Performance Analysis for a Heuristic p-Medians Algorithm

Gilsiley Henrique Dar[a]**, Gustavo Valentim**[a]
[a]Universidade Federal do Paran, Programa de Ps-Graduao em Mtodos Numricos, Centro Politcnico, Curitiba, Paran, Brasil

Abstract. Locations and clustering problems represent an important area to logistic and unsupervisoned learning. Both problems was solved by the p-Median problem. The exact formulation use mix integer programming and its solution growth exponentialy. Because this heuristics are developed. Analyses its characteristics became an important aspect to the area. Here are developed an heuristics and analysed the impact of size of the problem and number of medians over solution quality. For this is generated random samples over a set of parameters combination. This show that the propost heurist has a apresents a good quality solution, a linear time growth and a little degradation with over the size parameter. Some empiricus results based on simulations are discovered and main result is the little solution quality degradation over size.

Keywords: p-Medians, unsupervisioned learning, clustering, locations problems, heuristics.

* ghdaru@gmail.com

1 Introduction

The problem in which it is necessary to group entities or objects is called a cluster. Many types of location problems fall into this category. This problem is now also known as unsupervised learning.

The definition adopted here receives a set of points P and a distance matrix between these points M, select p points from P and call them p-median. For each different x point of p associate a p-median point that minimizes the total sum of x to this chosen p-median using the value of matrix M. The problem can be solved exactly using mixed linear programming. Mathematical formulas are

$$min \sum_{i=1}^{n} d_{ij}x_{ij} \tag{1}$$

s.a.

$$\sum_{j=1}^{n} y_j = p, \forall j \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \forall i \tag{3}$$

$$x_{ij} \leq y_j, \forall i, j \tag{4}$$

$$x_{ij} \in \{0, 1\}, \forall i, j \tag{5}$$

$$y_j \in \{0, 1\}, \forall j \tag{6}$$

where

d_{ij}: distance between points i and j, obtained from M y_j and x_{ij} are decision variables and means

$$y_j = \begin{cases} 1 & \text{,if point j is a p-median} \\ 0 & \text{,other case} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{,if point i is alocated to p-median j} \\ 0 & \text{,other case} \end{cases}$$

Equation (1) show the main objective that is minimize the distances between the points and their p-medians associated. How x_{ij} is zero or one, the distance is considered only when x_{ij} is equal to one.

Equation (2) warrant that only p points is selected like a median.

Equation 3 ensure that each point is associated with one and only one median point.

Equation 4 guarantee that the above association is with p-medians points. For instance, suppose that the point k is a selected like a median, Hence y_k is one. This restriction is relaxed and any point can be associated to y_k. However, if k isn't a median then this restriction force all x_{ik} be equals to zero.

Matrix M has n_2 entries, and the number of binary variables increase quadractily. This implies to solve using mathematical formulation for big problems in a razonable time is prohibitive. To solve this problem in a more fast form is necessary use heuristics. In this work, is studied some heuristics and compared their performance against optimal formulation. Also is compared the time necessary to solve some randomic instance for some pre-specified sizes.

2 Methodology

First of all, was developed an heuristic to solve the problem. This heuristic is showed in the code 01.

```
def heuristic(p):
    set iteration control variables
    initialize p-Medians with p first elements or randomly
    while improvement >= tol and iterations<max_iterations:
        for each point set the nearest median
        calculate improvement
        calculate centroid
        for each point:
            calculate distance until centroid
        for each cluster:
            set the centroid nearest point like new median
        update control variables
```

The main idea is generate a initial solution, attach for each point the nearest median, calculate the centroid for each cluster and set the nearest point to this centroid a new cluster median. Repeat these steps while there is improvement or for a pre specified number of iterations.

To availate the heuristic performance, was calculated time execution and how distance from optimal solution in percentage is the solution. For do this, was created some scenaries and simulated thirty times. For each scenary was considered two parameters. The first was the size problem which represents how many points has. The second was the number of clusters or medians to consider. For the first parameter, quantity of points, was used the values 50 until 400 by 50. This number was motivated by the total simulation time. The number of medians in set $\{2,3,4,5,10\}$ was choosen arbitrarily.

The language used was python 3.7. And the average time to execute a loop from 1 to one billion was fourty seconds(this is to permit others researchs compare this results in others languages or machines). The code is disponible in https://github.com/GHDaru/pMedianas

3 Results and Discussions

3.1 General Results

The tabel 1 shows for each size, how many runs was executed, the heuristic total time to execute all 150 runs and the time duration to execute optimal solution using linear programming for all 150 runs (OS Time). The last field is the heuristic time divided by optimal solution. The total time simulation, optimal plus heuristic was 1 days,2 hours, 18 minutes and 16 seconds. Note that the heuristic time for big problems became very little, last column on table. Still from table 1 it is possible note the linear growth for heuristic time and exponencial growth for optimal solution. This behaviour is showed in 1.

Table 1 Simulation summary by size

Size	# Samples	Time(s)	OS Time(s)	$\frac{Time}{OSTime}(\%)$
50	150	74	121	61.7
100	150	132	578	22.9
150	150	165	1953	8.5
200	150	179	2931	6.1
250	150	271	11200	2.4
300	150	363	28581	1.3
350	150	421	41519	1.0
Total	1050	1606	86883	1.8

The figure 1 shows in the left side (a) a scatter plot and regression line with size problem in x-axis and heuristic time in y-axis. The relationship is linear and from equation of regression is can be seen that for one increment by one in size the time increase by 1.15 or 15%. The pearson correlation is almos 98% that show an high correlation. However from right side (b) again is shown a scatter plot and a regression, where now y is log. This relation is exponential showing that time to solve optimal solution growth in a exponential form with the increment in size problem.

3.2 Quality of Heuristic Solution

To availate the quality of solution was generated many problems size shown in figure 2. It's possible idetificate that to size less than 150 solution has between 4 and 7%. For values above this value solution slowly decrease and variability decrease too. Showing that for size higher than 350 solution stabilizate between 3%.

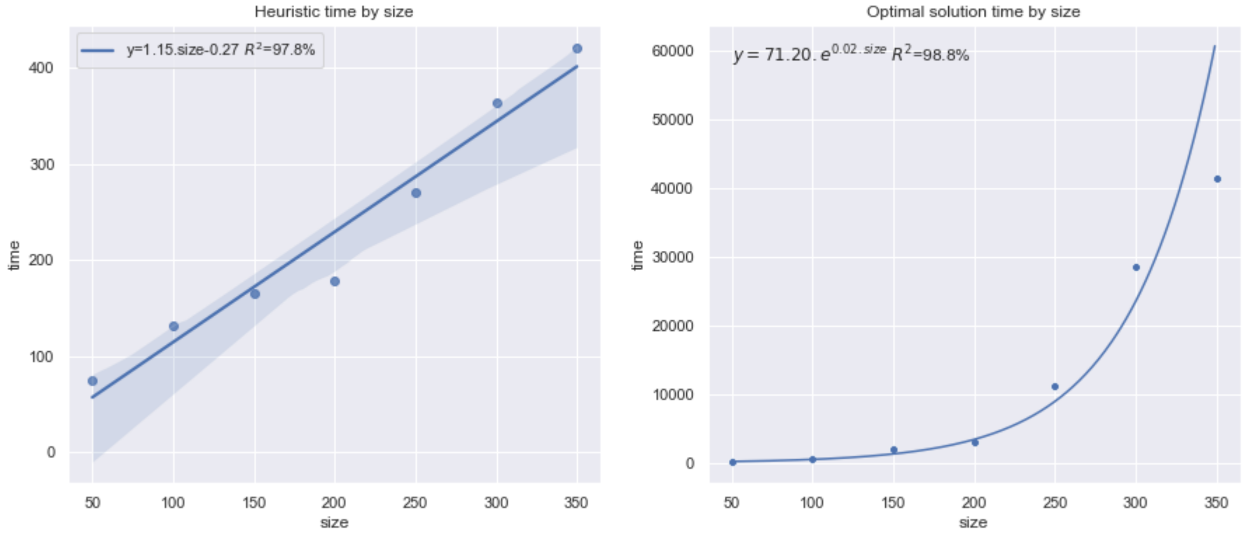

Fig 1 Heuristic time and optimal time to solve a problem of specific size

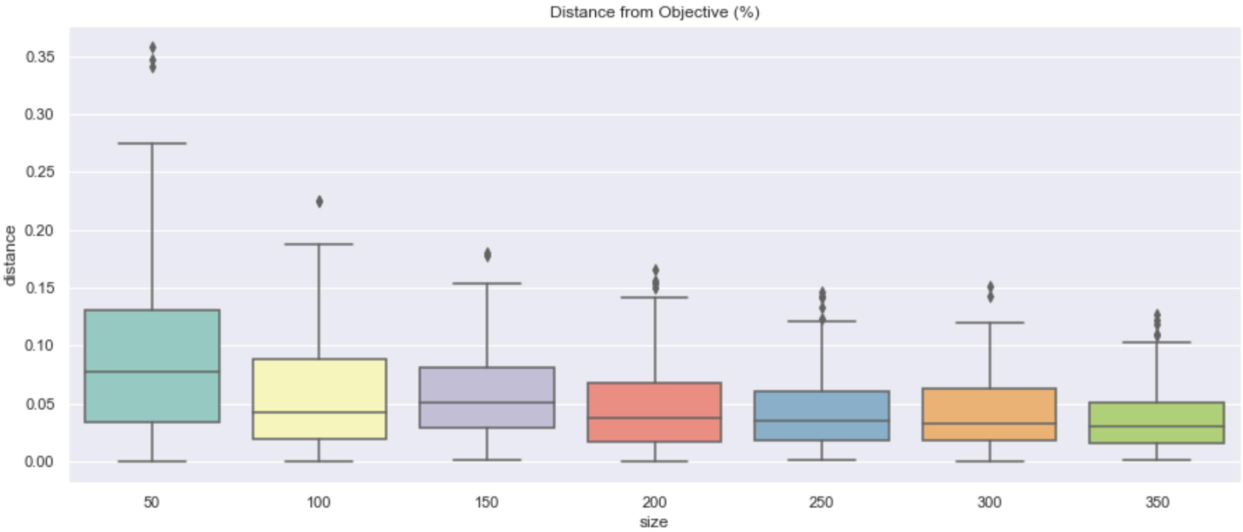

Fig 2 Heuristic time by size and heuristic solution distance to optimal (%) by size.

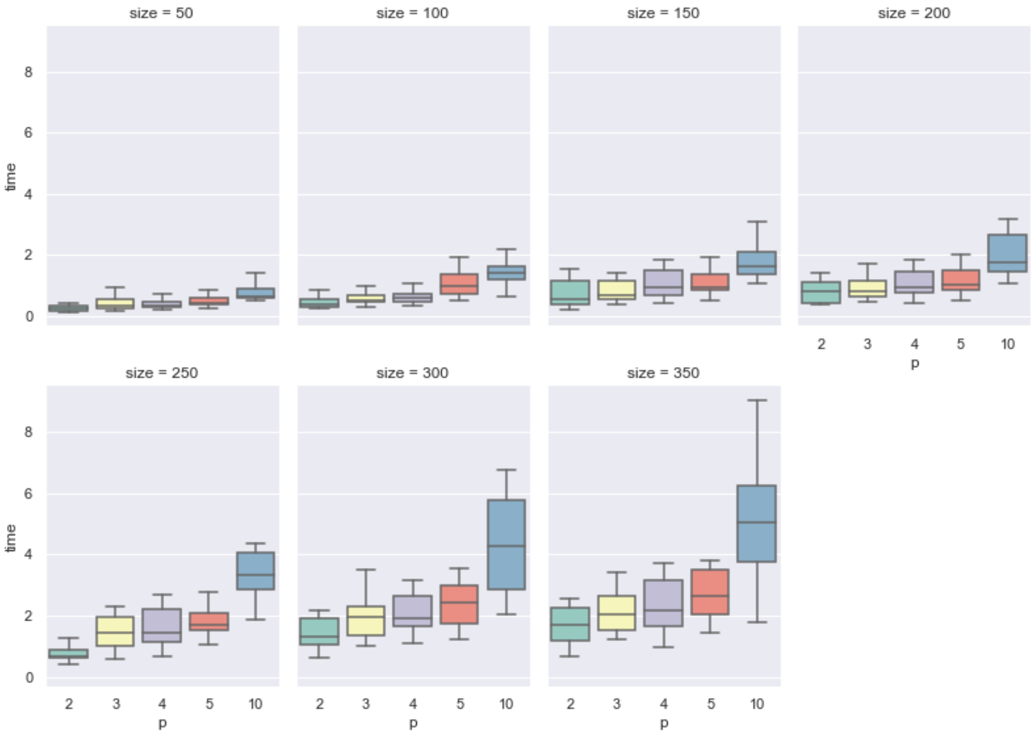

Fig 3 Heuristic time by size and number of medians.

3.3 Number of medians influence on time

The number of medians has influence over heuristic time solution. This is shown in figure 3, where for all sizes the time from biggest number of medians is higher than all before. This is explained directly from algorith, where there is a loop for each median.

3.4 Number of Medians Influence on Distance Solution from Optimal

Can be seen that for each size, the number of medians also increase the heuristic distance from optimal solution. This is a empirical results demonstrate by simultation. Figure 4 shows this result. While in the initials sizes 50 up to 200 the growth for number of medians increase more intensily the heuristic solution distance from optimal solution, this fact tend to became less intensive when size is higher than 200.

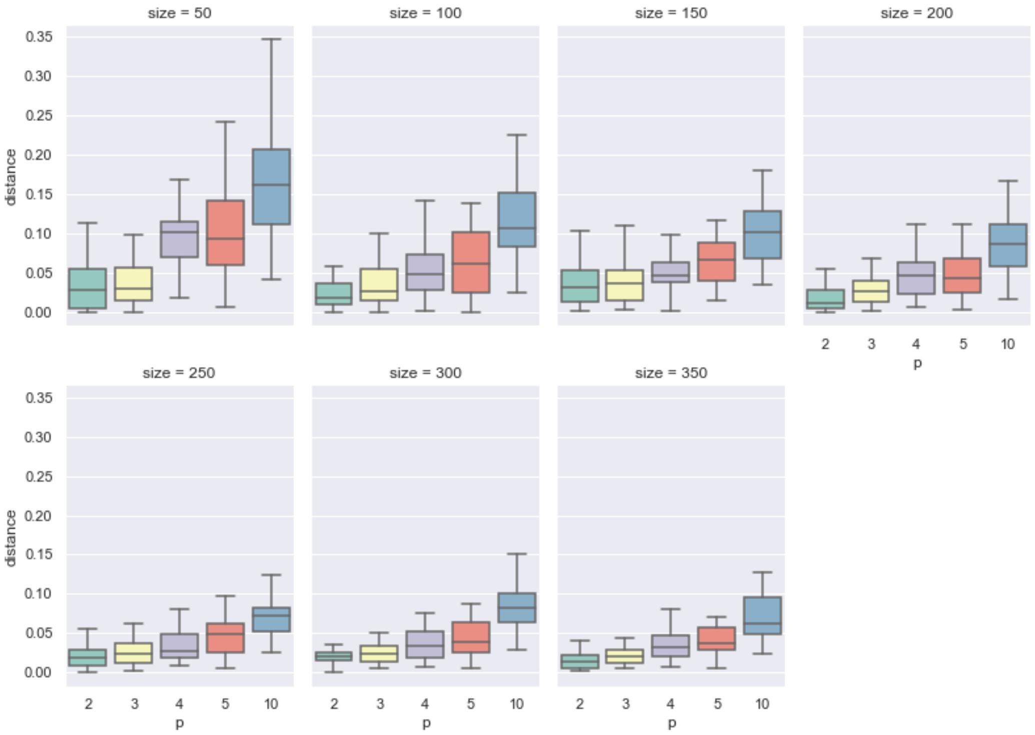

Fig 4 Heuristic solution distance from optimal by size and number of medians.

4 Conclusion

This paper shows the improvement in percentage with growth of size of problem, decreasing variability and that the time resolution increase linearly. Shows too that the variability decrease with size growth. This accomplish the balance between time and quality solution.

4.1 References

References