

experimento 6: dracarys - token limit - 19a5a42c2

USER

A base de dados anexada no arquivo dados_50_50.csv possui quatro variáveis: IDENTIFICADOR, CARGO, PERCENT_PRI E DISTANCE_KM. A primeira variável é o código que identifica o funcionário, a segunda apresenta os códigos dos cargos, a terceira é um índice que varia de 0 a 1, sendo o 0 o melhor e, por fim, a última variável é a distância em KM entre o funcionário e o cargo que ele poderá assumir, ou seja, quanto maior a distância, pior. Gostaria que me propusesse uma metodologia para solucionar meu problema de alocação de forma a minimizar o PERCENT_PRI ou a DISTANCE_KM ou os dois juntos. Determine e execute o código para solucionar o problema e indique o custo total minimizado.

BOT

Step-by-step Plan

- Define the optimization problem and the objective function to minimize PERCENT_PRI and DISTANCE_KM.
- Choose an appropriate algorithm or method to solve the optimization problem, such as linear programming or a heuristic approach.
- Implement the chosen method using a suitable programming language or tool, ensuring that the data from the CSV file is correctly read and processed.
- Execute the code, calculate the minimized cost, and present the results, including the total minimized cost and the allocation of employees to positions.

Certainly! Let's break down the problem and solve it step by step.

Step 1: Define the Optimization Problem

We need to minimize the following objectives:

1. **PERCENT_PRI**: An index that varies from 0 to 1, where 0 is the best.
2. **DISTANCE_KM**: The distance in kilometers between the employee and the position they will assume, where a smaller distance is better.

We can formulate this as a multi-objective optimization problem. However, for simplicity, we can combine these two objectives into a single objective function. One common approach is to use a weighted sum of the two objectives.

Step 2: Choose an Appropriate Algorithm

For this problem, we can use a linear programming approach. Linear programming is well-suited for problems where we need to minimize a linear objective function subject to linear constraints. We will use the `pulp` library in Python to solve this problem.

Step 3: Implement the Chosen Method

Let's implement the solution in Python. We will:

1. Read the data from the CSV file.
2. Define the decision variables.
3. Set up the objective function.
4. Define the constraints.
5. Solve the optimization problem.
6. Print the results.

Step 4: Execute the Code

Here is the Python code to solve the problem:

```
import pandas as pd
import pulp
```

```

# Step 1: Read the data from the CSV file
file_path = "datos_50_50.csv"
data = pd.read_csv(file_path, sep=";", header=None, names=["IDENTIFICADOR", "CARGO", "PERCENT_PRI", "DISTANCE_KM"])

# Step 2: Define the decision variables
# We will use a binary variable x[i, j] which is 1 if employee i is assigned to position j, and 0 otherwise.
employees = data['IDENTIFICADOR'].unique()
positions = data['CARGO'].unique()
x = pulp.LpVariable.dicts('x', [(i, j) for i in employees for j in positions], cat='Binary')

# Step 3: Define the problem
prob = pulp.LpProblem("Employee_Position_Allocation", pulp.LpMinimize)

# Step 4: Define the objective function
# We will use a weighted sum of PERCENT_PRI and DISTANCE_KM
weight_percent_pri = 0.5
weight_distance_km = 0.5

objective = pulp.lpSum(
    weight_percent_pri * data.loc[(data['IDENTIFICADOR'] == i) & (data['CARGO'] == j), 'PERCENT_PRI'].values[0] * x[i, j] +
    weight_distance_km * data.loc[(data['IDENTIFICADOR'] == i) & (data['CARGO'] == j), 'DISTANCE_KM'].values[0] * x[i, j]
    for i in employees for j in positions
)

prob += objective

# Step 5: Define the constraints
# Each employee must be assigned to exactly one position
for i in employees:
    prob += pulp.lpSum(x[i, j] for j in positions) == 1

# Each position must be assigned to exactly one employee
for j in positions:
    prob += pulp.lpSum(x[i, j])

```

USER

run the code

BOT