GeeksforGeeks

A computer science portal for geeks

Custom Search

COURSES

Login

HIRE WITH US

Python | How and where to apply Feature Scaling?

Feature Scaling or Standardization: It is a step of Data Pre Processing which is applied to independent variables or features of data. It basically helps to normalise the data within a particular range. Sometimes, it also helps in speeding up the calculations in an algorithm.

Package Used:

```
sklearn.preprocessing
```

Import:

```
from sklearn.preprocessing import StandardScaler
```

Formula used in Backend

Standardisation replaces the values by their Z scores.

$$z = \frac{x - \mu}{\sigma}$$

Mostly the **Fit** method is used for Feature scaling

```
fit(X, y = None)
Computes the mean and std to be used for later scaling.
```

```
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Read Data from CSV
data = read_csv('Geeksforgeeks.csv')
data.head()

# Initialise the Scaler
scaler = StandardScaler()

# To scale data
scaler.fit(data)
```

Why and Where to Apply Feature Scaling?

Real world dataset contains features that highly vary in magnitudes, units, and range. Normalisation should be performed when the scale of a feature is irrelevant or misleading and not should Normalise when the scale is meaningful.

The algorithms which use Euclidean Distance measure are sensitive to Magnitudes. Here feature scaling helps to weigh all the features equally.

Formally, If a feature in the dataset is big in scale compared to others then in algorithms where Euclidean distance is measured this big scaled feature becomes dominating and needs to be normalized.

Examples of Algorithms where Feature Scaling matters

1. **K-Means** uses the Euclidean distance measure here feature scaling matters.
2. **K-Nearest-Neighbours** also require feature scaling.
3. **Principal Component Analysis (PCA)**: Tries to get the feature with maximum variance, here too feature scaling is required.
4. **Gradient Descent**: Calculation speed increase as Theta calculation becomes faster after feature scaling.

Note: Naive Bayes, Linear Discriminant Analysis, and Tree-Based models are not affected by feature scaling.
In Short, any Algorithm which is **Not** Distance based is **Not** affected by Feature Scaling.

Recommended Posts:

ML | Feature Scaling - Part 1

ML | Feature Scaling – Part 2

Image Processing in Python (Scaling, Rotating, Shifting and Edge Detection)