Open | DSE

Introduction to Data Science with Python
Lecture 3: Unsupervised Learning

Vladimir Osin

Data Scientist/Engineer

Signify Research (formerly known as Philips Lighting)

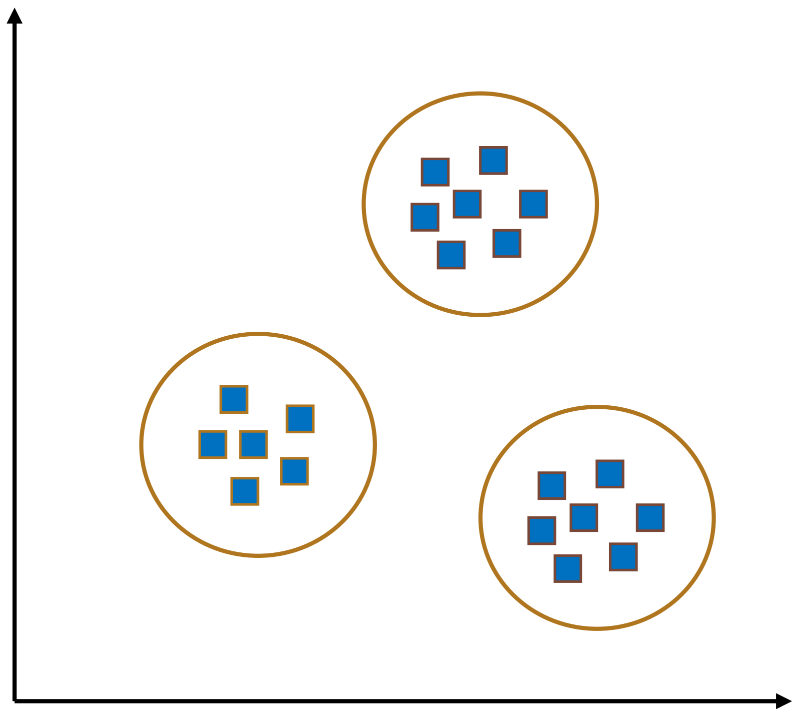
Outline

- Unsupervised Learning
  - Clustering
    - K-Means Clustering
    - DBSCAN
    - Clustering Validation
    - Scikit-learn Clustering Capabilities
  - Feature scaling
  - Dimensionality reduction
    - PCA
    - Random projection
    - Independent Component Analysis (ICA)

Unsupervised Learning

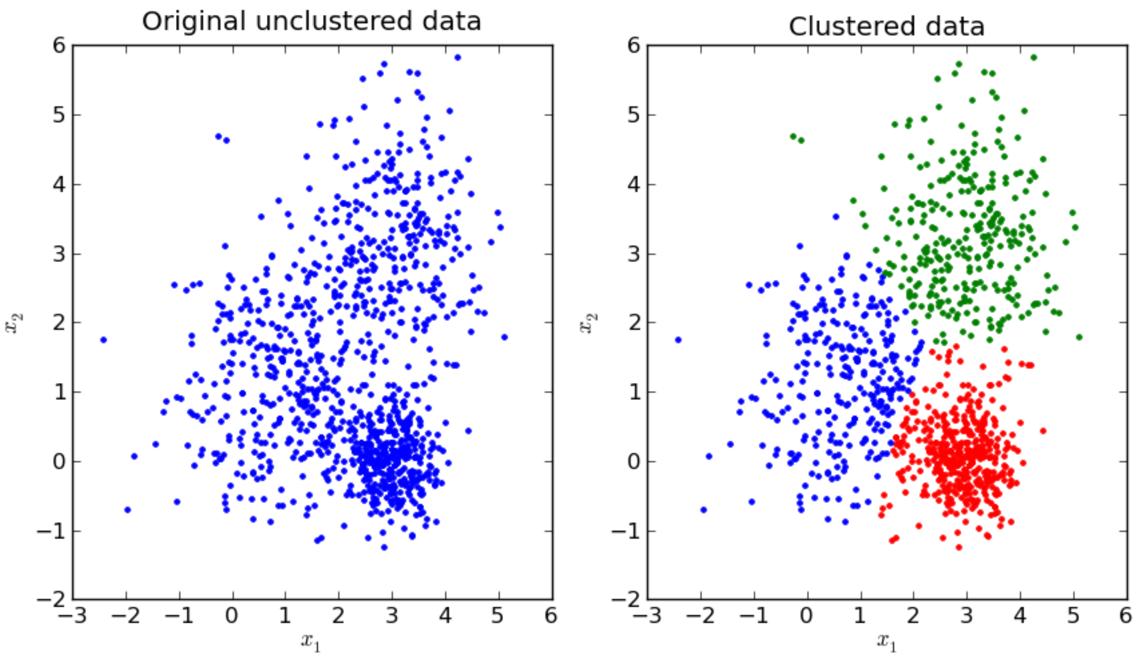
- The core idea of unsupervised learning is to find hidden patterns or structure in the unlabelled data.

- Types of tasks:
    - Clustering (**K Means**)
    - Dimensionality reduction (**PCA**)
    - Visualization (**T-SNE**)



K-Means Clustering

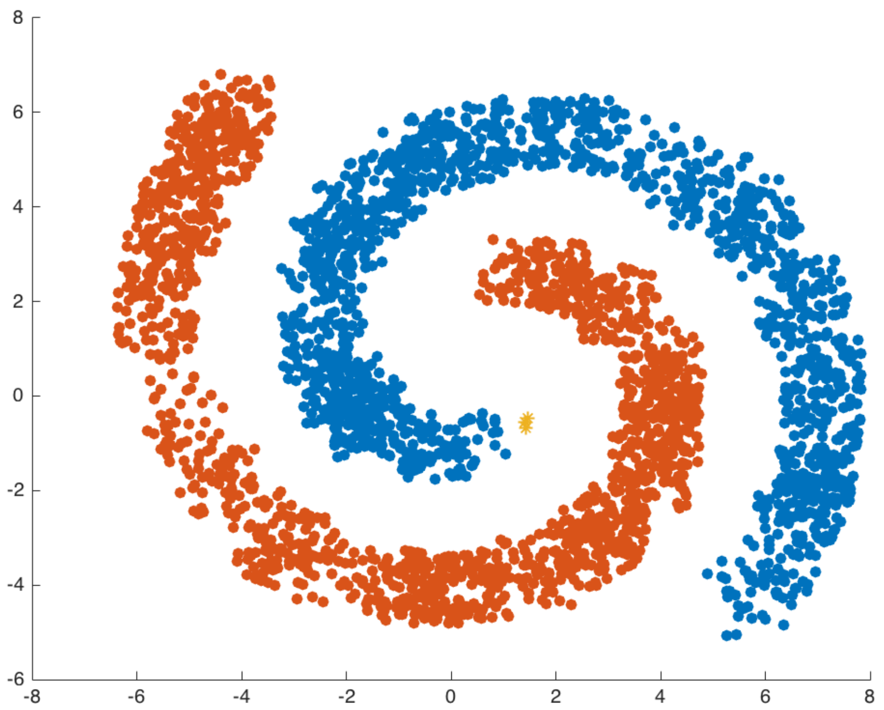
- The k-means algorithm captures the insight that each point in a cluster should be near to the center of that cluster.
- Works best on equally-sized and regularly shaped datasets
- Not capable to capture more complex types of data (works well when data points lie in the Euclidean space)
- Visualizing K-Means Clustering



Open | DSE

DBSCAN Clustering

- **Density-Based** Spatial Clustering of Applications with Noise captures the insight that clusters are dense groups of points. The idea is that if a particular point belongs to a cluster, it should be near to lots of other points in that cluster.
- <u>**Visualizing DBSCAN Clustering**</u>



Cluster Validation

- Silhouette coefficient
  - a – average distance to other samples in the same cluster
  - b – average distance to samples in the closest neighboring cluster
- Between -1 and 1
- Calculated for each data point at the dataset and then averaged
- Example

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Scikit Learn Clustering Capabilities

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers
Birch	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction.	Euclidean distance between points

Open | DSE

Open | DSE

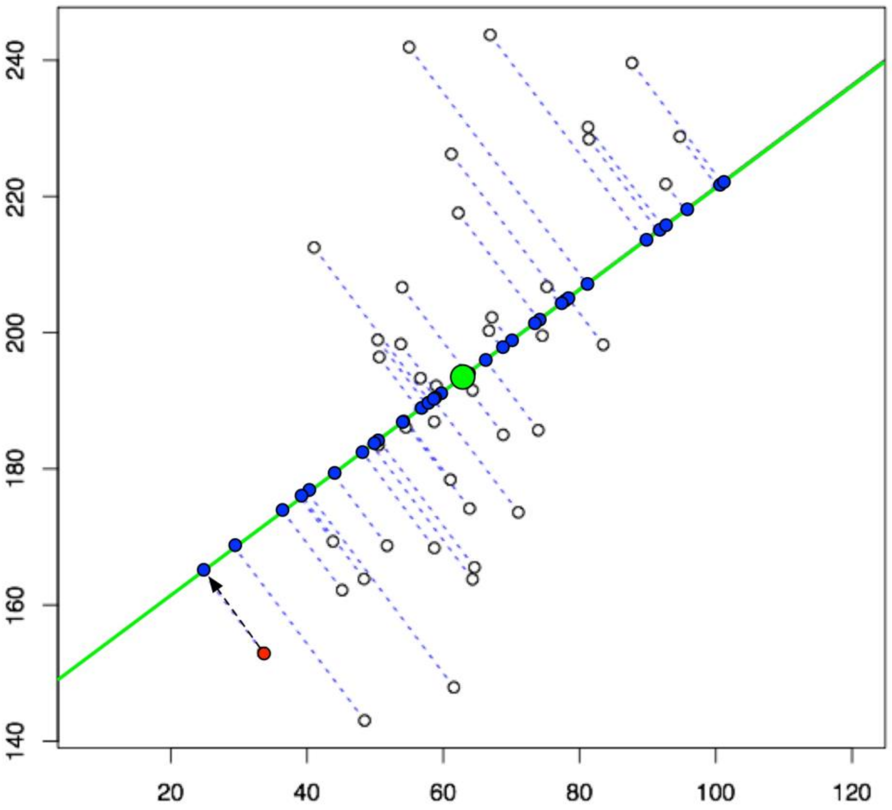
Feature scaling

- Usually your dataset contains features with different range, units and magnitudes.
- A lot of machine learning algorithms use Euclidian distance between data points in computations, which lead to errors without feature scaling.
- Some types of scaling:
  - Standardization, or mean removal and variance scaling
  - Scaling features to a range
  - Normalization (L1, L2 norms)
- When to scale?
  - K-nearest neighbor
  - Principal Component Analysis(PCA)
  - Speed up gradient descent

Open | DSE

Principal component analysis (PCA)

- Transformation of initial features to **principal components**
- Principal components are directions in data that maximize variance, when you perform projection to them.
- Idea is to maximize variance and minimize information loss
- Projection onto direction of maximal variance minimizes distance from high-dimensional data point to its new transformed value.
- When to use:
  - Dimensionality reduction
    - Visualizing high-dimensional data
    - Reduce noise (by throwing away less important principal components)
    - Prepare data for some ML algorithm
  - Latent variables that drive patterns at your data
  - Dataset is not too big



Random projection

- Almost same idea as PCA (but projection is **random**)
- Computationally more efficient than PCA on the larger datasets
- Based on Johnson–Lindenstrauss lemma

Independent component analysis (ICA)

- ICA assumes that features are mixtures of independent sources
- ICA trying to separate those sources
- Sources are statistically independent from each other

Assignment 3

- Experimenting with other clustering methods
- What about evaluation metrics, except Silhouette Coefficient ?
- Use provided notebook as reference
- Scikit-Learn Clustering