Unsupervised learning

Unsupervised Learning:
Given a set of input points \((X_i)_{i=1}^n\):

- **Clustering:** Construction of a grouping of the points into sets of similar points, the so called clusters.
- **Density Estimation:** Estimation of the distribution of the input points over the input space \(\mathcal{X}\). Related problem: Outlier detection.
- **Dimensionality Reduction:** Construction of a mapping \(\phi : \mathcal{X} \rightarrow \mathbb{R}^m\), where the dimensionality \(m\) of the target space is usually much smaller than that of the input space \(\mathcal{X}\). Generally, the mapping should preserve properties of the input space \(\mathcal{X}\) e.g. distances.
Clustering

- Goal of clustering,
- k-means clustering (prototype-based clustering)
- Spectral clustering (graph-based clustering),
- Agglomerative and hierarchical clustering,
- Density based clustering.

Clustering is one instance of unsupervised learning
What is clustering?

Clustering:
Construction of a grouping of the points into sets of similar points, the so-called clusters.

- no generally accepted objective for clustering $\implies$ without specifying a suitable objective clustering is ill-defined,
- clustering objective depends usually on application,
- in clustering the modeling aspect is even more important than in supervised learning $\implies$ do not use a clustering method if you have not understood what the objective implies!
Prototype based clustering

K-means clustering

- **Goal:** find prototypes $\mu_i$, $i = 1, \ldots, k$ which represent the data in an optimal way (what does that mean?),

- **Objective:** denote by $C_i$ the $i$-th cluster (set of points) which is represented by the prototype $\mu_i$,

$$\arg\min_{(C_1, \mu_1), \ldots, (C_k, \mu_k)} \sum_{i=1}^k \sum_{x_j \in C_i} \|x_j - \mu_i\|^2,$$

where $\|\cdot\|$ is the Euclidean norm,

- **True Goal:**
  1. finds sphere-like clusters in the data,
  2. heavily influenced by outliers,
  3. non-sphere like clusters are hard to fit.
K-means clustering:
- k-means is combinatorial optimization problem,
- simple iterative algorithm - converges fast but finds only local minimum.

Lloyd’s algorithm for k-means clustering:
1. initialize centers $\mu_i$,
2. do classify all samples according to closest $\mu_i$, $i = 1, \ldots, k$
3. recomputes $\mu_i$ as the mean of the points in cluster $C_i$ for $i = 1, \ldots, k$
4. while no change in $\mu_i$, $i = 1, \ldots, k$,
5. return $\mu_1, \ldots, \mu_k$.

Steps are optimal for fixed clusters resp. fixed centers.
Illustration of k−means for 3 Gaussians

Center 1
Center 2
Center 3
Middle: $k$ is chosen too large.

Right: The two moons dataset - clusters are not of spherical shape.

$$J(k) = \min_{(C_1, \mu_1), \ldots, (C_k, \mu_k)} \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2,$$

$\implies$ monotonically decreasing in $k$ - not useful for choosing $k$!
Spectral Clustering:

- an instance of graph-based clustering,
- First attempts can be traced back to Donath and Hoffman and Fiedler in 1973,
- very popular clustering algorithm since it can find clusters of almost arbitrary shape,
- rich theoretical background.

⇒ based on eigenvectors of the graph Laplacian.

In the following: we deal with weighted, undirected graphs $G = (V, W)$
⇒ symmetric weight matrix $w_{ij} = w_{ji}$,
⇒ degree of vertex $i$, $d(i) = \sum_{j=1}^{n} w_{ij}$, degree matrix $D_{ij} = d_i \delta_{ij}$. 
In the literature one can find three types of graph Laplacians:

**unnormalized:**
\[
(\Delta^{(u)}f)(i) = d(i)f(i) - \sum_{j=1}^{n} w_{ij} f(j),
\]
\[
(\Delta^{(u)}f) = (D - W)f,
\]

**normalized:**
\[
(\Delta^{(n)}f)(i) = f(i) - \sum_{j=1}^{n} \frac{w_{ij}}{\sqrt{d_i d_j}} f(j),
\]
\[
(\Delta^{(n)}f) = (\mathbb{1} - D^{-1/2}WD^{-1/2})f.
\]

**Caution:** often no distinction in the literature - each of them is just called graph Laplacian.
Relation to the continuous Laplacian

The **continuous Laplacian** is a second-order differential operator,

\[ \Delta f = \sum_{i=1}^{d} \frac{\partial^2 f}{\partial x_i^2}. \]

It is invariant under rotations and translations (⇒ image processing).

**Correspondence:** For a grid in \( \mathbb{R}^d \) the unnormalized graph Laplacian, \( \Delta^{(u)} = D - W \), corresponds up to the sign to the finite difference approximation of the continuous Laplacian.

For the real line with an equidistant discretization of size size \( h \), we get,

\[ \frac{d^2 f}{dx^2} \approx \frac{1}{h^2} \left( f(i+1) + f(i-1) - 2f(i) \right) = -d(i)f(i) + \sum_{j=1}^{m} w_{ij} f(j) = -(\Delta^{(u)} f)(i), \]

where in the grid each point connects to its nearest neighbors and the weights are \( 1/h^2 \) ⇒ degree of each grid point is \( 2/h^2 \).
Properties of the graph Laplacian

- All graph Laplacians are positive semi-definite and self-adjoint,
  \[ \langle f, \Delta g \rangle_{\mathcal{H}_V} = \langle g, \Delta f \rangle_{\mathcal{H}_V} . \]

- Associated regularization functionals (useful for SSL),
  \[ \langle f, \Delta^{(u)} f \rangle = \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2, \]
  \[ \langle f, \Delta^{(n)} f \rangle = \sum_{i,j=1}^{n} w_{ij} \left( \frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2. \]

- The eigenvectors of \( \Delta^{(u)} \) and \( \Delta^{(n)} \) define an orthonormal basis on \( \mathbb{R}^V \).
Key property for Spectral Clustering

- Algebraic connectivity of the graph:

**Theorem (Fiedler)**

The multiplicity of the first eigenvalue (the first eigenvalue is zero) of the graph Laplacians is equivalent to the number of connected components of the graph.

- Let $A_i, i = 1, \ldots, K$ be the connected components of the graph.
  - $\mathbb{1}_{j \in A_i}$ are eigenvectors of $\Delta^{(u)}$ to the eigenvalue 0.
  - $\sqrt{d_j} \mathbb{1}_{j \in A_i}$ are eigenvectors of $\Delta^{(n)}$ to the eigenvalue 0.

- **Caution:** there is no “first eigenvector” but we have an eigenspace to the eigenvalue zero which has dimension $K$.

A graph which resolves into disconnected components is the ideal clustering (already the graph reveals the cluster structure - no other clustering method necessary).
Spectral Clustering - Variant I

Choose the graph Laplacian: unnormalized or normalized and the number of clusters $k$.

- compute the graph Laplacian,
- compute the first $k$ eigenvectors $\{u_i\}_{i=1}^{k}$ (each eigenvector is normalized, $\|u_i\|_2 = 1$, $i = 1, \ldots, k$),
- Embedding $\phi : V \rightarrow \mathbb{R}^k$, of the $n$ vertices into $\mathbb{R}^k$ by $i \rightarrow z_i = (u_1(i), \ldots, u_k(i))$,
- clustering of the resulting $n$ points $\{z_i\}_{i=1}^{n}$ by $k$-means into clusters $C_1, \ldots, C_k$.

The embedding: $\phi : V \rightarrow \mathbb{R}^k$, $i \rightarrow \phi(i) = (u_1(i), \ldots, u_k(i))$ is basically the Laplacian eigenmap.
Central Questions

- Is the mapped data in the new space suited for k-means?
- Why should this yield a good clustering?

Three different motivations for spectral clustering:

1. Relaxation of graph cuts,
2. Markov random walks,
Motivation I - Graph Cuts

Partitioning of weighted, undirected graphs
Define: $C_i = V \setminus C_i$ and $\text{vol}(C_i) = \sum_{j \in C_i} d_j$ and

$$\text{cut}(C, D) = \sum_{i \in C, j \in D} w_{ij}.$$ 

Let $(C_1, \ldots, C_k)$ be a partition of $V$ ($\bigcup_{i=1}^{k} C_i = V$ and $C_i \cap C_j = \emptyset$, $i \neq j$)

Graph Cut Criteria:
- **MinCut**: $\text{MinCut}(C_1, \ldots, C_k) = \sum_{i=1}^{k} \text{cut}(C_i, \overline{C_i})$.
- **RatioCut**: $\text{RatioCut}(C_1, \ldots, C_k) = \sum_{i=1}^{k} \frac{\text{cut}(C_i, \overline{C_i})}{|C_i|}$.
- **Ncut (normalized Cut)**: $\text{NCut}(C_1, \ldots, C_k) = \sum_{i=1}^{k} \frac{\text{cut}(C_i, \overline{C_i})}{\text{vol}(C_i)}$.

Goal: find optimal (minimal) Min/Ratio/Normalized-cut among all possible partitions.
Partitioning of weighted, undirected graphs

- MinCut: yields often unbalanced partitions in particular single points become clusters.

- Ratio Cut and Normalized Cut are instances of balanced graph cut criteria
  \[ \rightarrow \text{enforces balanced partitions (what does balanced mean ?)} \]
  \[ \rightarrow \text{Ratio Cut prefers clusters of equal size,} \]
  \[ \rightarrow \text{Normalized Cut prefers clusters of equal volume.} \]

- **Problem:** All balanced graph cut criteria are NP-hard.

Spectral clustering is a relaxation of ratio/normalized cut!
Relaxation of Ratio Cut

Given a partition \((C, \bar{C})\) (two clusters, \(k = 2\)) define \(f^C : V \rightarrow \mathbb{R}\),

\[
f_i^C = \begin{cases} 
\sqrt{|\bar{C}|/|C|} & \text{if } i \in C, \\
-\sqrt{|C|/|\bar{C}|} & \text{if } i \in \bar{C}.
\end{cases}
\]

\[
\left< f^C, \Delta^{(u)} f^C \right> = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i^C - f_j^C)^2 = \sum_{i \in C, j \in \bar{C}} w_{ij} \left( \sqrt{\frac{|\bar{C}|}{|C|}} + \sqrt{\frac{|C|}{|\bar{C}|}} \right)^2
\]

\[
= \text{cut}(C, \bar{C}) \left( \frac{|\bar{C}|}{|C|} + \frac{|C|}{|\bar{C}|} + 2 \right) = \text{cut}(C, \bar{C}) \left( \frac{|C| + |\bar{C}|}{|C|} + \frac{|C| + |\bar{C}|}{|\bar{C}|} \right)
\]

\[
= |V| \text{cut}(C, \bar{C}) \left( \frac{1}{|C|} + \frac{1}{|\bar{C}|} \right) = |V| \text{RatioCut}(C, \bar{C})
\]

\[
\sum_{i=1}^{n} f_i^C = \sum_{i \in C} \sqrt{\frac{|\bar{C}|}{|C|}} - \sum_{i \in \bar{C}} \sqrt{\frac{|C|}{|\bar{C}|}} = 0, \quad \|f^C\|_2^2 = \sum_{i=1}^{n} (f_i^C)^2 = |C| \frac{|\bar{C}|}{|C|} + |\bar{C}| \frac{|C|}{|\bar{C}|} = n.
\]
Relaxation of ratio cut II

With the specific form of the function $f^C$ the optimal ratio cut can be written as:

$$\min_{C \subset V} \left\{ \left\langle f^C, \Delta^{(u)} f^C \right\rangle \mid \left\langle f^C, 1 \right\rangle = 0, \| f^C \| = \sqrt{n} \right\}.$$  

This is a discrete combinatorial optimization problem and is $NP$-hard $\Rightarrow$ relax problem by allowing $f$ to take arbitrary real values.

$$\min_{f \in \mathbb{R}^V} \left\{ \left\langle f, \Delta^{(u)} f \right\rangle \mid \left\langle f, 1 \right\rangle = 0, \| f \| = \sqrt{n} \right\}.$$  

- Rayleigh-Ritz principle $\Rightarrow$ If graph is connected, minimum is the second eigenvalue $\lambda_2$ and the minimizer is the second eigenvector $u_2$ of $\Delta^{(u)} = D - W$.
- Partitioning using optimal threshold $t$

$$C_t = \{ j \in V \mid u_2(j) > t \},$$

by optimizing the Ratio-Cut or alternatively k-means in the embedding.
Relaxation of normalized cut

Given a partition \((C, \overline{C})\) define the function,

\[
f^C_i = \begin{cases} \sqrt{\text{vol}(\overline{C})/\text{vol}(C)}, & i \in C, \\ -\sqrt{\text{vol}(C)/\text{vol}(\overline{C})}, & i \in \overline{C} \end{cases}
\]

\[
\langle f^C, \Delta^{(u)} f^C \rangle = \text{vol}(V) \text{NCut}(C, \overline{C}), \quad \langle f^C, Df^C \rangle = \text{vol}(V) = n, \quad \langle 1, Df^C \rangle = 0.
\]

The optimal normalized cut:

\[
\min_{C \subset V} \left\{ \langle f^C, \Delta^{(u)} f^C \rangle \mid \langle Df^C, 1 \rangle = 0, \langle f^C, Df^C \rangle = n \right\}.
\]

Relaxation of the normalized cut:

\[
\min_{f \in \mathbb{R}^V} \left\{ \langle f, \Delta^{(u)} f \rangle \mid \langle Df, 1 \rangle = 0, \langle f, Df \rangle = n \right\}.
\]

\[\Rightarrow\] generalized eigenproblem \(\Delta^{(u)} f = \lambda Df\).
The general case for the ratio cut

Given a partition \((C_1, \ldots, C_k)\) define the functions \(h_i,\)

\[
h_i(j) = \begin{cases} 
\frac{1}{\sqrt{|C_i|}} & j \in C_i, \\
0 & j \in \overline{C_i}.
\end{cases}
\]

General ratio cut:

\[
\min_{C_1, \ldots, C_k} \left\{ \text{Tr}(H \Delta^{(u)} H^T) \mid HH^T = \mathbb{1}_k \right\}
\]

- The minimizer of the relaxation to arbitrary \(H = \{h_1, \ldots, h_k\}\), that is \(H \in \mathbb{R}^{k \times n}\), yields the smallest \(k\) eigenvectors \(\{u_i\}_{i=1}^k\) of the unnormalized graph Laplacian \(\Delta^{(u)}\). The minimum is the sum of the \(k\)-smallest eigenvalues of \(\Delta^{(u)}\).
- The conversion of \(H = \{u_1, \ldots, u_k\}\) into a partition \((C_1, \ldots, C_k)\) can be done by \(k\)-means clustering of the rows of \(H \Rightarrow no approximation guarantees
Theoretical results for $k = 2$

- Let $\phi^* = \min_{C} \text{RCut}(C, \overline{C})$ and denote by $\phi_{SPECTRAL}$ the cut obtained by optimal thresholding of the second eigenvector. It holds

$$\phi^* \leq \phi_{SPECTRAL} \leq 2\sqrt{\max_i d_i} \sqrt{\phi^*}$$

There exist graphs which get close to upper bound.

- Better worst case guarantees for normalized/ratio cut for relaxation into a semi-definite program (Arora et al (2004)).


**Conclusion:** The graph cuts picture is only a part of the story of spectral clustering.
Spectral Clustering - Variant II (recursive bipartitioning)

Choose graph Laplacian and the number of clusters $k$.

- initialize: current partition $V$.
- do build on each element of the current partition the graph Laplacian,
  - compute the second eigenvector on each partition,
  - compute the optimal threshold for dividing each partition,
  - choose the cut which minimizes the total balanced cut criterion.
- while number of elements in the partition is less than $k$

Discussion:

- **Advantage:** uses original criterion to split - no k-means,
- **Disadvantage:** the embedding integrates global information about the data $\Rightarrow$ problem if first split is not optimal.
Markov random walk for an undirected, weighted graph:

- Stochastic matrix: \( P = D^{-1}W \).
- Stationary distribution: \( \pi_i = \frac{d_i}{\text{vol}(V)} \).

Proposition (Meila, Shi)

Let \( G \) be connected. Let \( X_0 \sim \pi \) be the random walk started in the stationary distribution and \( C \) be a subset of \( V \). Then the normalized cut can be written as

\[
\text{NCut}(C, \overline{C}) = \left[ P(X_1 \in \overline{C} \mid X_0 \in C) + P(X_1 \in C \mid X_0 \in \overline{C}) \right].
\]

Interpretation:

\( \rightarrow \) find a partitioning such that the random walk stays as long as possible in each cluster.
Motivation III - Perturbation theory

**Perfect clusters = disconnected graph**

- multiplicity of the eigenvalue, $\lambda = 0$, of the graph Laplacians is equal to the number $K$ of connected components of the graph.
- the $K$ eigenvectors for $\lambda = 0$ are constant on the connected component and zero elsewhere.

**Perturbation of the weight matrix - make the graph connected**

\[ \tilde{W} = W + \text{edges such that graph is connected}. \]

- only small change for the weight matrix, \( \implies \) first $K$ eigenvalues should still be very small, \( \implies \) first $K$ eigenvectors should be only very little perturbed
- each cluster is mapped to a single point (in the embedding).
- \( \implies \) rigorous statements using perturbation theory of symmetric matrices.
Practical issues

DemoSpectralClustering:

![Demo Spectral Clustering](image)

- **Data**
- **Symmetric KNN**
- **Current Clustering**

- **Eigenvalues**
- **Embedding in \( \mathbb{R}^2 \) - EV 2-4**

- **Number of connected components**: 1

- **Data Set**
  - Two Moons (balanced)

- **Dimension**
  - 3

- **Graph Type**
  - Symmetric KNN

- **Number of Neighbors**
  - \( K=15 \)

- **Kernel width**
  - \( \sigma=0.1 \)

- **Number of Clusters**
  - \( N=2 \)

---

Paul Swoboda (Lecture 18, 18.12.2018)  
Machine Learning  
26 / 27
For sparse graphs (\(k\)-NN graphs) the first few eigenvectors can be efficiently computed using the **power or Lanczos method** \(\Rightarrow\) spectral clustering can be done for **millions of points**.

- Spectral Clustering used for image segmentation (Shi and Malik),
- Check the spectrum of the graph Laplacian. Never cut the spectrum where two eigenvalues are close. Always cut at a gap. This can also be formally justified by the stability of eigenvectors and eigenvalues under perturbations.
- Spectral clustering is quite stable against high-dimensional noise.
- Use the normalized graph Laplacian.