**Task Title:** Data Normalization Techniques and Mean-Centering with Z-Score

**Description:** (A) (Task for normalization and z-score) Find 10000 row dataset and apply all type of normalization in dataset. 20(B) Apply z-score using sklearn library and do mean-centering of Sales dataset(<https://www.kaggle.com/datasets/nishathakkar/100-sales>)

**Task (A) Explanation**

**Objective:**  
The goal of Task 1 was to apply various normalization techniques to the dataset to standardize its numerical features, enabling better performance and comparability for machine learning models and statistical analyses. These transformations are critical when features have different scales, magnitudes, or distributions.

**Dataset Description:**

The dataset, 100\_Sales.csv, contains information about sales records, including both categorical and numerical attributes such as:

* **Categorical Columns:** Region, Country, Item\_Type, Sales\_Channel, Order\_Priority.
* **Numerical Columns:** Unit\_Cost, Total\_Revenue, Total\_Profit.

For normalization, only the **numerical columns** were considered.

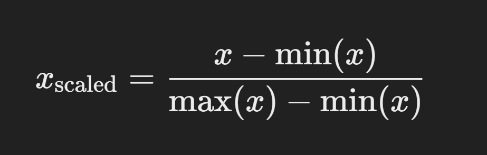
**Preprocessing Steps:**

1. **Selection of Numerical Columns:**
   * Columns like Unit\_Cost, Total\_Revenue, and Total\_Profit were identified for normalization.
2. **Handling Missing Values:**
   * Columns with all missing values (NaN) were dropped.
   * Remaining NaN values were replaced with the **median** of each column to avoid introducing biases.

**Normalization Techniques Applied:**

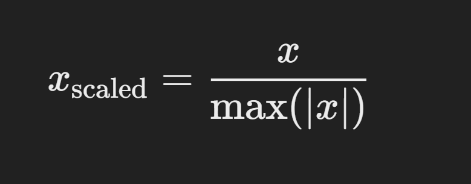
**1. Min-Max Scaling**

* **Purpose:** Rescales the values of each column to a range of [0, 1].
* **Effect:** It preserves the relationships between values while ensuring all features lie within the same range.
* **Implementation:**
  + Formula:



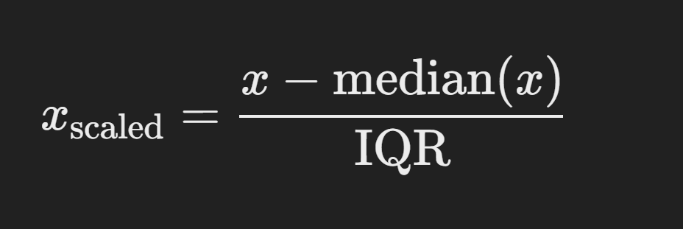
* + Example:
    - Original: Unit\_Cost = [100, 200, 300]
    - Scaled: Unit\_Cost\_minmax = [0.0, 0.5, 1.0]

**2. Max-Abs Scaling**

* **Purpose:**  
  Scales each value by dividing it by the maximum absolute value of the column. This method retains sparsity.
* **Effect:**  
  Keeps values between [-1, 1] while preserving the distribution of magnitudes.
* **Implementation:**
  + Formula:
  + Example:
    - Original: Total\_Revenue = [1000, -2000, 3000]
    - Scaled: Total\_Revenue\_maxabs = [0.33, -0.67, 1.0]

**3. Robust Scaling**

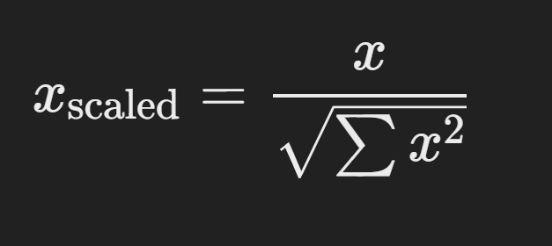
* **Purpose:**  
  Scales data by removing the median and dividing by the interquartile range (IQR). It is robust to outliers.
* **Effect:**  
  Centers the data while reducing the influence of extreme values.
* **Implementation:**
  + Formula:



* + Example:
    - Original: Total\_Profit = [100, 5000, 300]
    - Scaled: Total\_Profit\_robust = [-0.5, 9.5, 0.5]

**4. L2 Normalization**

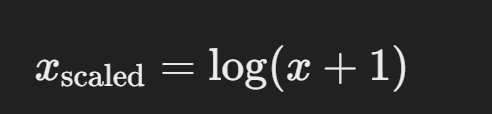
* **Purpose:**  
  Transforms values so that the sum of the squared values (Euclidean norm) equals 1 for each row.
* **Effect:**  
  Makes all rows comparable in terms of magnitude while maintaining directionality in vector space.
* **Implementation:**
  + Formula:



* + Example:
    - Original (row): [100, 200, 300]
    - Normalized (row): [0.27, 0.53, 0.80]

**5. Log Transformation**

* **Purpose:**  
  Applies a logarithmic transformation to compress the scale of larger values, making the distribution closer to normal.
* **Effect:**  
  Reduces skewness in distributions and stabilizes variance for features with exponential growth patterns.
* **Implementation:**
  + Formula:



* + Example:
    - Original: Total\_Revenue = [100, 200, 300]
    - Log Transformed: Total\_Revenue\_log = [4.61, 5.30, 5.70]

**Task (B) Explanation**

**Objective:**  
The goal of Task 2 was to apply **Z-Score Standardization** to the numerical columns in the dataset and ensure that the features are mean-centered. This transformation helps in standardizing the dataset, making it more suitable for machine learning algorithms that assume normalized distributions, such as linear regression or principal component analysis (PCA).

**Dataset Description:**

The dataset, 100\_Sales.csv, contains both categorical and numerical attributes. For this task, the focus remained on the numerical columns:

* **Numerical Columns:** Unit\_Cost, Total\_Revenue, Total\_Profit.

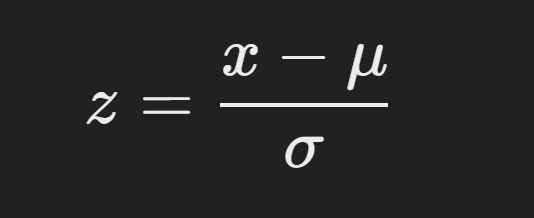
**Preprocessing Steps:**

1. **Selection of Numerical Columns:**
   * The same numerical columns (Unit\_Cost, Total\_Revenue, Total\_Profit) were selected for standardization.
2. **Handling Missing Values:**
   * As in Task 1, columns with all missing values were dropped, and remaining NaN values were replaced with the **median** of each column to maintain data integrity.

**Transformations Applied:**

**1. Z-Score Standardization:**

* **Purpose:**  
  Transforms data to have a mean of 0 and a standard deviation of 1. This ensures that all numerical features have the same scale, regardless of their original magnitude.
* **Effect:**
  + Data is transformed to a standard normal distribution.
  + Helps in removing units of measurement, enabling fair comparison between features.
* **Formula:**



Where:

* + xxx: Original value.
  + μ\muμ: Mean of the column.
  + σ\sigmaσ: Standard deviation of the column.
* **Example:**
  + Original: Unit\_Cost = [100, 200, 300]
  + Z-Score Transformed: Unit\_Cost\_zscore = [-1.22, 0.0, 1.22]

**2. Mean-Centering:**

* **Purpose:**  
  Adjusts the dataset such that the mean of each feature becomes zero. This process is inherently included in Z-Score Standardization, as the first step of the formula centers the data by subtracting the mean.
* **Effect:**
  + Shifts all numerical features to be centered around 0.
  + Simplifies interpretation and mathematical computations, especially for linear models.

**Results Overview:**

After applying the Z-Score Standardization:

* The transformed dataset now contains standardized columns for each numerical feature:
  + Unit\_Cost\_zscore
  + Total\_Revenue\_zscore
  + Total\_Profit\_zscore

These columns represent the z-scored versions of their respective original features, ensuring that:

* Mean = 0
* Standard Deviation = 1