# O problema das p-medianas

O problema das *p*-medianas constitui-se em:

* dado um conjunto de pontos e a matriz de distância entre eles
* determinar *p* pontos como sendo medianas
* alocar cada ponto a uma mediana
* minimizando a soma das distâncias entre cada ponto e a mediana à qual ele é alocado

# Modelo matemático exato

Parâmetros do modelo:

: distância entre os pontos e

Variáveis de decisão:

Modelo de Programação Linear Inteira

|  |  |
| --- | --- |
|  | (1) |

s.a

|  |  |
| --- | --- |
|  | (2) |

|  |  |
| --- | --- |
|  | (3) |

|  |  |
| --- | --- |
|  | (4) |

|  |  |
| --- | --- |
|  | (5) |

|  |  |
| --- | --- |
|  | (6) |

A função objetivo (1) minimiza a soma das distâncias. A restrição (2) estabelece que (3) pontos serão utilizados como medianas. O conjunto de restrições (3) impõe que cada ponto é alocado à exatamente uma mediana. O conjunto de restrições (4) garante que o ponto *i* só pode ser alocado à mediana *j* caso o ponto *j* seja escolhido como mediana. Os conjuntos de restrições (5) e (6) define que as variáveis de decisão são binárias.

# Método heurístico de construção de solução inicial

Passo 1. Escolha *p* pontos como medianas iniciais e aloque os mesmos

Passo 2. Escolha um ponto ainda não alocado

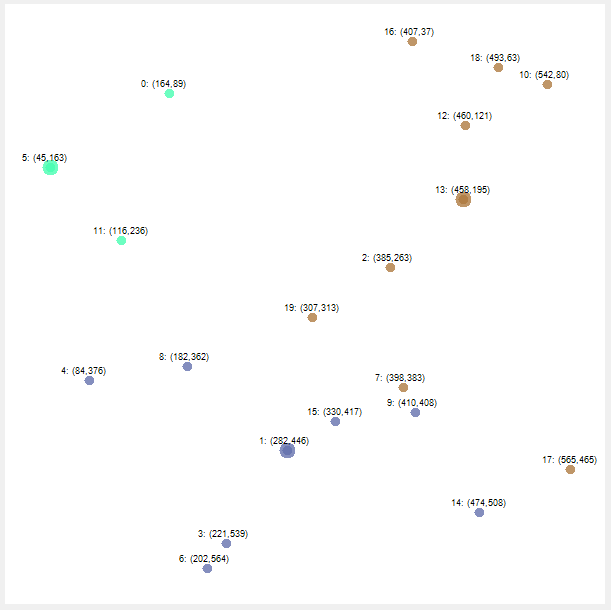
Passo 3. Encontre a mediana mais perto do ponto escolhido e realize a alocação do ponto à ela

Passo 4. Se ainda existe ponto não alocado, retorne ao passo 2. Caso contrário, fim.

Pergunta: Sobre o método descrito, quais as vantagens e desvantagens dele?

Pergunta: O que você alteraria no método para melhorar a qualidade da solução encontrada?

# Visualização de solução



# Representação de uma solução em estrutura de dados

class Mediana

{

public double Distancia;

public int ItemMediana;

public Color Cor;

public List<int> ItensAlocados;

}

class Item

{

public int X;

public int Y;

public int MedianaAlocada;

}

class ProblemaPmedianas

{

public int NumeroItens;

public int NumeroMedianas;

public double DistanciaTotal;

public Item[] Itens;

public Mediana[] Medianas;

public double[,] MatrizDistancias;

Métodos...

}

## Exemplo de representação de uma solução

Exercício: com base no código disponível em github.com/gustavovalentim/pMedianas, complete os quadros a seguir, os quais representam parte da estrutura de dados para representação de solução.

|  |  |  |  |
| --- | --- | --- | --- |
| Mediana | Distancia | ItemMediana | ItensAlocados |
| 0 |  | 5 | 0, 5, 11 |
| 1 |  | 1 | 1, 15, 3, 8, 9, 6, 14, 4 |
| 2 |  | 13 | 2, 19, 13, 7, 12, 16, 18, 10, 17 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Item | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Y |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Mediana Alocada |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Vizinhanças de solução

* Os agrupamentos são mantidos, mas o ponto que é a mediana é trocado
* Um ponto deixa de ser alocado à mediana atual e é alocado à uma nova mediana (shift)
* Troca de pontos entre medianas (exchange)

# Variable Neighborhood Descent

Exercício: pesquisar sobre o que é *Variable Neighborhood Descent* e verificar se em algum lugar do código disponibilizado ela é utilizada.