Power k-Means Clustering

Kathryn Carbone, Cindy Lyu, Erik Swanke

Paper Background

- Power k-Means Clustering
- Jason Xu, Kenneth Lange
- Proceedings of 36th International Conference on Machine Learning
- 2019

Motivation

An introduction to clustering

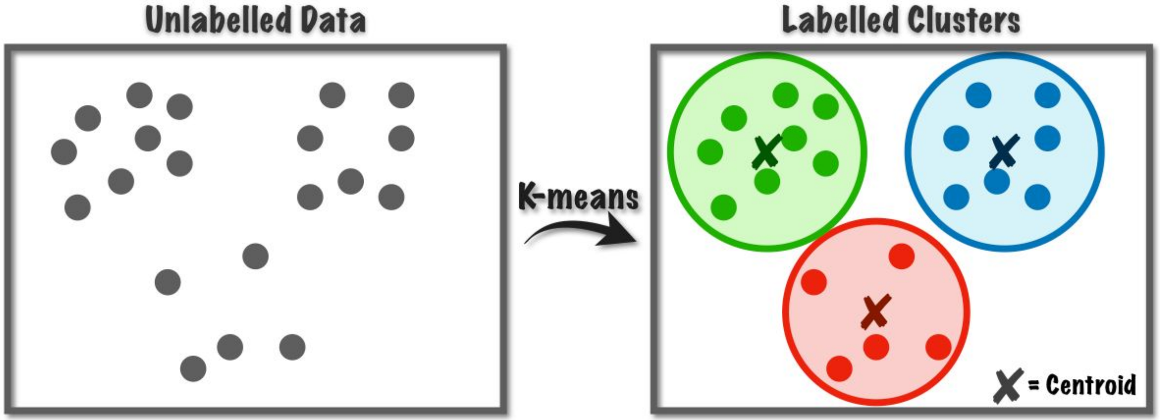
What is clustering?

$$f_{-\infty}(\Theta) = \sum_{j=1}^{k} \sum_{i \in C_j} ||x_i - \theta_j||^2$$

What is the problem?

Why is it interesting?

$$f_{-\infty}(\Theta) = \sum_{i=1}^{N} min_{1 \leq j \leq k} ||x_i - \theta_j||^2$$



Lloyd's Algorithm

Algorithm 1 Lloyd's Algorithm

Initialize Θ with k random datapoints
$X \leftarrow x$
$\Theta \leftarrow \theta$ Randomly initialize centroids
while Not Converged **do**
  Empty each C_j
  for $x_i \in X$ **do**
    $j = argmin_j\{||x_i - \theta_j||\}$
    $C_j \leftarrow C_j \cup x_i$ ▷ Reassign each datapoint to its nearest cluster
  end for
  for $j = 1 \cdots k$ **do**
    $\theta_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$ ▷ Recalculate each centroid as the average of points in its cluster
  end for
end while

Generalized and Power Means

- A simple way of describing a family of functions

$$M_g(\mathbf{y}) = g^{-1}\left[\frac{g(y_1) + \cdots + g(y_k)}{k}\right]$$

$$M_s(\mathbf{y}) = \left[\frac{y_1^s + \cdots + y_k^s}{k}\right]^{\frac{1}{s}}$$

$$\frac{\partial}{\partial y_j} M_s(\mathbf{y}) = \frac{1}{k}\left(\sum_{i=1}^{k} y_i^s\right)^{\frac{1}{s}-1} \frac{1}{k} y_j^{s-1} > 0$$

k-Harmonic Means

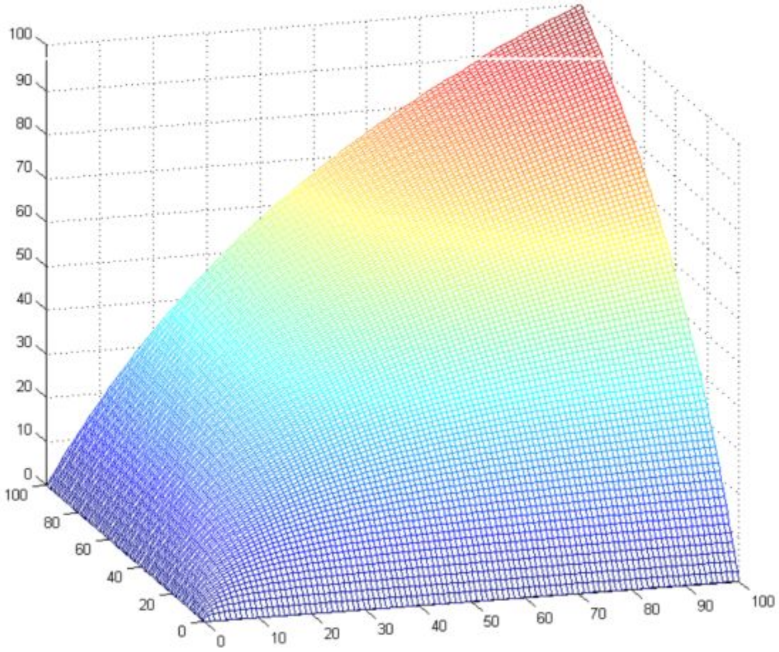
$$f_{-\infty}(\Theta) = \sum_{i=1}^{N} min_{1 \le j \le k} ||x_i - \theta_j||^2$$

- Replace k-means objective with proxy

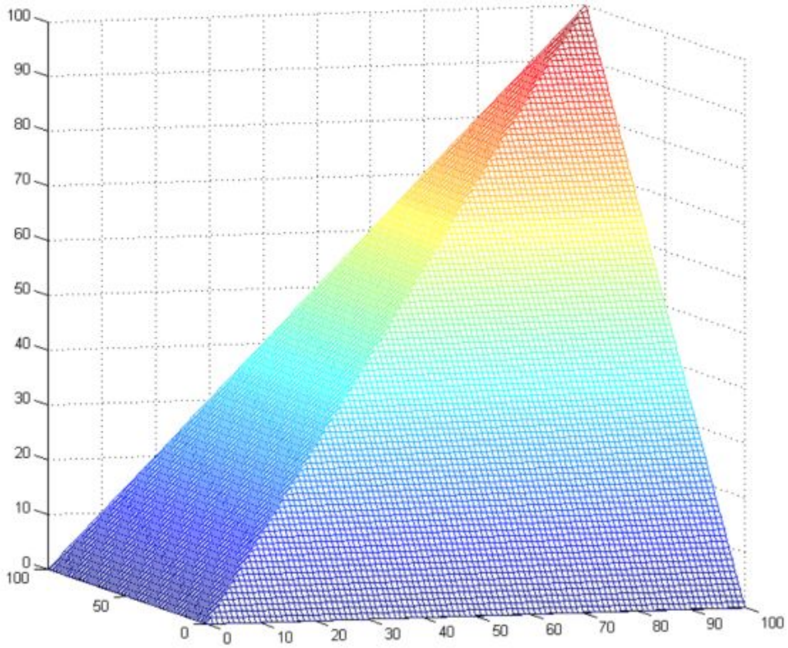
$$f_{-1}(\Theta) = \sum_{i=1}^{N} HA_{1 \le j \le k} ||x_i - \theta_j||^2 \qquad f_{-1}(\Theta) = \sum_{i=1}^{N} \frac{k}{\sum_{j=1}^{k} (||x_i - \theta_j||^2)^{-1}}$$

- Fewer local minima
- Curse of dimensionality

k-Harmonic Means

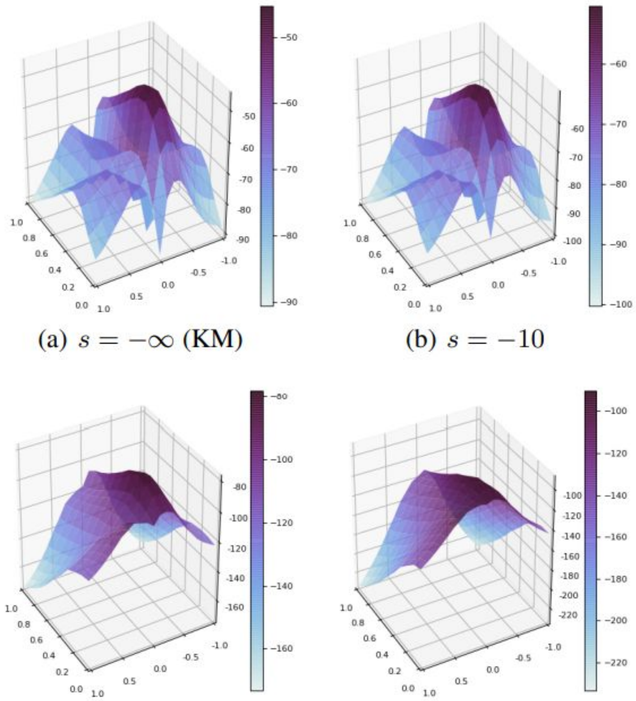


Harmonic Average



Min Function

Objective surfaces



(a) $s = -\infty$ (KM)

(b) $s = -10$

(c) $s = -1$ (KHM)

(d) $s = -0.2$

- Simulated data
- 100 data points
- 3 clusters
- 1 dimension

Power k-Means

Power K-Means: Objective Function

$$f_s(\Theta) = \sum_{i=1}^{n} M_s(||x_i - \theta_1||^2, \cdots, ||x_i - \theta_k||^2)$$

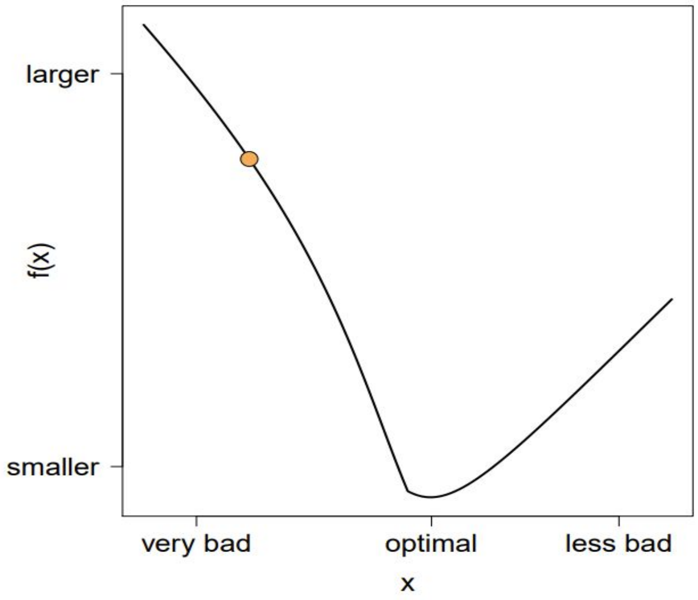
$$f_s(\Theta) = [\frac{(||x_i - \theta_1||^2)^s + \cdots + (||x_i - \theta_k||^2)^s}{k}]^{-s}$$

Expanded objective is general case of KHM

$$f_{-1}(\Theta) = \sum_{i=1}^{N} \frac{k}{\sum_{j=1}^{k}(||x_i - \theta_j||^2)^{-1}}$$

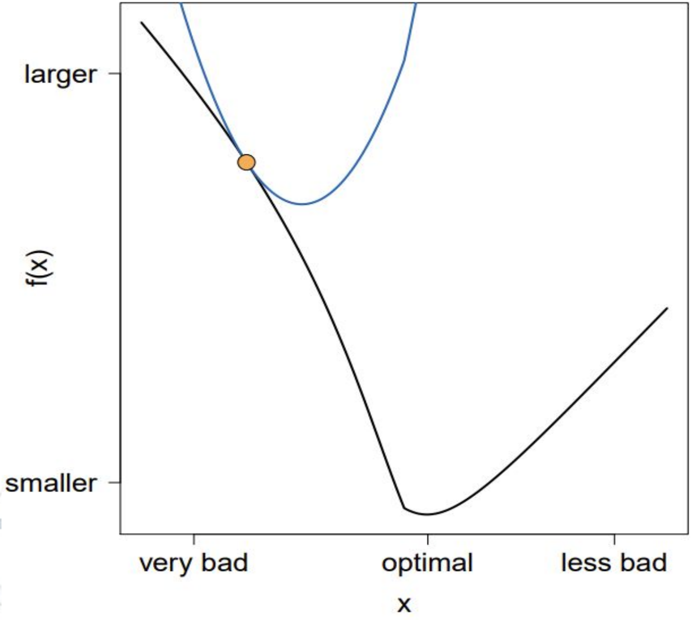
Power K-Means: Majorization Minimization (MM)

- Non-convex objective function f(x).



Power K-Means: Majorization Minimization (MM)

- Surrogate function $g(x|x_m)$

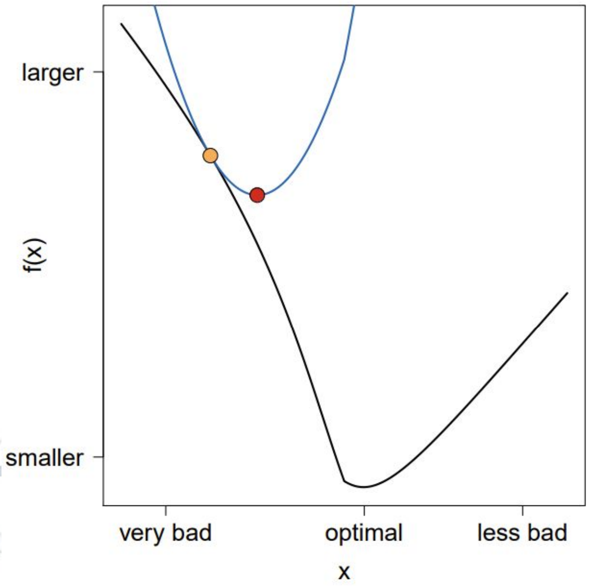


$$f(x_m) = g(x_m \mid x_m) \qquad \text{tangency at } x_m$$

$$f(x) \leqslant g(x \mid x_m) \qquad \text{domination for all } x$$

Power K-Means: Majorization Minimization (MM)

- Minimize the surrogate function to obtain the new estimate.

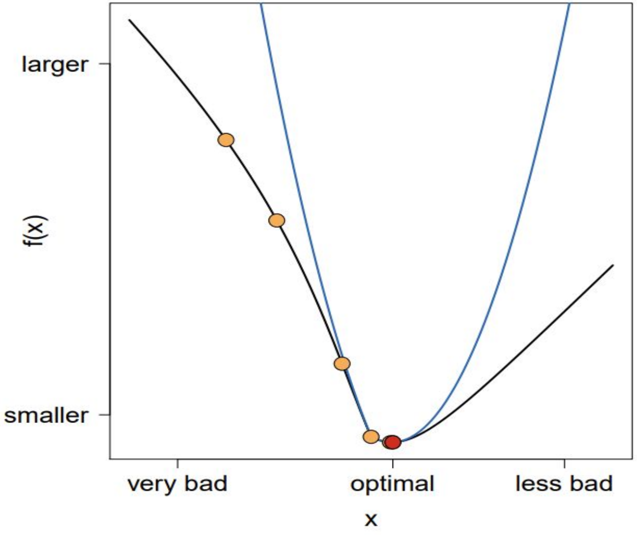


Updating Rule:

$$x_{m+1} = \arg\min_{x} g(x \mid x_m)$$

Power K-Means: Majorization Minimization (MM)

- Find a sequence of surrogate functions until convergence to a stationary point.



Power K-Means: The Algorithm

$$f_s(\Theta) = \sum_{i=1}^{n} M_s(||x_i - \theta_1||^2, \cdots, ||x_i - \theta_k||^2)$$

Algorithm 2 Power k-Means Algorithm

Input: Data matrix $X \in R^{n \times d}$, number of clusters k

1: Initialize $s_0 < 0$, Θ_0, $\eta > 1$ ▷ initial exponent s_0, exponent increment factor η

2: **repeat**

3: $\quad \omega_{m,ij} \leftarrow \left(\sum_{l=1}^{k} ||x_i - \theta_{m,l}||^{2s_m}\right)^{\frac{1}{s_m}-1} ||x_i - \theta_{m,l}||^{2(s_m-1)}$

4: $\quad \theta_{m+1,j} \leftarrow \left(\sum_{i=1}^{n} \omega_{m,ij}\right)^{-1} \sum_{i=1}^{n} \omega_{m,ij} x_i$

5: $\quad s_{m+1} \leftarrow \eta \cdot s_m$ ▷ Optional Annealing

6: **until** convergence

Power K-Means: The Algorithm

$$f_s(\Theta) = \sum_{i=1}^{n} M_s(||x_i - \theta_1||^2, \cdots, ||x_i - \theta_k||^2)$$

- Exploit the convexity by computing the Hessian:
  - S < 1, concave
  - Otherwise, convex
- Find the linear majorization:

$$M_s(\mathbf{y}) \leqslant M_s(\mathbf{y}_m) + \sum_{j=1}^{K} \nabla_{\mathbf{y}_j} M_s(\mathbf{y}_m)^T(\mathbf{y}_j - \mathbf{y}_{m,j}).$$

Power K-Means: The Algorithm

- Substitution and Summation:

$$f_s(\Theta) \leqslant f_s(\Theta_m) + \sum_{i=1}^{n}\sum_{j=1}^{K} \nabla_{y_j} M_s(y_m)\|x_i - \theta_j\|^2 - \sum_{i=1}^{n}\sum_{j=1}^{K} \nabla_{y_j} M_s(y_m)\|x_i - \theta_{m,j}\|^2.$$

- Focus on the term that is relevant in minimizing the right-hand side.

$$\omega_{m,ij} = \nabla_{y_j} M_s(y_m) = \nabla_{\theta_j} M_s(\|x_i - \theta_{m,k}\|^2)$$

Power K-Means: The Algorithm

- Do math to get the weights function (step 3) and the updating rule (step 4).

Algorithm 2 Power k-Means Algorithm

Input: Data matrix $X \in R^{n \times d}$, number of clusters k

1: Initialize $s_0 < 0$, Θ_0, $\eta > 1$ ▷ initial exponent s_0, exponent increment factor η

2: **repeat**

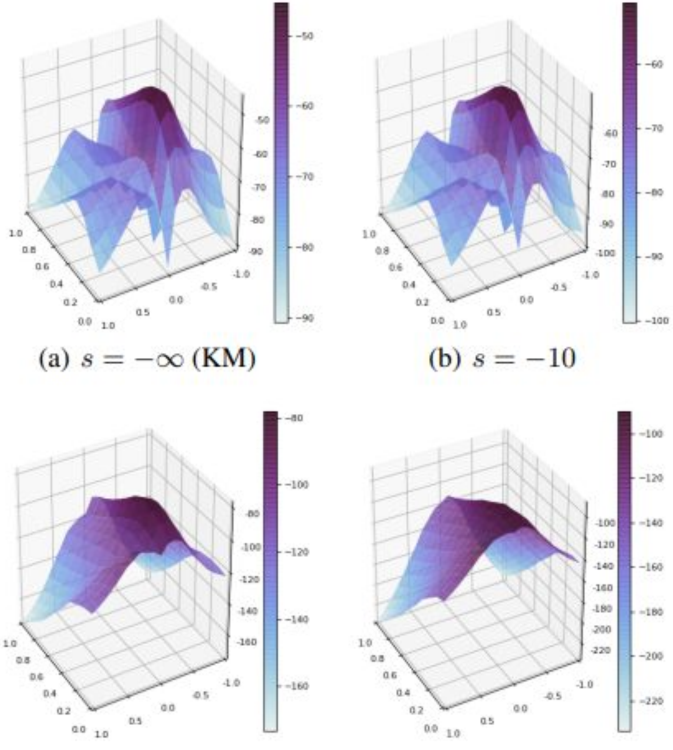
3: $\quad \omega_{m,ij} \leftarrow \left(\sum_{l=1}^{k} ||x_i - \theta_{m,l}||^{2s_m} \right)^{\frac{1}{s_m} - 1} ||x_i - \theta_{m,l}||^{2(s_m - 1)}$

4: $\quad \theta_{m+1,j} \leftarrow \left(\sum_{i=1}^{n} \omega_{m,ij} \right)^{-1} \sum_{i=1}^{n} \omega_{m,ij} x_i$

5: $\quad s_{m+1} \leftarrow \eta \cdot s_m$ ▷ Optional Annealing

6: **until** convergence

Annealing



(a) $s = -\infty$ (KM)

(b) $s = -10$

(c) $s = -1$ (KHM)

(d) $s = -0.2$

- Lloyd's (KM)
  - no annealing
- KHM
  - heuristic annealing
- Power-k Means
  - deterministic annealing
  - retains the benefits of annealing as dimension d increases

Performance Guarantee

$$f_s(\Theta) = \sum_{i=1}^{n} M_s(||x_i - \theta_1||^2, \cdots, ||x_i - \theta_k||^2)$$

- Altering the objective function at each step of an MM algorithm still guarantees the descent property and the advantages that follow from it.

Proposition: For any decreasing sequence $s_m \leqslant 1$, the iterates Θ_m produced by the MM updates (step 4) generates a decreasing sequence of objective values $f_{s_m}(\Theta_m)$ bounded below by 0.

Proof. By the power mean inequality, we have $M_{s_{m+1}}(y) \leqslant M_{s_m}(y)$ for $s_{m+1} \leqslant s_m$ where $\lim_{s \to -\infty} M_s(y) = \min(y1, ..., y_k)$. By summing over all data points, the descent property holds for chosen surrogate function $g = \omega$: $0 \leqslant f_{s_{m+1}}(\Theta_{m+1}) \leqslant g(\Theta_{m+1}) \leqslant g(\Theta_m) = f_{s_m}(\Theta_m)$.

Performance Guarantee

$$f_s(\Theta) = \sum_{i=1}^{n} M_s(||x_i - \theta_1||^2, \cdots, ||x_i - \theta_k||^2)$$

- Altering the objective function at each step of an MM algorithm still guarantees the descent property and the advantages that follow from it.
- We can freely choose a schedule for decreasing s in the initialization of the power k-means algorithm.
- Though the initial value of s does affect performance, it does not require careful tuning.

Experimental Results

Synthetic Data

- Binomial distribution
- Same amount of data points per cluster

ARI

- Adjusted Rand Index
- Accounts for chance

$$ARI = \frac{RI - ExpectedRI}{max(RI) - ExpectedRI}$$

ARI

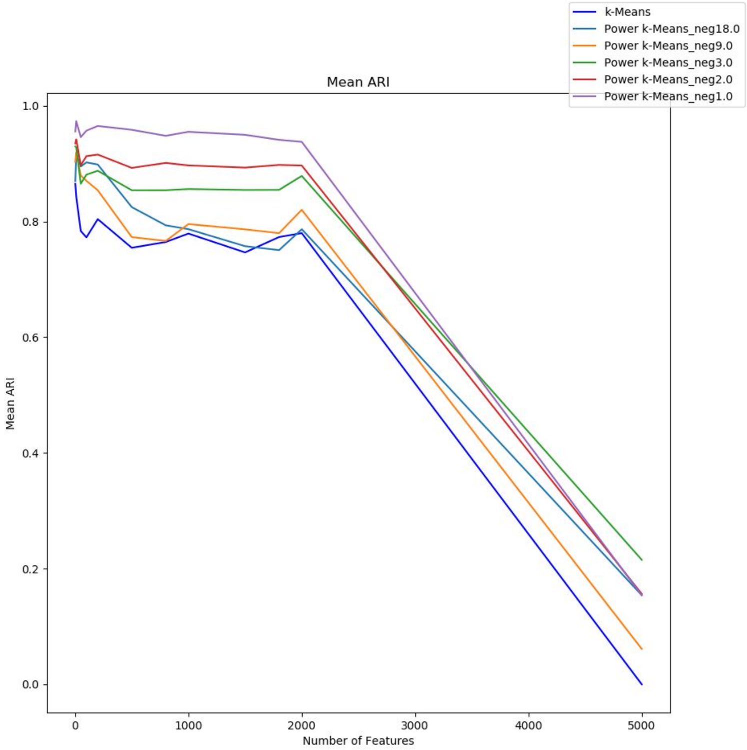
$$ARI = \frac{RI - ExpectedRI}{max(RI) - ExpectedRI}$$

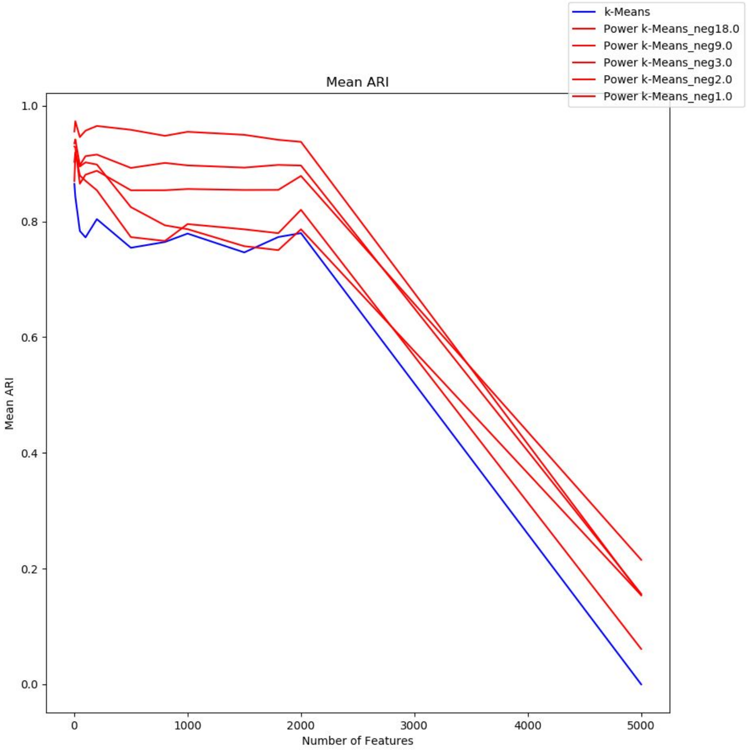
$$RI = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\frac{\sum_i \binom{a_i}{2} \times \sum_j \binom{b_j}{2}}{\binom{n}{2}}$$

True/Pred	C_1^T	$\cdots$	C_m^T	sum
C_1^P	$n_{1,1}$	$\cdots$	$n_{1,m}$	a_1
$\cdots$	$n_{i,1}$	$\cdots$	$n_{i,m}$	a_i
C_n^P	$n_{n,1}$	$\cdots$	$n_{n,m}$	a_n
sum	b_1	$\cdots$	b_m	

ARI


Mean ARI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean ARI
Number of Features
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
5000

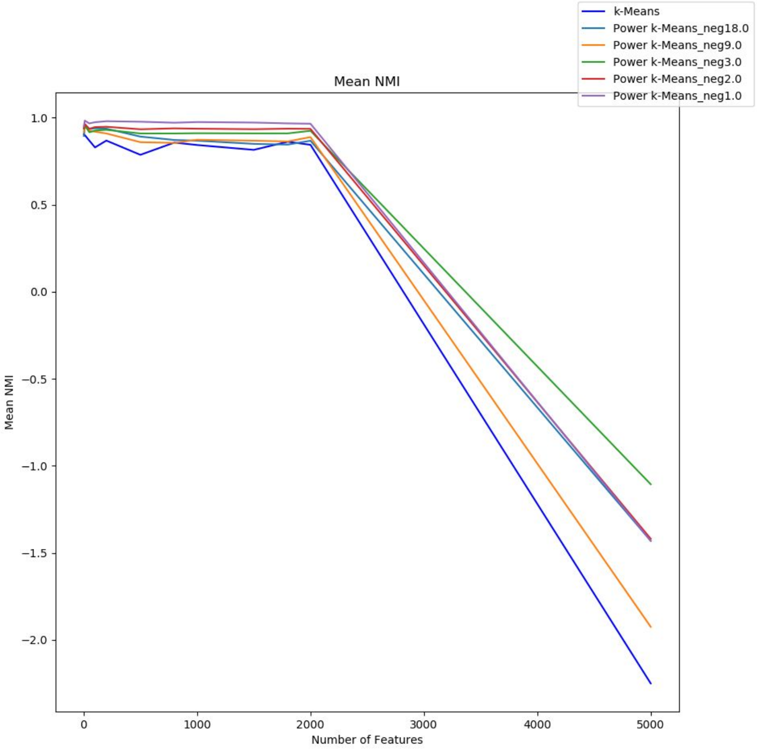

Mean ARI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean ARI
Number of Features
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
5000

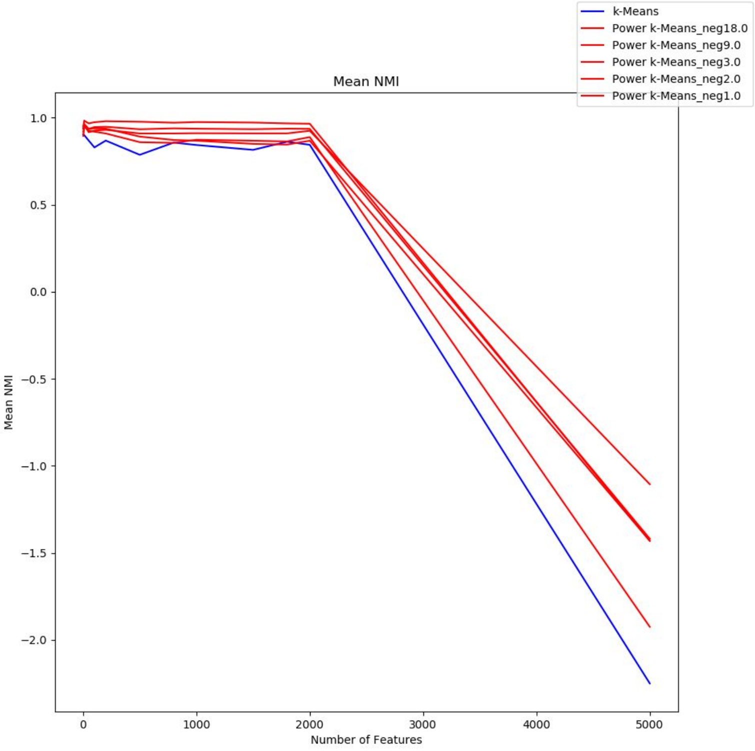
NMI

- Marginal Entropy $H(Pred) = -\sum^{k} p_{C_i} log(p_{C_i})$
- Conditional Entropy $H(Pred|True) = -\sum_{i=1}^{r}\sum_{j=1}^{k} p_{i,j} log(\frac{p_{i,j}}{p_{C_i}})$
- Mutual information $I(Pred, True) = H(Pred) - H(Pred|True)$
- Normalized Mutual Information

$$NMI(Pred, True) = \frac{I(Pred, True)}{\sqrt{H(True)H(Pred)}}$$

NMI


Mean NMI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean NMI
Number of Features
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
1000
2000
3000
4000
5000


Mean NMI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean NMI
Number of Features
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
1000
2000
3000
4000
5000

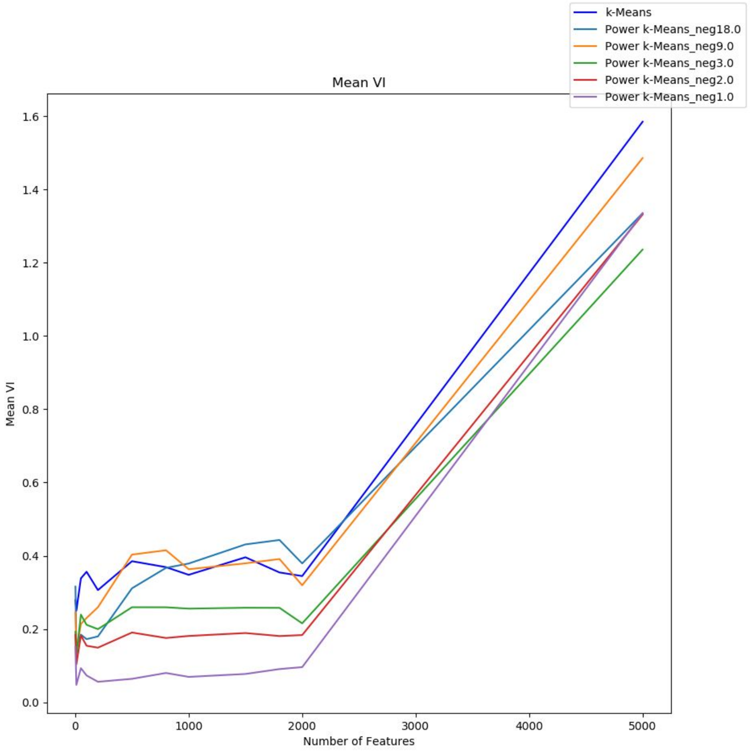
VI

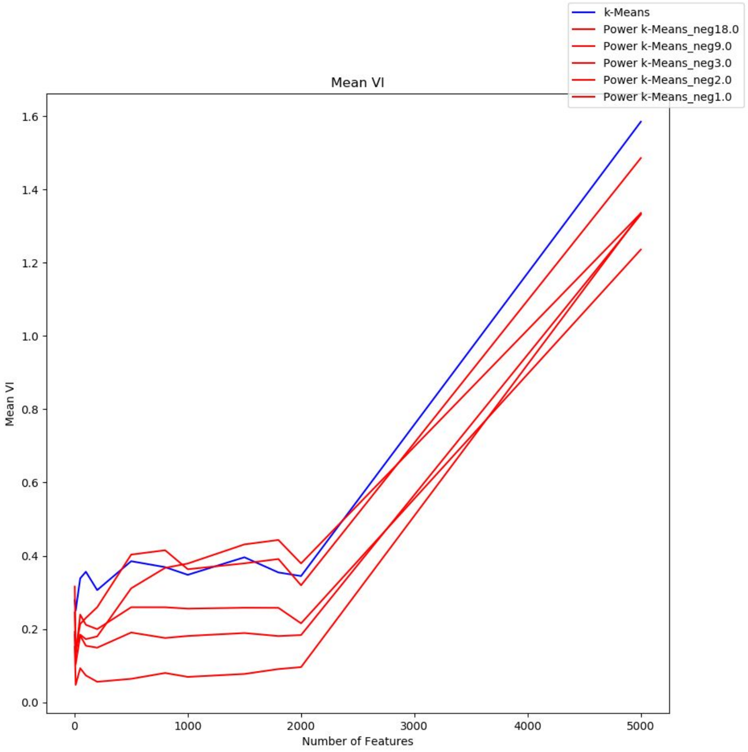
- Variation of information
- Uses conditional entropy

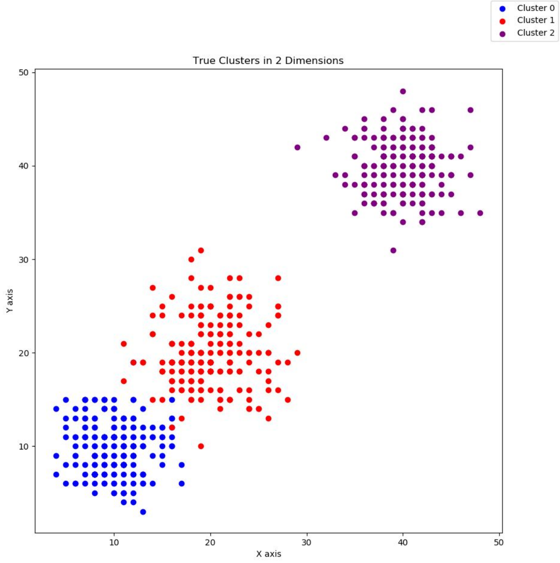
$$H(Pred|True) = -\sum_{i=1}^{r}\sum_{j=1}^{k} p_{i,j} log(\frac{p_{i,j}}{p_{C_i}})$$

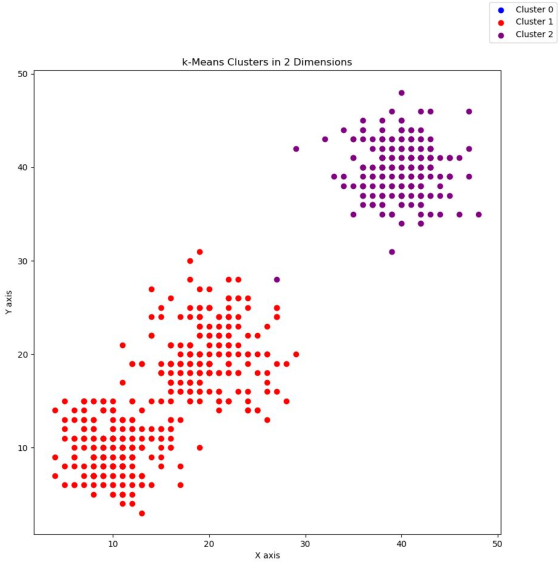
$$VI(True, Pred) = H(True|Pred) + H(Pred|True)$$

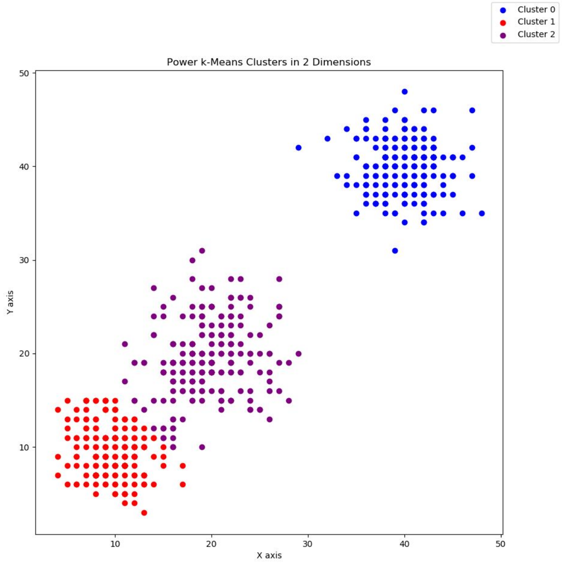
VI


Mean VI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean VI
Number of Features
0
1000
2000
3000
4000
5000
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6


Mean VI
k-Means
Power k-Means_neg18.0
Power k-Means_neg9.0
Power k-Means_neg3.0
Power k-Means_neg2.0
Power k-Means_neg1.0
Mean VI
Number of Features
0
1000
2000
3000
4000
5000
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6







3 clusters, 600 data points

Real Data

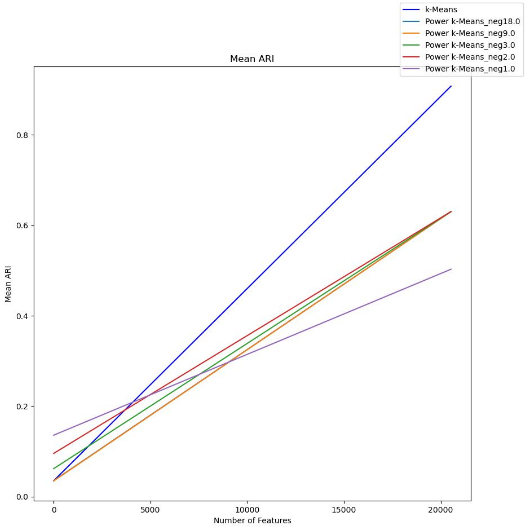
Low dimensional dataset: North Jutland 3d road network

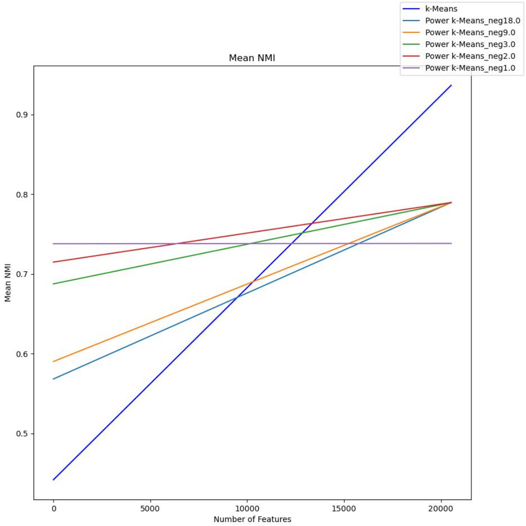
- 3 dimensions, large number of clusters (510)

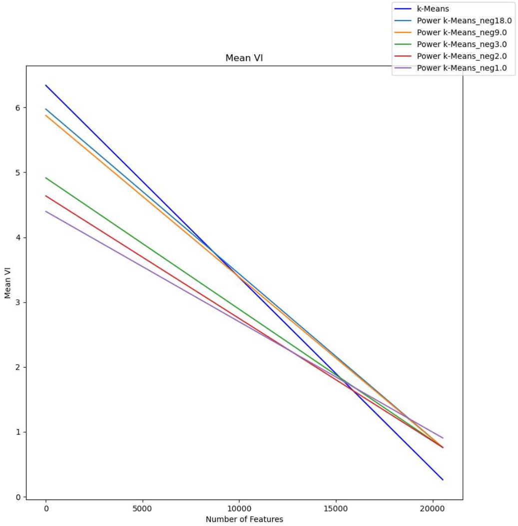
High dimensional dataset: Cancer gene expression

- >20,000 dimensions, 5 clusters

ARI, NMI, VI of the Real datasets







Sources

- Vellal, A., Chakraborty, S. & Xu, J.Q. (2022). Bregman Power k-Means for Clustering Exponential Family Data. *Proceedings of the 39th International Conference on Machine Learning*, in *Proceedings of Machine Learning Research* 162:22103-22119 Available from https://proceedings.mlr.press/v162/vellal22a.html.
- Xu, J. & Lange, K. (2019). Power k-Means Clustering. *Proceedings of the 36th International Conference on Machine Learning*, in *Proceedings of Machine Learning Research* 97:6921-6931 Available from https://proceedings.mlr.press/v97/xu19a.html.
- Zhang, B. (2001). Generalized K-Harmonic Means - Dynamic Weighting of Data in Unsupervised Learning. *SDM*.
- Zhang, B., Hsu, M., & Dayal, U. (1999). K-Harmonic Means - A Data Clustering Algorithm.