ML Hands-on Workshop @ Elec, SFIT

Instructor: Santosh Chapaneri

Jan 2022

K-Means Clustering

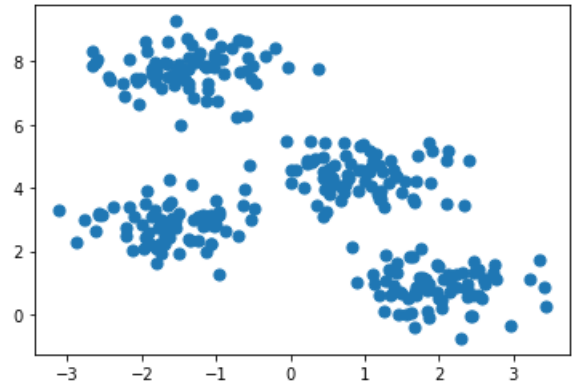
- Here we'll explore K Means Clustering, which is an unsupervised clustering technique.
- K-Means is an algorithm for unsupervised clustering: that is, finding clusters in data based on the data attributes alone (not the labels).
- K-Means is a relatively easy-to-understand algorithm. It searches for cluster centers which are the mean of the points within them, such that every point is closest to the cluster center it is assigned to.

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [3]:
```
from sklearn.datasets import make_blobs

X, y = make_blobs(n_samples=300, centers=4,
                  random_state=0, cluster_std=0.60)

plt.scatter(X[:, 0], X[:, 1], s=50);
```



- By eye, it is relatively easy to pick out the four clusters.
- If we were to perform an exhaustive search for the different segmentations of the data, however, the search space would be exponential in the number of points.
- Fortunately, there is a well-known Expectation Maximization (EM) procedure which scikit-learn implements, so that KMeans can be solved relatively quickly.

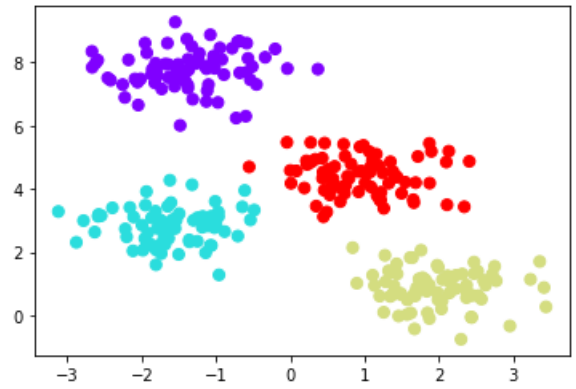
In [4]:
```
from sklearn.cluster import KMeans

est = KMeans(4)

est.fit(X)

y_kmeans = est.predict(X)

plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, s=50, cmap='rainbow');
```



In [5]:
```
from sklearn.metrics import calinski_harabasz_score

calinski_harabasz_score(X, y_kmeans)
```

Out[5]: 1210.0899142587816

- The algorithm identifies the four clusters of points in a manner very similar to what we would do by eye!