

Trabalho Prático: Ordenação
Disciplina: FGA0147 - Estruturas de Dados e Algoritmos
Prof. Dr. Nilton Correia da Silva
Faculdade UnB Gama - FGA
Universidade de Brasília - UnB

Aplicação de Algoritmos de Ordenação em Análises de Atrasos de Voos de Companhias Aéreas Norte Americanas.

Fonte de dados: Airlines Dataset to predict a delay



Gama, Distrito Federal

Sumário

1	Introdução	3
2	Dataset	4
2.1	Volumetria	4
2.2	Variáveis de Interesse	4
2.2.1	Tempo de Voo	4
2.2.2	Voo Atrasado ou Não	4
3	Objetivo	5
3.1	Variáveis do Arquivo de Saída	5
3.2	Escolha do Método de Ordenação	5
4	Condições de Contorno da Solução	5



Orientações

Trabalho Prático da disciplina de Estruturas de Dados e Algoritmos. Este trabalho foi elaborado considerando que sua resolução deve ser realizada de forma individual.

A solução deverá ser em linguagem C ou C++. No caso do acadêmico adotar C++, poderá se valer de uma solução orientada a objetos, contudo sua solução não deve usar classes e contêineres já prontos da linguagem C++ - por exemplo, algoritmos de ordenação, busca, objetos vector, list, matrix, etc.



1 Introdução

Algoritmos de ordenação são repetidamente demandados em diferentes tipos de análises de dados. Análises estatísticas que visam quantificar ou qualificar uma determinada característica, por vezes, requerem que o algoritmo realize ordenações em um dataset tendo como chave uma ou mais de suas variáveis - geralmente, as variáveis vêm em forma de colunas do dataset.

Neste trabalho teremos a oportunidade de exercitar algoritmos de ordenação em um caso concreto representado por um dataset que foi preparado para que engenheiros de IA (Inteligência Artificial) e outros profissionais pudessem apresentar soluções para um problema envolvendo o referido dataset. Nosso objetivo aqui, contudo, não será apresentar uma solução final para o problema apresentado no link da fonte de dados (veja a referência abaixo do título do trabalho), mas sim, apresentar uma solução mais simples que será detalhada nas seções posteriores.

Caso o dataset disponibilizado tenha uma quantidade de registros maior do que os seus recursos computacionais conseguem processar, você pode excluir registros ou colunas (desnecessárias à solução), deixando apenas a quantidade máxima de dados que seus recursos computacionais conseguem processar.



2 Dataset

O dataset escolhido para este trabalho refere-se ao registro de voos de diferentes companhias aéreas norte americanas. Ademais aos detalhes do voo, o dataset informa ainda se o voo atrasou ou não (variável booleana).

Desafio para o Engenheiro de IA: Gerar um modelo de IA que seja capaz de prever se um voo irá atrasar ou não.

2.1 Volumetria

Instâncias: 539.383 linhas

Atributos: 18 colunas

Tamanho: Compactado: 6,5MB, Descompactado: 18,51 MB

Formato: csv (valores separados por vírgula)

2.2 Variáveis de Interesse

Sua solução não demandará todas as variáveis do dataset. Veja abaixo o detalhamento das variáveis (colunas do dataset) que você precisará processar.

2.2.1 Tempo de Voo

Esta variável está na coluna **Length**. Esta coluna apresenta um número que representa o tempo do voo em minutos.

Trata-se de um valor inteiro.

2.2.2 Voo Atrasado ou Não

Esta variável está na coluna **Delay**. Esta coluna apresenta 0, caso voo não teve atraso ou 1, quando se tratar de voo com atraso.

Trata-se de um inteiro [0 | 1].



3 Objetivo

A solução deste trabalho consiste em responder à seguinte questão:

O tempo de voo influencia em atrasos?

Para responder a esta pergunta, você deve gerar um arquivo de saída contendo tempo de voo e sua respectiva indicação de atraso.

Segue abaixo o detalhamento do arquivo de saída que sua solução deverá gerar.

3.1 Variáveis do Arquivo de Saída

Tempo:. Esta coluna deve trazer o tempo do voo em minutos. Você encontrará esta informação no campo *Length* do arquivo de entrada.

Atraso:. Esta coluna deve trazer a indicação de o voo está atrasado (valor 1) ou não (valor 0). Você encontrará esta informação no campo *Delay* do arquivo de entrada..

Este arquivo de estar ordenado pela coluna *Tempo*

3.2 Escolha do Método de Ordenação

A ordenação é um ponto importante deste trabalho. Um dos métodos abaixo deverá ser adotado para a solução:

1. Insert Sort
2. Bubble Sort
3. Selection Sort
4. Quick Sort

Veja: Principais Algoritmos de Ordenação

4 Condições de Contorno da Solução

Sua solução deve atender às seguintes condições:

1. O arquivo de entrada (dataset) deve ser lido por seu programa em sua apresentação original - não deve ser ordenado antes por outro programa.
2. As linhas do arquivo de saída devem estar em ordem crescente pela coluna **Tempo**.

