



Aplicação 2

Introdução à Ciência de Dados e Análise de Desempenho Utilizando Árvores AVL e BST com o Dataset NetFlix

A ciência de dados é uma área interdisciplinar que combina métodos e técnicas de estatística, aprendizado de máquina, programação, matemática, e conhecimento de negócio para analisar, interpretar e obter *insights* a partir de dados brutos. Neste contexto, as principais etapas envolvidas no processo compreendem:

- Formulação de perguntas de pesquisa
- Coleta e preparação dos dados
- Exploração e análise dos dados
- Modelagem e seleção de algoritmos
- Avaliação e validação dos modelos
- Interpretação dos resultados
- Comunicação dos resultados

Uma representação gráfica do ciclo de vida do processo de Ciência de Dados está ilustrada na Figura 1 (AWARI, 2022). Na etapa *Data Exploration* (Exploração dos Dados ou podemos dizer Compreensão dos Dados) o cientista de dados faz uso de diversas técnicas de análise de dados, estatística e visualização para explorar o conjunto de dados obtido para melhor compreendê-lo. Segundo Facelli (2021), “a análise das características presentes em um conjunto de dados permite a descoberta de padrões e tendências que podem fornecer informações valiosas que ajudem a compreender o processo que gerou os dados”. Para uma leitura complementar sobre Ciência de Dados acesse: <https://www.heavy.ai/learn/data-science>

Em aplicações onde o volume de dados é grande, acessá-los diretamente em um arquivo pode ser um problema do ponto de vista do tempo. Memórias secundárias são muito mais lentas que a memória principal e isso pode comprometer o desempenho das aplicações. Neste sentido, mapear os dados em memória, através de estruturas de dados, significa otimizar o acesso aos dados e agilizar as análises.

Uma estrutura de dados em memória oferece várias vantagens sobre o acesso direto a um arquivo, tais como: velocidade de acesso, tempo de resposta, melhor desempenho em leituras e gravações, melhor desempenho em operações de busca/pesquisa etc.

Na Aplicação 2, faremos o uso de duas estruturas de dados, **AVL** e **BST**, que organizarão em memória dados que serão lidos de um *dataset* público com informações sobre o conteúdo disponível em uma das maiores plataformas de streaming por assinatura do mundo, a NetFlix¹. Ao utilizar os dois modelos de árvores, queremos comparar a eficiência/desempenho dessas duas

¹ Dataset de programas da NetFlix. Disponível em: <https://www.kaggle.com/datasets/victorsoeiro/netflix-tv-shows-and-movies>. Acesso em: 03 de outubro de 2023.

estruturas em um caso real. Os dados do *dataset* foram adquiridos em julho de 2022 nos Estados Unidos.

Figura 1.: Ciclo de Vida de Ciência de Dados



Fonte: <https://awari.com.br/tudo-sobre-ciencia-de-dados/>. Data da Consulta: 03/10/2023.

Ao acessar o site da Kaggle, fazer o *download* do arquivo compactado *arquivo.zip* e descompactá-lo, são encontrados dois *datasets*: **titles.csv** e **credits.csv** (O arquivo *arquivo.zip* também encontra-se dentro do Moodle na aba relativa as atividades Apl1 e Apl2). O arquivo *credits.csv* não será utilizado na nossa análise, somente **titles.csv**. Em **titles.csv** são encontradas 15 colunas com os atributos:

- id: o ID do título em JustWatch.
- título: O nome do título.
- show_type: programa de TV ou filme.
- descrição: Uma breve descrição.
- release_year: o ano de lançamento.
- age_certification: A certificação de idade.
- runtime: A duração do episódio (SHOW) ou filme.
- gêneros: Uma lista de gêneros.
- Production_countries: Uma lista de países que produziram o título.
- temporadas: Número de temporadas se for um SHOW.
- imdb_id: O ID do título no IMDB.
- imdb_score: Pontuação no IMDB.
- imdb_votes: Votos no IMDB.
- tmdb_popularity: popularidade no TMDB.
- tmdb_score: Pontuação no TMDB.

Atenção: A **chave** de inserção na BST e na AVL deve ser o id do título.

Deve ser criada uma classe, denominada ProgramaNetFlix, contendo todos os 15 atributos, de forma privada, e os métodos construtor, *getters* e *setters* públicos, para criação de objeto, leitura

e atualização dos valores. Além do atributo “id”, essa classe também será um atributo para as classes BST e AVL.

Posteriormente, elaborar um programa contendo opções de um menu para:

1. **Ler dados de arquivo:** no qual o arquivo original deve ser lido e as árvores BST e AVL montadas. Deve ser solicitado ao usuário a leitura do nome do arquivo de dados (*dataset*) a ser lido. Antes de realizar a inserção na árvore de cada programa Netflix verificar se todos os 15 atributos estão preenchidos, caso não estejam descartar e não inserir nas estruturas. Além disso, caso algum atributo não seja relevante para sua análise, descarte-o, porém não se esqueça de detalhar no relatório solicitado todas as decisões de alteração no *dataset*.
2. **Cinco opções contendo métodos para análise de dados:** que devem ser implementados somente na AVL, sendo todos **bem elaborados** e não contagens triviais apenas.

Exemplos de análises bem elaboradas podem ser: apresentar os *top* 10 títulos com *age_certification* = TV-14 que inclui o gênero “crime”; apresentar os N títulos ($N > 5$, fornecido pelo usuário) com os menores valores de *tmdb_score*; e/ou outras questões pertinentes para sua análise.

Cada grupo irá planejar as análises que deseja realizar para investigar questões sobre os dados mapeados na AVL. Formule suas questões antes! Lembre-se, a primeira etapa na ciência de dados é definir claramente as perguntas de pesquisa ou os problemas a serem resolvidos.

Os resultados obtidos em cada um dos cinco métodos devem ser devidamente formatados e apresentados. É necessário utilizar métodos de percurso diferentes (pré-ordem, em ordem simétrica, pós-ordem e largura) em pelo menos três das cinco opções.

3. **Inserir Programa:** os dados de um novo Programa Netflix devem ser inseridos em um novo nó das árvores BST e AVL. Para isso, crie um “id” para o programa Netflix, considerando o padrão de cada categoria (**ts** + número único ou **tm** + número único, onde **ts** = categoria SHOW; **tm** = categoria MOVIE; e “número único” é um número que identifica um programa de forma individual).
4. **Buscar Programa:** fazendo uso da BST e da AVL, solicitar do usuário o “id” do programa e apresentar: os dados desse programa; ou o título do programa; ou qualquer outro(s) dado(s) que desejar. Somente os resultados obtidos com a pesquisa na AVL devem ser apresentados. **No entanto, seu programa deve contabilizar o número de comparações realizadas para encontrar o nó e o tempo de execução dessa busca nas duas árvores: BST e a AVL**, mostrando os resultados das comparações e o tempo de execução para cada uma delas. Para contabilizar o tempo, use monitores de tempo para isso.
5. **Remover Programa:** a partir do ID do programa Netflix, fornecido pelo usuário, remover o nó das árvores BST e AVL correspondentes.
6. **Exibir a Altura das Árvores:** mostrar a altura das árvores AVL e BST.

7. **Salvar dados em arquivo:** salva os dados atualmente armazenados na árvore AVL em disco. Forneça a opção ao usuário de informar o nome do arquivo de gravação (é permitido gravar os novos dados no arquivo já existente).
8. **Encerrar a Aplicação:** os dados alocados das árvores BST e AVL são liberados e a aplicação desenvolvida é finalizada.

Observações:

1. O trabalho pode ser feito por grupos de até 4 pessoas.
2. Um único aluno do grupo deverá publicar o trabalho no Moodle.
3. Deverá ser entregue um relatório com os resultados da **“Atividade Aplicação 2”** deste projeto com base no *Template* disponibilizado contendo **obrigatoriamente**:

- Dados dos integrantes do grupo (nome e TIA).
- Decisões relativas ao dataset, por exemplo: remoção de objetos (motivo), eliminação de colunas (atributos - motivos), outros.
- Informações e detalhes sobre as cinco opções selecionadas pelo grupo para análise.
- *Printscreen* de testes de execução mostrando todas as opções do menu. Ao menos 2 testes de cada opção, se for permitido, caso contrário basta um único teste da opção.
- O relatório deve conter ao seu final um Apêndice contendo o código fonte desenvolvido (separado por arquivos, se for o caso). Em cada arquivo inserir:
 - um cabeçalho (comentário) com as identificações completas de todos os membros do grupo.
 - Documentação adequada e inclusão de comentários úteis e informativos.

4. Junto ao relatório também devem ser entregues os códigos fontes em Java e o *dataset* utilizado (contendo as modificações realizadas pelo grupo). A entrega deve ser realizada na data limite 23 de novembro 23h59min.

5. Deverá ser realizada uma apresentação do projeto no dia 23 de novembro no horário da aula:
 - O grupo deverá apresentar o processo de construção da solução do seu projeto, resultados obtidos e testes do menu de opções no tempo máximo de 6 (seis) minutos. Logo, fazer prévias para não ultrapassar o tempo limite, cada minuto que passar será descontado 0,25 ponto.

O projeto será avaliado de acordo com os seguintes critérios:

- Completude, clareza e ausência de erros de linguagem no relatório;
- Funcionamento correto da Aplicação;
- O trabalho deve ser desenvolvido na **linguagem Java** e será testado usando no Eclipse.
- O quão fiel é o programa quanto à descrição do enunciado;
- Indentação, comentários e legibilidade do código;
- Clareza na nomenclatura de variáveis e funções;
- Apresentação realizada com clareza, conhecimento e cumprimento do tempo estabelecido.

Para auxiliar na documentação do código e entendimento do que é um programa com boa legibilidade siga as dicas apresentadas nas páginas abaixo:

- <http://www.ime.usp.br/~pf/algoritmos/aulas/layout.html>
- <http://www.ime.usp.br/~pf/algoritmos/aulas/docu.html>

Como este trabalho pode ser feito em **grupo**, evidentemente você pode “*discutir*” o problema dado com outros **grupos**, inclusive as “*dicas*” para chegar às soluções, mas você deve ser responsável pela solução final e pelo desenvolvimento da sua aplicação.

Referências

- AWARI, Tudo sobre Ciência de Dados: o que é, como funciona e qual sua importância. Fevereiro, 2022. Endereço: <https://awari.com.br/tudo-sobre-ciencia-de-dados/>. Data da Consulta: 03/10/2023.
- FACELI, Katti et al. Inteligência artificial: uma abordagem de aprendizado de máquina. 2ª. Edição. Rio de Janeiro: LTC- Livros Técnicos e Científicos. 2012. Endereço da biblioteca do Mackenzie: [https://app.minhabiblioteca.com.br/reader/books/9788521637509/epubcfi/6/2\[%3Bvnd.vst.idref%3Dcover!\]/4/2/2%4051:3](https://app.minhabiblioteca.com.br/reader/books/9788521637509/epubcfi/6/2[%3Bvnd.vst.idref%3Dcover!]/4/2/2%4051:3). Data da consulta: 03/10/2023.