

# Implementação do algoritmo k-NN

Marco Cezar Moreira de Mattos<sup>1</sup>, Rômulo Manciola Meloca<sup>1</sup>

<sup>1</sup>DACOM – Universidade Tecnológica Federal do Paraná (UTFPR)  
Caixa Postal 271 – 87301-899 – Campo Mourão – PR – Brazil

{marco.cmm,rmeloca}@gmail.com

**Resumo.** *Relata o procedimento tomado para implementar o algoritmo k-NN e os testes feitos com ele sobre um conjunto de dados.*

## 1. O Problema

## 2. O algoritmo

meses do ano

1200 instâncias para o conjunto de teste com 24 características teste controlado uma vez que possuem respostas

3600 instâncias para o conjunto de treino com 24 características possui resposta uma vez que é um algoritmo supervisionado, isto é

utilizou-se a distância euclidiana

implementou-se em java. Diagrama de classes.

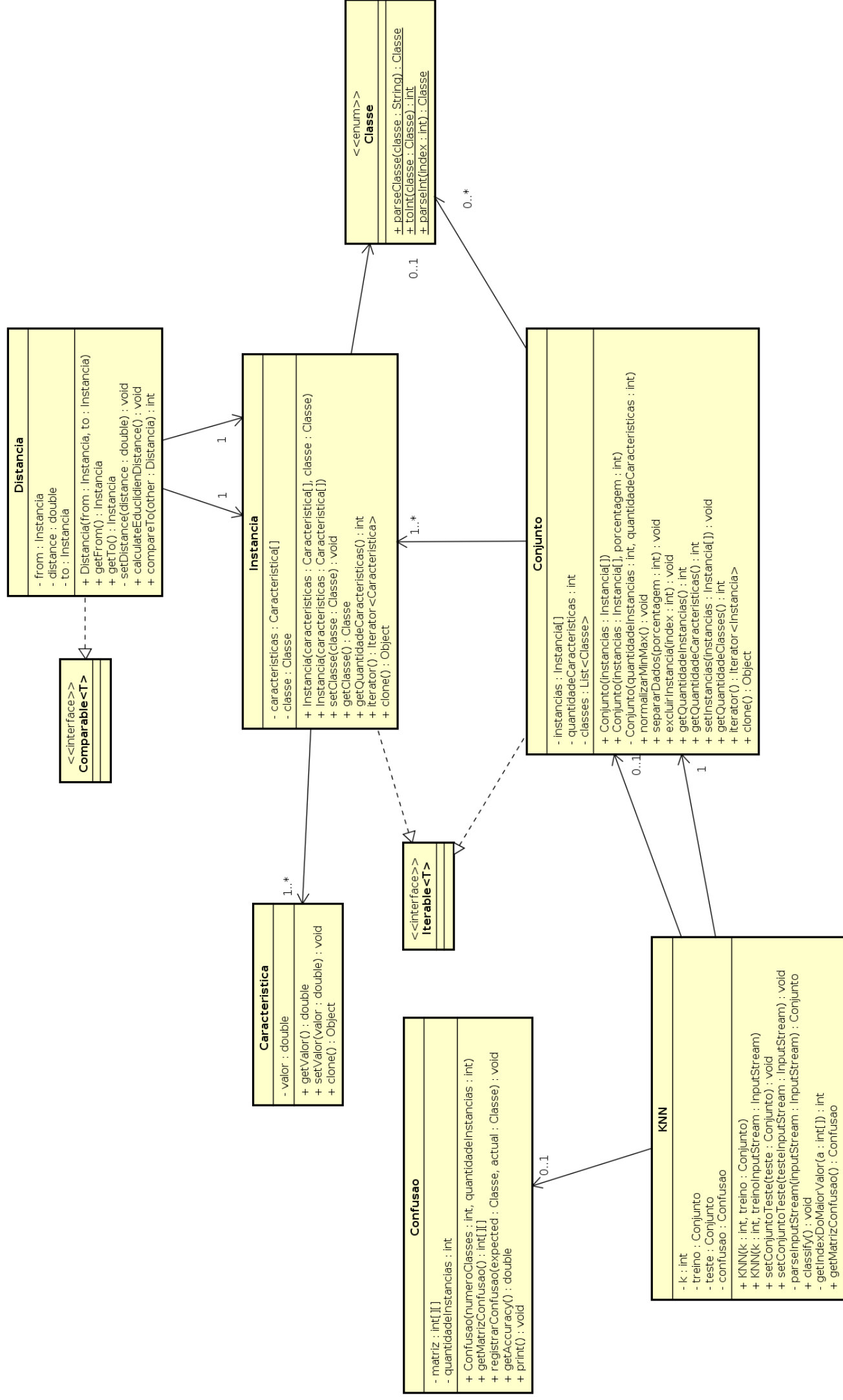


Figura 1. Diagrama de Classes

**Data:** Instância de um Puzzle a ser resolvida.

**Result:** Lista do caminho percorrido para solucionar o puzzle.

Insere a primeira instância na lista do caminho percorrido;

**while** *Puzzle não está resolvido* **do**

    Obtém o última instância do caminho;

    Obtém os possíveis movimentos da instância;

    Calcula a heurística para cada possível movimento;

    Escolhe a instância que possui melhor heurística;

    Adiciona a instância ao caminho percorrido;

**end**

**Algorithm 1:** Busca A\* para resolver 8-Puzzle

### 3. Resultados

- Seu algoritmo deve avaliar o desempenho para diferentes valores de k;
- Gerar a matriz de confusão;
- Usar a distância Euclidiana, Manhattan ou outra;
- Normalizar os dados com Min-Max ou Z-score;
- Separar o conjunto de treinamento (aleatoriamente) em 25%, 50% e 100% dos dados de treinamento. Avaliar qual o impacto de usar mais e menos instâncias no conjunto de treinamento.

### 4. Considerações Finais

### 5. Referências