Unsupervised Learning

Types of Unsupervised Learning

- Transformations of the dataset
  - Creates a new representation of data to make it easier for humans and other machine learning algorithms to understand.
  - Example : Principal Component Analysis (PCA)
- Clustering
  - Partition data into distinct groups of similar items
  - Example : Customer Segmentation
- Challenges in Unsupervised Learning
  - Evaluation

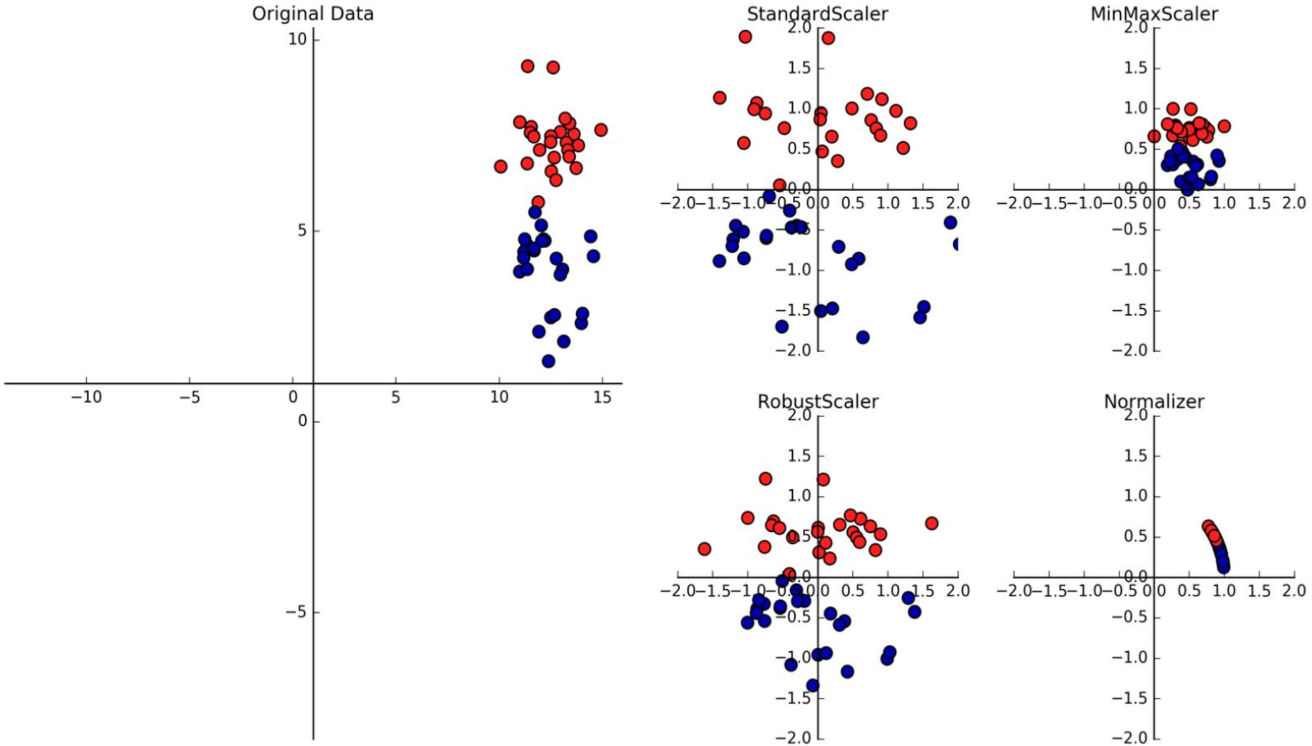
Why do we have to scale the data?

- Standardization of dataset is a common requirement for many Machine Learning Algorithms.
- The performance of these algorithms can be worse if the individual features don't look more or less like a **standard normally distributed data** (i.e Gaussian with Mean 0 and unit variance)
- Many elements used in the objective function of a learning algorithm assume that all the feature are centered around 0 have variance in the same order.
- If a feature has a variance that is orders of magnitude of others, it might dominate the objective function and make the estimator unable to learn from other features correctly as expected.

Different Types of Scaling

- Standard Scaler
- Robust Scaler
- MinMax Scaler
- Normalizer

Preprocessing and Scaling



Standard Scaler

- Standardize features by removing mean and scaling it to unit variance.
- Ensures that for each feature the Mean is 0 and Variance is 1.
- The standard score of a feature is calculated as:
    - $z = (x - u) / s$
        - u - mean of the training samples and
        - s - standard deviation of the training samples
- Centering and Scaling happens independently on each feature by computing the required statistics of the training dataset.
- This will not perform well if there are outlier in the dataset.

Standard Scaler Implementation

Robust Scaler

- Scales features using statistics that are robust to Outliers.
- This scaler removes Median and scales the data according to the quantile range (defaults to IQR)
- IQR - The range between 1st quartile(25th quantile) and the 3rd (75th quantile)
- Standardization of a dataset is a common requirement for many machine learning estimators.
- Typically this is done by removing the mean and scaling to unit variance. However, outliers can often influence the sample mean / variance in a negative way. In such cases, the median and the interquartile range often give better results.

Robust Scaler Implementation

MinMax Scaler

- Transform features by scaling each feature to a given range.
- Shifts the data such that all the features are between 0 and 1.
- For the two dimensional dataset this means all of the data is contained within in the rectangle created by X-axis between 0 and 1 and Y-axis between 0 and 1.
- This estimator scales and translates each feature individually such that it is in the given range on the training set, e.g. between zero and one.

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

MinMax Scaler Implementation

Normalizer

- Normalize samples individually to unit norm.
- Scaling inputs to unit norms is a common operation for text classification or clustering for instance.
- For instance the dot product of two l2-normalized TF-IDF vectors is the cosine similarity of the vectors and is the base similarity metric for the Vector Space Model commonly used by the Information Retrieval community.
- Types of Normalization

L1: $z = \|x\|_1 = \sum_{i=1}^{n} |x_i|$

L2: $z = \|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$

Max: $z = \max x_i$

- The options lead to different normalizations. if x is the vector of covariates of length n, and say that the normalized vector is $y=x/z$ then the three options denote what to use for z:

Dimensionality Reduction, Feature Extraction, and Manifold Learning

- Transforming data using unsupervised learning can have many motivations.
- The most common motivations are:
  - Visualization
  - Compressing the data, and finding a representation that is more informative for further processing.
- One of the simplest and most widely used algorithms for all of these is **Principal Component Analysis.**
- We'll also look at two other algorithms: **Non-Negative Matrix Factorization (NMF)**, which is commonly used for **feature extraction**, and **t-SNE**, which is commonly used for **visualization using two-dimensional scatter plots**.

Principal Component Analysis (PCA)

- Principal component analysis is a method that rotates the dataset in a way such that the rotated features are statistically uncorrelated.
- This rotation is often followed by selecting only a subset of the new features, according to how important they are for explaining the data.
- The following example (Figure 3-3) illustrates the effect of PCA on a synthetic two-dimensional dataset: