

# Problema do Caixeiro Viajante em Julia

...

Iann Takami Singo  
Júlio César Fagundes  
Luiza Gabriela da Silva  
Victor Gabriel Zerger

# Sobre o Problema



- Por que do tema;
  - O que é;
- Condicionais;



# Condicionais



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

$$\sum_{j=1}^n x_{ij} = 1, i = 1, \dots, n; i \neq j \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1, j = 1, \dots, n; j \neq i \quad (3)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1, S \subset \{1, \dots, n\}, 2 \leq |S| \leq n - 2 \quad (4)$$

$$x_{ij} = 0 \text{ ou } 1; i, j \in \{1, \dots, n\} \quad (5)$$

# Código 1

- Como conseguir criar um arquivo do tipo .csv;
- Como utilizar este tipo de arquivo no programa;
  - Pacotes necessários: CSV e DataFrames.

```
1  using CSV
2  using DataFrames
3  df = DataFrame(Name = ["Cidade 1", "Cidade 2", "Cidade 3", "Cidade 4", "Cidade 5"],
4  |               |               |               |               |
5  |               |               |               |               |
6  |               |               |               |               |
7  |               |               |               |               |
8  |               |               |               |               |
9  |               |               |               |               |
10 |               |               |               |               |
11 |               |               |               |               |
12 |               |               |               |               |
13 |               |               |               |               |
14 |               |               |               |               |
15 |               |               |               |               |
16 |               |               |               |               |
17 |               |               |               |               |
18 |               |               |               |               |
19 |               |               |               |               |
20 |               |               |               |               |
21 |               |               |               |               |
22 |               |               |               |               |
23 |               |               |               |               |
24 |               |               |               |               |
25 |               |               |               |               |
26 |               |               |               |               |
27 |               |               |               |               |
28 |               |               |               |               |
29 |               |               |               |               |
30 |               |               |               |               |
31 |               |               |               |               |
32 |               |               |               |               |
33 |               |               |               |               |
34 |               |               |               |               |
35 |               |               |               |               |
36 |               |               |               |               |
37 |               |               |               |               |
38 |               |               |               |               |
39 |               |               |               |               |
40 |               |               |               |               |
41 |               |               |               |               |
42 |               |               |               |               |
43 |               |               |               |               |
44 |               |               |               |               |
45 |               |               |               |               |
46 |               |               |               |               |
47 |               |               |               |               |
48 |               |               |               |               |
49 |               |               |               |               |
50 |               |               |               |               |
51 |               |               |               |               |
52 |               |               |               |               |
53 |               |               |               |               |
54 |               |               |               |               |
55 |               |               |               |               |
56 |               |               |               |               |
57 |               |               |               |               |
58 |               |               |               |               |
59 |               |               |               |               |
60 |               |               |               |               |
61 |               |               |               |               |
62 |               |               |               |               |
63 |               |               |               |               |
64 |               |               |               |               |
65 |               |               |               |               |
66 |               |               |               |               |
67 |               |               |               |               |
68 |               |               |               |               |
69 |               |               |               |               |
70 |               |               |               |               |
71 |               |               |               |               |
72 |               |               |               |               |
73 |               |               |               |               |
74 |               |               |               |               |
75 |               |               |               |               |
76 |               |               |               |               |
77 |               |               |               |               |
78 |               |               |               |               |
79 |               |               |               |               |
80 |               |               |               |               |
81 |               |               |               |               |
82 |               |               |               |               |
83 |               |               |               |               |
84 |               |               |               |               |
85 |               |               |               |               |
86 |               |               |               |               |
87 |               |               |               |               |
88 |               |               |               |               |
89 |               |               |               |               |
90 |               |               |               |               |
91 |               |               |               |               |
92 |               |               |               |               |
93 |               |               |               |               |
94 |               |               |               |               |
95 |               |               |               |               |
96 |               |               |               |               |
97 |               |               |               |               |
98 |               |               |               |               |
99 |               |               |               |               |
100 |               |               |               |               |
```

## Código 2

- Fórmula da distância não Euclidiana + regressão;
  - Pacotes específicos: Clp; JuMP e GLPK;
  - Otimização.

```

function PCV(p,gasolina,cpl,io::IO = stdout) ##p::os valores da tabela, gasolina::preço, cpl::quanto o caminhão faz por Litro
    m=length(p)
    s = rand(m,m)
    d = 0.0
    c = 0.0
    for i = 1:m
        for j= 1:m
            if ((cos((90-p[i][2])/180pi))cos((90-p[j][2])/180pi) + sin((90-p[i][2])/180pi)sin((90-p[j][2])pi/180)cos((p[i][3]-p[j][3])pi/180)) > 1
                d = sqrt((p[i][2]-p[j][2])^2+(p[i][3]-p[j][3])^2)
                c = 129.3799d -34.8839
                s[i,j] = c
            else
                x = 6371acos((cos((90-p[i][2])/180pi))cos((90-p[j][2])/180pi) + sin((90-p[i][2])/180pi)sin((90-p[j][2])pi/180)cos((p[i][3]-p[j][3])pi/180))1.15
                # formula de Haversine, utiliza as leis do cossenos considerando o modelo a curvatura da terra onde o raio é 6371km
                # poderíamos também utilizar a formula m(x)=129.3799x -34.8839 que foi adquirida atraves de uma regressão linear
                # onde meu x erra a distancia entre pontos (no caso distancia entre latitudes e longitudes) e o meu y era a distancia em km
                s[i,j] = x
            end
        end
    end
end

```

# Acertos e Dificuldades



- Colocar os custos;
- Calcular a distância;
- Printar a ordem certa;
- Todos fizeram sua parte.



- Falta de comunicação;
- Como utilizar os pacotes e seus comandos;
- Não conseguimos plotar o grafo das cidades;
- Problemas no Código 1.