Clustering
Clustering

Discover groups such that samples within a group are more similar to each other than samples across groups.
Discover groups such that samples within a group are more similar to each other than samples across groups.
Clustering

Discover groups such that samples within a group are more similar to each other than samples across groups.
Clustering

Discover groups such that samples within a group are more similar to each other than samples across groups.

Groups not known a priori (unsupervised learning)
Ingredients of clustering

Dissimilarity function between samples.
Ingredients of clustering

Dissimilarity function between samples.

Loss function to measure “goodness” of clustering
Ingredients of clustering

Dissimilarity function between samples.

Loss function to measure “goodness” of clustering

Algorithm for optimizing the loss function
Center-based clustering: K-means
K-means clustering

Distance function = Euclidean distance (squared)

Center-based clustering
K-means clustering

Distance function = Euclidean distance (squared)

Center-based clustering

\[
\min_{\mu} \sum_{i=1}^{k} \sum_{x \in C_i} | x - \mu_i |^2
\]
K-means clustering

How do we optimize the K-means objective function?
K-means clustering

How do we optimize the K-means objective function?

- Block-coordinate descent (alternating minimization)
  - Fix $\mu$, optimize $C$
  - Fix $C$, optimize $\mu$
- What are the corresponding updates? (derive on board)
Application:
K-means for image segmentation
K-means clustering

- 8 iterations of the K-means procedure, K=5
K-means clustering

- Effect of random initialization, $K=5$

- Effect of the choice of $K$

$K=3$  $K=5$  $K=8$  $K=15$
K-means clustering using intensity alone and color alone
K-means using color alone, 11 segments
K-means using color alone, 11 segments.
Questions about K-means

Is convergence guaranteed?

What is the corresponding running time per iteration?

Limitations/problems?
Questions about K-means

Is convergence guaranteed?

What is the corresponding running time per iteration?

Limitations/problems?

- Sensitive to initialization
- K is assumed as known
- Cannot handle well non-convex clusters
- Sensitive to outliers
- Hard assignments
Density-based clustering: Mean-shift
Density-based clustering

IDEA:

- Clusters are locations where data points have high density
- Assume data are IID samples from some underlying probability distribution
- Find local maxima (modes) of probability distribution
Density-based clustering

**IDEA:**

- Clusters are locations where data points have high density
- Assume data are IID samples from some underlying probability distribution
- Find local maxima (modes) of probability distribution

**PROBLEMS TO SOLVE:**

- What probability distribution to use?
- How to find its modes?
Mean Shift Theory

Slides credit: Ukrainitz & Sarel
What is Mean Shift?

A tool for:
Finding modes in a set of data samples, manifesting an underlying probability density function (PDF) in $\mathbb{R}^n$.

PDF in feature space
- Color space
- Scale space
- Actually any feature space you can conceive...

Data → Non-parametric Density GRADIENT Estimation (Mean Shift) → PDF Analysis
Non-Parametric Density Estimation

Assumption: The data points are sampled from an underlying PDF

Data point density implies PDF value!

Assumed Underlying PDF  Real Data Samples
Non-Parametric Density Estimation

Assumed Underlying PDF

Real Data Samples
Non-Parametric Density Estimation

Assumed Underlying PDF

Real Data Samples
**Parametric Density Estimation**

**Assumption**: The data points are sampled from an underlying PDF

\[
PDF(x) = \sum_{i} c_i \cdot e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}
\]

Assumed Underlying PDF

Real Data Samples

Estimate
Kernel Density Estimation
Parzen Windows - Function Forms

\[ P(x) = \frac{1}{n} \sum_{i=1}^{n} K(x - x_i) \]

A function of some finite number of data points \(x_1 \ldots x_n\)

In practice one uses the forms:

\[ K(x) = c \prod_{i=1}^{d} k(x_i) \quad \text{or} \quad K(x) = ck(\|x\|) \]

Same function on each dimension

Function of vector length only
Kernel Density Estimation

Various Kernels

\[ P(x) = \frac{1}{n} \sum_{i=1}^{n} K(x - x_i) \]

A function of some finite number of data points \(x_1 \ldots x_n\)

**Examples:**

- **Epanechnikov Kernel**
  \[ K_E(x) = \begin{cases} 
  c \left(1 - \|x\|^2\right) & \|x\| \leq 1 \\
  0 & \text{otherwise}
  \end{cases} \]

- **Uniform Kernel**
  \[ K_U(x) = \begin{cases} 
  c & \|x\| \leq 1 \\
  0 & \text{otherwise}
  \end{cases} \]

- **Normal Kernel**
  \[ K_N(x) = c \cdot \exp\left(-\frac{1}{2} \|x\|^2\right) \]
Kernel Density Estimation

Gradient

\[ \nabla P(x) = \frac{1}{n} \sum_{i=1}^{n} \nabla K(x - x_i) \]

Give up estimating the PDF! Estimate **ONLY** the gradient

Using the Kernel form:

\[ K(x - x_i) = ck \left( \frac{||x - x_i||^2}{h} \right) \]

We get:

\[ \nabla P(x) = \frac{c}{n} \sum_{i=1}^{n} \nabla k_i = \frac{c}{n} \left[ \sum_{i=1}^{n} g_i \right] \left[ \frac{\sum_{i=1}^{n} x_i g_i}{\sum_{i=1}^{n} g_i} - x \right] \]

\[ g(x) = -k'(x) \]
\[ \nabla P(x) = \frac{c}{n} \sum_{i=1}^{n} \nabla k_i = \frac{c}{n} \left[ \sum_{i=1}^{n} g_i \right] \left[ \frac{\sum_{i=1}^{n} x_i g_i}{\sum_{i=1}^{n} g_i} - x \right] \]

\[ g(x) = -k'(x) \]
Computing The Mean Shift

$$\nabla P(x) = \frac{c}{n} \sum_{i=1}^{n} \nabla k_i = \frac{c}{n} \left[ \sum_{i=1}^{n} g_i \right] = \frac{1}{n} \left[ \sum_{i=1}^{n} x_i g_i \right] \quad \text{subject to} \quad \sum_{i=1}^{n} g_i = 1$$

Yet another Kernel density estimation!

Simple Mean Shift procedure:
- Compute mean shift vector
  - $$m(x) = \left[ \frac{\sum_{i=1}^{n} x_i g \left( \frac{\|x - x_i\|^2}{h} \right)}{\sum_{i=1}^{n} g \left( \frac{\|x - x_i\|^2}{h} \right)} - x \right]$$
- Translate the Kernel window by $$m(x)$$

$$g(x) = -k'(x)$$
Intuitive Description

**Objective**: Find the densest region

Distribution of identical billiard balls
**Objective**: Find the densest region

Distribution of identical billiard balls
Objective: Find the densest region

Distribution of identical billiard balls
Intuitive Description

**Distribution of identical billiard balls**

**Objective**: Find the densest region

Distribution of identical billiard balls
Intuitive Description

Objective: Find the densest region

Distribution of identical billiard balls
Intuitive Description

Objective: Find the densest region
Distribution of identical billiard balls

Region of interest
Center of mass
Mean Shift vector
Intuitive Description

**Objective**: Find the densest region

Distribution of identical billiard balls
Mean Shift Properties

- Automatic convergence speed – the mean shift vector size depends on the gradient itself.
- Near maxima, the steps are small and refined
- Convergence is guaranteed for infinitesimal steps only → infinitely convergent, (therefore set a lower bound)
- For Uniform Kernel ( ), convergence is achieved in a finite number of steps
- Normal Kernel ( ) exhibits a smooth trajectory, but is slower than Uniform Kernel ( ).
Real Modality Analysis

Tessellate the space with windows
Run the procedure in parallel
The blue data points were traversed by the windows towards the mode.
Real Modality Analysis

An example

Window tracks signify the steepest ascent directions
Mean Shift Strengths & Weaknesses

**Strengths:**

- Application independent tool
- Suitable for real data analysis
- Does not assume any prior shape (e.g. elliptical) on data clusters
- Can handle arbitrary feature spaces
- Only ONE parameter to choose
- $h$ (window size) has a physical meaning, unlike K-Means

**Weaknesses:**

- The window size (bandwidth selection) is not trivial
- Inappropriate window size can cause modes to be merged, or generate additional “shallow” modes ➔ Use adaptive window size
Mean Shift Applications
Clustering

Cluster: All data points in the *attraction basin* of a mode

Attraction basin: the region for which all trajectories lead to the same mode

*Mean Shift: A robust Approach Toward Feature Space Analysis, by Comaniciu, Meer*
Clustering
Synthetic Examples

Simple Modal Structures

Complex Modal Structures
Clustering
Real Example

Feature space:
L*u*v representation

Initial window enters

M

pruning
Clustering
Real Example

L*u*v space representation
Clustering
Real Example

2D (L*u) space representation

Final clusters