

sklearn.metrics.silhouette_score

sklearn.metrics.silhouette_score(X, labels, *, metric='euclidean', sample_size=None, random_state=None, **kwargs)

[source]

Compute the mean Silhouette Coefficient of all samples.

The Silhouette Coefficient is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample. The Silhouette Coefficient for a sample is $(b - a) / \max(a, b)$. To clarify, b is the distance between a sample and the nearest cluster that the sample is not a part of. Note that Silhouette Coefficient is only defined if number of labels is $2 \leq n_labels \leq n_samples - 1$.

This function returns the mean Silhouette Coefficient over all samples. To obtain the values for each sample, use [silhouette_samples](#).

The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar.

Read more in the [User Guide](#).

Parameters:

X : {array-like, sparse matrix} of shape (n_samples, a, n_samples, a) if metric == "precomputed" or (n_samples, a, n_features) otherwise

An array of pairwise distances between samples, or a feature array.

labels : array-like of shape (n_samples,)

Predicted labels for each sample.

metric : str or callable, default='euclidean'

The metric to use when calculating distance between instances in a feature array. If metric is a string, it must be one of the options allowed by [pairwise_distances](#). If x is the distance array itself, use `metric="precomputed"`.

sample_size : int, default=None

The size of the sample to use when computing the Silhouette Coefficient on a random subset of the data. If `sample_size` is `None`, no sampling is used.

random_state : int, RandomState instance or None, default=None

Determines random number generation for selecting a subset of samples. Used when `sample_size` is not `None`. Pass an int for reproducible results across multiple function calls. See [Glossary](#).

****kwargs : optional keyword parameters**

Any further parameters are passed directly to the distance function. If using a `scipy.spatial.distance` metric, the parameters are still metric dependent. See the `scipy` docs for usage examples.

Returns:

silhouette : float

Mean Silhouette Coefficient for all samples.

References

[1]
[Peter J. Rousseeuw \(1987\). "Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis". Computational and Applied Mathematics 20: 53-65.](#)

[2]
[Wikipedia entry on the Silhouette Coefficient](#)

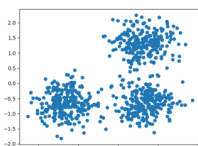
Examples

```
>>> from sklearn.datasets import make_blobs
>>> from sklearn.cluster import KMeans
>>> from sklearn.metrics import silhouette_score
>>> X, y = make_blobs(random_state=42)
>>> kmeans = KMeans(n_clusters=2, random_state=42)
>>> silhouette_score(X, kmeans.fit_predict(X))
0.49...
```

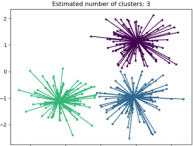
Examples using sklearn.metrics.silhouette_score



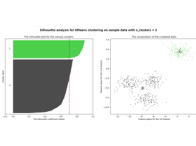
A demo of K-Means clustering on the handwritten digits data



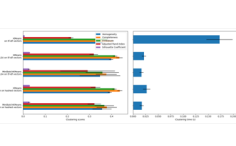
Demo of DBSCAN clustering algorithm



Demo of affinity propagation clustering algorithm



Selecting the number of clusters with silhouette analysis on KMeans clustering



Clustering text documents using k-means

