Luís Fernando Torres
linkedin.com/in/luuisotorres/

Data Scaling with Scikit-Learn

Luís Fernando Torres
linkedin.com/in/luuisotorres/

Data Scaling

- Data scaling is an important step to ensure that all features in a dataset are on the same scale (i.e., on the same range of values).
- This is essential for better performance in machine learning models.
- Scikit-learn library provides several rescaling methods like MinMaxScaler, MaxAbsScaler, StandardScaler, and RobustScaler.

Let's take a look at them!

Luís Fernando Torres
linkedin.com/in/luuisotorres/

MinMaxScaler

sklearn.preprocessing.MinMaxScaler

- Transforms features by scaling each feature to a given range within the dataset.

  std(x) = (x - x.min(axis=0)) ÷ (X.max(axis=0) - X.min(axis=0))

  x scaled = std(x) * (max - min) + min

- It's useful when the distribution is not Gaussian.
- By default, it scales features in the dataset between 0 and one [0,1].
- You can, however, specify a custom range.

Luís Fernando Torres
linkedin.com/in/luuisotorres/

MaxAbsScaler

sklearn.preprocessing.MaxAbsScaler

- Scale each feature individually by its maximum absolute value.

  x scaled = x ÷ max(abs(x))

- It scales the data to the range [-1,1] and it preserves the sparsity of the data.
- It works well on a mix of positive and negative values.
- It is considered robust to the presence of outliers.

Luís Fernando Torres
linkedin.com/in/luuisotorres/

StandardScaler

sklearn.preprocessing.StandardScaler

- Standardize features so they have a mean of 0 and a standard deviation of 1.

$$x_{scaled} = (x - mean(x)) \div std(x)$$

- It centers the data of each feature independently.
- It's particularly useful for data that has a Gaussian distribution, as well as linear models.
- It doesn't preserve the sparsity of the data.

Luís Fernando Torres
linkedin.com/in/luuisotorres/

RobustScaler

sklearn.preprocessing.RobustScaler

- Features are scaled using statistics robust to outliers, such as the Interquartile Range (IQR). It centers the data around the median.

$$x_{scaled} = (x - median(x)) \div IQR(x)$$

- By using the IQR, it is less affected by outliers compared to other scaling methods.
- It's also extremely useful with data that doesn't follow a normal distribution.

Luís Fernando Torres

linkedin.com/in/luuisotorres/

Thank you!

Follow for more content on Data Science, Machine Learning, Python and Quantitative Finance!