Mean-shift Algorithms for Manifold Denoising, Matrix Completion and Clustering

Weiran Wang
wwang5@ucmerced.edu
EECS Department, UC Merced
High dimensional dataset with **manifold structure**.

- Variations within the dataset can be modeled by a few latent variables.
- Small variation in latent space leads to small variation in data space.
- Local neighborhood of each data point can be approximated by a tangent space.
Small variations in translation, rotation, scaling and different writing styles change the image appearance slightly, and do not change the identity.
Mean-shift update

Given a set of data points $X = [x_1, \ldots, x_N] \subset \mathbb{R}^D$.

- Maximizes kernel density estimate (mode finding)

$$p(x) = \frac{1}{N} \sum_{n=1}^{N} G\left(\left\| \frac{x - x_n}{\sigma} \right\|^2\right), \quad G(t) = e^{-t/2}.$$

- Applies the mean-shift update (fixed point iteration) iteratively

$$p(n|x) = \frac{G\left(\left\| \frac{x - x_n}{\sigma} \right\|^2\right)}{\sum_{n'=1}^{N} G\left(\left\| \frac{x - x_{n'}}{\sigma} \right\|^2\right)}, \quad x \leftarrow f(x) = \sum_{n=1}^{N} p(n|x)x_n$$

- Gradient ascent. Linear convergence rate.
Paths followed by GMS for various starting points.
Mean-shift clustering

- **Gaussian Mean-shift (GMS):** points that converge to the same mode/centroid define a cluster. Number of clusters depends on $\sigma$.

- **Gaussian Burring Mean-shift (GBMS):** update dataset after each mean-shift step, has much faster (cubic) convergence rate and strong (isotropic) denoising effect.
Outline

– Manifold Blurring Mean-shift (MBMS) algorithm for manifold denoising

– MBMS for matrix completion

– $K$-modes algorithm for clustering

– Laplacian $K$-modes algorithm for clustering
We develop an algorithm that denoises the dataset, and acts as a preprocessing step for unsupervised/supervised learning.
Manifold Blurring Mean-Shift

- **Predictor averaging step:** local clustering with GBMS, moves data point to the kernel average of its neighbors

\[
x_n \leftarrow \sum_{m \in \mathcal{N}_n} \frac{G\left(\|x_n - x_m\|_2^2 / \sigma^2\right)}{\sum_{m' \in \mathcal{N}_n} G\left(\|x_n - x_{m'}\|_2^2 / \sigma^2\right)} x_m
\]

- **Corrector projective step:** estimate local tangent space with PCA, gives the best linear \( L \)-dimensional manifold in terms of reconstruction error (orthogonal projection on the manifold)

\[
\min_{\mu, U} \sum_{m \in \mathcal{N}_n} \left\| x_m - (UU^T(x_m - \mu) + \mu) \right\|^2
\]

- **User parameters:** \( \sigma, K, L \).
Variations of MBMS:
- **MBMSf/MBMSk**: use full/knn graph in predictor step.
- **Local Tangent Projection (LTP)**: MBMSk with $\sigma = \infty$.
- **GBMS**: $L = 0$, no corrector step.

User parameters can be determined by cross-validation for supervised problem.

Stopping criteria: orthogonal variance $\lambda_\perp$ (sum of the trailing $D - L$ eigenvalues of $x_n$’s local covariance) is small.
Denoising a noisy spiral with outliers over iterations.
Experiment: preprocessing for spectral methods

\[ \tau = 0 \quad \tau = 1 \quad \tau = 2 \quad \tau = 3 \quad \tau = 5 \]

Dimensionality reduction with Isomap and LTSA for iterations of MBMSk.
Experiment: preprocessing for classifying MNIST

We denoise images of each digit separately using MBMSk.

Sample pairs of (original, denoised) images from the training set.
Experiment: preprocessing for classifying MNIST

Classify test set using denoised training set and Nearest Neighbor.

Some misclassified images. Each triplet is (test, original-nearest-neighbor, denoised-nearest-neighbor) and the corresponding label is above each image, with errors highlighted.
Experiment: preprocessing for classifying MNIST

Top 3 plots: 5–fold cross-validation error (%) curves with a nearest-neighbor classifier on training set using MBMSk.

Bottom left plot: denoising and classification of the MNIST test set, by training on the entire training set and smaller subsets.
Conclusion

- Very effective at denoising in a handful of iterations.
- Nonparametric and deterministic.
- Causing very small shrinkage or distortion.
- Able to handle large noise and extreme outliers.
Outline

- Manifold Blurring Mean-shift (MBMS) algorithm for manifold denoising
- MBMS for matrix completion
- $K$-modes algorithm for clustering
- Laplacian $K$-modes algorithm for clustering
Problem Setting

❖ Given a set of data points $\mathbf{X} = [x_1, \ldots, x_N] \subset \mathbb{R}^D$, where each point may contain missing entries.

✦ $\mathbf{X}^M$ and $\mathbf{X}^P$ indicate the selection of missing or present entries $\mathbf{X}$, where $\mathcal{P} \subset \mathcal{U}$, $\mathcal{M} = \mathcal{U} - \mathcal{P}$ and $\mathcal{U} = \{(d, n) : d = 1, \ldots, D, n = 1, \ldots, N\}$.

✦ Indices $\mathcal{P}$ and values $\mathbf{X}^P$ of the present entries are the data of the problem.

❖ An ill-posed problem. Very important in industrial applications.

<table>
<thead>
<tr>
<th>5</th>
<th>1</th>
<th>?</th>
<th>2</th>
<th>3</th>
<th>?</th>
<th>?</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>2</td>
<td>?</td>
<td>4</td>
<td>1</td>
<td>?</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>?</td>
<td>?</td>
<td>1</td>
<td>2</td>
<td>?</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>5</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>?</td>
<td>4</td>
<td>2</td>
<td>?</td>
<td>5</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>3</td>
<td>?</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
Motivation

❖ Popular approaches for matrix completion
  ✦ Low-rank: \( \min_X \|X\|_* \quad \text{s.t.} \quad X_{\mathcal{P}} = \overline{X}_{\mathcal{P}} \)
  ✦ Matrix factorization (probabilistic and nonlinear extensions):
    \[
    \min_{L,R} \sum_{(i,j) \in \mathcal{P}} (X_{ij} - L_i R_j^T)^2 + \lambda(\|L\|_{\text{Fro}}^2 + \|R\|_{\text{Fro}}^2).
    \]

❖ Globally low-rank assumption is too restrictive for nonlinear manifold. We use \textbf{locally low-rank} assumption instead.

![Data](image1.png)  ![SVP](image2.png)

- p. 18
MBMS for matrix completion

- GBMS maximizes the following objective function by taking parallel steps of the mean-shift form for each point:

\[ E(X) = \frac{1}{N} \sum_{n,m=1}^{N} G \left( \frac{\| x_n - x_m \|}{\sigma} \right)^2 \]

- Apply GBMS to matrix completion by adding the constraints given by the present values \( X_P = \overline{X}_P \).

- We iteratively carry out a GBMS denoising step on \( X \) and refill \( X_P \) to the present values; equivalent to a gradient projection algorithm.

- MBMS can be applied instead to prevent shrinkage.

- Hyperparameters and number of iterations can be cross-validated on held out present entries.
Synthetic example

Denoising effect of different algorithms on 100D swissroll.
Experiment: Mocap

Running sequence with 148 samples of 150D sensor readings.

- frame 2 (leg distance)
- frame 10 (foot pose)
- frame 147 (leg pose)

Sample reconstructions when 85% percent data is missing. Row 1: initialization. Row 2: init+GBMS. Row 3: init+MBMS. Color indicates different initialization: original data, nlPCA, SVP, Gaussian.
Results on Mocap dataset. Mean of errors (RSSE) of 5 runs obtained by different algorithms for varying percentage of missing values.
Experiment: MNIST digit 7

6,265 greyscale images of size $28 \times 28$, 50% entries missing.

<table>
<thead>
<tr>
<th>Methods</th>
<th>RSSE</th>
<th>mean</th>
<th>stdev</th>
</tr>
</thead>
<tbody>
<tr>
<td>nIPCA</td>
<td>7.77</td>
<td>26.1</td>
<td>42.6</td>
</tr>
<tr>
<td>SVP</td>
<td>6.99</td>
<td>21.8</td>
<td>39.3</td>
</tr>
<tr>
<td>+ GBMS (400,140,0,1)</td>
<td>6.54</td>
<td>18.8</td>
<td>37.7</td>
</tr>
<tr>
<td>+ MBMS (500,140,9,5)</td>
<td>6.03</td>
<td>17.0</td>
<td>34.9</td>
</tr>
</tbody>
</table>

Reconstruction errors of different algorithms at their optimal parameters.
### Experiment: MNIST digit 7

<table>
<thead>
<tr>
<th>Orig</th>
<th>Missing</th>
<th>nlPCA</th>
<th>SVP</th>
<th>GBMS</th>
<th>MBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Digit 7" /></td>
<td><img src="image2" alt="Digit 7" /></td>
<td><img src="image3" alt="Digit 7" /></td>
<td><img src="image4" alt="Digit 7" /></td>
<td><img src="image5" alt="Digit 7" /></td>
<td><img src="image6" alt="Digit 7" /></td>
</tr>
<tr>
<td><img src="image7" alt="Digit 7" /></td>
<td><img src="image8" alt="Digit 7" /></td>
<td><img src="image9" alt="Digit 7" /></td>
<td><img src="image10" alt="Digit 7" /></td>
<td><img src="image11" alt="Digit 7" /></td>
<td><img src="image12" alt="Digit 7" /></td>
</tr>
<tr>
<td><img src="image13" alt="Digit 7" /></td>
<td><img src="image14" alt="Digit 7" /></td>
<td><img src="image15" alt="Digit 7" /></td>
<td><img src="image16" alt="Digit 7" /></td>
<td><img src="image17" alt="Digit 7" /></td>
<td><img src="image18" alt="Digit 7" /></td>
</tr>
</tbody>
</table>

Selected reconstructions of MNIST block-occluded digits ‘7’.
Conclusion

❖ We propose new denoising paradigm for matrix completion, which generalizes the commonly used assumption of low rank.

❖ MBMS-based algorithm bridges the gap between pure denoising (GBMS) and local low rank.

❖ Denoising works due to the fundamental fact that a missing value can be predicted by averaging nearby present values, a common approach in recommender systems.
Outline

– Manifold Blurring Mean-shift (MBMS) algorithm for manifold denoising
– MBMS for matrix completion
– $K$-modes algorithm for clustering
– Laplacian $K$-modes algorithm for clustering
Motivation

- Given a dataset $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^D$, centroids-based clustering
  - partition data points into groups,
  - estimate a representative $\mathbf{c}_k \in \mathbb{R}^D$ of each cluster $k$.

- Popular algorithms of this type: $K$-means, mean-shift, $K$-medoids.

- No $K$-modes algorithm exists. Mode $\Rightarrow$ high density $\Rightarrow$ representativeness.
K-means algorithm

Optimizes over assignment \( Z \) and centroids \( C \)

\[
\min_{Z,C} \sum_{k=1}^{K} \sum_{n=1}^{N} z_{nk} \| x_n - c_k \|^2
\]

s.t. \( z_{nk} \in \{0, 1\}, \sum_{k=1}^{K} z_{nk} = 1, \text{ for } n = 1, \ldots, N. \)

- Efficient algorithm alternates \( Z \)-step (computes assignment) and \( C \)-step (computes mean).
- Can only produce convex clusters (Voronoi tessellation).
- Cluster mean may not be valid pattern.
- Sensitive to noise and outliers.
\( K \)-modes: objective function

\[
\max_{Z, C} \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} G \left( \frac{\left\| x_n - c_k \right\|^2}{\sigma} \right)
\]

\[\text{s.t.} \quad z_{nk} \in \{0, 1\}, \quad \sum_{k=1}^{K} z_{nk} = 1, \text{ for } n = 1, \ldots, N,\]

- Sum of KDE but separately for each cluster.
- Combines the notions of assignment and density estimation.
- Two limit cases: “\( K \)-medoids” when \( \sigma \to 0 \), \( K \)-means when \( \sigma \to \infty \).
- Alternating optimization with guaranteed convergence
  - \( Z \)-step: decouples over points, same assignment rule as \( K \)-means.
  - \( C \)-step: decouples over clusters, mode-finding within each cluster.
$K$-modes: homotopy algorithm

Start with $\sigma = \infty$ ($K$-means), gradually decrease $\sigma$ while running $J$ iterations of the fixed-$\sigma$ $K$-modes algorithm for each value of $\sigma$, until reach a target value $\sigma^*$.  

- A deterministic algorithm given local optimum found by $K$-means.  
- Follows an optimum path $(Z(\sigma), C(\sigma))$ for $\sigma \in [\sigma^*, \infty)$.  
- Homotopy techniques tends to find better optima than starting directly at the target value $\sigma^*$.  
- Representative, valid centroids are obtained for a wide range of intermediate $\sigma$ values.
\(K\)-modes: homotopy algorithm

\[ \sigma = \infty \]

\[ \sigma = 0.1 \]

\[ K = 2. \text{ No value of } \sigma \text{ results in two modes that separate the (nonconvex) moons.} \]
Experiment: misspecification of $K$

3 natural clusters, but use $K = 2$. ☞

\[ \sigma = \infty \quad \text{and} \quad \sigma = 1 \]
Experiment: handwritten digit images

\( K \)-means result \((K = 10, \sigma = \infty)\)

- Centroids are average of different classes.
- Neighborhoods are not homogeneous/pure.
Experiment: handwritten digit images

$K$-modes result ($K = 10$, $\sigma = 1$)

- Centroids are very representative.
- Neighborhoods are homogeneous/pure.
Experiment: handwritten digit images

Mean-shift result ($\sigma = 1.8369$)

- In high dimensions, many modes have very few associated points.
**Summary**

- $K$-modes is more robust than $K$-means and GMS to outliers and parameter misspecification.

- $K$-modes will return exactly $K$ modes (one per cluster) no matter the value of $\sigma$, and whether the dataset KDE has more or fewer than $K$ modes.

- Centroids are representative, valid patterns.
Outline

– Manifold Blurring Mean-shift (MBMS) algorithm for manifold denoising

– MBMS for matrix completion

– $K$-modes algorithm for clustering

– Laplacian $K$-modes algorithm for clustering
Motivation

- Limitation of $K$-modes assignment: can only find convex clusters.
- In addition to representative centroids and density estimate, we want more flexible assignment.

$K$-modes ($\sigma = 0.1$)

Laplacian $K$-modes ($\sigma = 0.1$)
Laplacian smoothing

Key to separate clusters with manifold structure: nearby data points should have similar assignment.

1. Relax the assignment to be continuous, but constrain them to probabilities.
2. Build a graph on the dataset, let $w_{mn}$ be the weight between $x_m$ and $x_n$.
3. Add Laplacian smoothing term $\frac{\lambda}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} w_{mn} \|z_m - z_n\|^2$. 
Laplacian $K$-modes: objective function

\[
\min_{Z, C} \frac{\lambda}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} w_{mn} \|z_m - z_n\|^2 - \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} G\left(\left\|\frac{x_n - c_k}{\sigma}\right\|^2\right)
\]

s.t. \[\sum_k z_{nk} = 1, \text{ for } n = 1, \ldots, N,\]

\[z_{nk} \geq 0, \text{ for } n = 1, \ldots, N, \ k = 1, \ldots, K.\]

❖ Obtain hard assignment by choosing largest assignment probability.
❖ Alternating optimization
  ✦ $C$-step: decouples over clusters, mode-finding within each cluster.
  ✦ $Z$-step: convex quadratic program, solved with gradient projection.
❖ Homotopy in $(\sigma, \lambda)$ can be done similarly as in $K$-modes.

- p. 40
Effect of Laplacian smoothing

- $K$-modes
- Laplacian $K$-modes
- KDE

$K=5$. $K$-modes assignment rule can never separate the spirals.
Out-of-sample problem

- Optimize assignment \( z \) of new point \( x \) given \( Z \) and \( C \) from training.
- The out-of-sample problem is equivalently

\[
\min_z \quad \frac{1}{2} \| z - \bar{z} - \gamma q \|^2,
\]

s.t. \( z^\top 1_K = 1 \), \( z \geq 0 \),

where \( \bar{z} \) is the weighted mean of training assignments, \( q \) is soft distance to centroids.

- Projection of \( \bar{z} + \gamma q \) onto the probability simplex.
- It is a mixture of two assignment rules and a nonlinear mapping.
Out-of-sample problem

\[ K = 2. \text{ Homotopy in } \sigma \text{ for Laplacian } K\text{-modes.} \]
## Clustering analysis

### Statistics of datasets.

<table>
<thead>
<tr>
<th>dataset</th>
<th>size ($N$)</th>
<th>dimensionality ($D$)</th>
<th># of classes ($K$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST (digit image)</td>
<td>2000</td>
<td>768</td>
<td>10</td>
</tr>
<tr>
<td>COIL20 (object image)</td>
<td>1440</td>
<td>1024</td>
<td>20</td>
</tr>
<tr>
<td>TDT2 (document)</td>
<td>9394</td>
<td>36771</td>
<td>30</td>
</tr>
</tbody>
</table>

### Clustering accuracy (%).

<table>
<thead>
<tr>
<th>dataset</th>
<th>$K$-means</th>
<th>$K$-modes</th>
<th>GMS</th>
<th>NCut</th>
<th>GNMF</th>
<th>DCD</th>
<th>Lap. $K$-modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>58.2</td>
<td>59.2</td>
<td>15.9</td>
<td>65.5</td>
<td>66.2</td>
<td>69.4</td>
<td>70.5</td>
</tr>
<tr>
<td>COIL–20</td>
<td>66.5</td>
<td>67.2</td>
<td>27.2</td>
<td>79.0</td>
<td>75.3</td>
<td>71.5</td>
<td>81.0 (81.5)</td>
</tr>
<tr>
<td>TDT2</td>
<td>68.9</td>
<td>70.0</td>
<td>N/A</td>
<td>88.4</td>
<td>88.6</td>
<td>55.1</td>
<td>91.4</td>
</tr>
</tbody>
</table>

### Normalized Mutual Information (%).

<table>
<thead>
<tr>
<th>dataset</th>
<th>$K$-means</th>
<th>$K$-modes</th>
<th>GMS</th>
<th>NCut</th>
<th>GNMF</th>
<th>DCD</th>
<th>Lap. $K$-modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>53.3</td>
<td>53.6</td>
<td>6.51</td>
<td>66.9</td>
<td>64.9</td>
<td>65.6</td>
<td>68.8</td>
</tr>
<tr>
<td>COIL–20</td>
<td>75.3</td>
<td>75.9</td>
<td>38.9</td>
<td>88.0</td>
<td>87.5</td>
<td>77.6</td>
<td>87.3 (88.0)</td>
</tr>
<tr>
<td>TDT2</td>
<td>75.3</td>
<td>75.8</td>
<td>N/A</td>
<td>83.7</td>
<td>83.7</td>
<td>68.6</td>
<td>88.8</td>
</tr>
</tbody>
</table>
Centroids found by different algorithms on MNIST.
Centroids found by different algorithms on COIL–20.
Comparison of different clustering algorithms.

<table>
<thead>
<tr>
<th></th>
<th>$K$-means</th>
<th>$K$-medoids</th>
<th>Mean-shift</th>
<th>Spectral clustering</th>
<th>$K$-modes</th>
<th>Laplacian $K$-modes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Centroids</strong></td>
<td>likely invalid</td>
<td>“valid”</td>
<td>“valid”</td>
<td>N/A</td>
<td>valid</td>
<td>valid</td>
</tr>
<tr>
<td><strong>Nonconvex clusters</strong></td>
<td>no</td>
<td>depends</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Density</strong></td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Assignment</strong></td>
<td>hard</td>
<td>hard</td>
<td>hard</td>
<td>hard</td>
<td>hard</td>
<td>soft</td>
</tr>
<tr>
<td><strong>Cost/iteration</strong></td>
<td>$KnD$</td>
<td>$Kn^2D$</td>
<td>$N^2D$</td>
<td>$N^2 \sim N^3$</td>
<td>$KnD$</td>
<td>$KnD$</td>
</tr>
</tbody>
</table>
Conclusion

❖ We develop mean-shift algorithms to analyze dataset with low degrees of freedom.

❖ Future directions:
  ✦ Theoretical analysis
  ✦ Speedup training and testing
  ✦ Incorporating more domain knowledge


Denoising a complex shape with nonuniform density and noise with MBMSf using different affinity (left: normal, middle: diffusion maps, right: entropic affinity).
MBMS Experiment: Robustness to parameters choice

For swissroll dataset, there is a wide range for each parameter in which MBMS works well.

\[ \frac{1}{ND} \sum_{i=1}^{N} \| X_i - true X_i \| \]

\[ \sigma = \infty, L = 2 \]

\[ \sigma = \infty, K = 30 \]

Behavior of LTP for different parameters $K$ and $L$. Error decreases for all parameter choices.
$K$-modes Experiment: heavy tailed distribution

$\sigma = 200$  $\sigma = 29.86$  $\sigma = 15.14$  $\sigma = 3.89$  $\sigma = 1$

$K = 2$. Separating mixture of a Gaussian component and a power-law component.
Laplacian $K$-modes: alternating optimization

- **C-step**: decouples over different cluster. For cluster $k$, solve
  \[
  \max_{c_k} \sum_{n: z_{nk} > 0} z_{nk} G \left( \frac{\|x_n - c_k\|}{\sigma} \right)
  \]
  with mean-shift updates.

- **Z-step**: no longer decouples over different points.
  \[
  \min_{Z} \quad \lambda \text{tr} (Z^\top LZ) - \text{tr} (B^\top Z)
  \]
  \[
  \text{s.t.} \quad Z1_K = 1_N, \quad Z \geq 0,
  \]

  where $B_{nk} = G \left( \frac{\|x_n - c_k\|}{\sigma} \right)$, $L$ is the graph Laplacian.

- Quadratic program of $NK$ variables.
- Interior point method is too slow for large dataset.
- We use first order method instead.
The ISTA/FISTA framework (gradient proximal method):

- Solves \( \min_x f(x) = g(x) + h(x) \). \( g \) is convex and has Lipschitz continuous gradient (with constant \( L \)). \( h \) is convex and not necessarily differentiable.

\[ x_{n+1} = \arg \min_y \frac{L}{2} \left\| y - (x_n - \frac{1}{L} \nabla g(x_n)) \right\|^2 + h(y). \]

- Convergence: \( f(x_T) - f(x^*) \approx O\left(\frac{1}{T}\right) \) for constant stepsize \( \frac{1}{L} \).
- Nesterov’s acceleration scheme improves the rate to \( O\left(\frac{1}{T^2}\right) \).

Apply to our \( Z \)-step:

- \( g \) is the quadratic objective function, with \( L = 2\lambda \sigma_1(L) \).
- \( h \) is the indicator function of probability simplex, therefore the proximal step is computing Euclidean projection.
Accelerated gradient projection for $Z$-step

**Input:** Initial $Z_0 \in \mathbb{R}^{N \times K}$, $s = \frac{1}{2\lambda \sigma_1(L)}$.

1. Set $Y_1 = Z_0$, $t_1 = 1$, $k = 1$.
2. repeat
3. Compute gradient at $Y_k$: $G_k = 2\lambda LY_k - B$,
4. $Z_k = \text{SimplexProj}(Y_k - sG_k)$ where SimplexProj() projects each row of the argument onto the probability simplex,
5. $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$,
6. $Y_{k+1} = Z_k + \left(\frac{t_k-1}{t_{k+1}}\right)(Z_k - Z_{k-1})$,
7. $k = k + 1$,
8. until convergence.

**Output:** $Z_k$ is the solution of $Z$ given $C$. 

- p. 55
Projection onto the probability simplex

**Input:** A vector \( \mathbf{v} \in \mathbb{R}^K \)

1: Sort \( \mathbf{v} \) into \( \mathbf{u} : u_1 \geq u_2 \geq \cdots \geq u_K \)

2: Find \( \rho = \max \{1 \leq j \leq K : u_j - \frac{1}{j} (\sum_{r=1}^{j} u_r - 1) > 0\} \)

3: Define \( \theta = \frac{1}{\rho} (\sum_{r=1}^{\rho} u_r - 1) \)

**Output:** \( \mathbf{w} \) s.t. \( w_i = \max \{v_i - \theta, 0\} \)

**Computational complexity:** \( \mathcal{O}(K \log K) \).
Laplacian $K$-modes: occluder segmentation

$K = 5$. Each pixel is connected with nearby eight pixels with edge weighted using heat kernel.