

Feature Scaling

May 11, 2021

1 Feature Scaling (Normalization v/s Standardization)

1.1 1) Normalization

```
[1]: import pandas as pd
```

```
[2]: df = pd.read_csv('iris_csv.csv')  
df.head()
```

```
[2]:      sepallength  sepalwidth  petallength  petalwidth      class  
0           5.1           3.5           1.4           0.2  Iris-setosa  
1           4.9           3.0           1.4           0.2  Iris-setosa  
2           4.7           3.2           1.3           0.2  Iris-setosa  
3           4.6           3.1           1.5           0.2  Iris-setosa  
4           5.0           3.6           1.4           0.2  Iris-setosa
```

```
[3]: x = df.iloc[:, :-1]
```

```
[4]: from sklearn.preprocessing import MinMaxScaler
```

```
[5]: sc = MinMaxScaler()  
x_scaler = sc.fit_transform(x)
```

```
[6]: x_scaler[:5]
```

```
[6]: array([[0.22222222, 0.625      , 0.06779661, 0.04166667],  
          [0.16666667, 0.41666667, 0.06779661, 0.04166667],  
          [0.11111111, 0.5        , 0.05084746, 0.04166667],  
          [0.08333333, 0.45833333, 0.08474576, 0.04166667],  
          [0.19444444, 0.66666667, 0.06779661, 0.04166667]])
```

1.2 2) Standardization

```
[7]: from sklearn.preprocessing import StandardScaler
```

```
[8]: sc = StandardScaler()  
x_scaled = sc.fit_transform(x)
```

```
[9]: x_scaled[:5]
```

```
[9]: array([[ -0.90068117,  1.03205722, -1.3412724 , -1.31297673],  
          [-1.14301691, -0.1249576 , -1.3412724 , -1.31297673],  
          [-1.38535265,  0.33784833, -1.39813811, -1.31297673],  
          [-1.50652052,  0.10644536, -1.2844067 , -1.31297673],  
          [-1.02184904,  1.26346019, -1.3412724 , -1.31297673]])
```