Unsupervised ML with K-Means

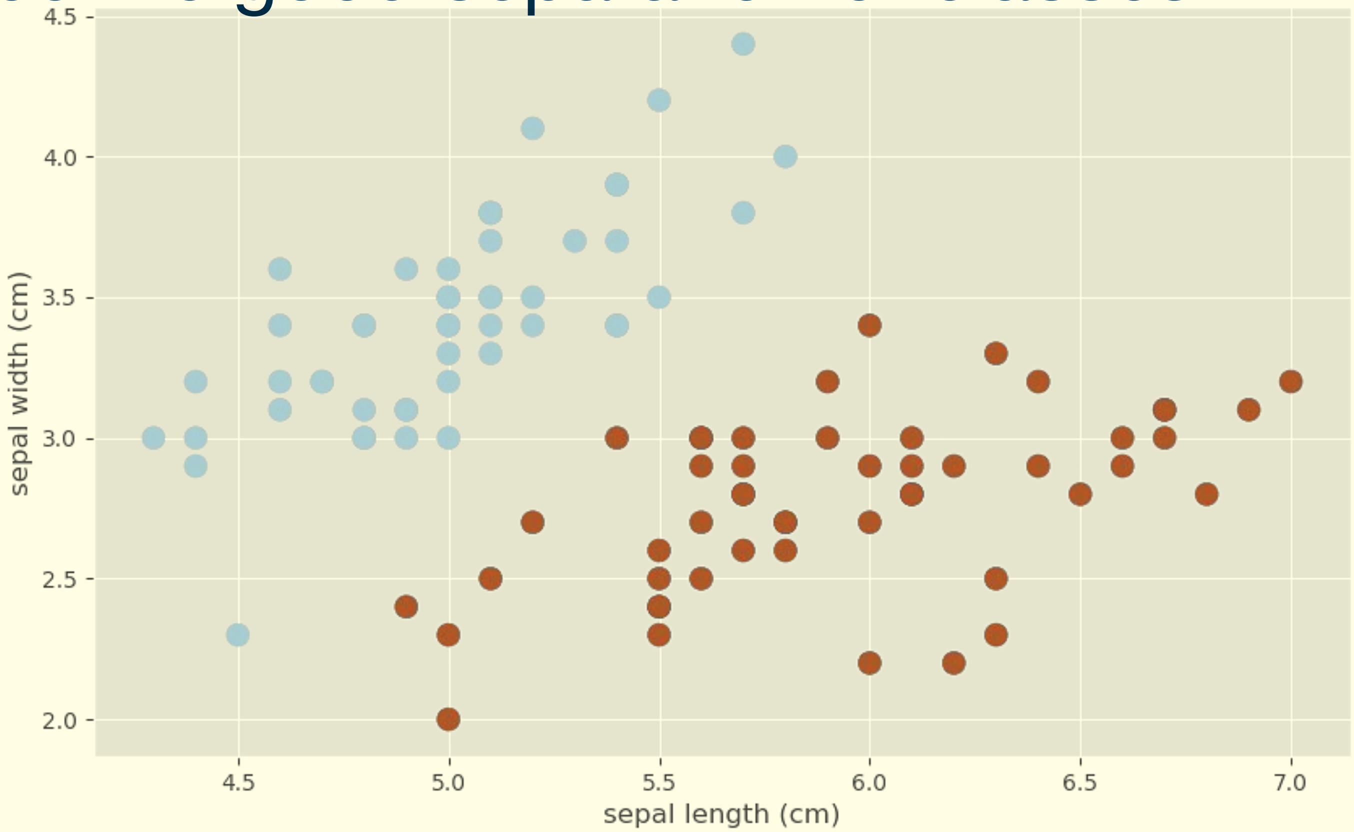
by Dane Brown

Easy Data Visualization

Visualize the Data

Check it out -> CV_ML

- Limit the plotting to the first two features
  - sepal length
  - sepal width
- Notice the good separation of classes in the figure



Why K-Means Clustering

- Many clustering algorithms are available in Scikit-Learn

- but k-Means is easy to understand

- k-Means searches for a given number of clusters within an unlabeled multidimensional dataset

- It simply defines the optimal clustering

How K-Means Clustering

- The **cluster centre** is the arithmetic **mean** of all the points belonging to that cluster
- Each point is closer to its own cluster centre than to other cluster centres

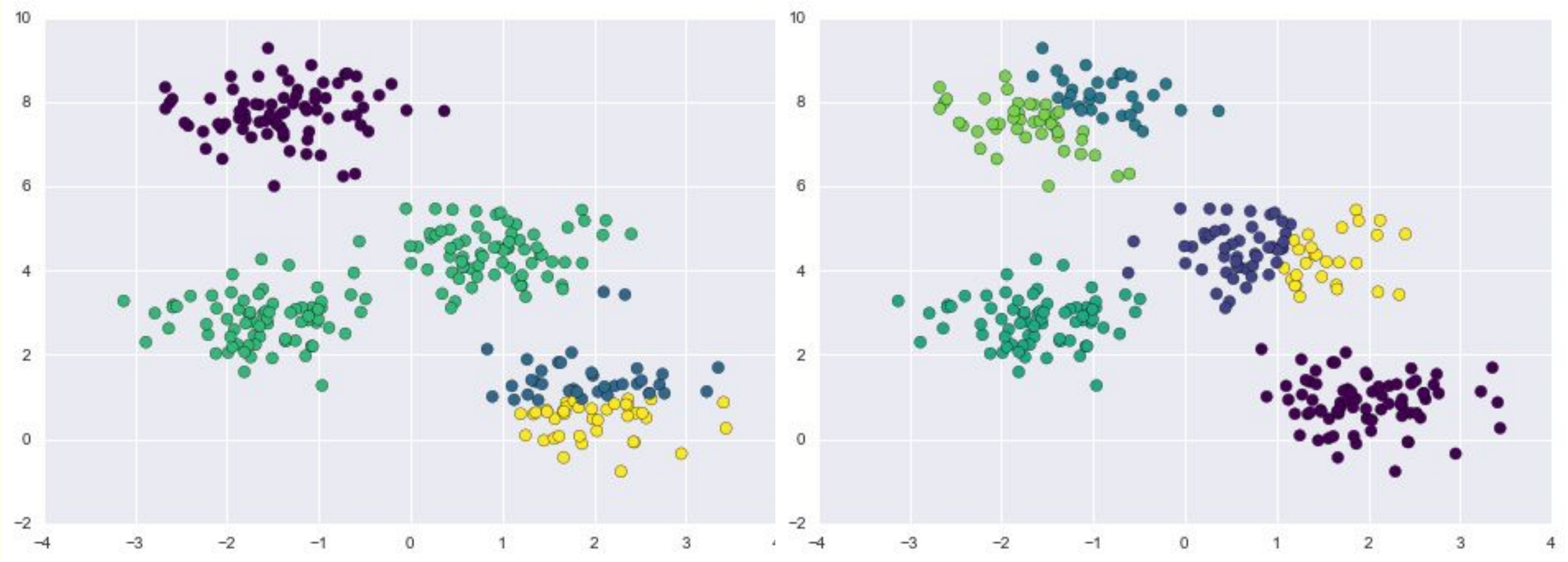
- **Algorithm:**
  - Exhaustive search is not necessary instead, use an iterative approach **expectation–maximization**
  - **E (expectation) step**
  - **M (maximization) step**
  - each iteration will always result in a better estimate of the cluster characteristics

How K-Means Clustering

1. Guess some cluster centres
2. Repeat until converged
   a. E-Step: assign points to the nearest cluster centre
   b. M-Step: set the cluster centres to the mean

Not a Global Optimum Solution

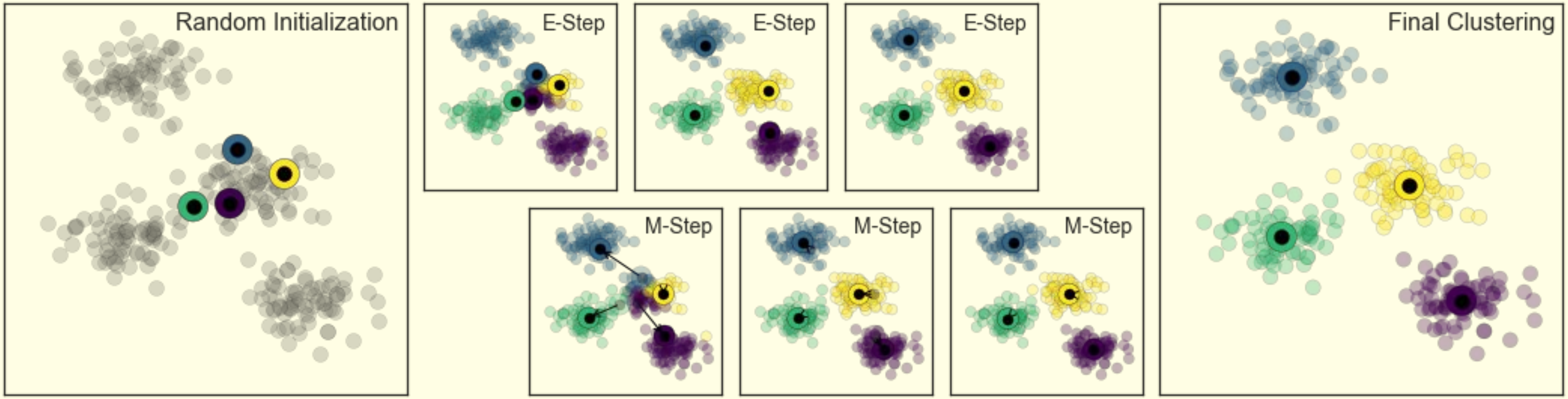
- Many k-Means implementations run for multiple starting guesses to find a more optimal solution
- Scikit avoids the problems below



Check it out -> CV_ML

How K-Means Clustering

- Only three iterations



Disadvantages of k-Means

- The final result may not always be optimal, as the starting point differs based on random seed

- k must be specified

- limited to linear cluster boundaries

- low for large numbers of samples

Check it out -> CV_ML

k-Means for Colour Compression

- Clustering can be used for colour compression within images.
- In many images
    - a large number of the colours will be unused
    - many pixels will have similar/identical colours.