





```
[17]: # average shift
# Load libraries
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import MeanShift

#Load data
iris = datasets.load_iris()
features = iris.data

#standard
scaler = StandardScaler()
features_std = scaler.fit_transform(features)
# Creat classifier

cluster = MeanShift(n_jobs=-1)

# Train a model
model = cluster.fit(features_std)
```

```
[19]: # DBSCAN
# Load libraries
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import DBSCAN

#Load data
iris = datasets.load_iris()
features = iris.data

#standard
scaler = StandardScaler()
features_std = scaler.fit_transform(features)
# Creat classifier

cluster = DBSCAN(n_jobs=-1) # parameters eps, min_samples, metric
↳(euclidean,minkowski)

# Train a model
model = cluster.fit(features_std)
```

```
[23]: model.labels_
# tow clusters are considered 0 and 1, -1 is considered as noise
```

```
[23]: array([[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, -1, -1,  0,
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, -1, -1,
         0,  0,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  1,
```

```

    1, 1, 1, 1, 1, -1, -1, 1, -1, -1, 1, -1, 1, 1, 1, 1,
    -1, 1, 1, 1, -1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    -1, 1, -1, 1, 1, 1, 1, 1, -1, 1, 1, 1, 1, -1, 1, -1, 1,
    1, 1, 1, -1, -1, -1, -1, -1, 1, 1, 1, 1, -1, 1, 1, -1, -1,
    -1, 1, 1, -1, 1, 1, -1, 1, 1, 1, -1, -1, -1, 1, 1, 1, -1,
    -1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, -1, 1],
dtype=int64)

```

```

[27]: # Hierarchical clustering
# Load libraries
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import AgglomerativeClustering

#Load data
iris = datasets.load_iris()
features = iris.data

#standard
scaler = StandardScaler()
features_std = scaler.fit_transform(features)
# Creat classifier

cluster = AgglomerativeClustering(n_clusters=3) # parameters linkage(ward,
↳complete,average..), affinity (euclidean,minkowski..)

# Train a model
model = cluster.fit(features_std)

```

```
[29]: model.labels_
```

```

[29]: array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 0, 0, 0, 2, 0, 2, 0, 2, 0, 2, 2, 0, 2, 0, 2, 0,
    2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 0, 2, 0, 0, 2,
    2, 2, 2, 0, 2, 2, 2, 2, 2, 0, 2, 2, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int64)

```

```
[ ]:
```