

## Methods and Algorithms

**MSc Data Science**

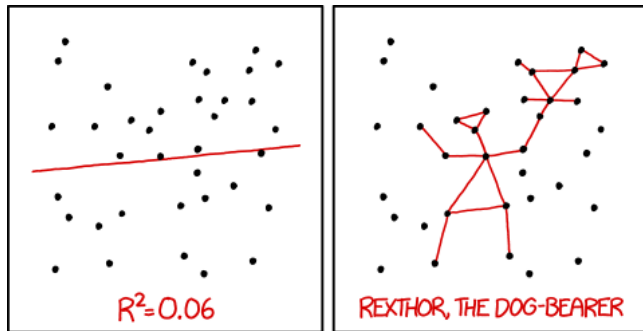
Spring 2026

By the end of this session, you will be able to:

- 1 **Understand** the ordinary least squares (OLS) framework
- 2 **Apply** gradient descent for parameter optimization
- 3 **Interpret** regression coefficients in business contexts

**Finance Application:** House price prediction, factor models

*Foundation for all supervised learning methods*



I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER  
TO GUESS THE DIRECTION OF THE CORRELATION FROM THE  
SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.

*XKCD #1725 by Randall Munroe (CC BY-NC 2.5) – Always check if your data supports a linear fit!*

## Finance Use Case: Predicting House Prices

- Banks need accurate property valuations for mortgages
- Insurance companies assess property risk
- Investors evaluate real estate portfolios

**The Question:** Given features (size, location, age), what price?

### Why Linear Regression?

- Interpretable coefficients (price per square meter)
- Fast, well-understood method
- Strong baseline for comparison

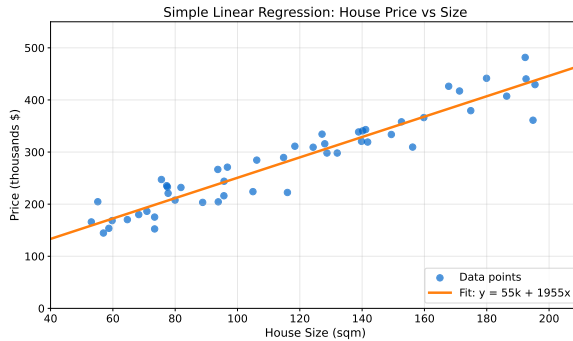
*Linear regression: the “hello world” of machine learning*

**Core Idea:** Model the relationship between inputs and a continuous output

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon \quad (1)$$

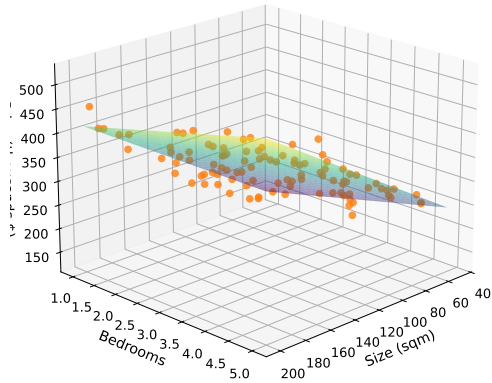
- $y$ : Target variable (house price)
- $x_1, \dots, x_p$ : Features (size, bedrooms, age)
- $\beta_0, \dots, \beta_p$ : Coefficients to learn
- $\varepsilon$ : Random error term

*Goal: Find coefficients that minimize prediction error*



*The best-fit line minimizes the sum of squared distances from points*

Multiple Regression: Price =  $f(\text{Size, Bedrooms})$



*With multiple features, we fit a hyperplane to minimize squared errors*

**Objective:** Minimize the sum of squared errors (SSE)

$$\min_{\beta} \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \min_{\beta} \|y - X\beta\|^2 \quad (2)$$

### Why Squared Errors?

- Penalizes large errors more than small errors
- Differentiable (enables gradient-based optimization)
- Leads to closed-form solution

*This objective function defines “ordinary least squares” (OLS)*



## 1. Closed-Form (Normal Equation)

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

### Pros:

- Exact solution
- One computation

### Cons:

- Slow for large  $p$
- Memory intensive

## 2. Gradient Descent

(3)

$$\beta_{t+1} = \beta_t - \alpha \nabla L(\beta_t)$$

(4)

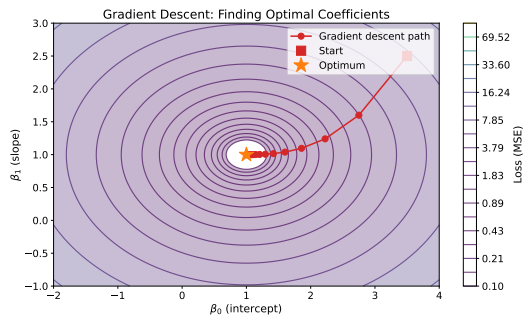
### Pros:

- Scales to big data
- Memory efficient

### Cons:

- Requires tuning  $\alpha$
- Iterative process

*Both methods yield the same solution for OLS*



*Iteratively update parameters in the direction of steepest descent*

### Example: House price model

$$\text{Price} = 50,000 + 200 \times \text{Size} + 15,000 \times \text{Bedrooms} - 1,000 \times \text{Age} \quad (5)$$

#### Interpretation:

- $\beta_0 = 50,000$ : Base price (all features = 0)
- $\beta_1 = 200$ : Each extra sqm adds \$200
- $\beta_2 = 15,000$ : Each bedroom adds \$15,000
- $\beta_3 = -1,000$ : Each year of age subtracts \$1,000

*Coefficients show marginal effect, holding others constant*

## Key Metrics:

- $R^2$  (Coefficient of Determination): Variance explained

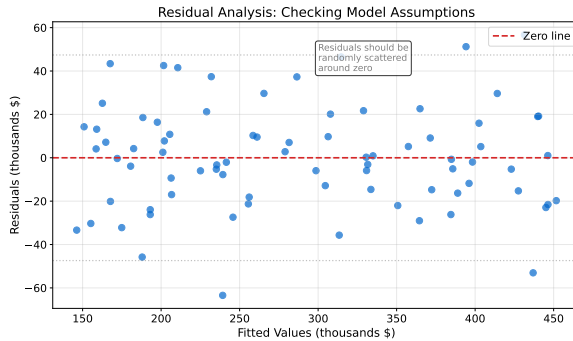
$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (6)$$

- RMSE (Root Mean Squared Error): Prediction accuracy

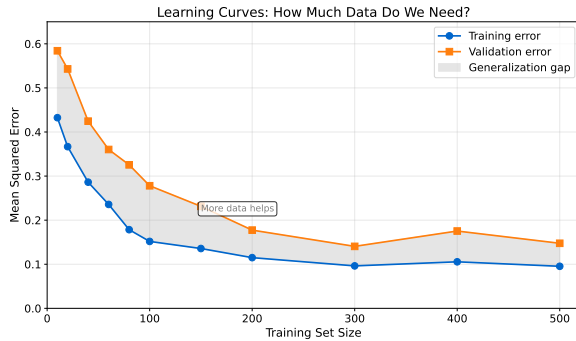
$$\text{RMSE} = \sqrt{\frac{1}{n} \sum (y_i - \hat{y}_i)^2} \quad (7)$$

**In Practice:**  $R^2 = 0.75$  means model explains 75% of price variance

*Always evaluate on held-out test data*



*Good residuals are random; patterns suggest model problems*



*Learning curves help diagnose underfitting vs overfitting*

### Use When:

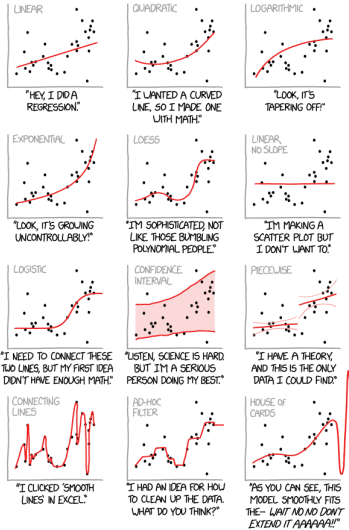
- Target is continuous
- Linear relationships expected
- Interpretability matters
- Fast inference needed

### Avoid When:

- Target is categorical
- Strong non-linearities
- High multicollinearity
- Many outliers present

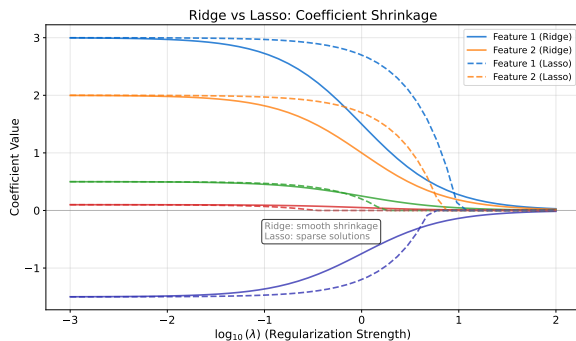
Pros	Cons
Interpretable	Assumes linearity
Fast training	Sensitive to outliers
Well-understood	Limited flexibility

## CURVE-FITTING METHODS AND THE MESSAGES THEY SEND

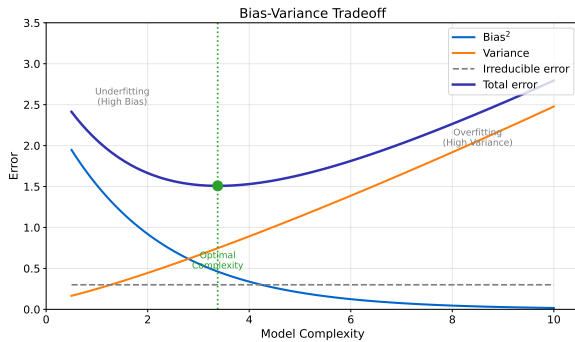


XKCD #2048 – Simpler models often generalize better (CC BY-NC 2.5)



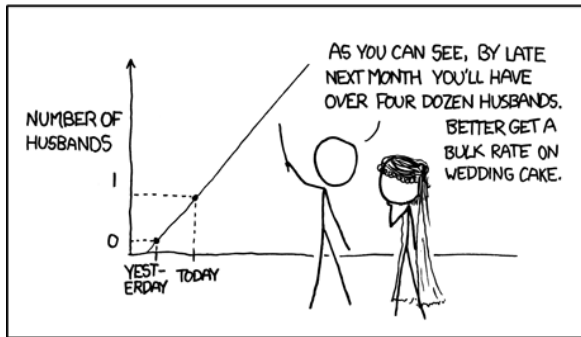


*Ridge shrinks all coefficients; Lasso can set some to exactly zero*



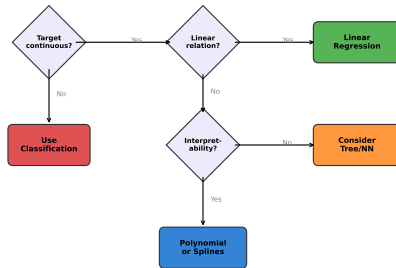
*Model complexity controls the tradeoff between bias and variance*

## MY HOBBY: EXTRAPOLATING



XKCD #605 by Randall Munroe (CC BY-NC 2.5) – Extrapolation is dangerous!

### Linear Regression Decision Guide



*Use this flowchart to decide when linear regression is appropriate*

- 1 Linear regression models continuous outcomes as weighted sum of features
- 2 OLS minimizes sum of squared errors
- 3 Solve via normal equation (small data) or gradient descent (big data)
- 4 Coefficients are directly interpretable as marginal effects
- 5 Evaluate using  $R^2$  (variance explained) and RMSE (prediction error)

**Next:** Deep dive into mathematics and implementation

#### References:

- ISLR Chapter 3: Linear Regression
- ESL Chapter 3: Linear Methods for Regression