



HAL
open science

Exact enumeration of local minima for kmedoids clustering in a 2D Pareto Front

Nicolas Dupin

► **To cite this version:**

Nicolas Dupin. Exact enumeration of local minima for kmedoids clustering in a 2D Pareto Front. HUGO 2022 - XV. Workshop on Global Optimization, Sep 2022, Szeged, Hungary. hal-03695180

HAL Id: hal-03695180

<https://hal.science/hal-03695180>

Submitted on 14 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Exact enumeration of local minima for k-medoids clustering in a 2D Pareto Front

Nicolas Dupin,¹

¹*Laboratoire Interdisciplinaire des Sciences du Numérique, Université Paris-Saclay, Gif-sur-Yvette, France, nicolas.dupin@universite-paris-saclay.fr*

Abstract K-medoids clustering is solvable by dynamic programming in $O(N^3)$ time for a 2D Pareto Front (PF). A key element is a interval clustering optimality. This paper proves this property holds also for local minima for k-medoids. It allows to enumerate the local minima of k-medoids with the same complexity than the computation of global optima for k=2 ou k=3. A pseudo-polynomial enumeration scheme is designed, for small values of k . This allows to understand results obtained by local search approaches in a 2D PF.

Keywords:

Local minimums, k-medoids, clustering, Pareto Front,

1. Problem statement, notation

Definition 1. A 2D PF is defined as a set E of N points in \mathbb{R}^2 , indexed with $E = \{(x_k, y_k)\}_{k \in \llbracket 1, N \rrbracket}$ such that $k \in \llbracket 1, N \rrbracket \mapsto x_k$ is increasing and $k \in \llbracket 1, N \rrbracket \mapsto y_k$ is decreasing.

Let $\Pi_K(E)$ the set of partitions of E in K subsets. K -medoids clustering is a combinatorial optimization problem indexed by $\Pi_K(E)$. It minimizes the sum for all the K clusters of the dissimilarity measure minimizing the sum of the squared distances from one chosen point of P , the *medoid*, to the other points of P :

$$\min_{\pi \in \Pi_K(E)} \sum_{P \in \pi} \min_{c \in P} \sum_{x \in P} \|x - c\|^2 \quad (1)$$

Combinatorial optimization problem (1) defines global optimums. For this paper, local minima are defined using neighborhoods of the PAM (Partitioning Around Medoids) heuristic for k-medoids, adaptation of seminal Lloyd's heuristic for the k-means problem [6].

Definition 2 (Local minima of K -medoids). *For all $K \in \mathbb{N}^*$, local minima of K -medoids (for PAM) are characterized by the encoding of partitioning subsets P_1, \dots, P_K and their respective medoids c_1, \dots, c_K , with the property:*

$$\forall k \in \llbracket 1; K \rrbracket, \forall p \in P_k, \forall k' \neq k, \quad \|p - c_k\| \leq \|p - c_{k'}\| \quad (2)$$

2. Local optimality and interval clustering

A global minimum of optimization problem (1) can be computed in a 2D PF by dynamic programming in $O(N^3)$ time [1]. A key element is a necessary condition: interval optimality as in 1D clustering problems [3, 5]. The interval clustering property holds also for local minima for k-medoids:

Theorem 3. *We suppose that points (x_i) are sorted as in Definition 1. Each local minimum of K -Medoids in a 2dPF is only composed of clusters $\mathcal{C}_{i,i'} = \{x_j\}_{j \in \llbracket i, i' \rrbracket} = \{x \in E \mid \exists j \in \llbracket i, i' \rrbracket, x = x_j\}$. As a consequence, there is at most $\binom{N}{K}$ local optima for K -Medoids in a 2d PF of size N .*

Note that this optimality property is specific for k-medoids clustering. For k-center problems, global minima exist with nested clusters [2]. For the k-means problem, we can give a counter example of local minima which do not fulfill the interval clustering property.

3. Enumeration algorithms of local optima

Theorem 3 allows to enumerate the $\binom{N}{K}$ possible local minimums. Lemmas help to decrease the complexity of such operation. Enumeration schemes require to prove that a candidate solution is a local minimum fulfilling 2, which can be checked easily using properties related to interval clustering:

Lemma 4. *Interval clustering $\mathcal{C}_{1,f_1}, \mathcal{C}_{f_1+1,f_2}, \dots, \mathcal{C}_{f_{K-2}+1,f_{K-1}}, \mathcal{C}_{f_{K-1}+1,N}$ with medoids c_1, \dots, c_K is a local minimum if and only if both properties are fulfilled:*

$$\forall k \in \llbracket 1; K-1 \rrbracket, \quad \|z_{f_k} - c_k\| \leq \|z_{f_k} - c_{k+1}\| \quad (3)$$

$$\forall k \in \llbracket 1; K-1 \rrbracket, \quad \|z_{1+f_k} - c_{k+1}\| \leq \|z_{1+f_k} - c_k\| \quad (4)$$

For enumeration schemes, it requires to compute the medoids $c_{j,j'}$ for each cluster $\mathcal{C}_{j,j'}$ with $j \leq j'$. Lemmas 5 and 6 allows to do it efficiently for our enumeration schemes:

Lemma 5. *Computing medoids (and costs) of clusters $\mathcal{C}_{j',j}$ for all $j' \in \llbracket 1; j \rrbracket$ for a given $j \in \llbracket 1; N \rrbracket$ runs in $O(j^2)$ time and uses $O(j)$ memory space.*

Lemma 6. *Computing medoids (and costs) of clusters $\mathcal{C}_{j,j'}$ for all $j' \in \llbracket j; N \rrbracket$ for a given $j \in \llbracket 1; N \rrbracket$ runs in $O((N - j)^2)$ time and uses $O(N - j)$ memory space.*

For $K = 2$ and $K = 3$, straightforward enumeration of local minima has the same complexity than computing only a global optimum:

Proposition 7. *Enumerating the $O(N)$ local minima of 2-Medoids in a 2d PF of size N runs in $O(N^2)$ time using $O(N)$ additional memory space.*

Proposition 8. *Enumerating the $O(N^2)$ local minima of 3-Medoids in a 2d PF of size N runs in $O(N^3)$ time using $O(N)$ additional memory space.*

For a general algorithm, we use $O(N^2)$ additional memory space, computing first the medoids $c_{j,j'}$ of each interval cluster $\mathcal{C}_{j,j'}$ in $O(N^3)$ time. Then, a backtracking algorithm enumerates only partial feasible solutions for Lemma 4: if $\mathcal{C}_{1,f_1}, \mathcal{C}_{f_1+1,f_2}, \dots, \mathcal{C}_{f_{k-2}+1,f_{k-1}}, \mathcal{C}_{f_{k-1}+1,f_k}$ with medoids c_1, \dots, c_k partially fulfills (3) and (4), the next point $f_{k+1} > f_k$ induces a new medoid $c_{f_{k-1}+1,f_k}$ such that $\|z_{f_k} - c_{f_{k-1}+1,f_k}\| \leq \|z_{f_k} - c_{f_k+1,f_{k+1}}\|$ and $\|z_{1+f_k} - c_{f_k+1,f_{k+1}}\| \leq \|z_{1+f_k} - c_{f_{k-1}+1,f_k}\|$. The following recursive algorithm calls `ENUMLOCMIN(0, [], [], 0)` to print all the local minimums. We use functional programming notations, as in OCaml, $x :: l$ pop first element x in top of list l , $l' = x :: l = (\text{hd } l') :: (\text{tl } l')$, $[]$ is empty list.

4. Conclusions and perspectives

If k -medoids clustering in a 2D PF is solvable by dynamic programming in $O(N^3)$ time with an interval clustering property, this paper proves that this property holds also for local minima for k -medoids. It allows to enumerate the local minima of k -medoids with the same complexity than the computation of global optima for $k = 2$ ou $k = 3$. A pseudo-polynomial enumeration scheme is designed, for small values of k . Perspectives are to understand results obtained by local search approaches in a 2D PF and why local minima of a poor quality can be found by PAM local search in a 2D PF [4]. This problem and these properties are also an interesting for landscape analysis approaches.

Algorithm : Exhaustive search of Local Minima of K -Medoids

Input: N points of a 2D PF, $E = \{z_1, \dots, z_N\}$, $K \leq N$

The medoids $c_{j,j'}$ of each interval cluster $C_{j,j'}$ are computed in pre-processing

```

ENUMLOCMIN(last, ptList, medList, k)
  if k ==  $K - 1$ 
    c := hd medList
    if  $\|z_{last} - c\| \leq \|z_{last} - c_{last+1,N}\| \ \&\& \ \|z_{1+last} - c_{last+1,N}\| \leq \|z_{1+last} - c\|$ 
      then print(ptList)
    end if
  else
    for next =  $1 + last$  to  $N - K + k + 1$ 
      c' :=  $c_{1+last,next}$ 
      if k == 0 then ENUMLOCMIN (next, next::[], c'::[], 1)
      else
        c := hd medList
        if  $\|z_{last} - c\| \leq \|z_{last} - c'\| \ \&\& \ \|z_{1+last} - c'\| \leq \|z_{1+last} - c\|$ 
          then ENUMLOCMIN (next, next::ptList, c'::medList,  $1 + k$ )
        end if
      end if
    end for
  end if
end

```

References

- [1] N. Dupin, F. Nielsen, and E. Talbi. k-medoids clustering is solvable in polynomial time for a 2d Pareto front. In *World Congress on Global Optimization*, pages 790–799. Springer, 2019.
- [2] N. Dupin, F. Nielsen, and E. Talbi. Unified polynomial dynamic programming algorithms for p-center variants in a 2d pareto front. *Mathematics*, 9(4):453, 2021.
- [3] A. Grønlund et al. Fast exact k-means, k-medians and Bregman divergence clustering in 1d. *arXiv:1701.07204*, 2017.
- [4] J. Huang, Z. Chen, and N. Dupin. Comparing local search initialization for k-means and k-medoids clustering in a planar Pareto Front, a computational study. In *International Conference on Optimization and Learning*. Springer, 2021.
- [5] F. Nielsen and R. Nock. Optimal interval clustering: Application to Bregman clustering and statistical mixture learning. *IEEE Signal Processing Letters*, 21(10):1289–1292, 2014.
- [6] E. Schubert and P. Rousseeuw. Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms. *arXiv:1810.05691*, 2018.