Wasserstein Barycenter Applied to K-Means Clustering

Carol Long
Email Address: carol.long@nyu.edu

Advisor: Yunan Yang
Agenda

1. K-Means Clustering
2. Optimal Transport and Wasserstein Distance/Barycenter
3. Modified K-Means Algorithm with Wasserstein Distance
4. Implementation - Shape Experiment
5. Conclusion and Future Work
K-Means Clustering - Applications

**Business**
Customer segmentation

**Computer Vision**
Image Classification

**Medical Research**
Cancer Signature

![Business and Computer Vision Diagrams](image)

K-Means Clustering - Algorithm

Algorithm

1. Initialization: randomly pick k centroids from the samples as initial cluster centers;
2. Expectation Step: Assign each sample to its nearest centroid \( \mu_j, j \in \{1, ..., k\} \);
3. Maximiaization Step: Move the centroid to the center of samples that were assigned to it;
4. Repeat steps 2 and 3 until the cluster assignments do not change/ convergence/ max itr reached.

K-Means Clustering - Algorithm

K-Means Algorithm

1. Initialization: randomly pick k centroids from the samples as initial cluster centers;
2. Expectation Step: Assign each sample to its nearest centroid $\mu_j, j \in \{1, ..., k\}$;
3. Maximization Step: Move the centroid to the center of samples that were assigned to it;
4. Repeat steps 2 and 3 until the cluster assignments do not change/ convergence/ max itr reached.

Motivating Questions/Observation

- Samples - In the form of distributions
- Does Euclidean norm / Frobenius norm capture distance well?
- Does centroids produce good prototypes?
Optimal Transport and Wasserstein Distance

Kantorovich Formulation of OT:

\[ OT(a, b) = \min_{\gamma} \sum_{i,j} \gamma_{i,j} M_{i,j} \]

s.t. \( \gamma 1 = a; \gamma^T 1 = b; \gamma \geq 0 \)

Wasserstein Distance

\[ W_p(a, b) = (\min_{\gamma} \sum_{i,j} \gamma_{i,j} \|x_i - y_j\|_p)^{\frac{1}{p}} \]

s.t. \( \gamma 1 = a; \gamma^T 1 = b; \gamma \geq 0 \)

Ref: Python Optimal Transport Documentation
Barycenter vs Centroid

Wasserstein Barycenter

$$\min_{\mu} \sum_k w_k \mathcal{W}(\mu, \mu_k)$$

Centroid

$$\sum_k w_k \mu_k$$

Captures the geometric shape of distributions

Img source: Fast Computation of Wasserstein Barycenter, Marco Cuturi
Barycenter Example - Covid Testing Sites

**Input Measure**
- From JHU Covid-19 Repository, April 1st, 2020 cumulative data, before the shelter-at-home order

**Original Centers**
- Considered as a mean of the distribution with n supports
- Only include temporary testing sites, only use the resources that the state can deploy

**Wasserstein Barycenter**
- Note: more support than original centers, counter the effect of disproportionately large density around NYC area
- Still, resemblance between the two means

Data source: JHU Covid-19 Repository
K-Means with Wasserstein Distance/Barycenter

K-Means Algorithm

1. **Initialization**: randomly pick k centroids from the samples as initial cluster centers;
2. **Expectation Step**: Assign each sample to its nearest centroid;
3. **Maximization Step**: Move the centroid to the center of samples that were assigned to it;
4. **Repeat** steps 2 and 3 until the cluster assignments do not change/convergence/max itr reached.

### Traditional K-Means

- **Sets**: \( S_1 \ldots S_k \)
- **Mean distributions**: \( m_1 \ldots m_k \)
- **Samples**: \( x_1 \ldots x_n \)

**Expectation Step**

for \( p \) in 1\ldots n:

\[
\argmin_j ||x_p - m_j||^2
\]

**Maximization Step**

for \( j \) in 1\ldots k:

\[
m_j = \frac{1}{|S_j|} \sum_{x_p \in S_j} x_p
\]

### K-Means with Wasserstein

- **Sets**: \( S_1 \ldots S_k \)
- **Mean distributions**: \( m_1 \ldots m_k \)
- **Samples**: \( x_1 \ldots x_n \)

**Expectation Step**

for \( p \) in 1\ldots n:

\[
\argmin_j W(x_p, m_j)^2
\]

**Maximization Step**

for \( j \) in 1\ldots k:

\[
m_j = \min_{m_j} \frac{1}{|S_j|} \sum_{x_p \in S_j} W(x_p, m_j)^2
\]
Implementation - Shape Experiment

Problem Setup

99 Shape Dataset: 9 classes x 11 images

Turned into probability distributions and shuffled

Objective:

Classify into 9 sets and find their means

K-means: Good for clustering

Barycenter: No way to find multiple at once but good for capturing geometric shape

Data source: Shape Indexing of Image Dataset (SIID), Brown University
Computing Wasserstein Distance

**Kantorovich Formulation**

\[ OT(a, b) = \min_{\gamma} \sum_{i,j} \gamma_{i,j} M_{i,j} \]
\[ s.t. \gamma 1 = a; \gamma^T 1 = b; \gamma \geq 0 \]

**Regularized Wasserstein Distance**

\[ \gamma^* = \arg\min_{\gamma} \sum_{i,j} \gamma_{i,j} M_{i,j} + \lambda \Omega(\gamma) \]
\[ s.t. \gamma 1 = a; \gamma^T 1 = b; \gamma \geq 0 \]
\[ \Omega(\gamma) = \sum_{i,j} \gamma_{i,j} \log(\gamma_{i,j}) \]

**Challenges**

- Need to find transport map, matrix of real numbers, very expensive (NP-hard!)
- Prev experiment takes half a day to run

**Sliced Wasserstein Distance**

\[ \widehat{SW}_p^p(\eta, \mu) = \frac{1}{N} \sum_{n=1}^{N} W_p((\Pi_{v_n}) \# \eta, (\Pi_{v_n}) \# \mu) \]

where \( v_1, \ldots, v_N \overset{\text{i.i.d.}}{\sim} \text{Unif}(S^{d-1}) \).

Ref: Python Optimal Transport Documentation, “Orthogonal Estimation of Wasserstein Distance”, M. Roland
Results - Shape Experiment

Traditional K-Means

K-Means w/ Regularized Wasserstein Dist

K-Means w/ Sliced Wasserstein Dist
Results - Shape Experiment

K-Means w/ Sliced W-2
Results - Shape Experiment

Traditional K-Means
Conclusion and Future Work

Benefits of Wasserstein Barycenter

- Captures some geometric features in distributions rather than giving a naive mean

Limitations

- Sliced Wasserstein Distance still slightly more costly than K-means
- Only captures characteristic shape, neglect details, selective on problems that fits

Future Work

- Further optimize K-means with sliced W-2 distance algorithm
- Find other suitable problems for this method

Barycenters of Lung CT Images with and without Pneumonia
Thank you!

Special thanks to:
- Dr. Yunan Yang
- Prof. Robert Kohn
- Prof. Antoine Cerfon & SURE Program
- Prof. Matthew Leingang