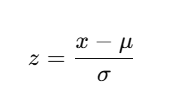
Standardization and normalization are preprocessing techniques used to scale data so that it is in a comparable range. This step is crucial for many machine learning algorithms to perform efficiently and effectively. Here’s an explanation of **why we do standardization and normalization**:

**1. Why Standardization?**

Standardization transforms data to have a mean of 0 and a standard deviation of 1, using the formula:



Where:

* xxx: Original data point
* μ: Mean of the feature
* σ: Standard deviation of the feature

**Reasons for Standardization:**

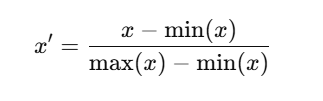
1. **Algorithms Sensitive to Scale:**
   * Algorithms like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Principal Component Analysis (PCA), and linear regression rely on distance or variance. Features with larger scales can dominate smaller-scale features.
2. **Ensures Equal Contribution:**
   * Standardization ensures that all features contribute equally to the model, regardless of their original scale.
3. **Faster Convergence:**
   * Gradient descent optimization (used in neural networks, logistic regression, etc.) converges faster when features are standardized.

**Example of Unstandardized Data Issue:**

* Feature A (age): Ranges from 0 to 100.
* Feature B (income): Ranges from 10,000 to 100,000.
* Without standardization, income would dominate the model's learning process due to its higher magnitude.

**2. Why Normalization?**

Normalization rescales data to a fixed range, typically [0, 1], using the formula:



Where:

* x: Original data point
* min(x): Minimum value of the feature
* max(x): Maximum value of the feature

**Reasons for Normalization:**

1. **Algorithms with Assumptions on Range:**
   * Algorithms like neural networks and KNN perform better with normalized data since they expect inputs to be in a specific range.
2. **Feature Comparability:**
   * Ensures features with different units (e.g., height in cm and weight in kg) are scaled to the same range for comparability.
3. **Probability Interpretations:**
   * Normalized values often align with probabilities, which are bounded between 0 and 1.

**Example of Unnormalized Data Issue:**

* Feature A (height): Ranges from 100 to 200 cm.
* Feature B (weight): Ranges from 30 to 150 kg.
* Without normalization, larger feature ranges could disproportionately influence the results.

**Key Differences:**

| **Feature** | **Standardization** | **Normalization** |
| --- | --- | --- |
| **Formula** |  |  |
| **Range** | Mean = 0, Standard Deviation = 1 | Scaled to [0, 1] (or another range) |
| **When to Use** | Distance-based models (e.g., SVM, KNN, PCA) | When inputs must be bounded (e.g., neural networks) |
| **Handling Outliers** | Less sensitive | More sensitive |

**Practical Use Cases**

1. **Standardization is preferred when:**
   * Data follows a Gaussian (normal) distribution.
   * The algorithm uses distance-based metrics (e.g., Euclidean distance).
2. **Normalization is preferred when:**
   * Data does not follow a normal distribution.
   * Algorithms require input within a fixed range (e.g., image pixel intensities for neural networks).