THINKFUL

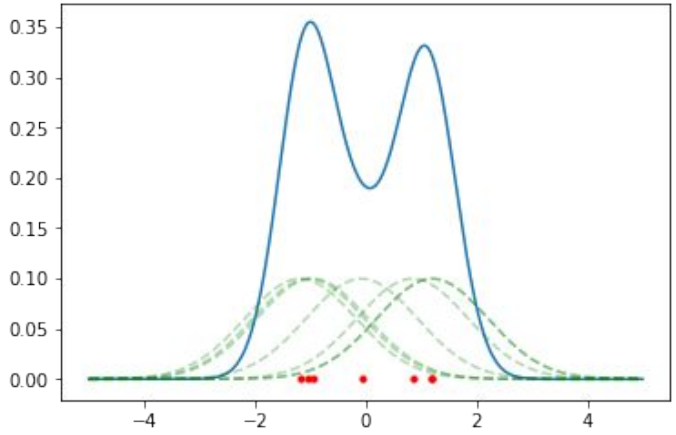
Mean Shift

DATA SCIENCE

Warm Up

Consider the following questions:

- A common technique for estimating the probability density function of a random variable is Kernel Density Estimation (KDE). Take the 1-D example below, which contains a synthetic dataset and the corresponding kernel density estimate. How could we use the estimated density function to identify clusters?



Agenda

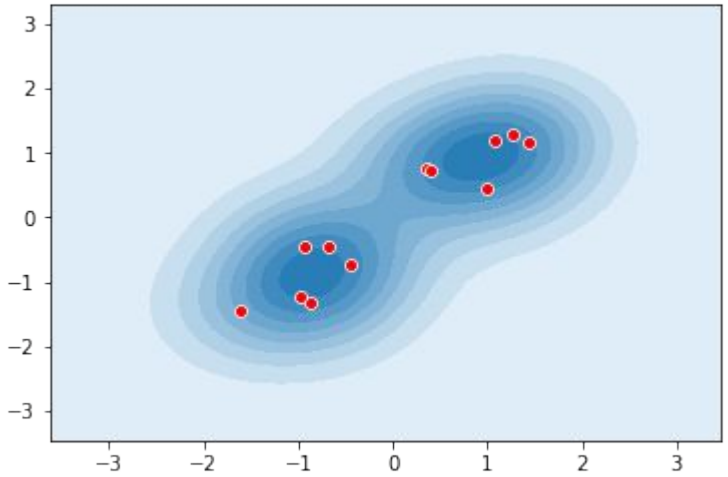
- Overview
- Kernel Density Estimation
- Mean Shift algorithm steps
- Advantages and Disadvantages

Overview

Mean Shift Clustering is built on the more general mean-shift technique, which is used to find the maxima of a density function

- Intuitively, a good density estimator will place the highest densities on the regions with the most points
- The mean-shift algorithm then forces each observation to “climb the hills” of the estimated density function toward a local maximum, which corresponds to a cluster
- Does not require any pre-specification of the number of clusters, only the bandwidth of the kernel function chosen to estimate the density

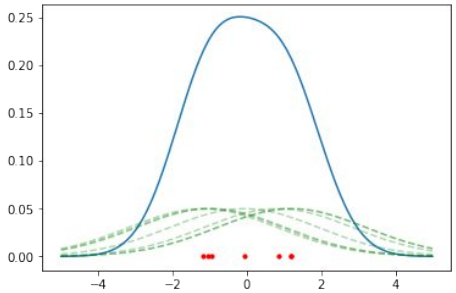
Overview



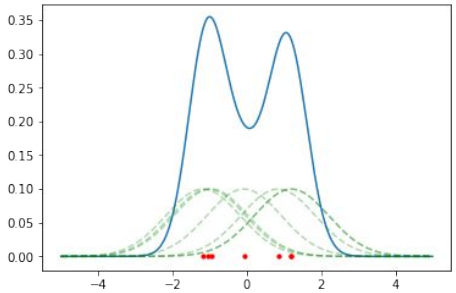
A contour map of the density function estimated by KDE, illustrating the “hill-climbing” aspect of mean-shift clustering

Kernel Density Estimation

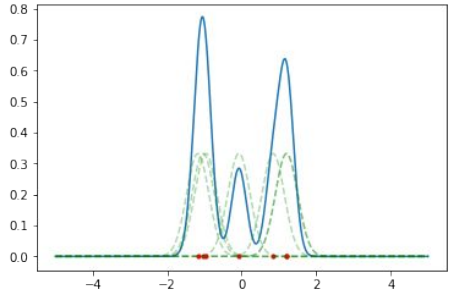
Mean shift builds on the idea of kernel density estimation (KDE), which is a non-parametric method for estimating the probability density function of a random variable. The kernel density estimator is characterized by the choice of kernel function and bandwidth (a smoothing parameter).



High Bandwidth/Smoothing



Medium Bandwidth/Smoothing

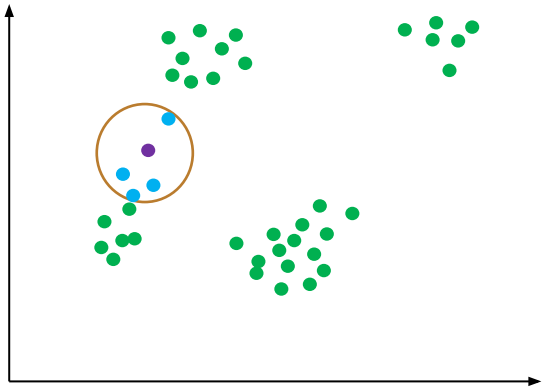


Low Bandwidth/Smoothing

The Mean-Shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

1. For a given observation x, determine which other observations are in the neighborhood of x (within some radius r using the distance function D)



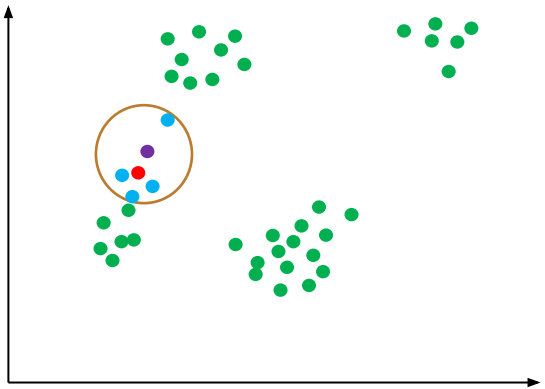
The Mean-Shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

2. Calculate the weighted mean density of all points in the neighborhood of x, $N(x)$, using the kernel K:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x, x_i) x_i}{\sum_{x_i \in N(x)} K(x, x_i)}$$

- Original Data Point
- Weighted Mean Density Of Neighboring (Blue) Points

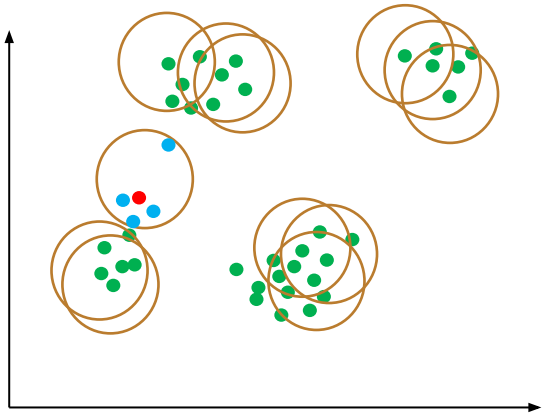


The Mean-Shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

3. Update x to $m(x)$. The difference between the two values is called the mean-shift.
4. Repeat steps 1 to 3 for all observations in the data set. This constitutes one iteration of the algorithm.
5. Repeat steps 1 to 4 until convergence.

- Mean-shifted update of original observation



UP NEXT

Additional Considerations

Kernels & Distance Functions

The most commonly chosen distance function is Euclidean distance

- Typical of algorithms that rely on calculating centroids, particularly means (think back to k-means)

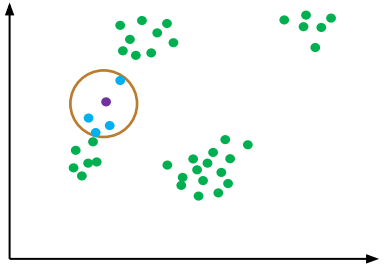
Two popular choices for kernels:

- Flat

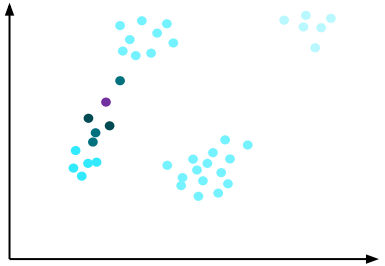
$$K(x) = \begin{cases} 1 & if\ \|x\| < \lambda \\ 0 & if\ \|x\| > \lambda \end{cases}$$

- Gaussian/RBF

$$K(x) = e^{-\frac{x}{2\sigma^2}}$$



Flat: every point within the neighborhood is equally weighted



Gaussian/RBF: every point in the dataset is weighted; points that are further away receive lower weight

Comparison To K-Means & DBSCAN

The mean-shift algorithm utilizes techniques that are strongly reminiscent of previously covered techniques, such as K-Means and DBSCAN

Like K-Means, the mean-shift algorithm involves calculating centroids and updates based on the new centroid locations

- However, the centroids are calculated in small neighborhoods around each observation, rather than a more global/cluster-level
- Furthermore, the centroid calculation has more flexibility than a regular mean, thanks to kernels

Comparison To K-Means & DBSCAN

Like DBSCAN, the density of points in the neighborhood of each observation drives the resulting clustering results

- However, DBSCAN uses density to establish clusters through connectivity (i.e. notions of density-reachability), while mean-shift uses density to push observations towards the peaks of the estimated density function (through KDE)

Advantages & Disadvantages

Advantages

- A low number of parameters (typically just one, the window size/bandwidth r, which often corresponds to a physical quantity such as real-world distance)
- Can identify a variable number of clusters, no need to specify the number upfront
- Can identify somewhat irregularly shaped clusters
- Relatively robust to outliers

Disadvantages

- Choosing r can sometimes be difficult
  - Large bandwidth leads to a smooth, flat density estimate with potentially very large clusters
  - Small bandwidth leads to sharp, pointy density estimates with many local maxima and possibly an undesirably large number of clusters
  - Output can be very sensitive to value of r
- Scales poorly to high-dimensional datasets; tends towards $O(n^2)$
- Generally not suitable for categorical or mixed data

Summary

- Mean-shift clustering extends the notion of kernel density estimation (KDE) to a clustering context, seeking to shift points towards the local maxima of the estimated density function
- The algorithm follows an iterative process that estimates the weighted mean density within a small neighborhood of each point, then updates the observation to the estimated mean (a mean-shift)
- Mean-shift clustering has only one parameter and doesn’t need the number of clusters to be specified in advance, but is fairly slow and unsuitable for categorical and mixed data

EXERCISE

1. Jupyter notebook
2. Data set: Financial well-being survey results
3. Isolate the “score” subset of questions
4. Perform mean-shift clustering and evaluate the results
5. Tune the model and re-evaluate the results

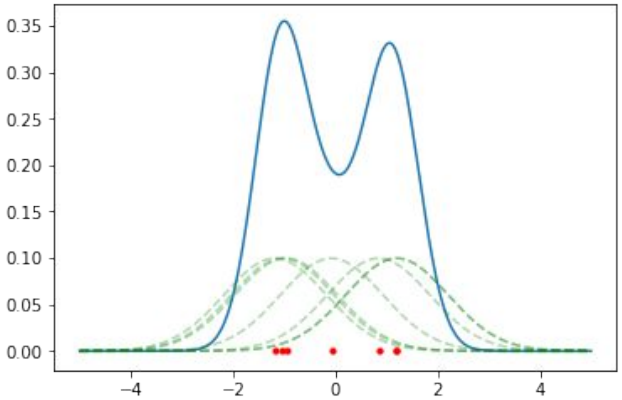
Thank You

Miscellaneous Clustering Algorithms, Part 1: Mean Shift

Warm Up

Consider the following questions:

- A common technique for estimating the probability density function of a random variable is Kernel Density Estimation (KDE). Take the 1-D example below, which contains a synthetic dataset and the corresponding kernel density estimate. How could we use the estimated density function to identify clusters?



High Level Agenda

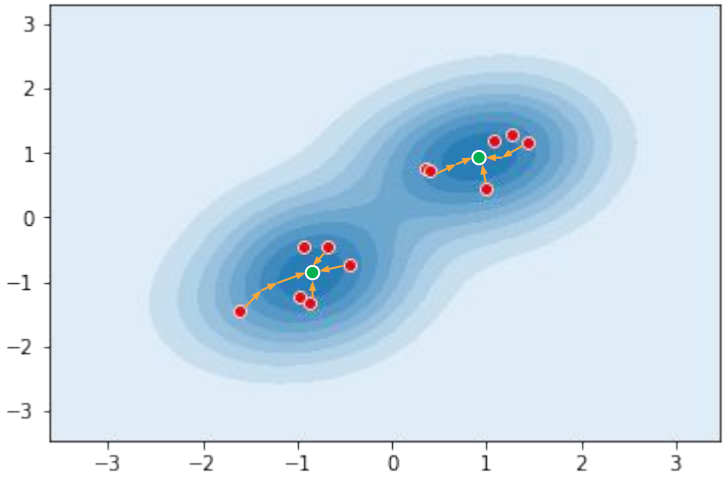
- Overview
- Kernel Density Estimation
- Mean Shift algorithm steps
- Advantages and Disadvantages

Overview

Overview

Mean Shift Clustering is built on the more general mean-shift technique, which is used to find the maxima of a density function

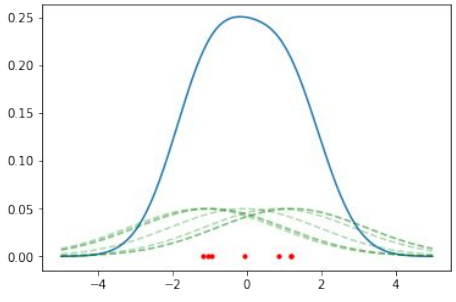
- Intuitively, a good density estimator will place the highest densities on the regions with the most points
- The mean-shift algorithm then forces each observation to “climb the hills” of the estimated density function toward a local maximum, which corresponds to a cluster
- Does not require any pre-specification of the number of clusters, only the bandwidth of the kernel function chosen to estimate the density



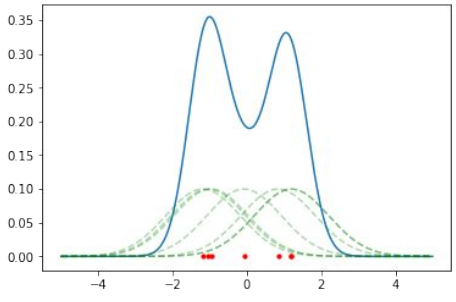
A contour map of the density function estimated by KDE, illustrating the “hill-climbing” aspect of mean-shift clustering

Kernel Density Estimation

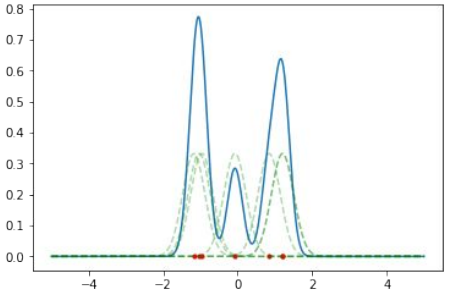
Mean shift builds on the idea of kernel density estimation (KDE), which is a non-parametric method for estimating the probability density function of a random variable. The kernel density estimator is characterized by the choice of kernel function and bandwidth (a smoothing parameter).



High Bandwidth/Smoothing



Medium Bandwidth/Smoothing

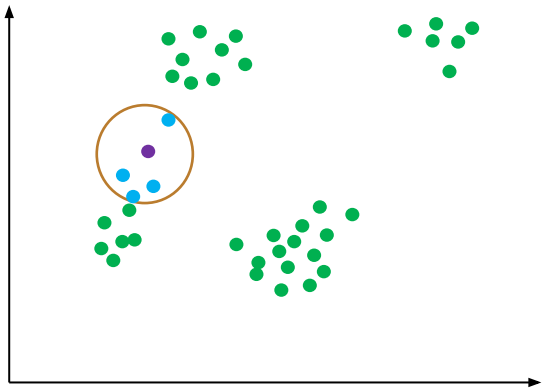


Low Bandwidth/Smoothing

The Mean-shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

1. For a given observation x, determine which other observations are in the neighborhood of x (within some radius r using the distance function D)



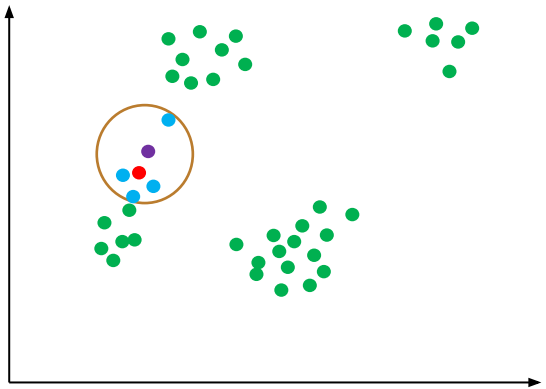
The Mean-shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

2. Calculate the weighted mean density of all points in the neighborhood of x, $N(x)$, using the kernel K:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x, x_i) x_i}{\sum_{x_i \in N(x)} K(x, x_i)}$$

- Original Data Point
- Weighted Mean Density Of Neighboring (Blue) Points

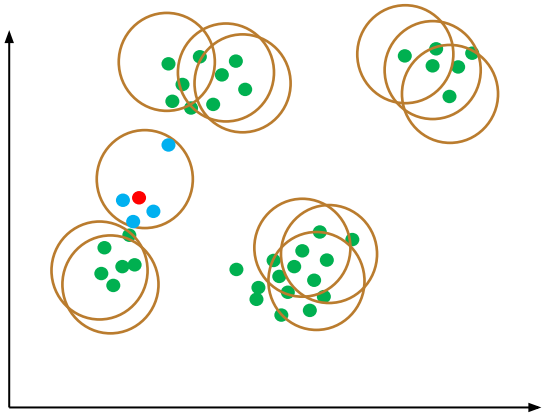


The Mean-shift Algorithm

Mean-shift clustering is an iterative algorithm; after choosing a distance function D and a kernel K, the algorithm works as follows:

3. Update x to $m(x)$. The difference between the two values is called the mean-shift.
4. Repeat steps 1 to 3 for all observations in the data set. This constitutes one iteration of the algorithm.
5. Repeat steps 1 to 4 until convergence.

- Mean-shifted update of original observation



Additional Considerations

Kernels and Distance Functions

The most commonly chosen distance function is Euclidean distance

- Typical of algorithms that rely on calculating centroids, particularly means (think back to k-means)

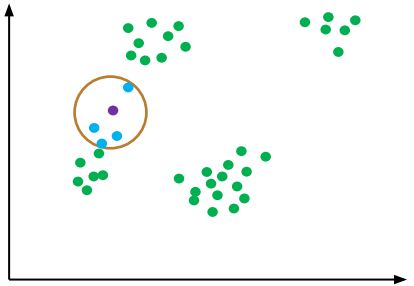
Two popular choices for kernels:

- Flat

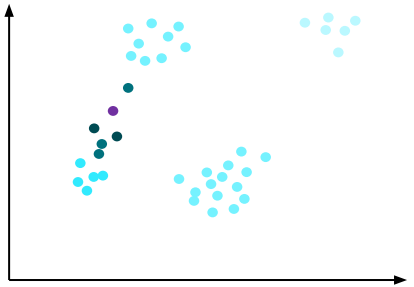
$$K(x) = \begin{cases} 1 & if\ \|x\| < \lambda \\ 0 & if\ \|x\| > \lambda \end{cases}$$

- Gaussian/RBF

$$K(x) = e^{-\frac{x}{2\sigma^2}}$$



Flat: every point within the neighborhood is equally weighted



Gaussian/RBF: every point in the dataset is weighted; points that are further away receive lower weight

Comparison to K-Means and DBSCAN

The mean-shift algorithm utilizes techniques that are strongly reminiscent of previously covered techniques, such as K-Means and DBSCAN

Like K-Means, the mean-shift algorithm involves calculating centroids and updates based on the new centroid locations

- However, the centroids are calculated in small neighborhoods around each observation, rather than a more global/cluster-level
- Furthermore, the centroid calculation has more flexibility than a regular mean, thanks to kernels

Like DBSCAN, the density of points in the neighborhood of each observation drives the resulting clustering results

- However, DBSCAN uses density to establish clusters through connectivity (i.e. notions of density-reachability), while mean-shift uses density to push observations towards the peaks of the estimated density function (through KDE)

Advantages and Disadvantages

Advantages and Disadvantages

Advantages

- A low number of parameters (typically just one, the window size/bandwidth r, which often corresponds to a physical quantity such as real-world distance)
- Can identify a variable number of clusters, no need to specify the number upfront
- Can identify somewhat irregularly shaped clusters
- Relatively robust to outliers

Disadvantages

- Choosing r can sometimes be difficult
  - Large bandwidth leads to a smooth, flat density estimate with potentially very large clusters
  - Small bandwidth leads to sharp, pointy density estimates with many local maxima and possibly an undesirably large number of clusters
  - Output can be very sensitive to value of r
- Scales poorly to high-dimensional datasets; tends towards $O(n^2)$
- Generally not suitable for categorical or mixed data

Recap

- Mean-shift clustering extends the notion of kernel density estimation (KDE) to a clustering context, seeking to shift points towards the local maxima of the estimated density function
- The algorithm follows an iterative process that estimates the weighted mean density within a small neighborhood of each point, then updates the observation to the estimated mean (a mean-shift)
- Mean-shift clustering has only one parameter and doesn’t need the number of clusters to be specified in advance, but is fairly slow and unsuitable for categorical and mixed data

Exercise:
Mean Shift

- Jupyter notebook
- Data set: Financial well-being survey results
- Isolate the “score” subset of questions
- Perform mean-shift clustering and evaluate the results
- Tune the model and re-evaluate the results