitable trade	Opportunity cost
Overestimate risk	Excess capital reserve	Reduced leverage, lower ROE
Underestimate risk	Insufficient reserves	Regulatory penalty, ruin

The choice of error metric must match the business cost structure.

Metric selection is a business decision, not a statistical one.

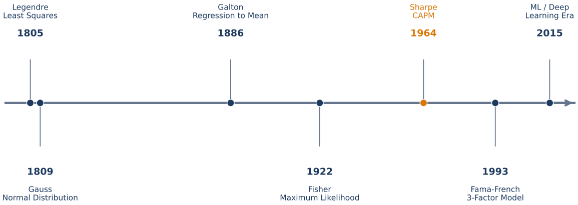
What you see: regression highlighted within the machine-learning taxonomy.



Regression is the continuous-output branch of supervised learning.

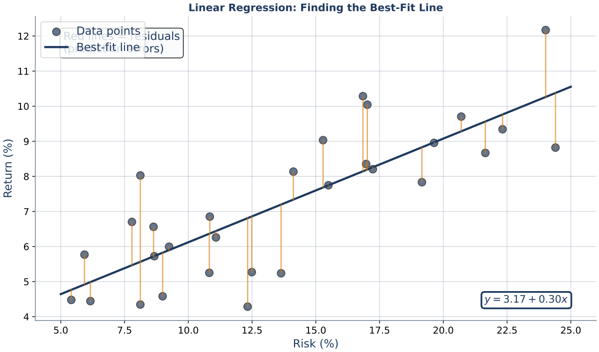
A Brief History of Regression

What you see: from Gauss (1805) through Tikhonov to modern machine learning.



Least squares is over 200 years old—still the workhorse of quantitative finance.

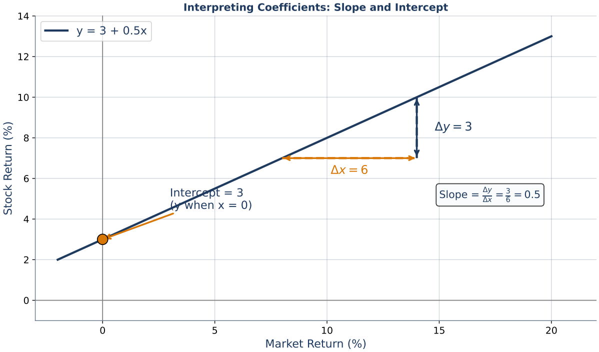
What you see: a scatter plot of data points with a fitted regression line.



One line, two parameters—this is where every regression journey begins.

Reading the Coefficients

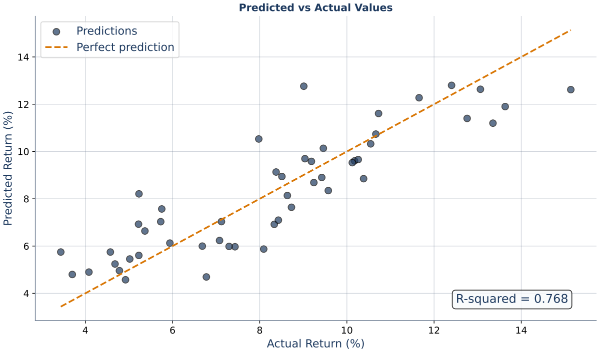
What you see: the meaning of slope (rate of change) and intercept (baseline).



Slope = "for each unit increase in x, y changes by β_1." Intercept = y when $x = 0$.

Predicted vs. Actual

What you see: the 45-degree line where perfect predictions would fall.



Deviation from the diagonal reveals systematic bias.

From Intuition to Algorithm

What we have established so far:

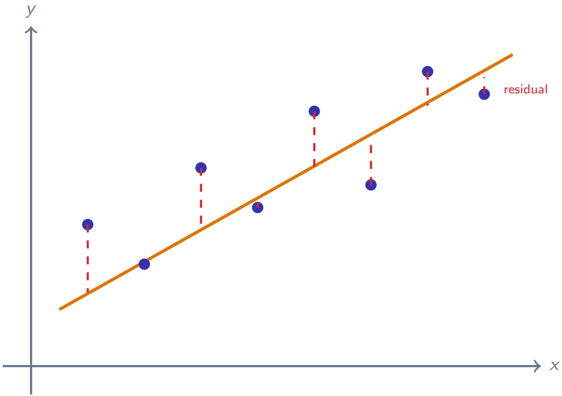
- Regression predicts a continuous value from features
- Coefficients have direct financial interpretation (beta, alpha)
- Prediction errors carry real business costs

Next question: *How does the algorithm actually find the best line?*

The answer is **Ordinary Least Squares (OLS)**—minimise the sum of squared residuals.

Regression is everywhere in finance. But how does OLS find the line?

OLS: Minimising Squared Errors

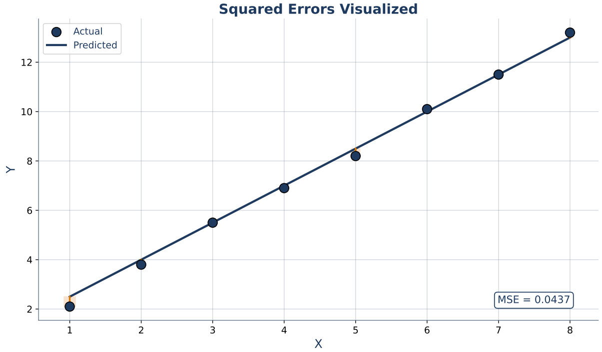


$$\min_{\beta_0,\,\beta_1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad \hat{y}_i = \beta_0 + \beta_1 x_i$$

OLS minimises the total area of the red squares—hence “least squares.”

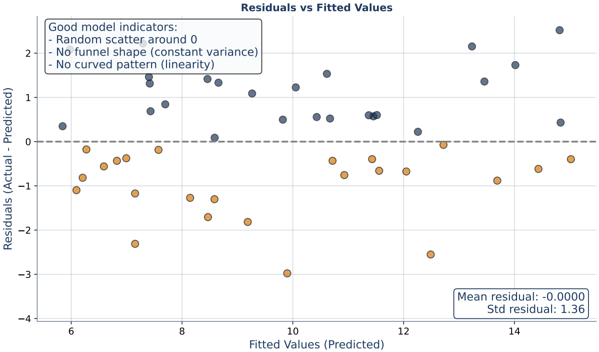
Squared Errors as Literal Squares

What you see: each residual drawn as an area square—larger errors dominate the loss.



Squaring penalises large errors disproportionately—one big miss outweighs many small ones.

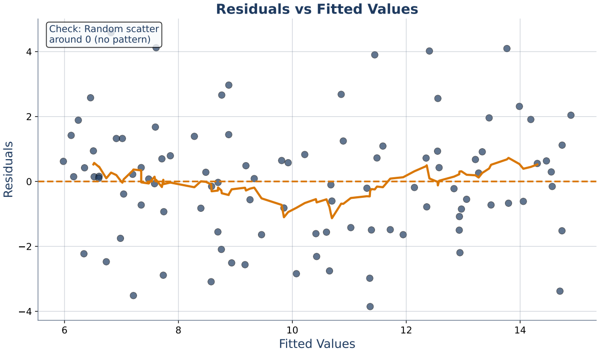
What you see: residuals scattered around zero indicate a well-specified model.



Patterns in residuals signal missing variables or wrong functional form.

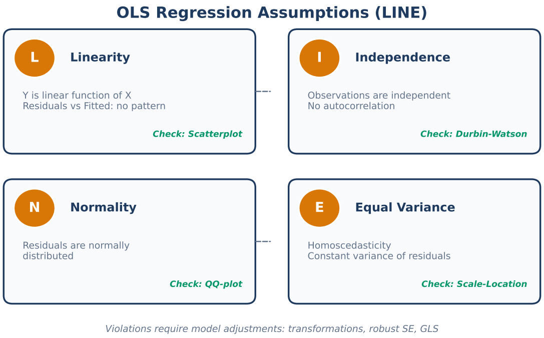
Residuals vs. Fitted Values

What you see: random scatter means OLS assumptions hold; structure means trouble.



A fan shape indicates heteroscedasticity; a curve indicates non-linearity.

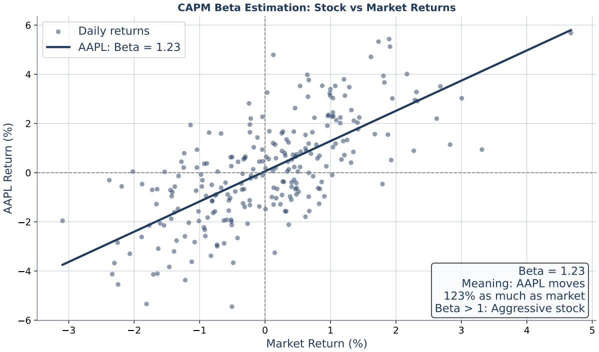
What you see: the four key assumptions visualised side by side.



Linearity, independence, homoscedasticity, normality—violate any and OLS estimates degrade.

Finance Application: CAPM Beta

What you see: a stock's excess return regressed on the market—slope is beta.



$R_i - R_f = \alpha + \beta(R_m - R_f) + \varepsilon$. **Beta measures systematic risk.**

OLS Recap and What Comes Next

What OLS gives us:

- Closed-form solution: $\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$
- Unbiased estimates under Gauss–Markov conditions
- Direct interpretation: slope, intercept, R^2

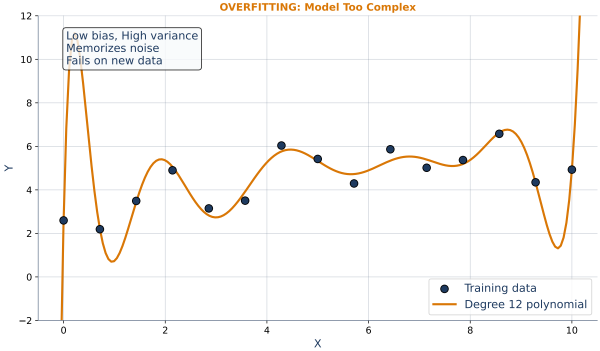
The problem: When we add many features, OLS fits noise as eagerly as signal.

Next: What happens when the model fits too well?

Overfitting turns a good model into an expensive random-number generator.

Overfitting: Memorising Noise

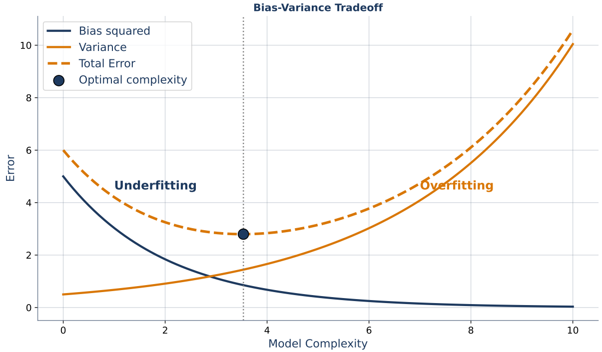
What you see: a model that passes through every training point but fails on new data.



A perfect training score is a warning, not a victory.

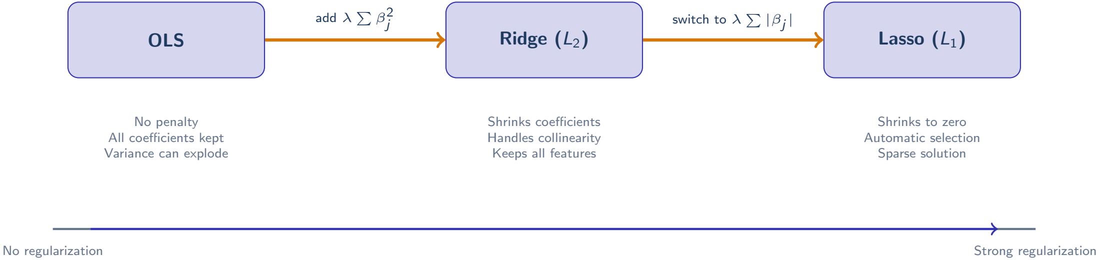
Bias–Variance Tradeoff

What you see: as model complexity grows, bias falls but variance rises.



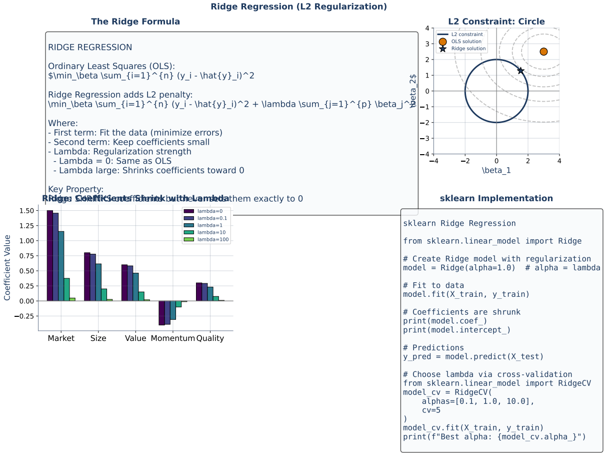
The sweet spot lies where total error (bias^2 + variance) is minimised.

The Regularization Spectrum



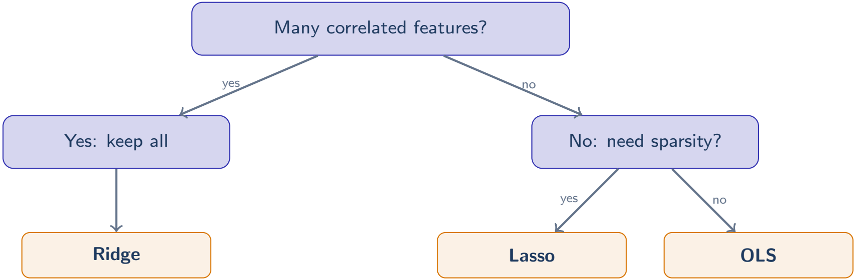
Regularization trades a little bias for a large reduction in variance.

What you see: Ridge shrinks all coefficients toward zero but keeps every feature.



Ridge is ideal when many features contribute small, correlated effects.

Think–Pair–Share: Ridge or Lasso?

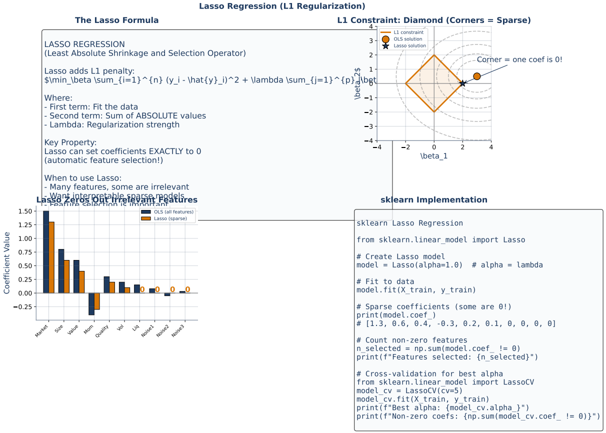


Prompt: You have 50 correlated sector factors. Ridge or Lasso? Discuss for 2 minutes.

Correlated features → Ridge keeps all; irrelevant features → Lasso drops them.

Lasso Regression: Select and Shrink

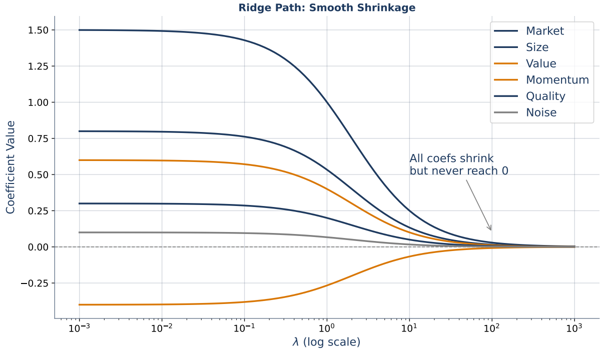
What you see: Lasso drives irrelevant coefficients exactly to zero.



Lasso performs automatic feature selection—fewer features, simpler model.

Coefficient Paths Under Ridge

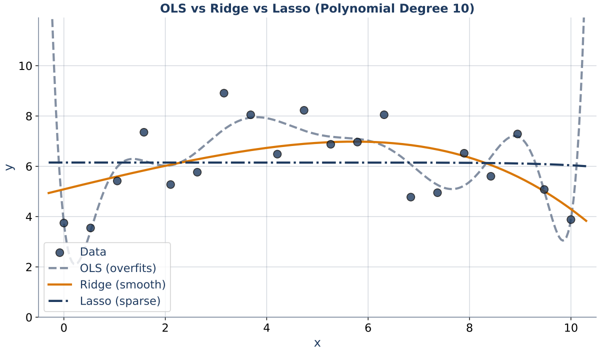
What you see: as λ increases, coefficients shrink smoothly toward zero.



No coefficient reaches zero—Ridge never eliminates a feature entirely.

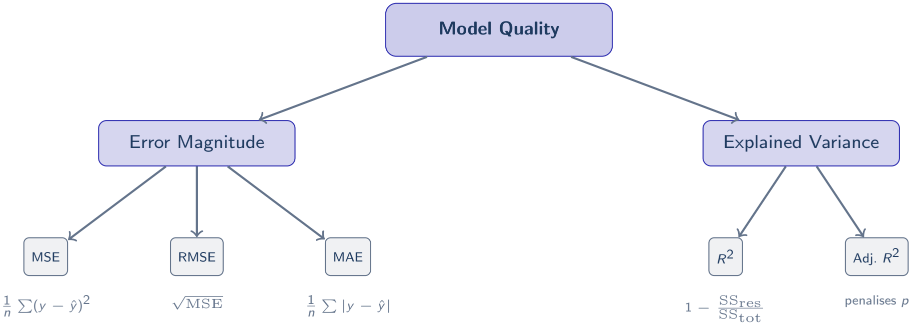
Side-by-Side: OLS, Ridge, Lasso

What you see: how the three methods produce different coefficient profiles on the same data.



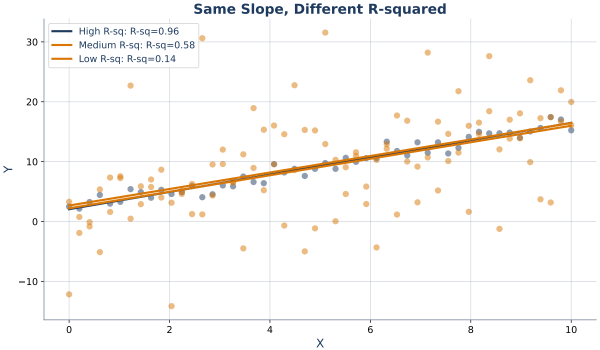
Same data, three philosophies: unpenalised, shrunk, sparse.

Regression Metrics at a Glance



RMSE speaks in the same units as y; R^2 tells you what fraction of variance you explain.

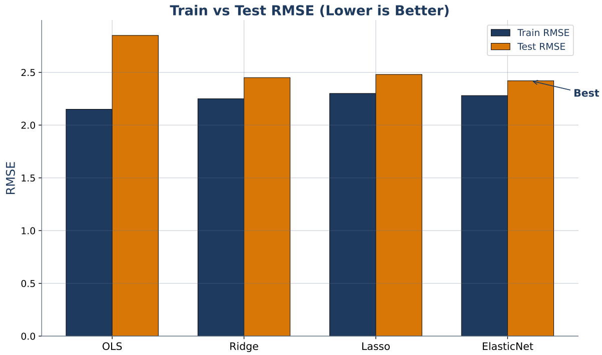
What you see: three models with $R^2 = 0.3$, 0.7, and 0.95—tighter scatter means better fit.



In finance, $R^2 = 0.05$ on daily returns can still be profitable.

Train vs. Test RMSE

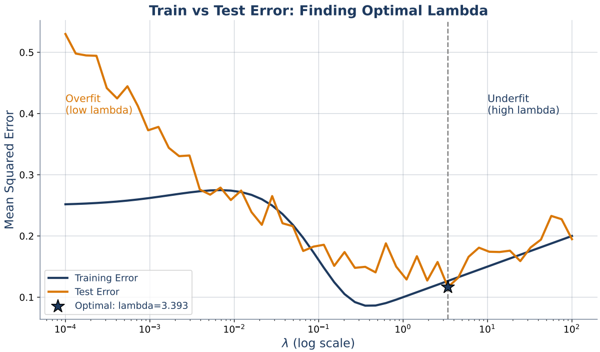
What you see: a growing gap between training and test error signals overfitting.



If training RMSE keeps dropping but test RMSE rises, stop adding complexity.

The U-Shaped Test Error Curve

What you see: test error first falls then rises—the classic overfitting signature.



Optimal complexity sits at the bottom of the U—not too simple, not too complex.

Why Standard CV Fails for Finance

The problem: k-fold cross-validation shuffles time.

- Fold 3 may train on 2024 data and test on 2022 data
- This is "future peeking"—test performance is artificially high
- Financial returns are serially correlated and regime-dependent

The solution: Respect temporal order.

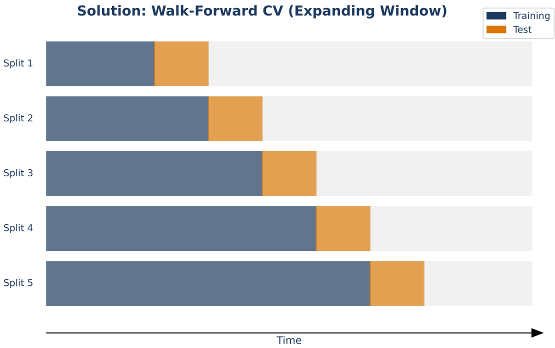
- Always train on the past, test on the future
- Expanding or rolling windows preserve causality

Next: walk-forward cross-validation.

In finance, time is not exchangeable—validation must respect the arrow of time.

Walk-Forward Cross-Validation

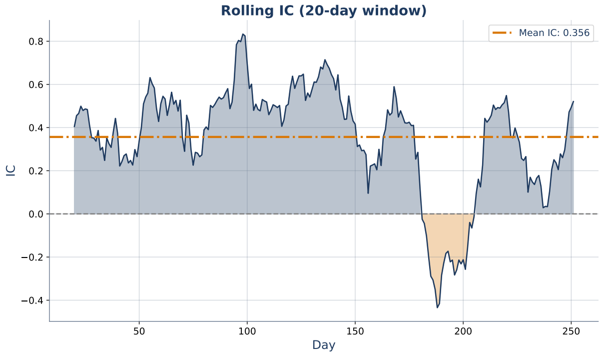
What you see: expanding training windows that always predict forward in time.



Each fold adds one more period of history—the model sees progressively more data.

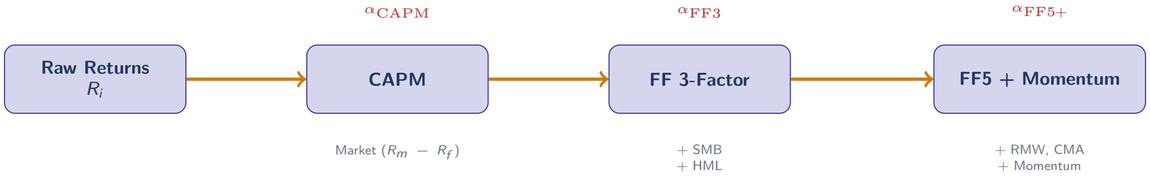
Information Coefficient Over Time

What you see: rolling IC shows whether your model's predictive power is stable or decaying.



A decaying IC means the signal is crowded or the regime has changed.

Factor Models: Building Up Complexity



Each model absorbs more return variation. What remains is alpha—skill or luck?

Adding factors shrinks alpha; only genuine skill survives a richer model.

Discussion: Skill or Factor Exposure?

Scenario: A small-cap fund reports these results:

- CAPM alpha: 3.2% per year (statistically significant)
- Fama–French 3-factor alpha: 0.5% per year (not significant)

Questions for discussion (3 minutes):

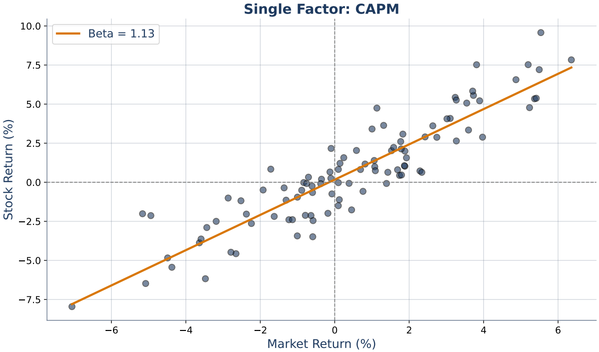
1. Where did the "alpha" go when we added SMB and HML?
2. Is the fund manager skilled, or just tilted to small-cap value?
3. Should investors pay active fees for factor exposure they can get cheaply?

This is both a statistical and an ethical question.

Most "alpha" disappears when the right factors are included—Fama & French (1993).

CAPM in Practice

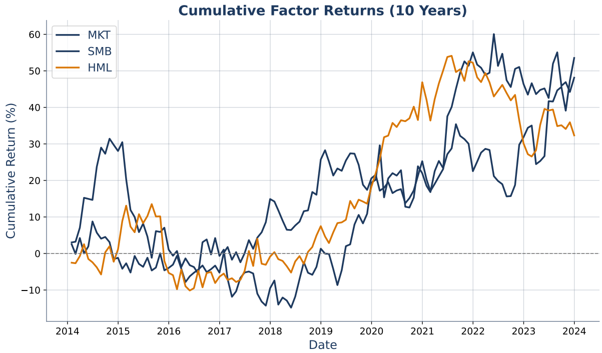
What you see: a stock's returns regressed on the market—slope is beta, intercept is alpha.



Alpha above zero suggests outperformance after adjusting for market risk.

Factor Performance Over Time

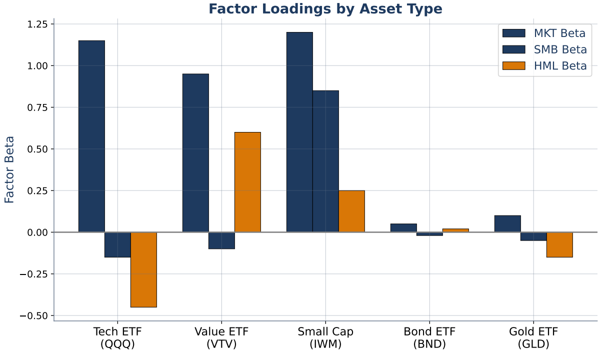
What you see: cumulative returns of major factors—some persist, others decay.



Value (HML) and momentum have delivered long-run premia with different timing.

Factor Fingerprints

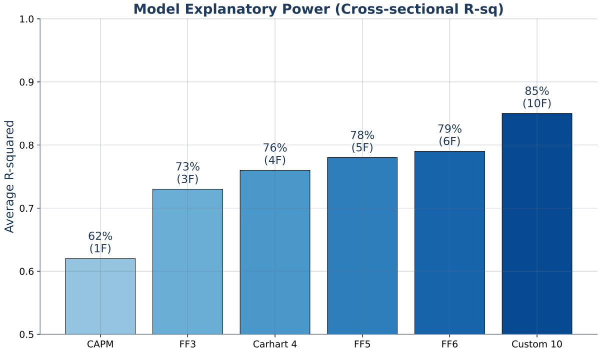
What you see: each asset's exposure (loading) on multiple factors—its "fingerprint."



Two assets can have similar returns but very different factor profiles.

More Factors, Better R^2?

What you see: R^2 increases as we move from CAPM to multi-factor models.



Higher R^2 means more return variation explained—but watch for overfitting with too many factors.

Four Limitations to Remember

No model is complete. Keep these in mind:

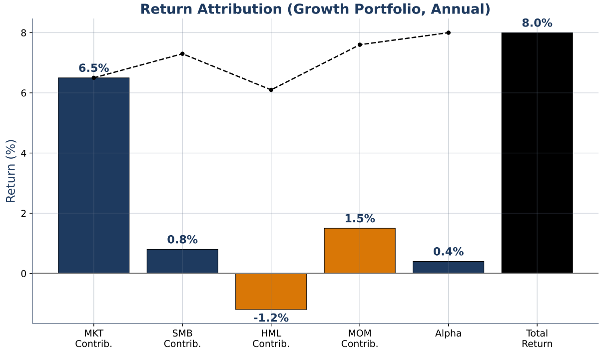
1. **Linearity:** OLS assumes a linear relationship—if the true function is curved, the fit is biased
2. **Stationarity:** Financial relationships change over time; a model fit on 2015 data may fail in 2025
3. **Multicollinearity:** Highly correlated features inflate coefficient variance and make interpretation unreliable
4. **Outliers:** Squared-error loss gives extreme observations outsized influence

Knowing when your model breaks is as important as knowing when it works.

Every assumption violation is a source of risk—quantify it before you trade on it.

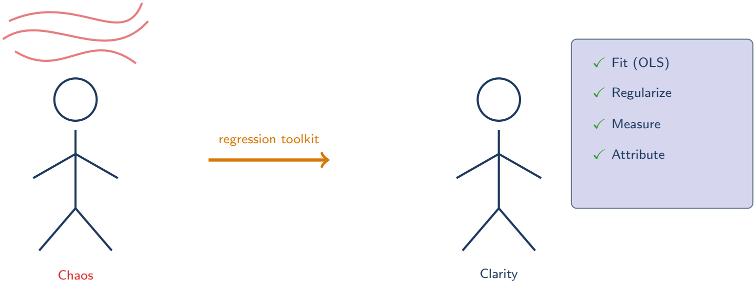
Return Attribution: Where Did Performance Come From?

What you see: a return decomposition showing each factor's contribution to total return.



Attribution answers the fund board's question: "Was it skill or exposure?"

From Confusion to Clarity



"Prediction, not prophecy."

A disciplined regression workflow turns market noise into actionable insight.

Five Takeaways

1. **OLS is the foundation:** minimise squared errors, check assumptions, interpret coefficients
2. **Regularization prevents overfitting:** Ridge for correlated features, Lasso for sparsity
3. **Metrics must match the question:** RMSE for magnitude, R^2 for explained variance, MAE for robustness
4. **Time-aware validation is non-negotiable:** walk-forward CV, rolling windows, never shuffle dates
5. **Know your model's boundaries:** linearity, stationarity, collinearity, and outlier sensitivity

Regression gives you a prediction and a framework for questioning it.