Lesson 21: Linear Regression

Data Science with Python – BSc Course

Data Science Program

BSc Course

45 Minutes

The Story of Regression

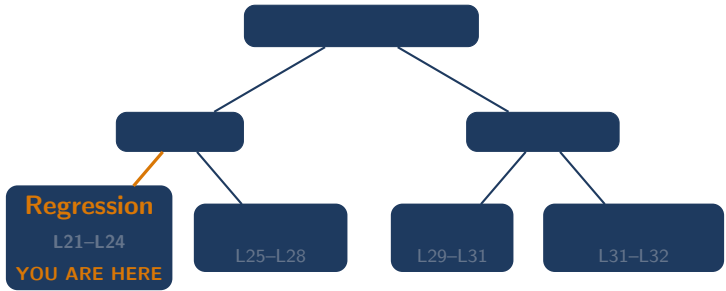
A 150-Year Quest for the Best-Fit Line

In 1886, Francis Galton measured sweet pea seeds and discovered something surprising: tall parents tend to have *shorter* children, and short parents have *taller* children. He called it **"regression to the mean."**

But the mathematics behind fitting a line started even earlier...

Legendre and Gauss fought over who invented least squares. Galton gave regression its name.

Where Are We? The ML Map



We are beginning **Module 4: Supervised Learning – Regression.** Over the next four lessons, you will learn to predict continuous values, regularize models, and evaluate predictions.

Regression predicts numbers; classification predicts categories. Both learn from labeled data.

Learning Objectives

The Problem: A portfolio manager needs to understand how stocks respond to market movements. How do we quantify systematic risk?

After this lesson, you will be able to:

- Understand OLS estimation and the least squares principle
- Fit linear models using sklearn's `LinearRegression`
- Interpret coefficients (slope as beta, intercept as alpha)
- Estimate CAPM beta to classify stocks by risk profile

Finance Application: Stock classification for portfolio construction

Math Notation Primer

Symbols You'll See in This Module:

- $\sum$ (sigma) = "add up all of these" – e.g., $\sum_{i=1}^{3} x_i = x_1 + x_2 + x_3$
- x_i = "the i-th value" – subscript i is an index (1st, 2nd, 3rd...)
- $\bar{x}$ (x-bar) = mean of x – e.g., $\bar{x} = \frac{x_1+x_2+x_3}{3}$
- $\hat{y}$ (y-hat) = predicted value of y (the "hat" means estimate)

Greek Letters:

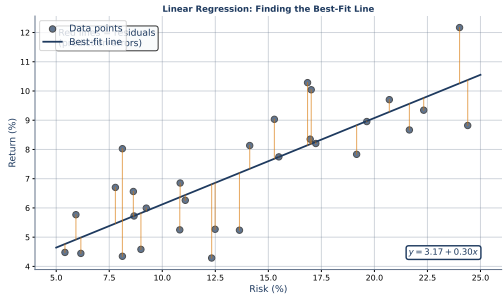
- β (beta) = slope coefficient – pronounced "BAY-tuh"
- α (alpha) = intercept – pronounced "AL-fuh"
- λ (lambda) = regularization strength – pronounced "LAM-duh"

Review L13 for variance (σ^2) and L16 for covariance formulas

The Core Question

Look at this scatter plot. What line would YOU draw?

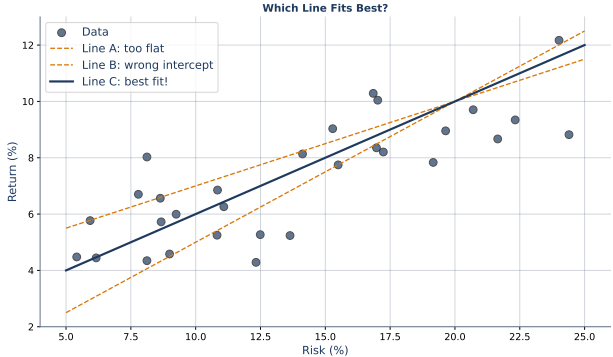
Every dot is one day: x = market return, y = stock return. There is clearly a relationship — but how do we find the *best* line through this cloud?



Take 30 seconds. Imagine drawing a line. Where would you place it?

Which Line Fits Best?

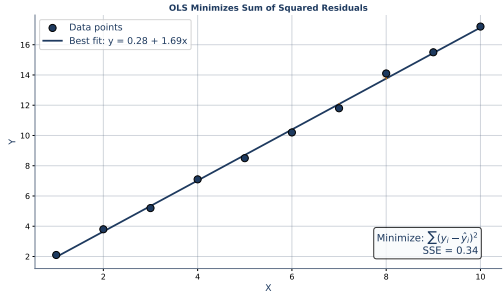
Three candidate lines through the same data — all look plausible. How do we pick **the** best one? We need a precise definition of "best."



Without a rule, different people draw different lines. OLS gives a unique answer.

The Answer: Minimize Squared Errors

OLS: The "best" line minimizes **total squared vertical distance.** Why squared? (1) Makes errors positive, (2) Penalizes large errors severely, (3) Has a unique closed-form solution.



OLS: the same method Legendre published in 1805 and Gauss claimed he knew first

Building the Formula: Intuition

What Does the Slope Formula Actually Calculate?

$$\beta_1 = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2} = \frac{\text{Cov}(X, Y)}{\text{Var}(X)}$$

- **Numerator** = covariance: "When X goes up, does Y go up too?"
- **Denominator** = variance: "How spread out is X?"
- **Ratio** = "For each unit X moves, how much does Y respond?"

Plain English: Slope = Responsiveness. A slope of 1.5 means Y moves 1.5 units for every 1 unit change in X.

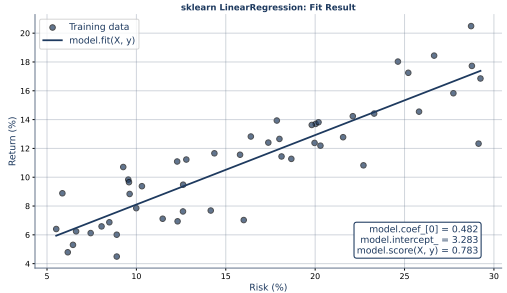
The OLS slope is simply: covariance(X,Y) / variance(X). Review L13 and L16.

From Math to Code: sklearn

Three Lines of Python — That's It

- `from sklearn.linear_model import LinearRegression`
- `model = LinearRegression().fit(X, y)`
- `predictions = model.predict(X_new)`

Access results: `model.coef_[0]` → slope, `model.intercept_` → intercept

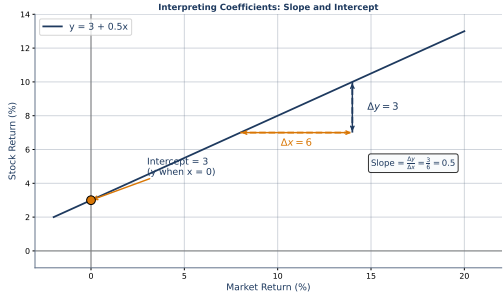


Pattern: fit(X, y) then predict(X_new) — this same API works for ALL sklearn models

Reading the Coefficients

What Do the Numbers Mean?

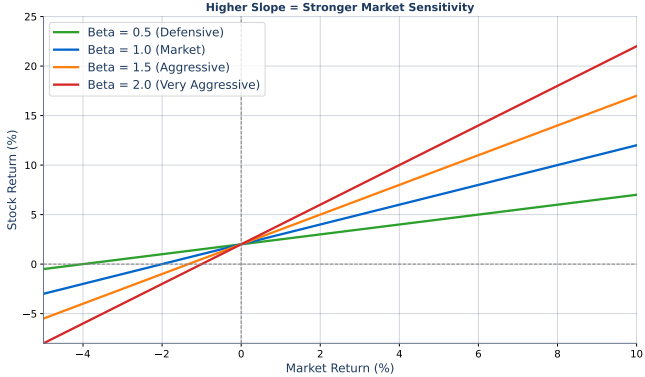
- **Slope (β_1):** For each 1% market move, stock moves β_1%
- **Intercept (β_0):** Stock return when market return is zero
- Example: $\beta_1 = 1.5$, market rises 2% $\Rightarrow$ stock rises $1.5 \times 2\% = 3\%$



Finance translation: Slope = beta (systematic risk), Intercept = alpha (skill or luck)

Different Slopes Tell Different Stories

Each line below represents a different stock's relationship to the market. Steeper slopes mean the stock amplifies market moves; flatter slopes mean it dampens them.

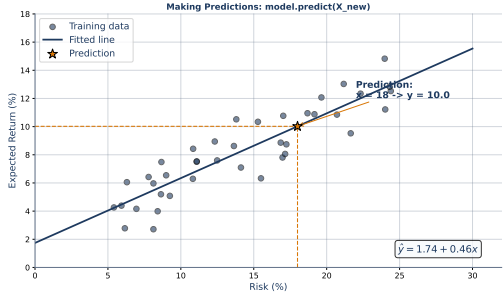


Higher beta = stock amplifies market moves. Lower beta = more defensive.

Making Predictions (and Their Limits)

Using the Fitted Model

- Predict: `model.predict([[0.03]])` $\rightarrow$ expected return if market rises 3%
- The prediction sits on the regression line
- **Caution:** Predictions assume the relationship persists. Markets change.



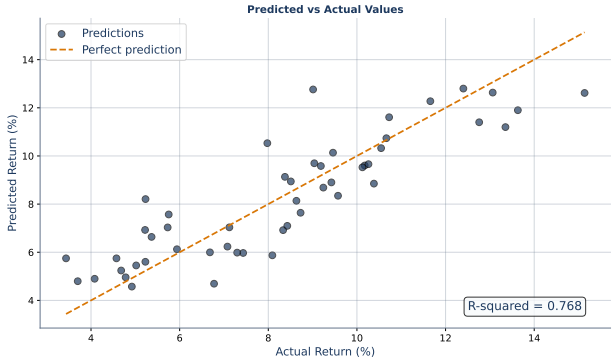
Always validate: does the model still hold for recent data?

How Good Are Our Predictions?

Plot predicted vs actual values. Perfect model = all points on diagonal.

R-squared (R^2): Fraction of variance in Y explained by X.

- $R^2 = 1.0$: perfect predictions $R^2 = 0.0$: no better than guessing the mean



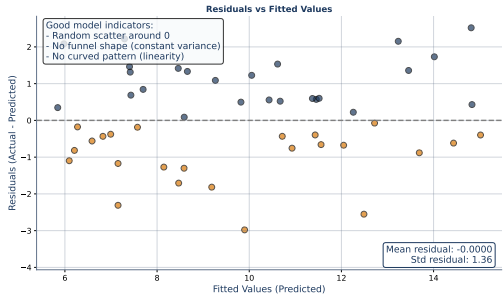
Points on the diagonal = perfect. Wider scatter = lower R^2.

Residuals: The Model's Report Card

Residual = Actual − Predicted $(e_i = y_i - \hat{y}_i)$

Good residuals look like **random noise**: no pattern, centered at zero.

- Always positive for large X? Model underpredicts there.
- Funnel shape? Variance is not constant (heteroscedasticity).



Plot residuals vs predicted values: patterns indicate model misspecification

Checkpoint: Test Your Understanding

Q1: If $\beta_1 = 1.5$ and the market rises 2%, what does the stock do?

Q2: What should a residual plot look like for a good model?

Q3: Why do we use squared errors instead of absolute errors?

A1: Rises 3%. **A2: Random scatter, no pattern.** **A3: Closed-form solution + penalizes large errors.**

The LINE Assumptions

Four Assumptions for Valid Linear Regression:

1. **L**inearity: The true relationship is a straight line
2. **I**ndependence: Each observation is unrelated to others
3. **N**ormality: Residuals follow a bell curve
4. **E**qual variance: Residual spread is constant (homoscedasticity)

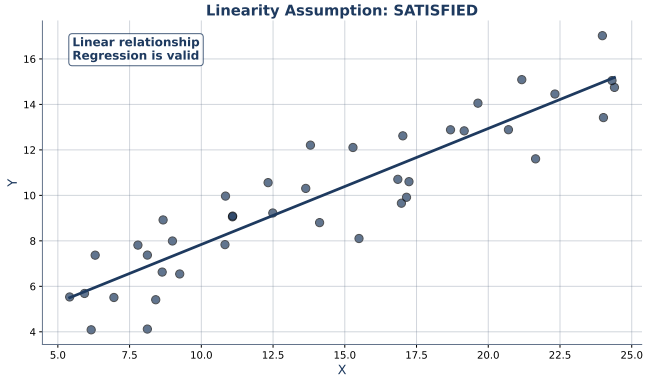
Why do they matter?

- Violated assumptions $\rightarrow$ biased coefficients or wrong confidence intervals
- You cannot trust a model's predictions if the foundations are broken
- We will check each one visually in the next slides

Mnemonic: LINE — Linearity, Independence, Normality, Equal variance

Assumptions: Linearity (Good)

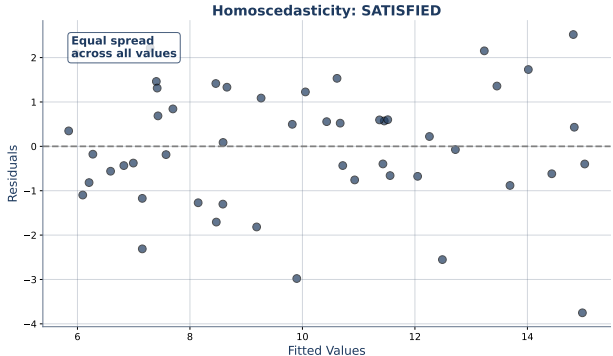
When the true relationship *is* linear, the regression line captures the pattern well and residuals show no curvature.



Linear relationship confirmed: regression works well here

Assumptions: Equal Variance (Good)

Residuals spread evenly across the range of predicted values. No funnel shape, no clusters — just uniform scatter.



Constant variance (homoscedasticity): standard errors and confidence intervals are reliable

Common Mistakes in Regression

△ **Extrapolation:** Predicting beyond your data range. A model trained on calm markets knows nothing about crashes.

△ **Correlation $\neq$ Causation:** Stocks may move together due to a common factor, not because one drives the other.

△ **Ignoring Assumptions:** Violated assumptions produce numbers that *look* precise but are not trustworthy.

These three mistakes account for the majority of regression misuse in practice

From Theory to Finance: CAPM

Everything We Just Learned IS the CAPM

Sharpe, Lintner, and Mossin (1960s) asked: "How should the market price risk?" Their answer — the **Capital Asset Pricing Model**:

$$E[R_i] = R_f + \beta_i\big(E[R_m] - R_f\big)$$

- R_f = Risk-free rate (Treasury bill, $\approx$ 2–5%)
- β_i = Stock i's sensitivity to market (the OLS slope!)
- $E[R_m] - R_f$ = Market risk premium ($\approx$ 6% historically)

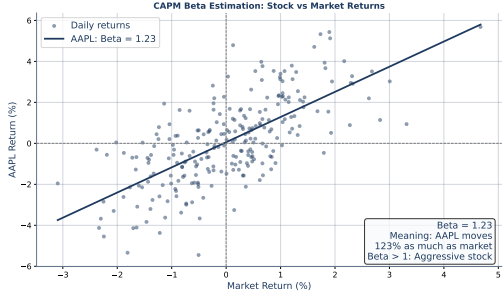
Plain English: Expected return = safe rate + (sensitivity $\times$ reward for bearing market risk).

CAPM won William Sharpe the 1990 Nobel Prize in Economics

CAPM Beta: Classifying Stocks

Beta Tells You What Kind of Stock You Own

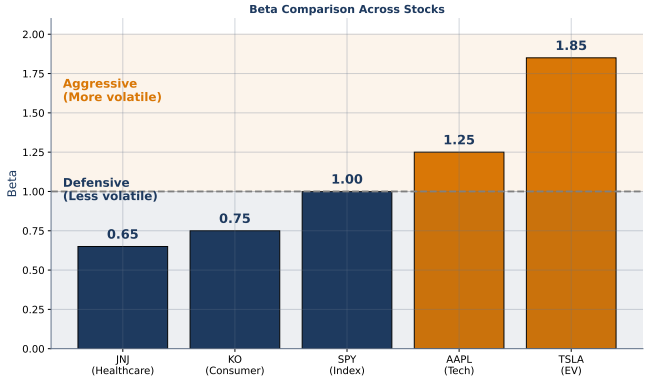
- $\beta > 1$: Aggressive — amplifies market moves (tech stocks)
- $\beta < 1$: Defensive — dampens volatility (utilities, staples)
- $\beta = 1$: Moves in lockstep with the market (index fund)



Alpha (β_0): Positive alpha = outperformance after adjusting for risk

Beta Comparison Across Stocks

Different stocks, different betas. A portfolio manager blends defensive and aggressive stocks based on the client's risk tolerance.



Mix low-beta (stability) and high-beta (growth) based on investment horizon and risk appetite

Hands-On Exercise (25 min)

Task: Estimate Beta for Your Favorite Stock

1. Download 1 year of daily returns for a stock (e.g., MSFT) and SPY
2. Fit: `model.fit(spy_returns, stock_returns)`
3. Extract and interpret: What is the beta? What is the alpha?
4. Plot the regression line with actual data points

Deliverable: Scatter plot with regression line, annotated with β.

Extension: Compare beta estimates using 1-year vs 5-year windows. Is the beta stable over time?

Reference: `inclass/L21_inclass_noise_effects.ipynb`

Extension: Compare betas across sectors — which industries are most market-sensitive?

Lesson Summary + What's Next

Problem Solved: We can now quantify systematic risk using CAPM beta via linear regression.

Key Takeaways:

- OLS finds the unique line that minimizes squared errors
- sklearn pattern: `LinearRegression().fit(X, y)` — three lines of code
- Slope = beta (market sensitivity), Intercept = alpha (skill or luck)
- Always check the LINE assumptions before trusting the model

Next Lesson (L22): Regularization
What happens when we have *too many* features? The model starts memorizing noise instead of learning signal. Regularization is the cure.

From Galton's sweet peas to Wall Street betas — regression is everywhere