Clustering with General Costs

A Digression from “Factor Discovery Through Optimal Transport”

Given by Daniel Wang on 07/27/2023
Mentors: Esteban G. Tabak, Andrew Lipnick, Nina Mortensen, Ryan Shìjié Dù

The Roadmap

- Clusters as Discrete Factors
  - Relaxing The Problem to k-Means
- k-Means: The Standard Algorithm
- k-GenCenters: An Extension of k-Means to General Costs
  - Introduce k-Medians
  - k-Means ***vs*** k-Medians
  - Improving initialization
  - Must-link constraints

Clusters as Discrete Factors

A relaxation of the factor discovery problem

We seek factors z and a map $y = T(x, z)$ that solve

$$\max_{z} \left\{ \min_{y=T(x,z)} \int c(x, y)\rho(x \,|\, z)\gamma(z)\, dx\, dz \quad \mathrm{s.t.} \quad y \perp z \right\}.$$

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

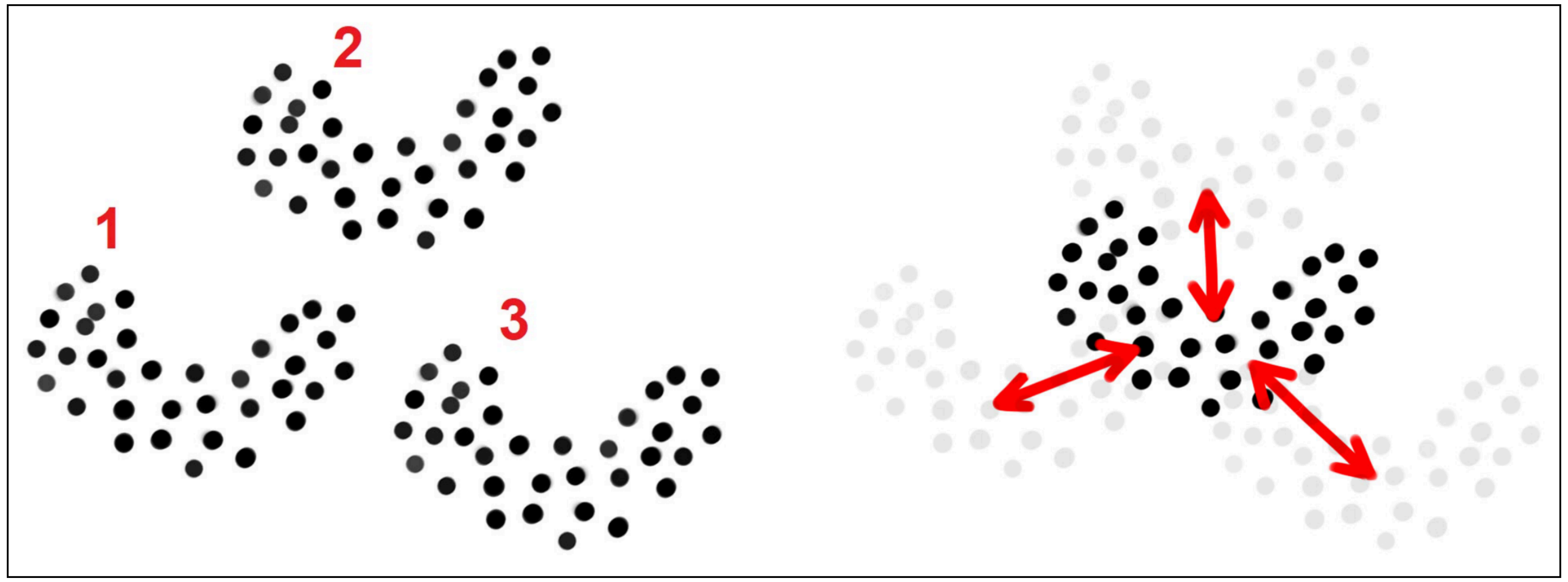
A relaxation of the factor discovery problem

We seek factors z and a map $y = T(x, z)$ that solve

$$\max_{z} \left\{ \min_{y=T(x,z)} \int c(x, y)\rho(x \mid z)\gamma(z)\, dx\, dz \quad \text{s.t.} \quad y \perp z \right\}.$$

In the case of a discrete-valued z, a natural relaxation of the independence condition is that

$$\bar{y} = \bar{y}(z)\ \forall\ z$$



Yang, H., & Tabak, E. G. (2020). Conditional Density Estimation, Latent Variable Discovery and Optimal Transport.

Clusters as Discrete Factors
A relaxation of the factor discovery problem

We seek factors z and a map $y = T(x, z)$ that solve

$$\max_{z} \left\{ \min_{y=T(x,z)} \int c(x, y)\rho(x \,|\, z)\gamma(z)\, dx\, dz \quad \text{s.t.} \quad y \perp z \right\}.$$

Premises of relaxation:

1) $\bar{y} = \bar{y}(z) \ \forall \ z$

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

A relaxation of the factor discovery problem

We seek factors z and a map $y = T(x, z)$ that solve

$$\max_{z} \left\{ \min_{y=T(x,z)} \int c(x, y)\rho(x \mid z)\gamma(z)\, dx\, dz \quad \text{s.t.} \quad y \perp z \right\}.$$

Premises of relaxation:

1) $\bar{y} = \bar{y}(z)\ \forall\ z$
2) $c(x, y) = \|x - y\|^2$

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

A relaxation of the factor discovery problem

We seek factors z and a map $y = T(x, z)$ that solve

$$\max_{z} \left\{ \min_{y=T(x,z)} \int c(x, y)\rho(x \,|\, z)\gamma(z)\, dx\, dz \quad \text{s.t.} \quad y \perp z \right\}.$$

Premises of relaxation:

1) $\bar{y} = \bar{y}(z) \ \forall \ z$
2) $c(x, y) = \|x - y\|^2$

We seek I_k that solve the data-driven formulation,

$$\max_{I_k} \left\{ \sum_{k=1}^{p} [I_k] \, \|\bar{y} - \bar{x}(z_k)\|^2 \right\},$$

where I_k is a set containing the identities of points attributable to the class z_k.

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

A relaxation of the factor discovery problem

We seek I_k that solve the data-driven formulation,

$$\max_{I_k} \left\{ \sum_{k=1}^{p} [I_k] \, \|\bar{y} - \bar{x}(z_k)\|^2 \right\}.$$

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

A relaxation of the factor discovery problem

We seek I_k that solve the data-driven formulation,

$$\max_{I_k} \left\{ \sum_{k=1}^{p} [I_k] \, \|\overline{y} - \overline{x}(z_k)\|^2 \right\}.$$

Since $\overline{y} = \overline{x}$ and since

$$\sum_{k=1}^{p} \sum_{i \in I_k} \left\| x^i - \bar{x}(z_k) \right\|^2 + \sum_{k=1}^{p} [I_k] \, \| \bar{x} - \bar{x}(z_k) \|^2 = \sum_{i=1}^{n} \left\| x^i - \bar{x} \right\|^2 ,$$

our problem is equivalent to

$$\min_{I_k} \sum_{k=1}^{p} \sum_{i \in I_k} \|x^i - \overline{x}(z_k)\|^2.$$

Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

Clusters as Discrete Factors

A refresher on the arithmetic mean

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=1}^{N}$, the arithmetic mean is

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x^i$$

Clusters as Discrete Factors

A refresher on the arithmetic mean

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=1}^N$, the arithmetic mean is

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x^i$$
$$= \underset{\hat{x}}{\operatorname{argmin}} \sum_{i=1}^{N} |x^i - \hat{x}|^2,$$

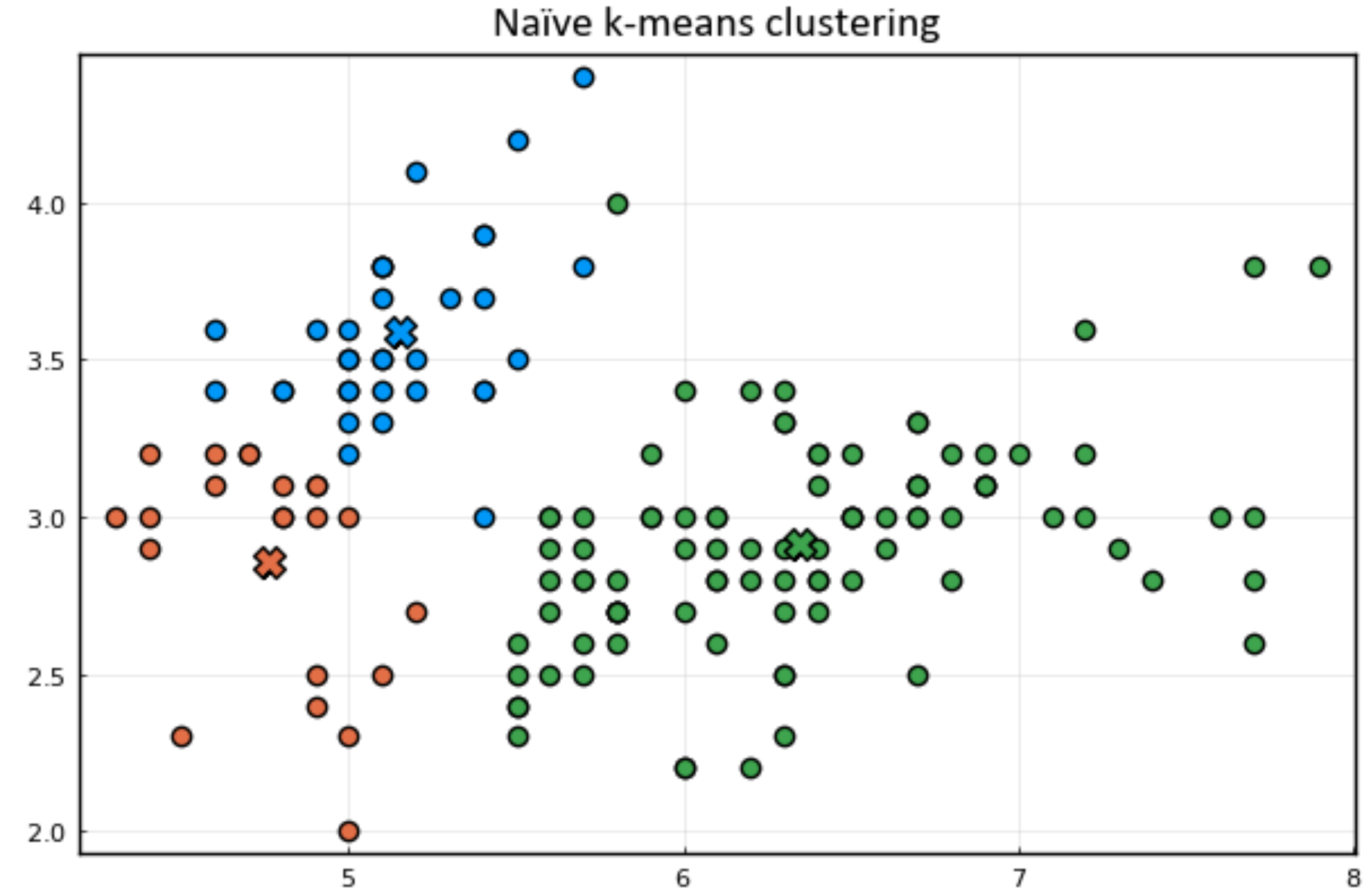
which, performed component-wise with vectors x^i, is precisely the centroid from k-Means.

In fact, our relaxed, data-driven optimization problem is equivalent to k-Means.

$$\min_{I_k} \sum_{k=1}^{p} \sum_{i \in I_k} \|x^i - \bar{x}(z_k)\|^2$$

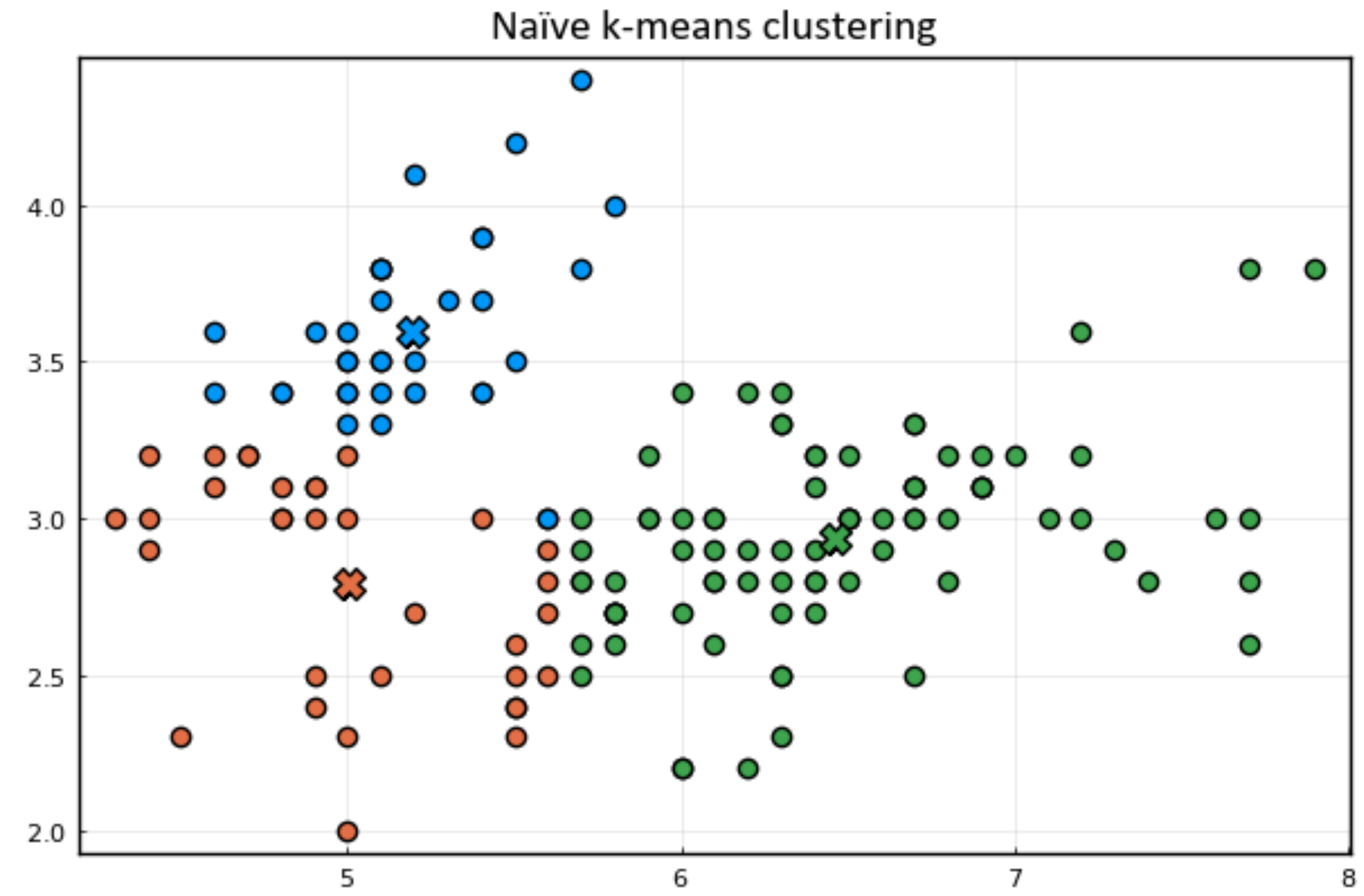
Tabak, E. G. (2023, June 7). Factor Discovery Through Optimal Transport [Draft].

k-Means
the standard algorithm



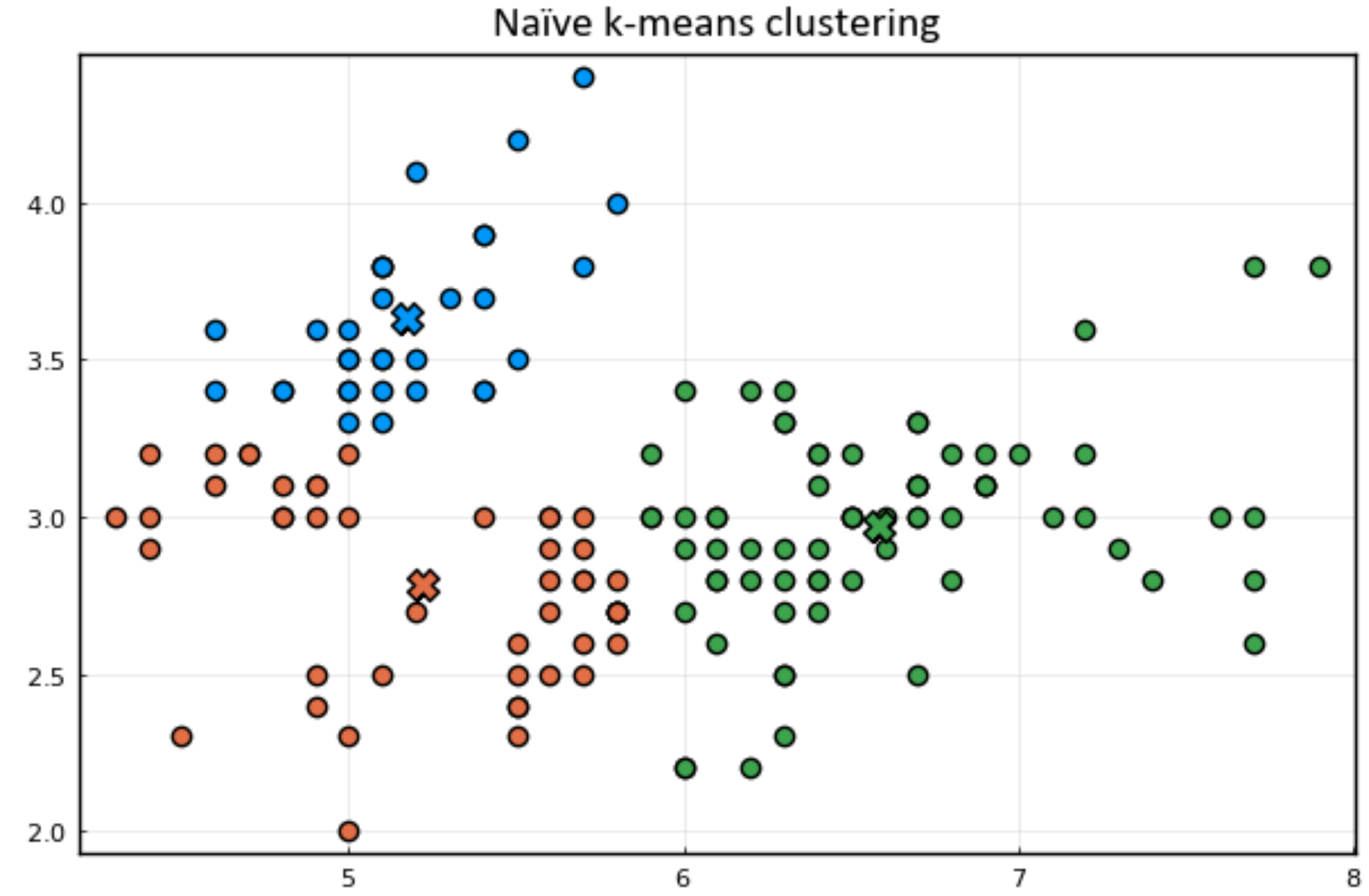
Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

k-Means
the standard algorithm



Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

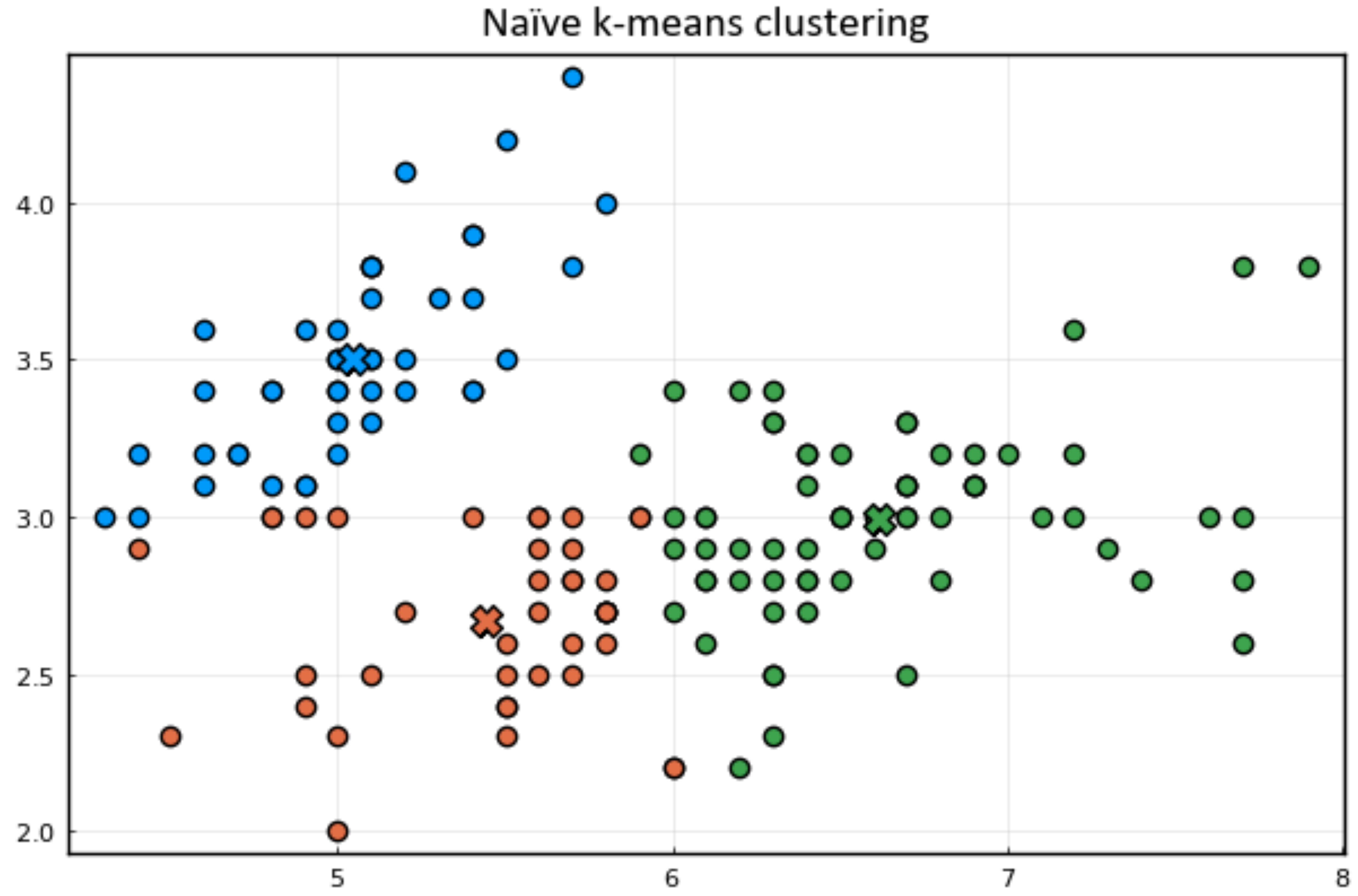
k-Means
the standard algorithm



Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

k-Means

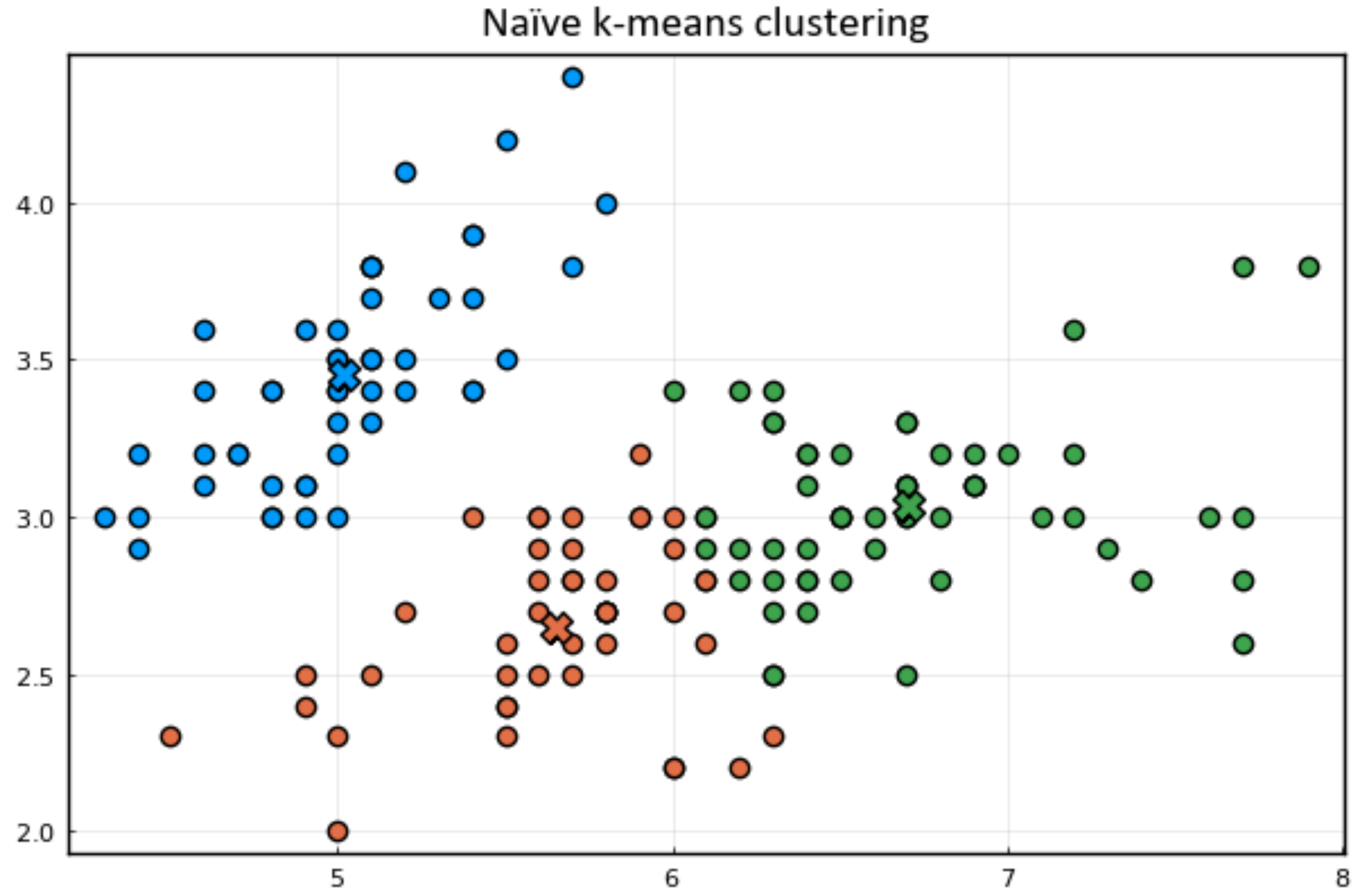
the standard algorithm



Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

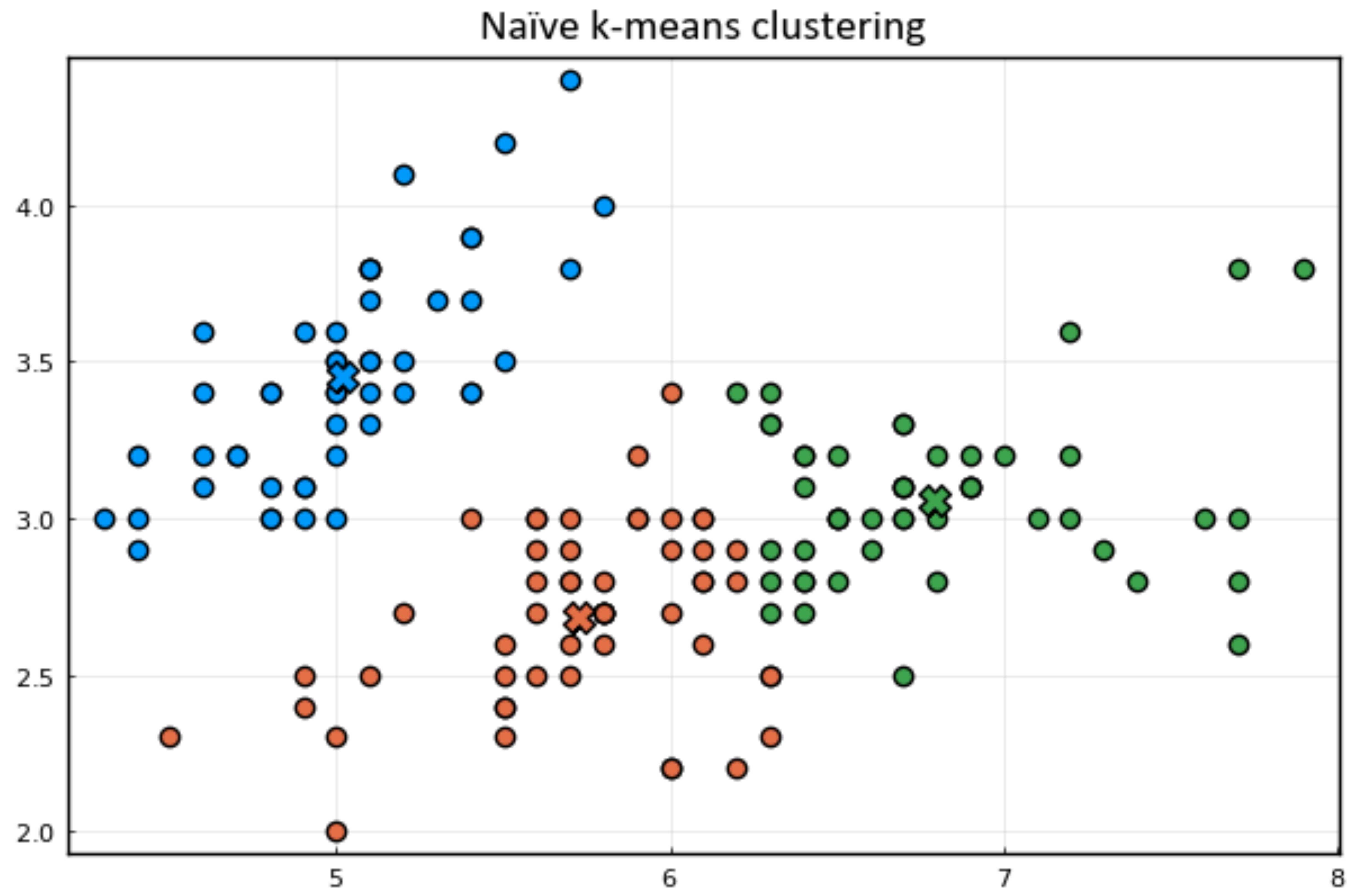
k-Means

the standard algorithm



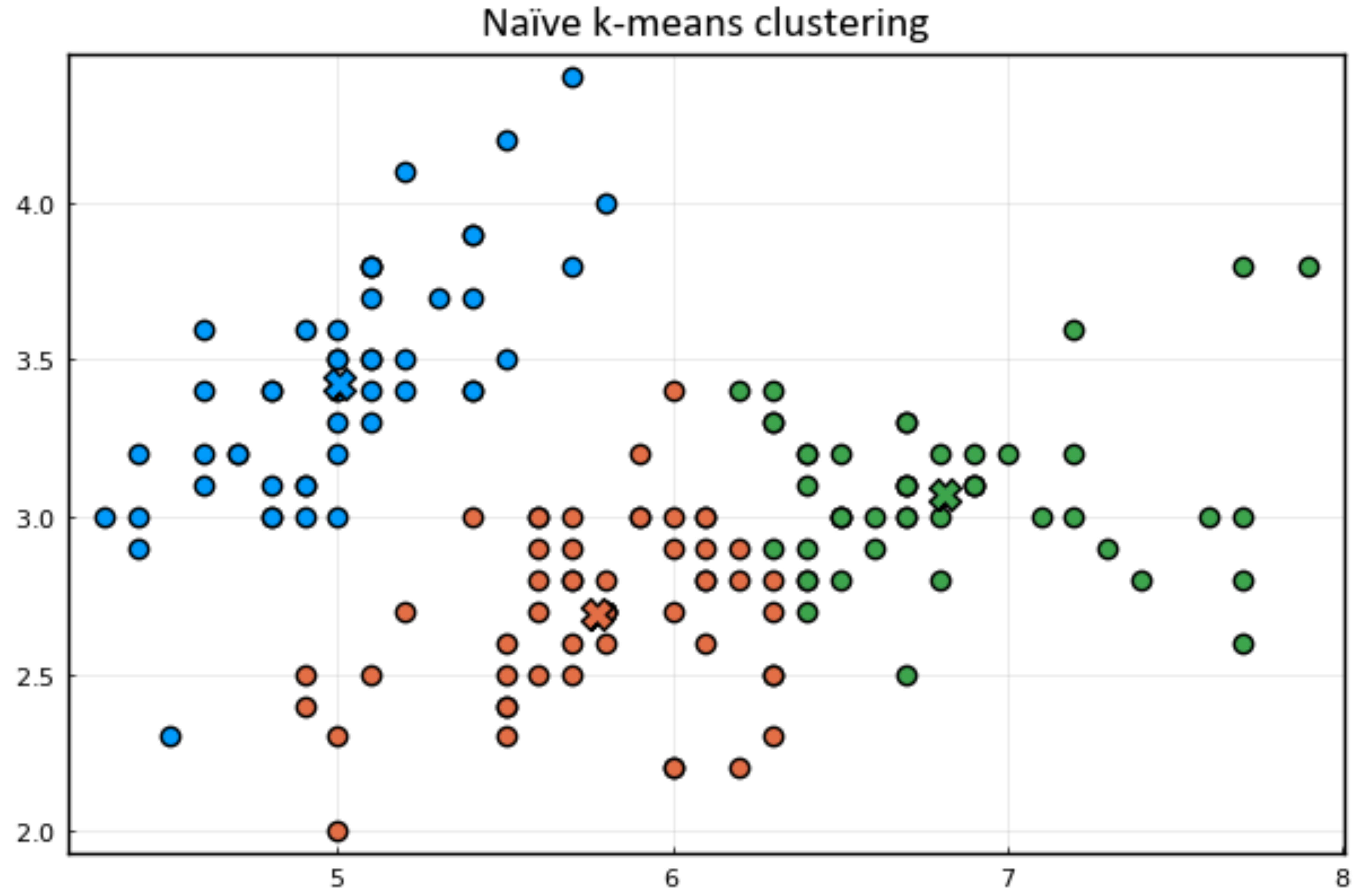
Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

k-Means
the standard algorithm



Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

k-Means
the standard algorithm



Mroavi. (2023). Naive K-Means Implementation in Julia. Julia Forem. https://forem.julialang.org/mroavi/naive-k-means-39dk

k-GenCenters

Coming To A Repository Near You…

k-GenCenters

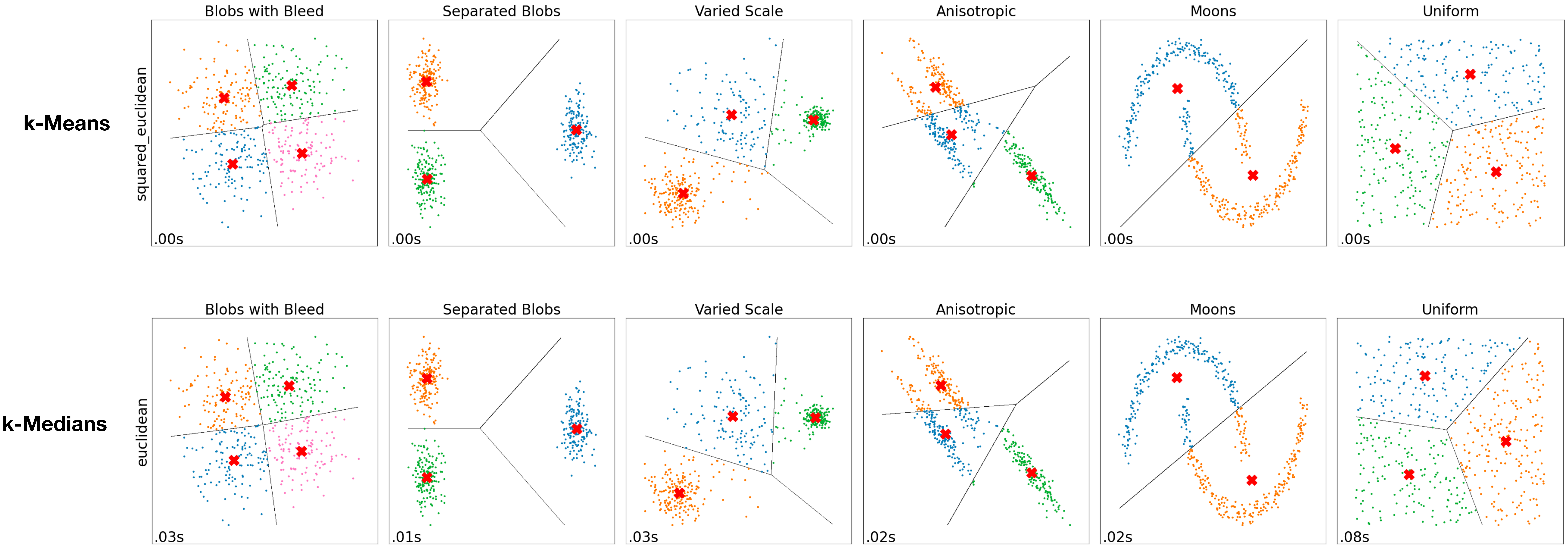
An Extension of k-Means to General Costs

The k-GenCenters module…

- is styled after sklearn.cluster.KMeans
- has multiple initialization options, including kGenCenters++
- can perform variations on k-Means using
    - ✦ Any L^p norm $c(x, \hat{x}) = \left(\sum_{d=1}^{D} |x_d - \hat{x}_d|^p\right)^{\frac{1}{p}}$
    - ✦ Any power of the Euclidean distance $c(x, \hat{x}) = \|x - \hat{x}\|_2^n$
    - ✦ Any future cost functions contributed by the community
- can generate Voronoi diagrams

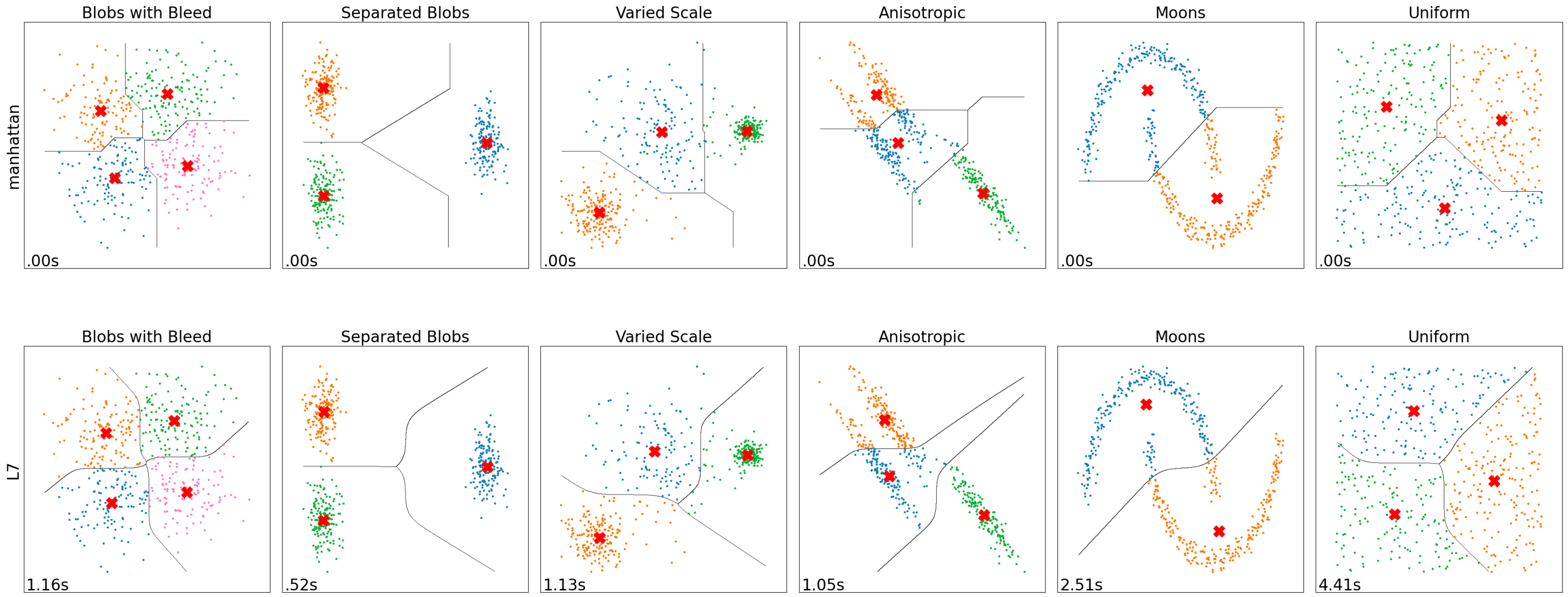
k-GenCenters

An Extension of k-Means to General Costs



k-GenCenters

An Extension of k-Means to General Costs



k-Medians

A refresher on the median

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=0}^{N}$, the median is

$$\begin{aligned}\text{Median} &= x^{\frac{N}{2}} \\ &= \operatorname*{argmin}_{\hat{x}} \sum_{i=0}^{N} |x^i - \hat{x}|.\end{aligned}$$

k-Medians

A refresher on the median

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=0}^{N}$, the median is

$$\begin{aligned}\text{Median} &= x^{\frac{N}{2}} \\ &= \underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} | x^i - \hat{x} | .\end{aligned}$$

Naïve k-Medians finds the component-wise medians of vectors x^i, solving

$$\underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} \|x^i - \hat{x}\|_1 .$$

k-Medians

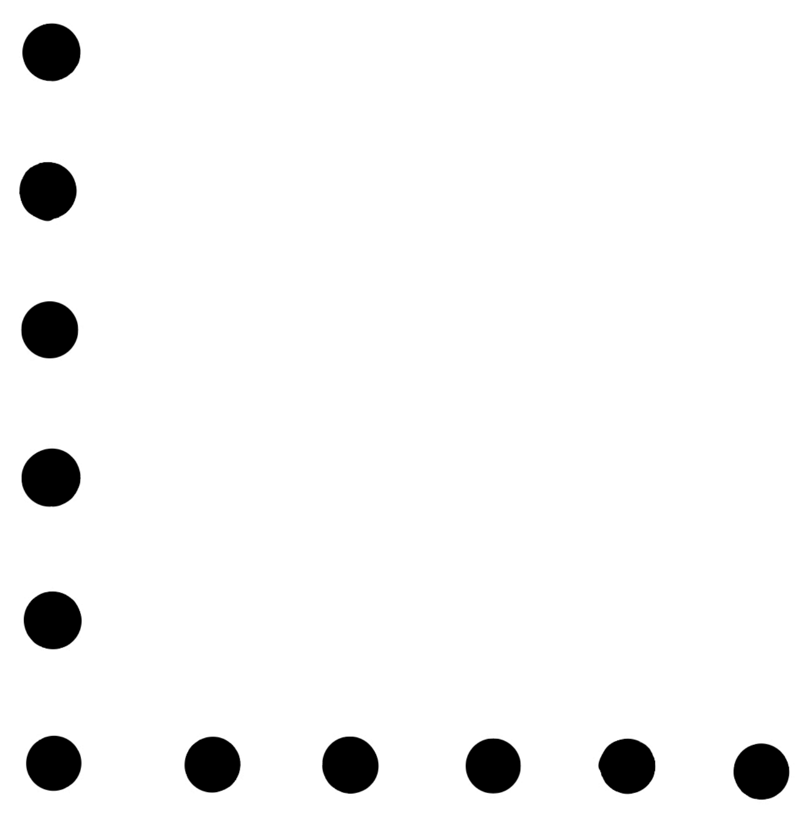
A refresher on the median

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=0}^{N}$, the median is

$$\begin{aligned}\text{Median} &= x^{\frac{N}{2}} \\ &= \underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} |x^i - \hat{x}|.\end{aligned}$$

Naïve k-Medians finds the component-wise medians of vectors x^i, solving

$$\underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} \|x^i - \hat{x}\|_1.$$



k-Medians

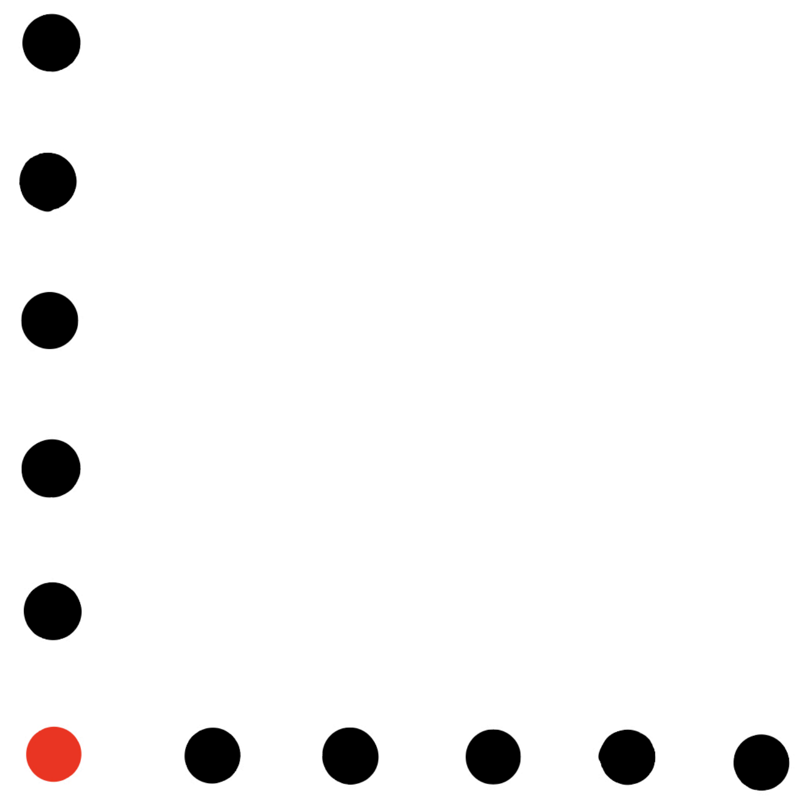
A refresher on the median

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=0}^{N}$, the median is

$$\begin{aligned}\text{Median} &= x^{\frac{N}{2}} \\ &= \operatorname*{argmin}_{\hat{x}} \sum_{i=0}^{N} |x^i - \hat{x}|.\end{aligned}$$

Naïve k-Medians finds the component-wise medians of vectors x^i, solving

$$\operatorname*{argmin}_{\hat{x}} \sum_{i=0}^{N} \|x^i - \hat{x}\|_1.$$



k-Medians

A refresher on the median

Given a set of numbers $\{x^i \in \mathbb{R}\}_{i=0}^{N}$, the median is

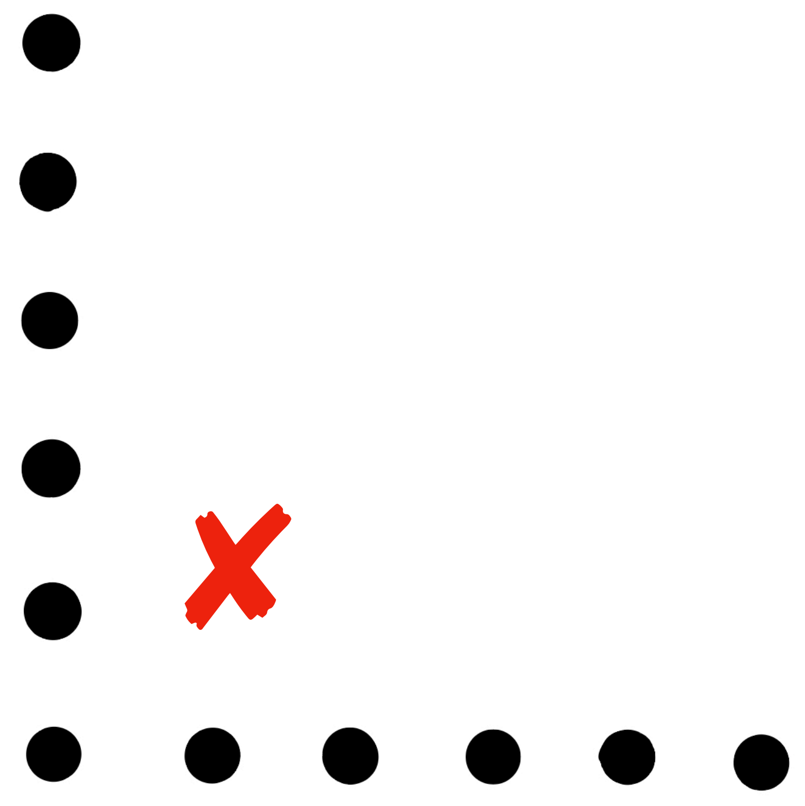
$$\begin{aligned} \text{Median} &= x^{\frac{N}{2}} \\ &= \underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} | x^i - \hat{x} | . \end{aligned}$$

Naïve k-Medians finds the component-wise medians of vectors x^i, solving

$$\underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} \| x^i - \hat{x} \|_1 .$$

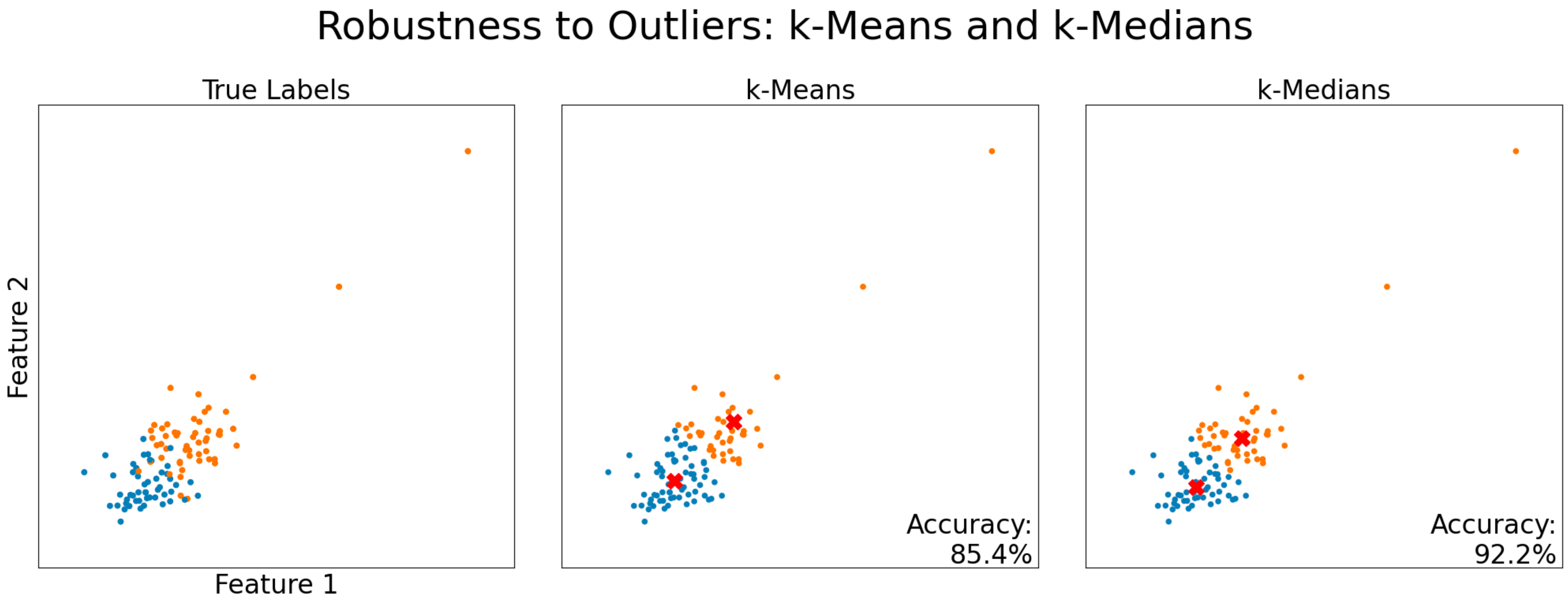
We want the more natural "geometric median":

$$\underset{\hat{x}}{\operatorname{argmin}} \sum_{i=0}^{N} \| x^i - \hat{x} \|_2$$



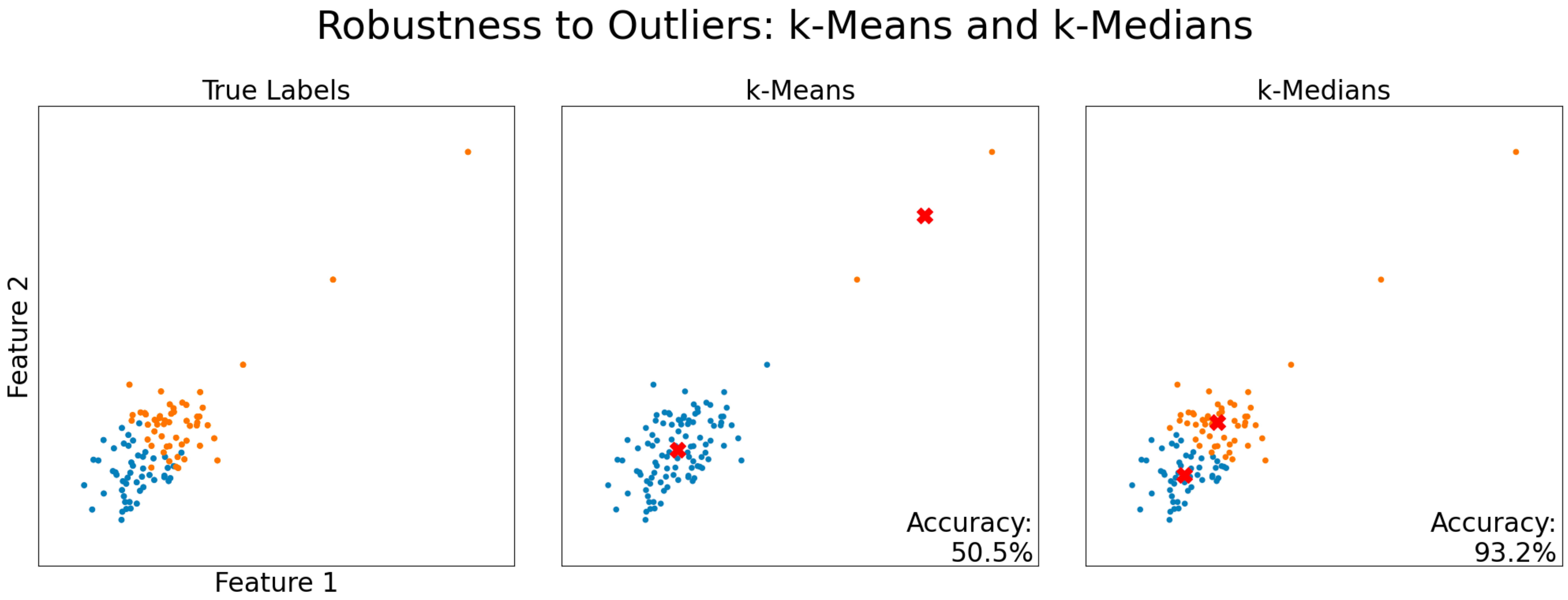
k-Means *vs* k-Medians

Performance with outliers



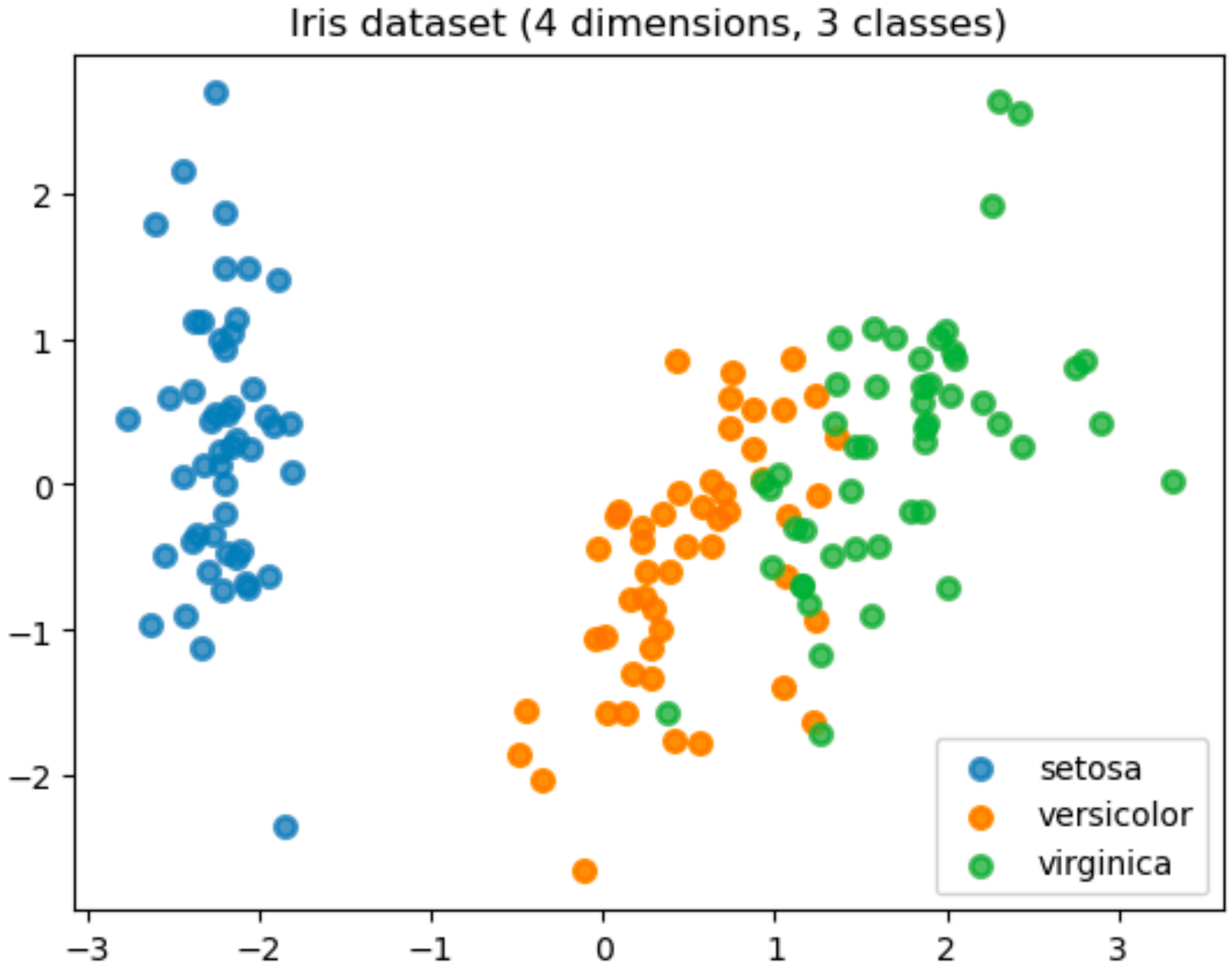
k-Means *vs* k-Medians

Performance with outliers



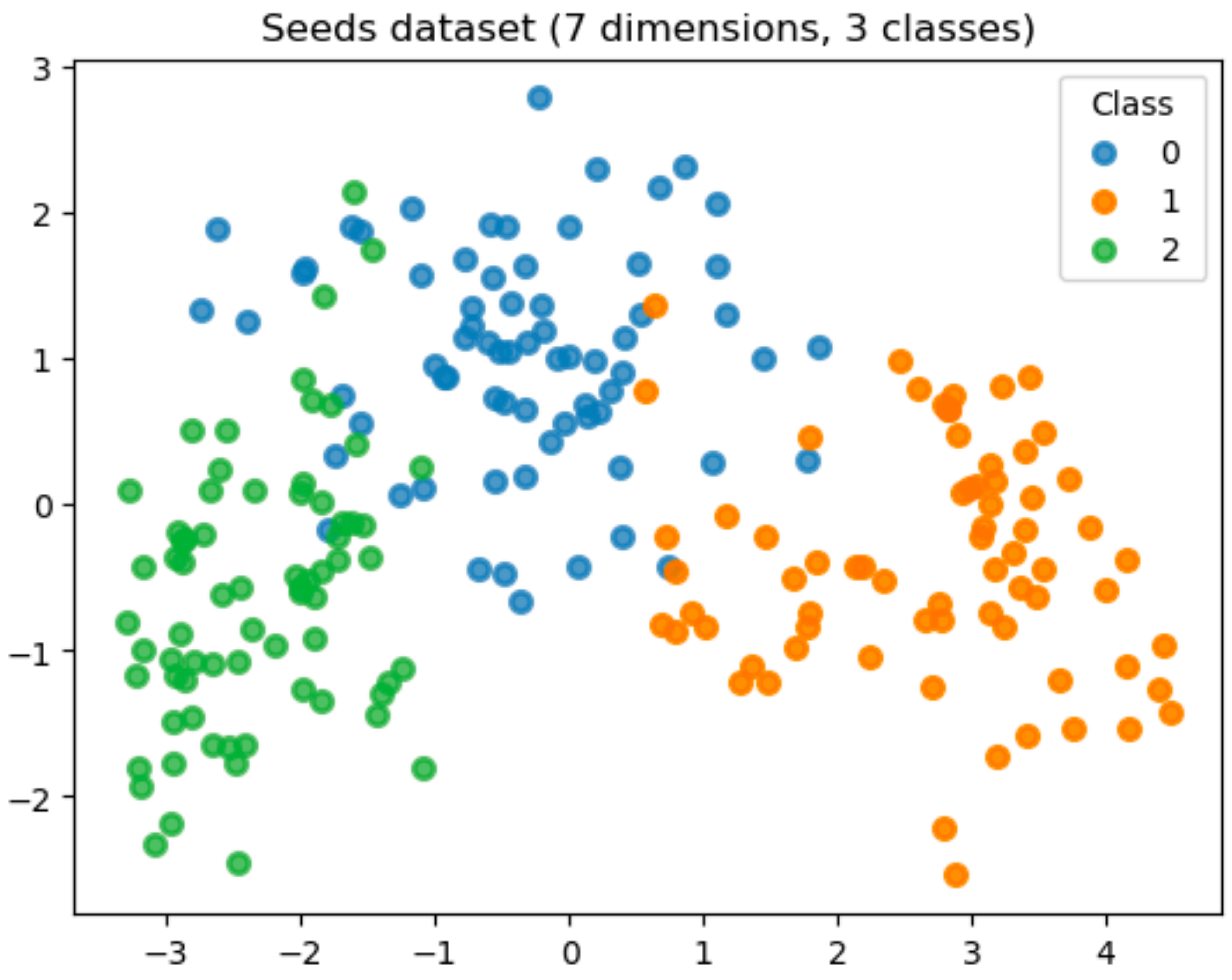
k-Means *vs* k-Medians

Performance on real-world datasets



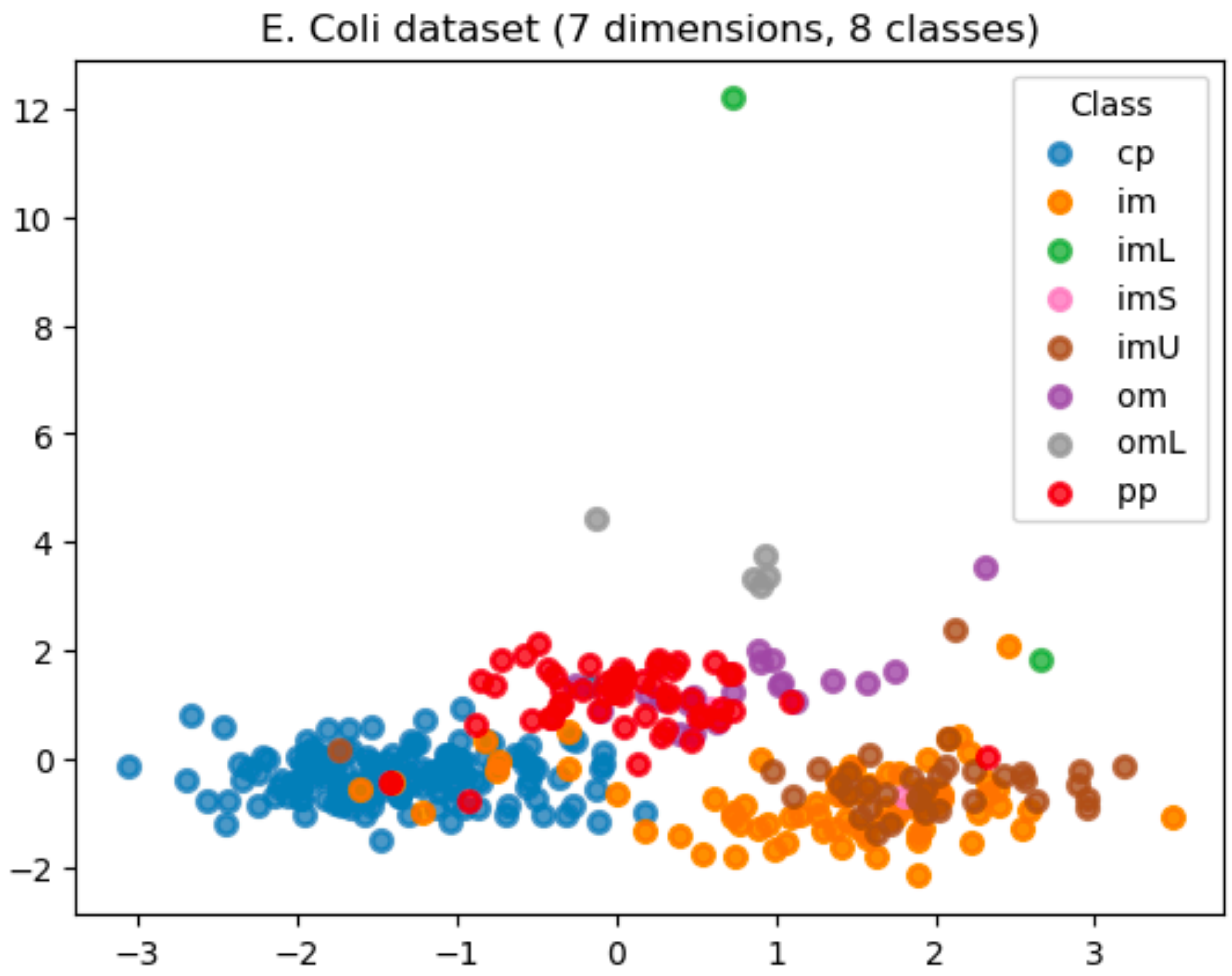
k-Means *vs* k-Medians

Performance on real-world datasets



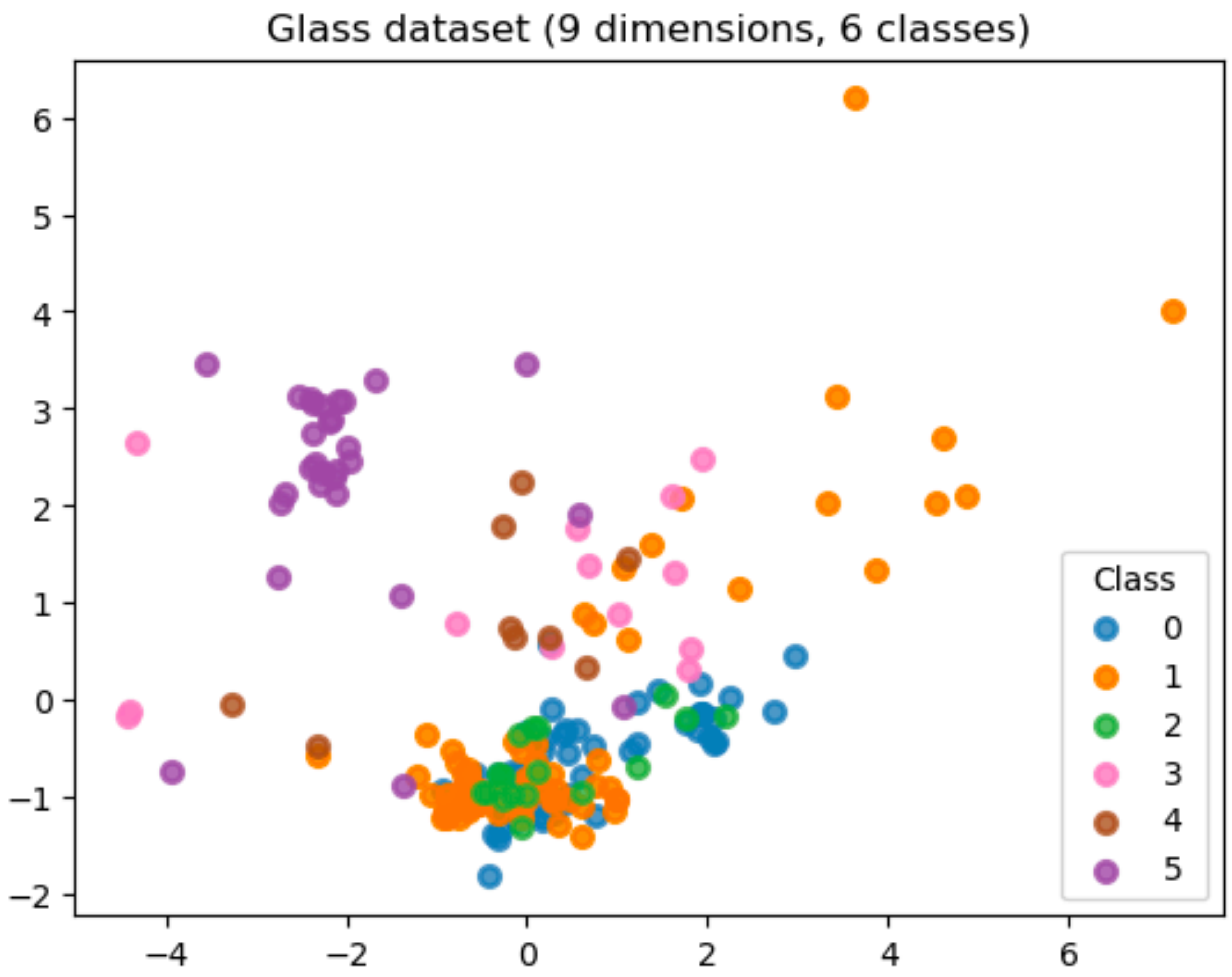
k-Means *vs* k-Medians

Performance on real-world datasets



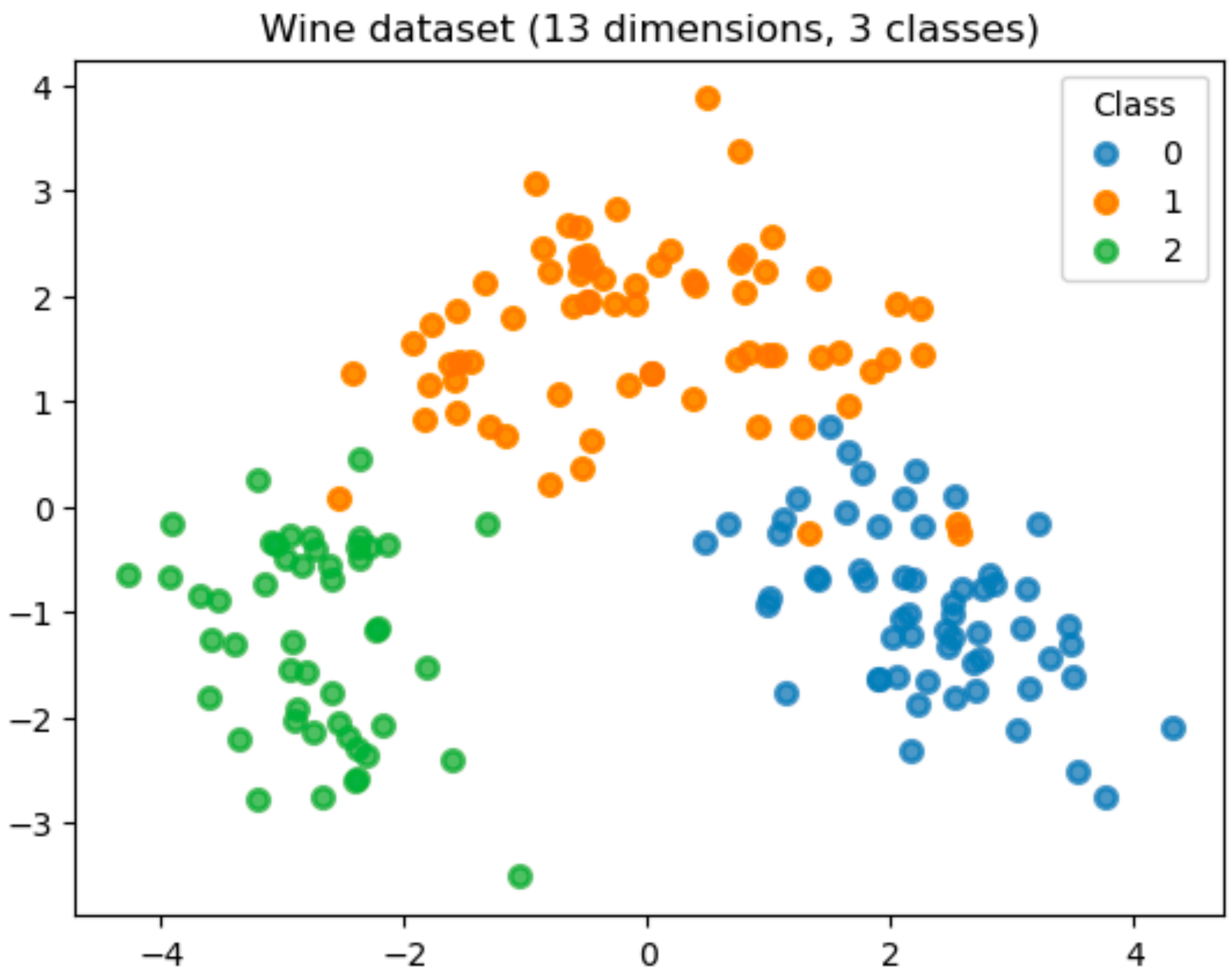
k-Means *vs* k-Medians

Performance on real-world datasets



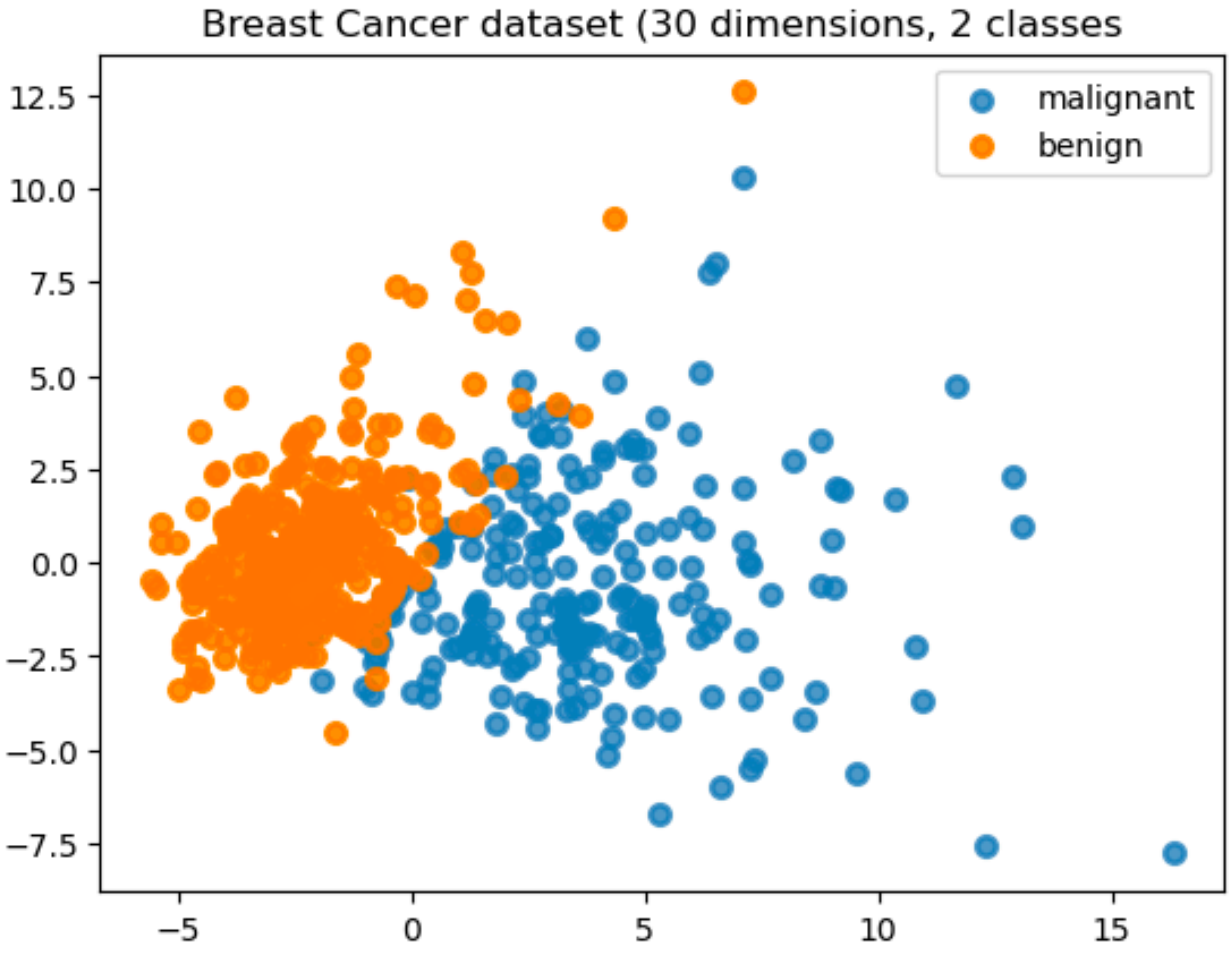
k-Means *vs* k-Medians

Performance on real-world datasets



k-Means *vs* k-Medians

Performance on real-world datasets



k-Means *vs* k-Medians

A Performance Comparison

Average Accuracy of k-Means and k-Medians (over 100 trials)

	k-Means	**k-Medians**
Iris	79.57%	78.73%
Seeds	91.71%	90.82%
E. Coli	74.47%	74.85%
Glass	44.84%	42.29%
Wine	94.31%	95.82%
Breast Cancer	90.80%	91.62%

Improving Initialization

Forgy

- Initialize the **centers** randomly

Random Partition

- Initialize the **assignments of the data points** randomly

k-Means++ (improved Forgy)

- Initialize the **centers**
- Incentivize distance from existing centers
- $\mathbb{P}(z_k = x^i) = \dfrac{\min\limits_{z_{j<k}} \|x^i - z_j\|^2}{\sum_{w=1}^{N} \min\limits_{z_{j<k}} \|x^w - z_j\|^2}$

Improving Initialization

Forgy

- Initialize the **centers** randomly

Random Partition

- Initialize the **assignments of the data points** randomly

k-Means++ (improved Forgy)

- Initialize the **centers**
- Incentivize distance from existing centers
- $\mathbb{P}(z_k = x^i) = \dfrac{\min\limits_{z_{j<k}} \|x^i - z_j\|^2}{\sum_{w=1}^{N} \min\limits_{z_{j<k}} \|x^w - z_j\|^2}$

k-GenCenters++

- Incentivize a ***custom cost*** from existing centers
- $\mathbb{P}(z_k = x^i) = \dfrac{\min\limits_{z_{j<k}} c(x^i, z_j)}{\sum_{w=1}^{N} \min\limits_{z_{j<k}} c(x^w, z_j)}$

Improving Initialization

Breast Cancer Dataset:
Accuracies of Various Initializations (avg. over 100 trials)

	k-Means	**k-Medians**
Forgy	90.80%	91.62%
Random Partition	90.73%	91.61%
++	90.94%	91.64%
Euclidean ++	90.90%	91.64%
Euclidean^2 ++	90.94%	91.65%
Euclidean^3 ++	90.99%	91.68%

Must-link Constraints

A form of semi-supervised clustering

Points *known to share a factor* (e.g. belonging to the same person) are “must-link”.

Must-link Constraints

A form of semi-supervised clustering

Points *known to share a factor* (e.g. belonging to the same person) are “must-link”.

At every assignment step, for every set of must-link identities I_p, we assign the whole set to

$$\operatorname*{argmin}_z \sum_{i \in I_p} c(x^i, z).$$

Must-link Constraints

A form of semi-supervised clustering

Points *known to share a factor* (e.g. belonging to the same person) are “must-link”.

At every assignment step, for every set of must-link identities I_p, we assign the whole set to

$$\underset{z}{\operatorname{argmin}} \sum_{i \in I_p} c(x^i, z).$$

By this approach,

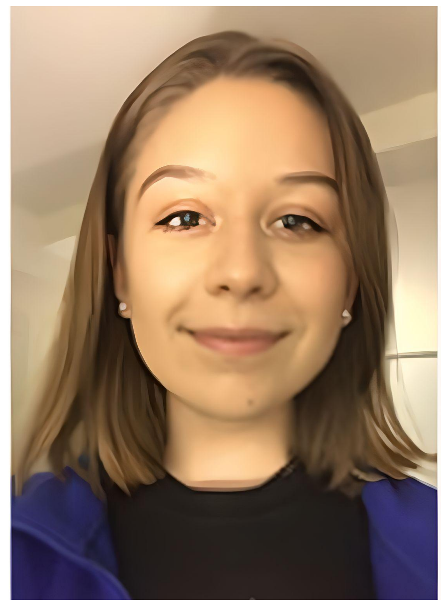
- the centers may be more robust against local minima
- the algorithm can be expected to converge faster
  - ✦ Further speedups may be accessible using a weighted surrogate

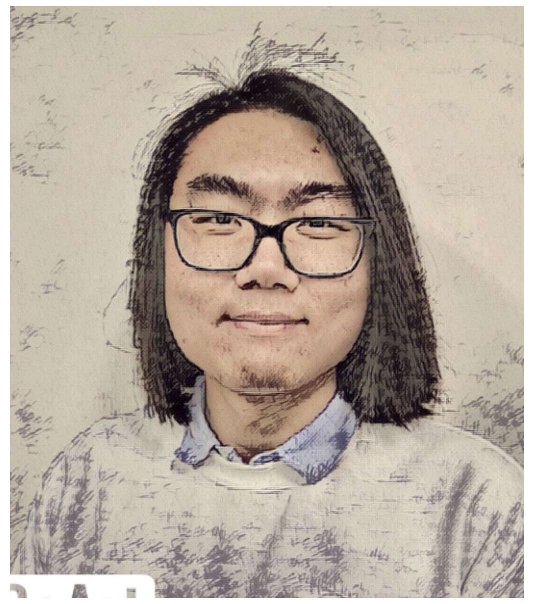
Acknowledgements

Prof. Esteban Tabak!

Andrew Lipnick and Nina Mortensen!

Olivia, Fortino, and Ryan!





Prof. Aleksandar Donev!