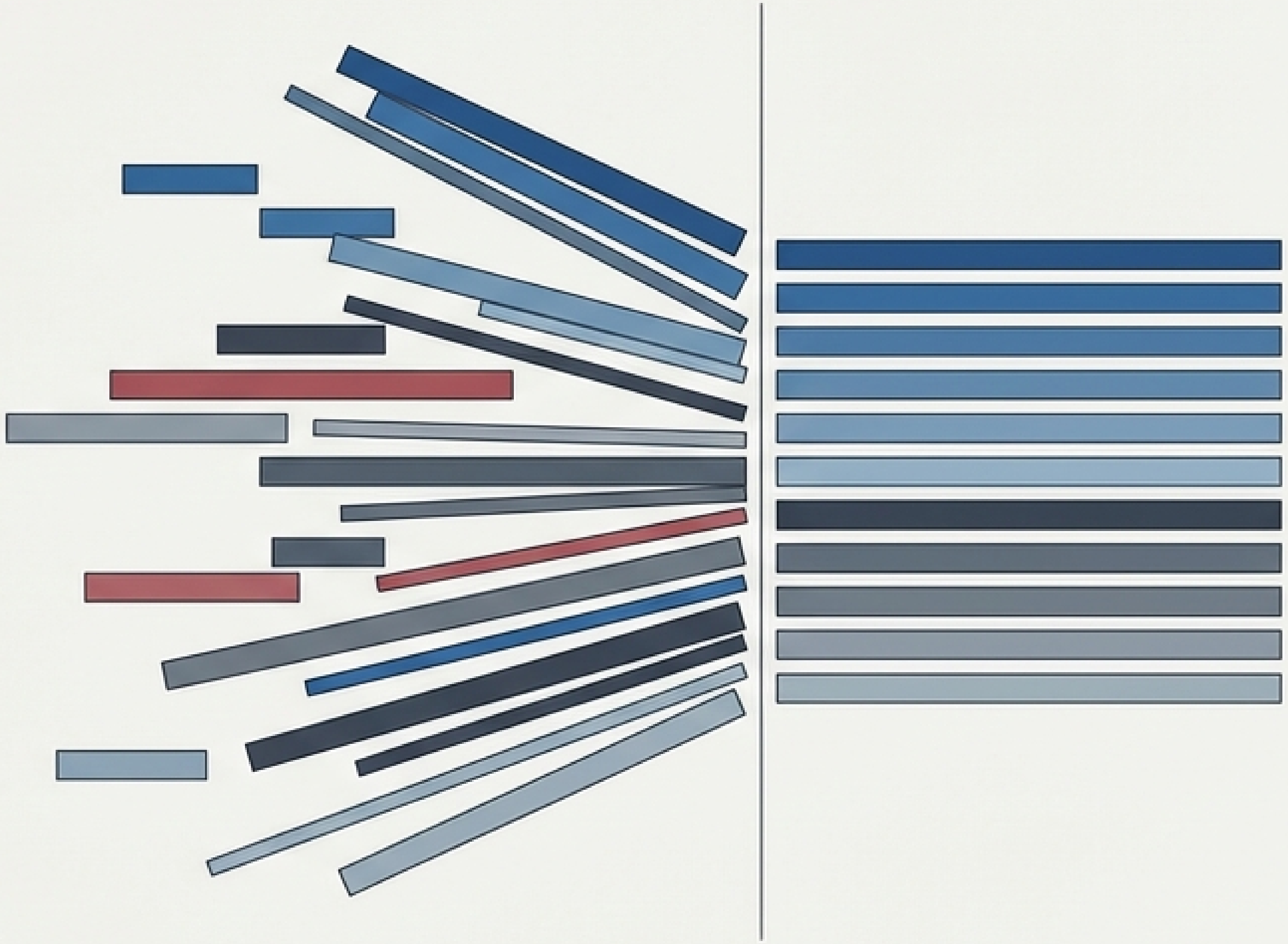
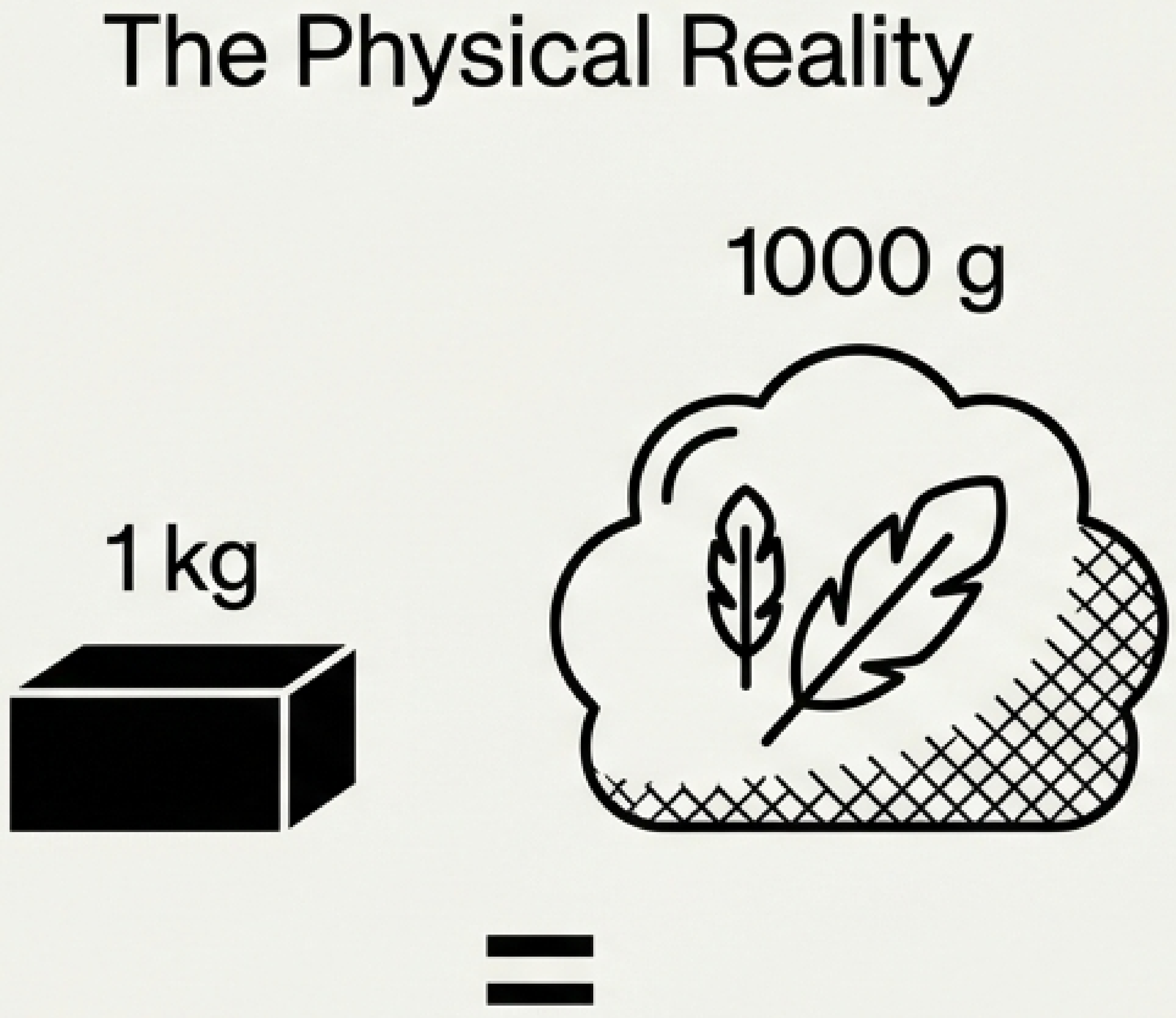
Feature Normalisation: The Practitioner’s Guide

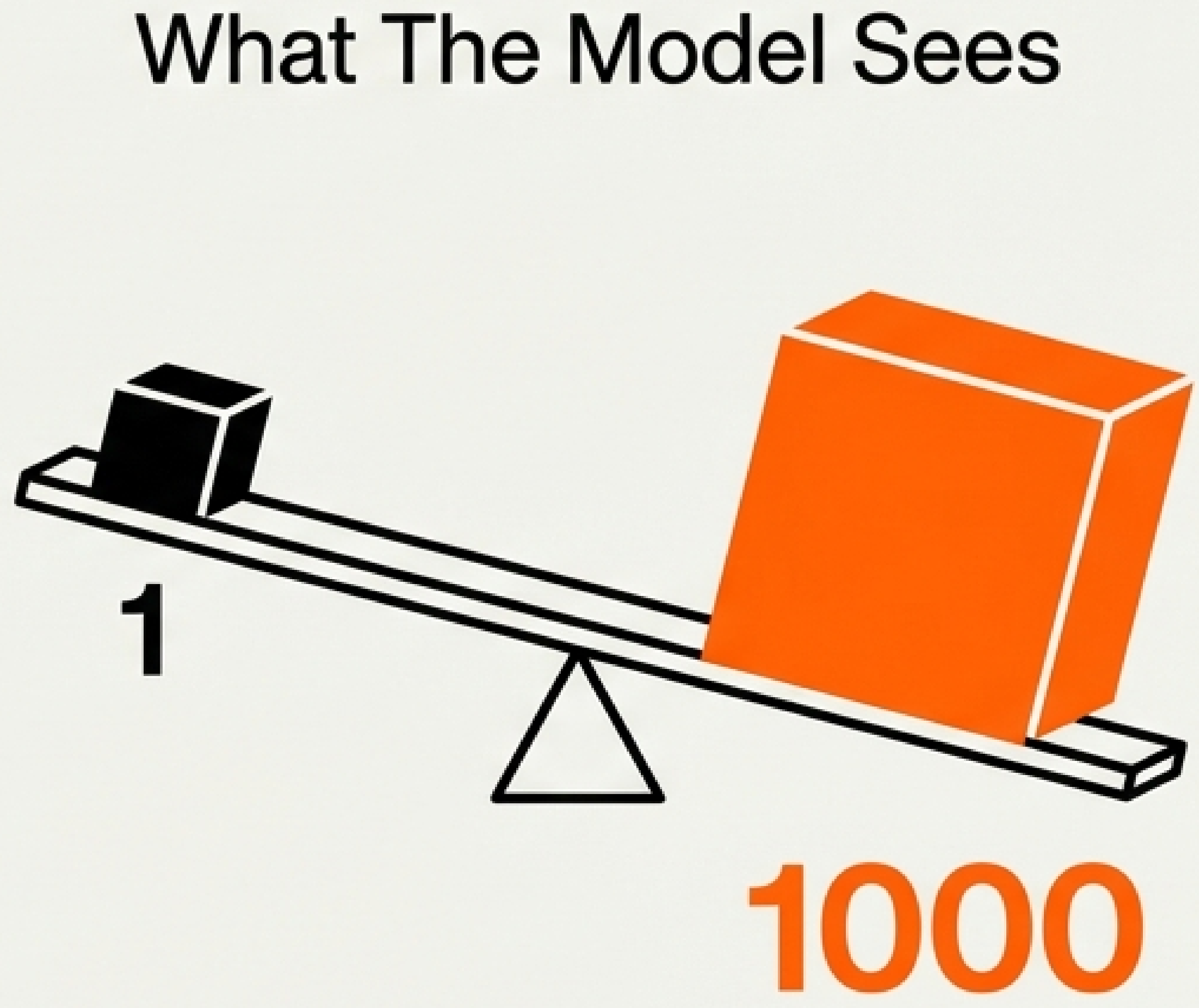
From Geometric Intuition to Strategic Application



Machine learning models struggle when features operate on vastly different scales. This guide explores Normalisation—the art of bringing data to a common scale without distorting the information it holds.

The Problem of Scale: Magnitude vs. Unit

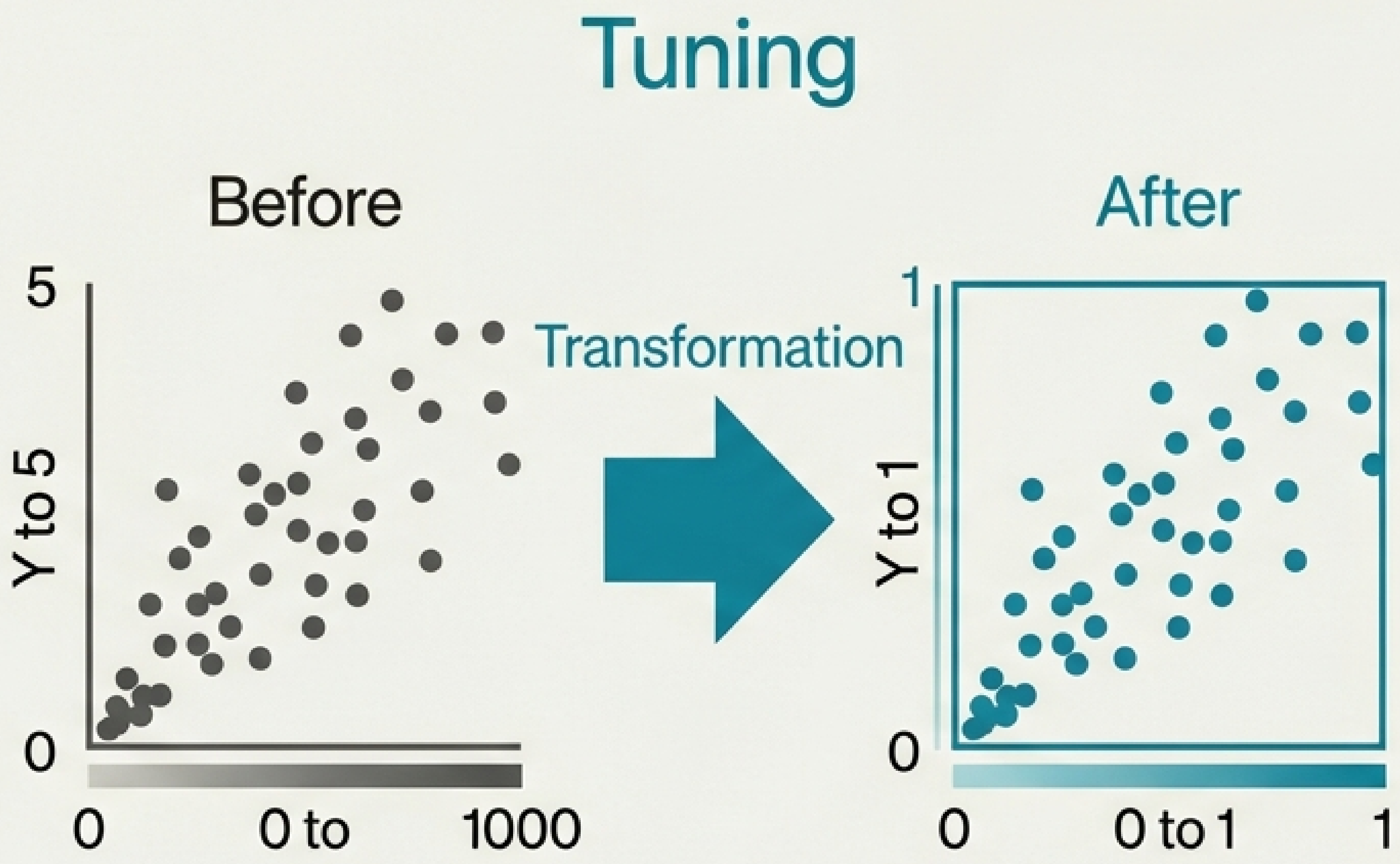




Machine learning algorithms see numbers, not units. A value of 1000 dominates a value of 1, causing biased predictions and slow convergence, even if they represent the same physical quantity.

Defining Normalisation

Definition
Normalisation is a data preparation technique that changes the values of numeric columns to a common scale.

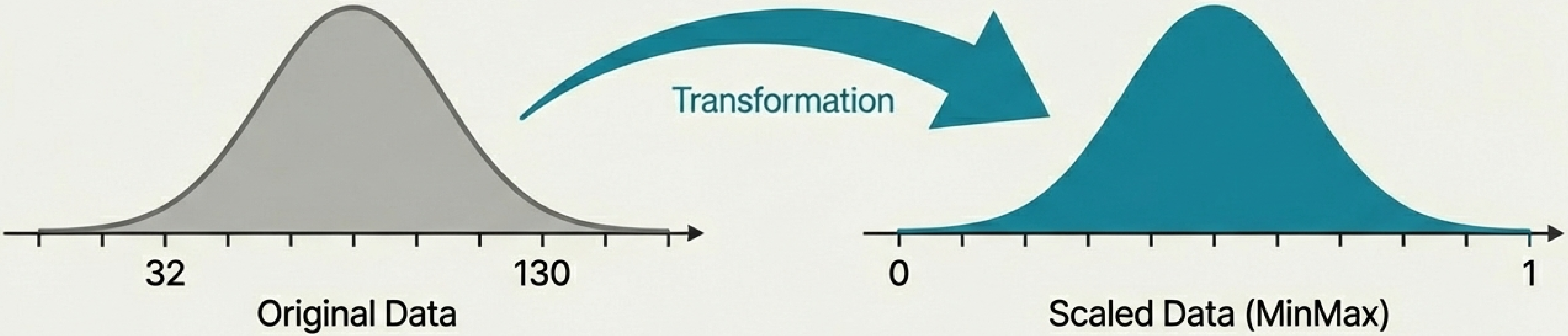


The Objective: Unit Independence
Eliminate the bias introduced by arbitrary units (e.g., kg vs g). The model should focus on relative differences within the data, not magnitude.

Key Constraint
Must be done without distorting the shape of the distribution or losing information.

The Primary Tool: MinMax Scaling

The industry standard, used in approx. 90% of normalisation cases.



- Transforms features by scaling them to a fixed range, typically [0, 1].
- The Minimum value (X_{min}) becomes 0.
- The Maximum value (X_{max}) becomes 1.
- Preserves the exact shape of the original distribution.

The Mathematics of MinMax

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

The Shift: Moves the distribution so the minimum value starts at 0.

The Scale: Divides by the total range to compress the data.

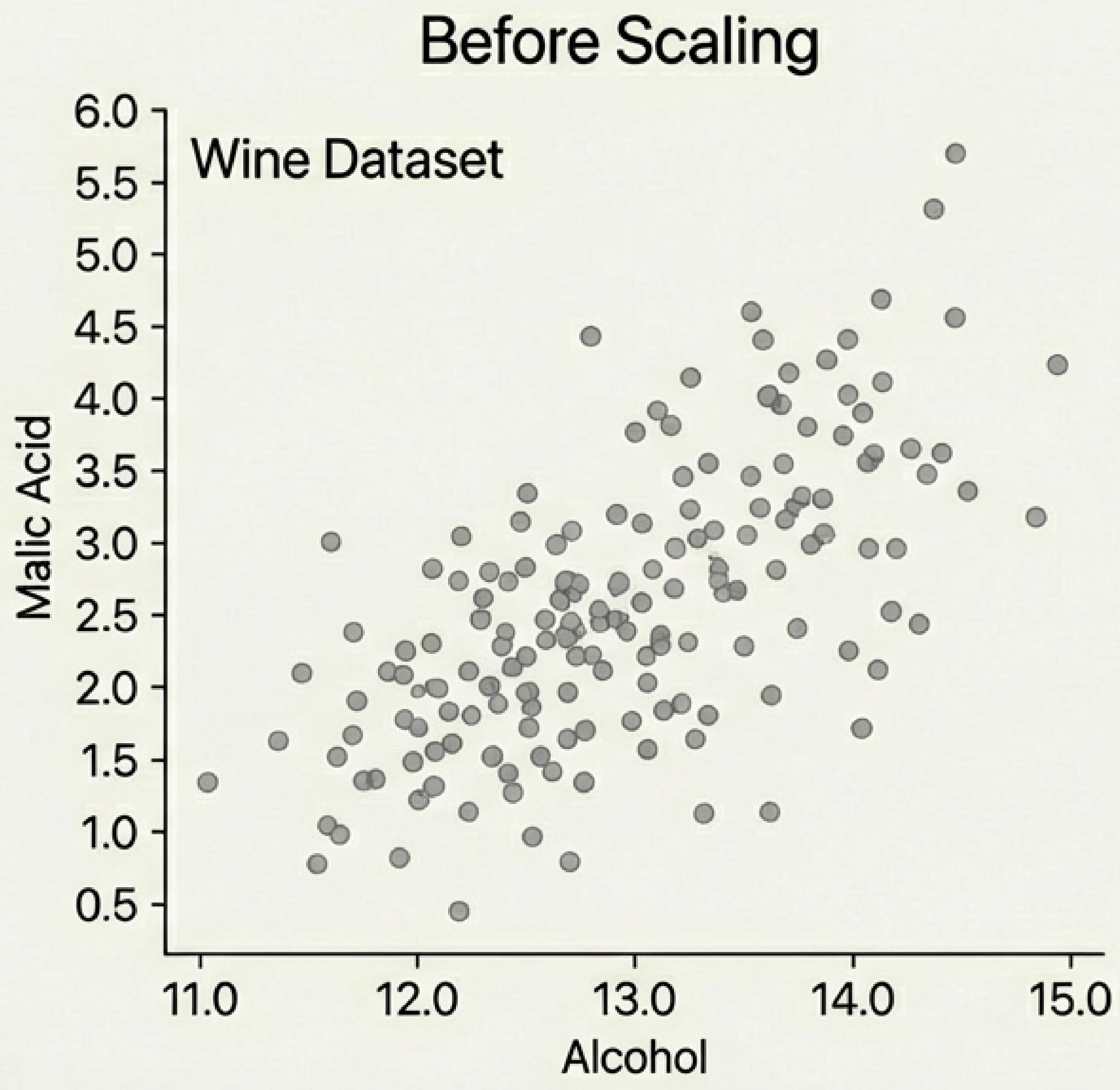
Example Calculation

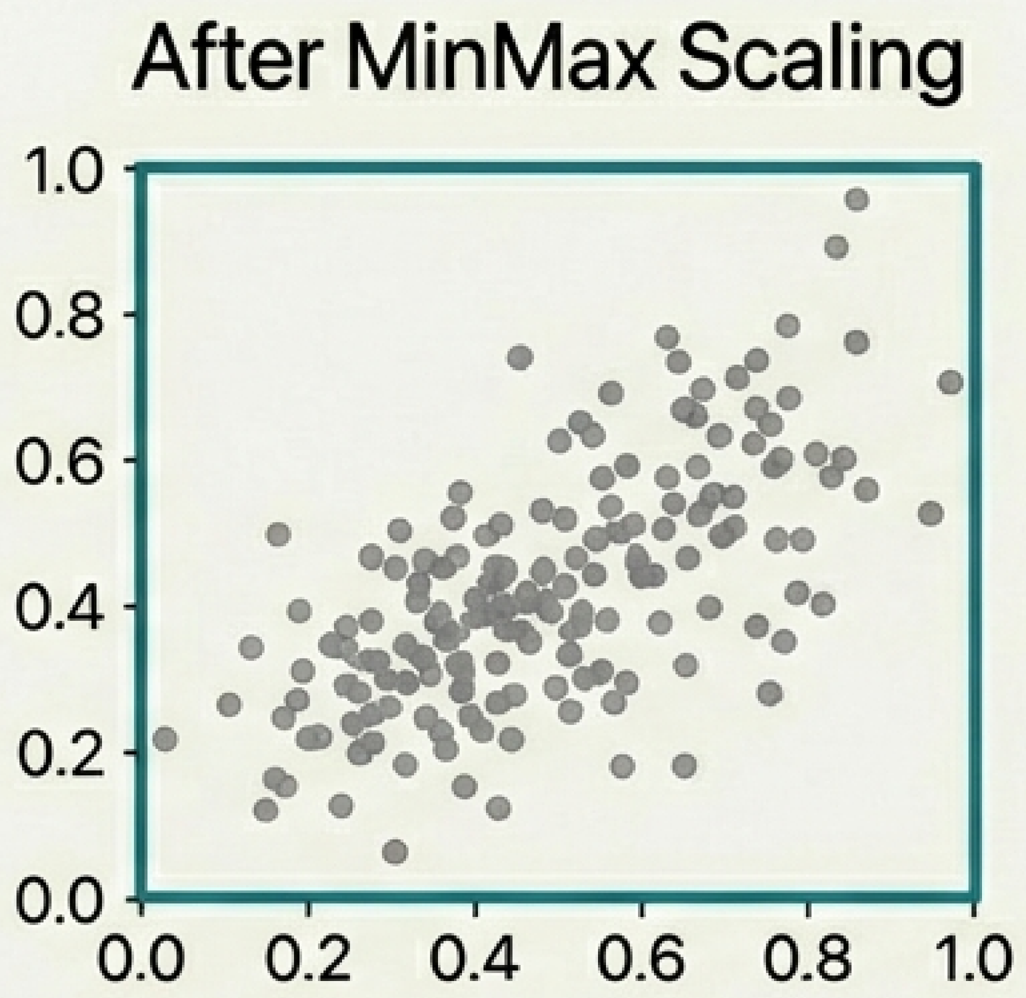
Dataset Range: Min = 32, Max = 130.

Data point to scale: 130

Calculation: $\frac{130 - 32}{130 - 32} = \frac{98}{98} = 1$

Geometric Intuition: The Unit Box

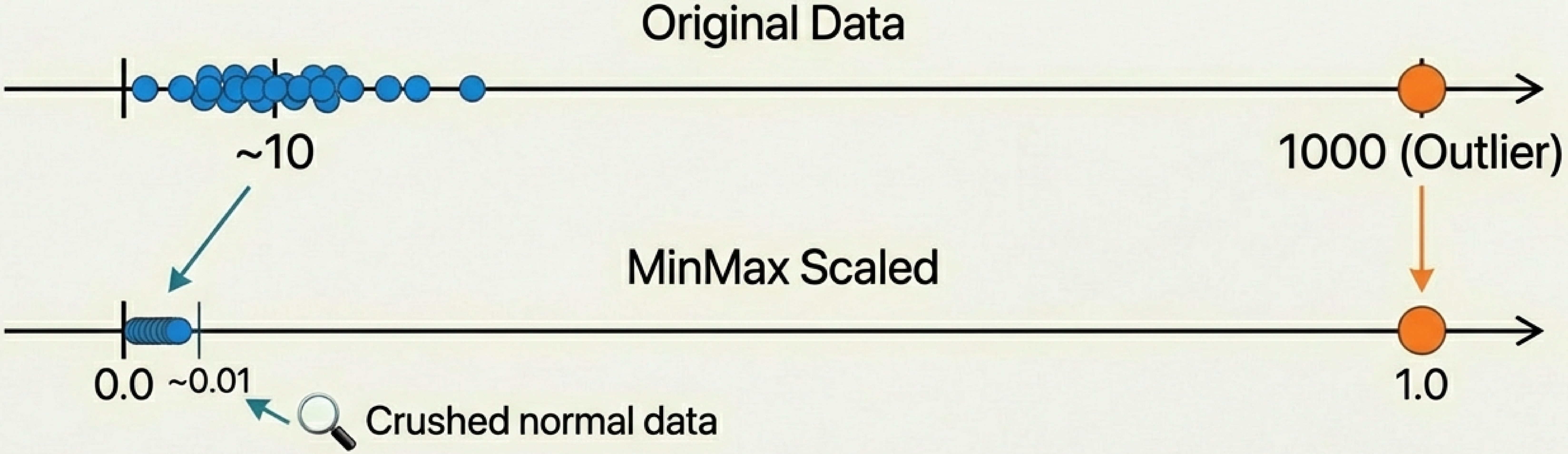




The Squashing Effect
Imagine a bounding box defined by the min and max. MinMax scaling compresses the entire dataset into a 'Unit Structure'.

- 2D Data → Unit Rectangle
- 3D Data → Unit Cube
- n-Dimensions → Unit Hypercube

The Vulnerability: Impact of Outliers



Fragility in Noisy Datasets
MinMax uses the absolute min and max to define the range. A single extreme outlier stretches the scale.

Result: Useful variance in normal data is suppressed into an insignificant interval.

Alternative 1: Mean Normalisation

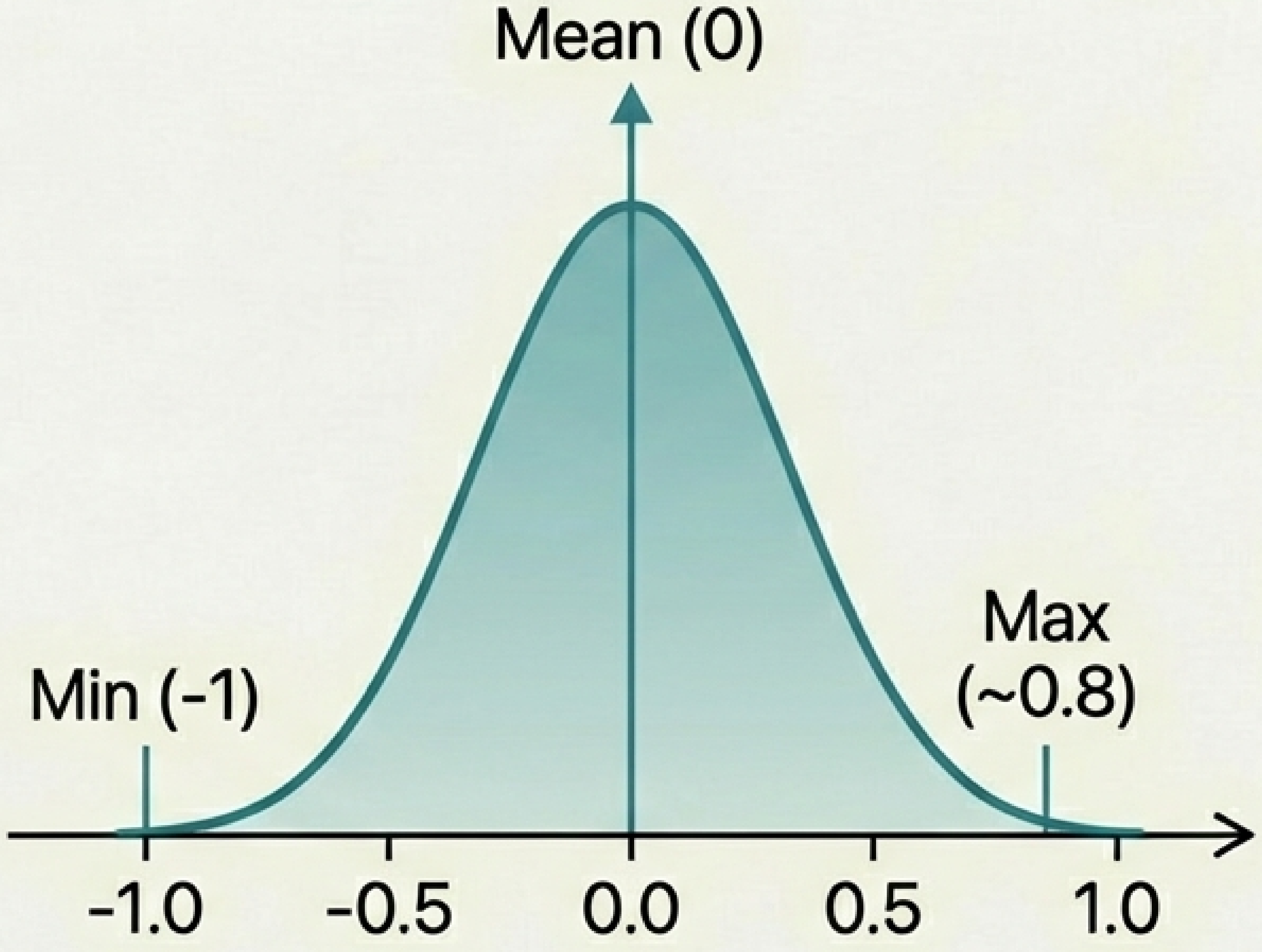
Concept:
Similar to MinMax, but centers data around the mean (0) rather than setting the minimum to 0.

Formula:

$$X_{\text{new}} = \frac{X - X_{\text{mean}}}{X_{\text{max}} - X_{\text{min}}}$$

Characteristics:

- Range: Typically [-1, 1]
- Mean: Centred at 0
- **Note:** Rarer in practice. Scikit-learn has no dedicated class for this; requires manual implementation.



Alternative 2: Max Absolute Scaling

Pro Tip: Best for Sparse Data

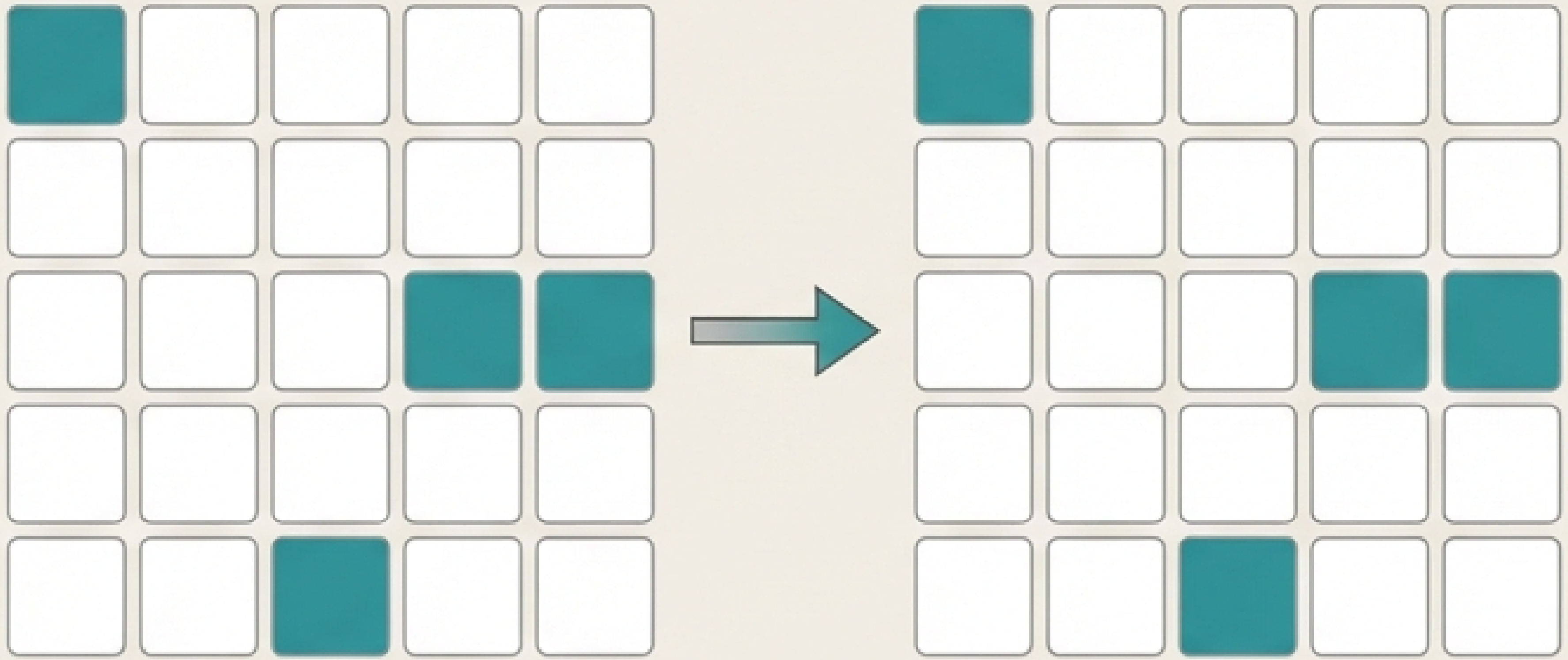
Formula:

$$X_{new} = \frac{X}{|X_{max}|}$$

The Use Case: Sparse Matrices

Datasets containing mostly zeros (Sparse Data) require special handling. Centering techniques (like Mean Normalisation) shift data, turning 0s into non-zero values and destroying sparsity.

MaxAbs divides by the maximum absolute value, ensuring that **0 remains 0**.



Structure Preserved (Zeros stay Zeros)

Alternative 3: Robust Scaling

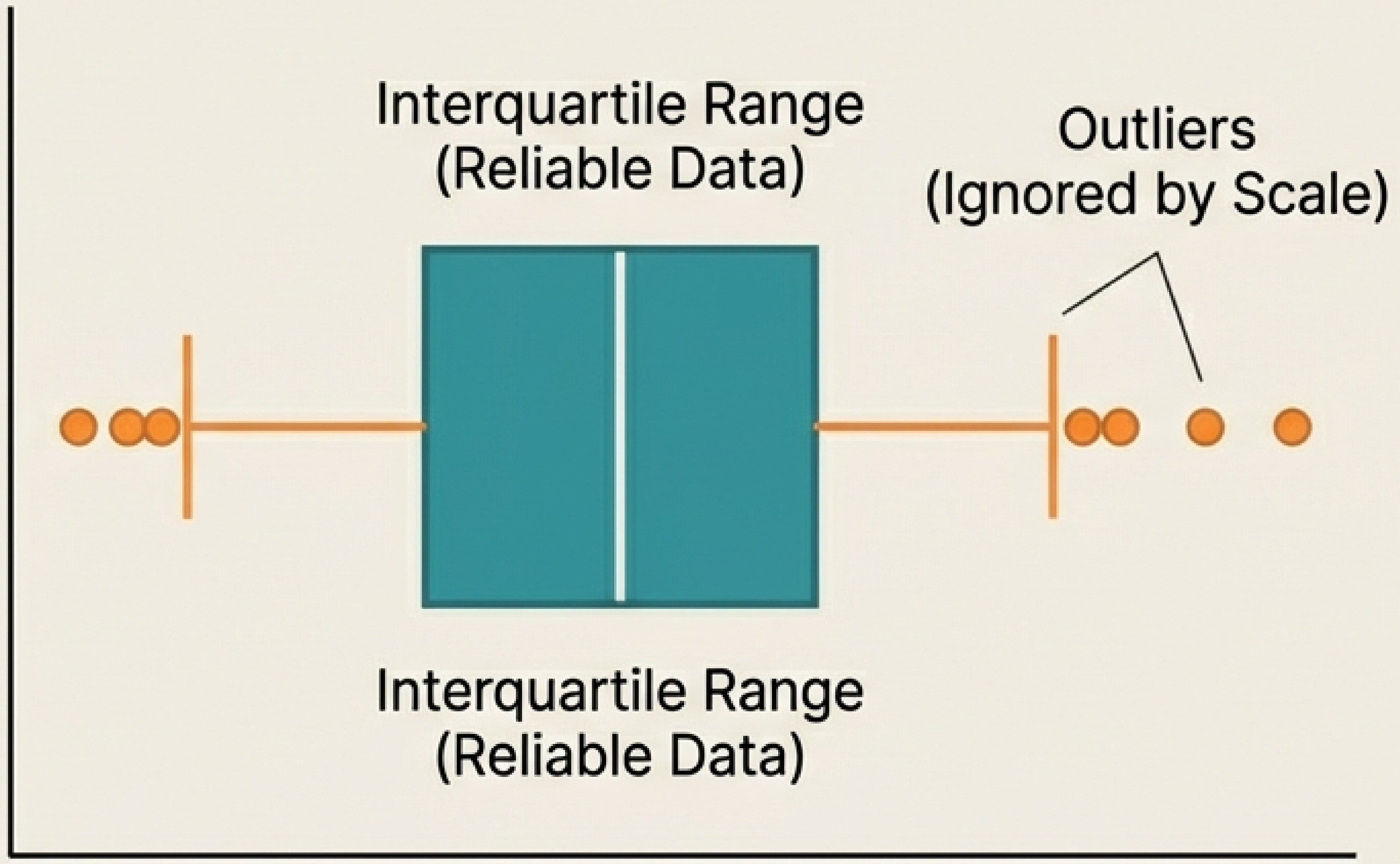
The solution for outliers.

The Mechanism:
Instead of using the vulnerable Min/Max, Robust Scaling uses statistical measures that are resistant to extremes: the Median and Interquartile Range (IQR).

Formula:

$$X_{\text{new}} = \frac{X - X_{\text{median}}}{IQR}$$

*(IQR = 75th Percentile - 25th Percentile)



Scaling focuses on the middle 50% of the data.

The Landscape of Techniques

Technique	Key Formula Component	Range	Primary Use Case
MinMax Scaling	Min & Max	[0, 1]	Bounded Data, Image Processing (CNNs)
Mean Normalisation	Mean	[-1, 1]	Centred Data (Rare)
MaxAbs Scaling	Max Absolute	[-1, 1]	Sparse Data (Preserving Zeros)
Robust Scaling	Median & IQR	Variable	Data with Outliers

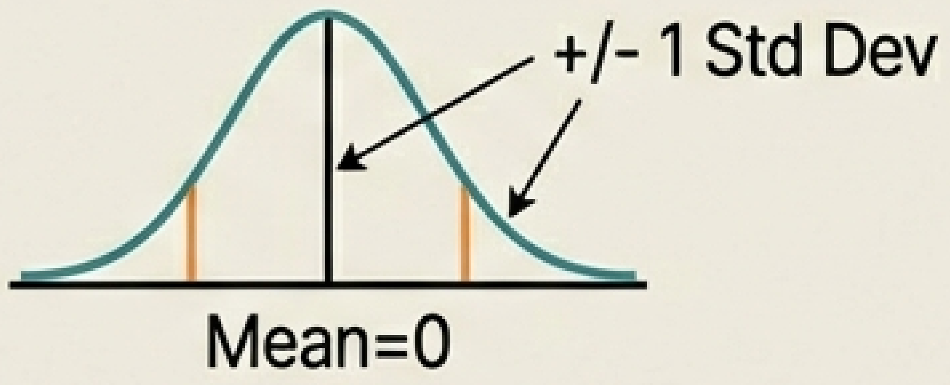
Strategic Choice: Normalisation vs. Standardisation

The Confusion

Terms are often used interchangeably, but are distinct.

Standardisation (Z-Score):
Mean = 0, Std Dev = 1. No fixed bounds.

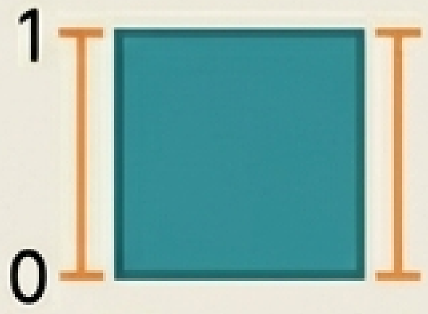
Normalisation:
Scales to a fixed range (e.g., 0 to 1).



Expert Insight

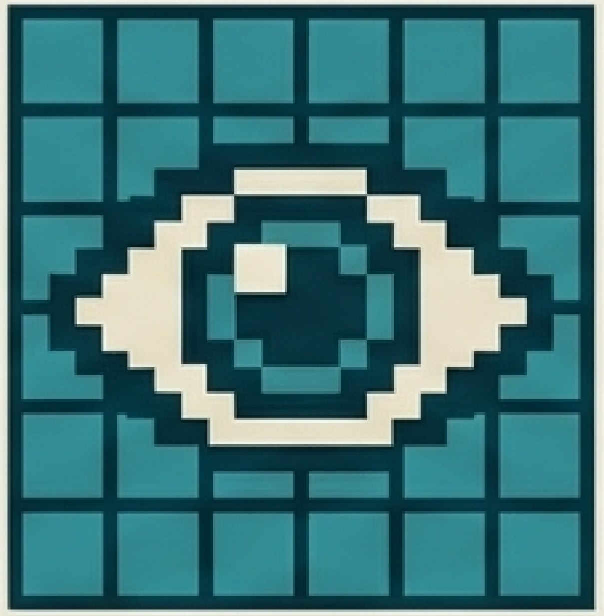
1. **The Default:** Standardisation is generally the preferred starting point for most algorithms.

2. **The Exception:** Normalisation is required for specific cases (Neural Networks, Image Processing) or when data does not follow a Gaussian distribution.



NotebookLM

Rules of Thumb: When to Use What

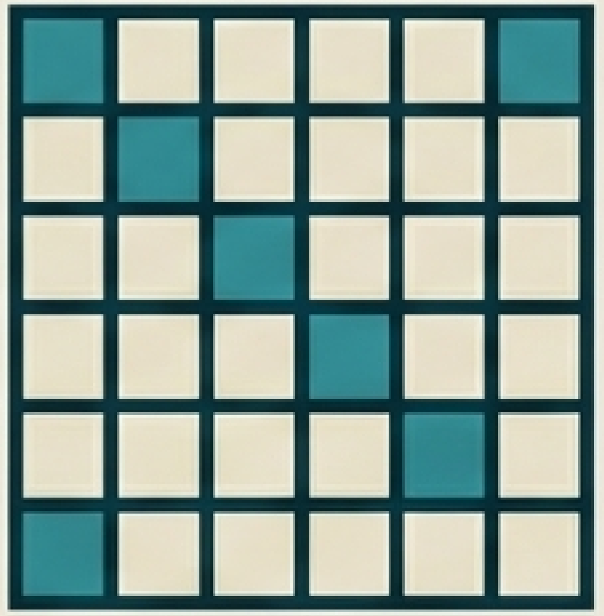


Known Bounds

Use when data has hard boundaries. E.g., Image Processing (Pixel intensities are always 0-255).

Outliers Present

Use when data contains significant noise or extremes that shouldn't skew the scale.



Sparse Data

Use when the input matrix has many zeros that must be preserved for efficiency.

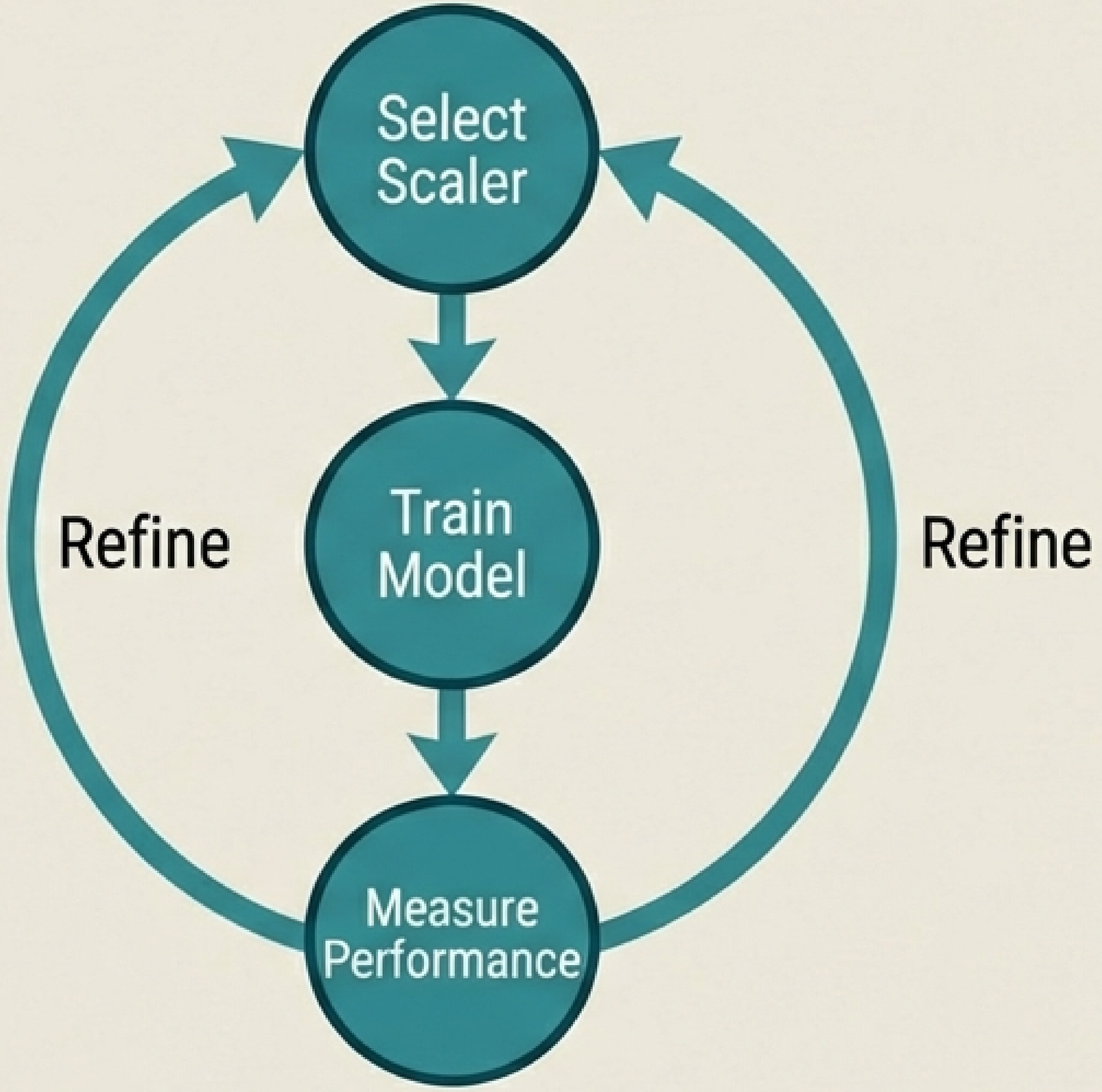
The Philosophy of Experimentation

There is no perfect theory.

Data Science is empirical. You cannot always predict which scaler will perform best on unseen data.

The Strategy:

Treat scalers as tools in a toolkit. Apply different techniques, measure model accuracy, and select the one that yields the best results for your specific problem.



"Machine Learning is all about performing experimentation."

Key Takeaways

01. Goal: Normalisation achieves unit independence, preventing large magnitudes from biasing the model.

02. Geometry: Visualize the process as squashing data into a Unit Hypercube.

03. The Standard: MinMax is the workhorse (used in ~90% of cases), scaling data to [0, 1].

04. The Specialists: Use Robust Scaling for outliers and MaxAbs for sparse matrices.

05. The Mindset: Start with Standardisation, but experiment with Normalisation techniques to optimise performance.