Introdução à Ciência de Dados

usando Árvores

Enzo Benedetto Proença, 10418579

Gabriel Alves de Freitas Spinola Sucupira, 10418133

Lucas Pires de Camargo Sarai, 10418013

Vitor Alves Pereira, 10410862

Sumário

[Introdução 3](#_Toc183451415)

[Descrição do Dataset 4](#_Toc183451416)

[Modelagem dos Dados 5](#_Toc183451417)

[Descrição da Implementação das Estruturas de Dados e as Operações Realizadas 7](#_Toc183451418)

[Gráficos e Análises de Resultados 14](#_Toc183451419)

[Comparação de Desempenho das Árvores BST e AVL 22](#_Toc183451420)

[Testes e Resultados 23](#_Toc183451421)

[Conclusão 28](#_Toc183451422)

[Reflexões Finais 29](#_Toc183451423)

[Referências 30](#_Toc183451424)

Introdução

O intuito deste trabalho é realizar uma análise de dados, usando das estruturas de árvore AVL e BST, assim como utilizando do site Governo Aberto SP, a fim de integrar o conhecimento desenvolvido em sala com os Objetivos de Desenvolvimento Sustentável (ODS), criados pelas Nações Unidas. Com isso, escolhemos o objetivo 8 : “Trabalho decente e crescimento econômico” e trabalharemos com um banco de dados sobre a variação do PIB no setor da agropecuária do período entre 2002 e 2024.

Descrição do Dataset

O *Dataset* escolhido foi “Taxas de crescimento do Valor Adicionado, por setores e subsetores de atividade econômica, dos Impostos Líquidos de Subsídios e do Produto Interno Bruto, no mês em relação ao mesmo mês do ano anterior”. Ele , mostra a variação percentual do PIB por setor econômico de um mês em relação ao mesmo mês do ano anterior. Ou seja, se janeiro de 2003 possui uma variação de +15,8% no setor da agropecuária, isso indica que em relação a janeiro de 2002, esse setor do PIB aumentou em 16%, aproximadamente.

Dentre os setores, optamos por analisar somente a agropecuária, visto que poderíamos nos aprofundar mais se focássemos em apenas um, e escolhemos esse setor mais especificamente por variar bastante e possuir vários fatos que podem ser analisados.

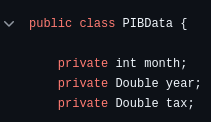
Abaixo, segue trecho da base de dados:

Interface gráfica do usuário, Tabela

Descrição gerada automaticamente

Modelagem dos Dados

Quanto aos nós (elementos inseridos nas árvores), criamos um objeto PIBData, que contém uma taxa de crescimento no setor agropecuário, além do mês e ano do dado analisado.



Para cada árvore, desenvolvemos dois tipos de organização dos dados: pela taxa de crescimento do setor agropecuário e pelas datas mês/ano de cada dado.

Texto

Descrição gerada automaticamente

Para o primeiro, usamos a taxa de crescimento por não ter valores repetidos, visto que são valores do tipo *double* e possuem muitas casas decimais. Seguindo a lógica de BST que é aplicada também na AVL, as taxas de crescimento menores que uma taxa em específico estarão à sua esquerda, enquanto as outras estarão à direita. Esse tipo de árvore nos auxiliou a descobrir os *outliers*, ou seja, as taxas de crescimento muito distantes do padrão observado. Sendo que ao percorrer até o filho mais distante à esquerda, encontramos o menor valor, e repetindo o processo na direção oposta, encontramos o maior valor.

Para o segundo tipo, usamos o ano e o mês, nessa ordem, para organizar os valores. Então as operações começavam procurando pelo ano, sabendo que anos menores estariam à esquerda, enquanto anos maiores, à direita. No entanto, temos vários valores com o mesmo ano, só que em meses diferentes; Então, usando a ideia de “chave primária composta” de banco de dados, onde um valor é diferenciado de outro pela combinação de chaves primárias e não por apenas uma única, usamos o mês para diferenciar valores de mesmo ano, utilizando o valor inteiro representativo de cada mês (01 para janeiro, 02 para fevereiro etc.), dado que se usássemos o valor texto como “Janeiro” ou “Fevereiro”, os dados seriam ordenados pela ordem lexicográfica, logo “Abril”, por exemplo, seria tratado como o primeiro mês do ano, por começar com “A”. Assim sendo, os valores à esquerda de um nó possuem não somente o valor de ano menor ou igual ao desse nó, como possuem o valor de mês menor. Esse segundo tipo nos auxiliou a identificar as médias anuais de anos específicos, pois tínhamos como procurar de forma eficiente os dados de acordo com a data.

Descrição da Implementação das Estruturas de Dados e as Operações Realizadas

Texto

Descrição gerada automaticamente

Como podemos observar na estrutura de pastas acima, temos a estrutura base de uma Árvore Binária e de um Nó de árvore binária. Além disso, definimos um tipo de dado *PIBData*, que vai ser armazenado no campo “data” do Nó,. Sendo assim, os nós serão do tipo PIBData.

Depois de montarmos a estrutura base, definimos as estruturas da BST e da AVL, usando o recurso de Herança em três ocasiões:

* BST é uma especialização de BinaryTree
* AVL é uma especialização de BST
* AVLNode é uma especialização de Node

Ainda separamos em uma pasta o arquivo CSV com os dados que trabalhamos, e a classe Main para a execução principal.

Vamos analisar cada estrutura individualmente:

PIBData

**Atributos:**

* **Month:** Mês do dado analisado em valor numérico (1 para janeiro, 2 para fevereiro etc.)
* **Year:** Ano do dado analisado, com valor numérico de 4 dígitos
* **Tax:** Taxa de variação desse dado em relação ao mesmo mês do ano anterior.

**Métodos básicos =** *Getters, Setters*, Construtor e *toString*

**Métodos específicos =** Nenhum.

Node

**Atributos:**

* **Data:** Dado armazenado no Nó, que é do tipo PIBData.
* **Parent:** Referência em memória ao *Node* armazenado como pai.
* **Left:** Referência em memória ao *Node* armazenado como filho esquerdo.
* **Right:** Referência em memória ao *Node* armazenado como filho direito.

**Métodos básicos:** *Getters, Setters*, Construtor e *toString*

**Métodos específicos:**

* getDegree(): Retorna o grau do nó, que é definido pela contagem de referências aos filhos que sejam não nulas.
* getHeight(): Retorna a altura do nó, que é a maior altura entre as subárvores acrescido de 1.
* isLeaf(): Retorna um valor booleano indicando se um nó representa uma folha, verificando se a referência de suas subárvores são a mesma, o que só ocorre quando ambas são nulas.

AVLNode

**Atributos:**

* **fb:** Mês do dado analisado em valor numérico (1 para janeiro, 2 para fevereiro etc.)

**Métodos básicos:** *Getter* e Construtor

**Métodos específicos:**

* setFb(): Calcula o fator de balanceamento, o qual é definido pela diferença de altura entre a subárvore direita e a subárvoren esquerda.

BinaryTree

**Atributo:**

* **Root:** Referência à raiz.

**Métodos básicos: *Getter, Setter* e Construtor**

**Método específico:**

* getHeight(): Retorna a altura da árvore, que é o retorno da altura da raiz.

BST

**Atributos:** Herdados de BinaryTree

**Métodos básicos:** Herdados de BinaryTree

**Métodos específicos:**

* insertAsTax(chave): Insere um nó utilizando o atributo *tax* como critério. Então para um dado nó, os nós à esquerda terão valores de *tax* menores, e à direita, maiores.
* insertAsDate(chave): Insere um nó utilizando os atributos *month* e *year* como critérios. Então quando dois nós pertencem ao mesmo ano, utiliza-se o mês como critério de desempate.
* remove(chave): Remove um nó realizando a busca pela chave passada.
* search(chave): Realiza a busca por um nó específico com base na chave passada. Para realizar essa busca, avalia-se o valor de cada nó, vendo se a chave é menor ou maior que o valor do nó. Se a chave de busca é maior, vai para direita, caso contrário, vai para esqueda. Se encontrar o nó, retorna sua referência, senão, retorna null; Essa lógica é aplicada nas operações de inserção e remoção.
* getSucessor(): Obtém o nó de menor valor de taxa ou data da subárvore direita.
* getTotalFromYear(chave): Retorna a somatória das taxas de um ano “chave”. Esse método é privado, pois é utilizado apenas para calcular a média.
* getAverageFromYear(chave): Após chamar o método de getTotalFromYear(chave), divide por 12 meses e obtém-se a média anual.
* getMaxIncrease() e getMaxDecrease(): Retornam a maior taxa de crescimento e a maior taxa de queda mensal, respectivamente. As lógicas são semelhantes, apenas direcionadas em subárvores diferentes. O primerio busca pelo nó mais à direita da raiz, o qual terá o maior valor da BST. O outro, pelo nó mais à esquerda, o qual terá o menor valor.
* isTaxBST() e isDateBST(): Retornam um booleano indicando se a árvore montada pode ser classificada como uma BST, mostrando se a montagem foi correta. O primeiro realiza a comparação considerando que os valores de *tax* de cada nó são diferentes e que à esquerda tem que ter apenas valores menores, enquanto à direita, apenas os maiores. Para a árvore baseada na data, segue a mesma lógica de comparação, porém com um acréscimo de comparar o mês, visto que existem mais de um nó com o mesmo valor do atributo *year*.
* Outros métodos: herdados de Tree.

AVL

**Atributos:** Herdados de BST.

**Métodos básicos:** Herdados de BST.

**Métodos específicos:**

* nodeIsRoot(): Retorna se um nó em específico representa a raiz da AVL. Esse nó costuma ser implementado como método de *Node* e cuja lógica se baseia em verificar se a referência de *parent*(“pai” desse nó) é nula. Entretanto, como não usamos essa referência em nenhum nó, então utilizamos outra abordagem: verificar se a referência do nó passado corresponde à referência do nó no atributo *root* e, para isso, precisamos definir o método dentro de uma classe com o atributo *root*. Como esse método só foi usado para os métodos de rotação, então o definimos diretamente na AVL.
* rotateRight(): Realiza a rotação simples à direita de um nó. Para tanto, as seguintes etapas são seguidas:
  + O filho à esquerda desse nó se torna seu “pai”.
  + O filho à direita do filho à esquerda se torna o novo filho à esquerda do nó.
  + Se o nó que sofreu a rotação era a raiz, atualiza a referência da raiz.

Esse método é chamado quando a árvore cresce muito para a esquerda.

* rotateLeft(): Realiza a rotação simples à esquerda de um nó. Para tanto, as seguintes etapas são seguidas:
  + O filho à direita desse nó se torna seu “pai”.
  + O filho à esquerda do filho à direita se torna o novo filho à direita do nó.
  + Se o nó que sofreu a rotação era a raiz, atualiza a referência da raiz.

Esse método é chamado quando a árvore cresce muito para a direita. Ambos os métodos de rotação visam rebalancear a árvore para manter a eficiência das operações.

* insertBalancedAsTax(chave); insertBalancedAsDate(chave): Insere um nó seguindo a lógica de inserção nas BSTs baseadas em taxa e data, respectivamente. O que possui de específico é o cálculo do fator de balanceamento para cada nó visitado durante a inserção, com o rebalanceamento da árvore com as rotações quando necessário.
* deleteBalanced(chave): Remove um nó assim como na BST, com o acréscimo do cálculo dos fatores de balanceamento e o rebalanceamento caso necessário.
* Outros métodos: Herdados de BST.

**Estruturas em Main.java:**

* List<PIBData> list: Armazena os dados a serem inseridos nas árvores baseadas na taxa.
* List<PIBData> unsortedList: Armazena os dados a serem inseridos nas árvores baseadas em data. Essa lista foi criada dado que a lista original estava organizada por data, o que geraria os piores casos nas operações feitas na BST, visto que se tornaria uma lista encadeada. Então embaralhamos a lista dividindo-a em três grupos com um terço da lista original. Em seguida, inserimos um elemento do segundo grupo, depois do primeiro e depois do terceiro. Dado que o terceiro grupo ficou com um tamanho um pouco menor, utilizamos uma condição para verificar se ainda existem elementos nesse grupo a serem inseridos.

Texto

Descrição gerada automaticamente

* HashMap<String,Integer> months: Faz uma conversão de um valor de texto de mês para um valor numérico, utilizando a numeração padrão: Janeiro = 1, Fevereiro = 2 etc.
* TreeMap<Double,Double> averageDiffFromYears: Faz uma relação entre ano e a diferença entre sua média anual e a média do ano anterior. Para montá-la, chamamos a função de calcular média anual para cada ano entre 2003 e 2023, sempre subtraindo da média do ano anterior. Utilizamos essa diferença como chave para que conseguíssemos obter a maior e a menor de forma mais rápida.

Texto

Descrição gerada automaticamente

Gráficos e Análises de Resultados

Texto

Descrição gerada automaticamente

• **Pergunta Exploratória 1**: *“Qual foi o motivo do maior crescimento percentual em todo o período (2003-2024)?*

Em todo o período de 2003 a 2024, o maior crescimento no estado de São Paulo ocorreu entre 2014 e 2016, período em que houve uma recessão econômica em nível nacional, porém o setor agropecuário paulista conseguiu crescer por diversos fatores: diversificação na produção, liderança na produção de cana-de-açúcar e etanol no país e desvalorização do real são alguns fatores.

O primeiro fator auxiliou o estado a compensar a queda em alguns produtos agrícolas com o crescimento na produção de outros, se mantendo competitivo em nível nacional e internacional.

A liderança na produção de cana e etanol também foi importante, pois esses produtos se mantiveram em alta no período, e sendo o estado o líder na produção, por consequência, continuou crescendo.

Com a desvalorização do real frente ao dólar, a venda de produtos em dólar aumentou os ganhos em real, além de os produtos se tornarem mais baratos e, assim, mais procurados e competitivos em nível internacional.

Portanto, mesmo com o PIB geral em queda, o setor cresceu muito no estado durante esse período, como podemos ver nos gráficos abaixo:

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Gráfico

Descrição gerada automaticamente com confiança média

• **Pergunta Exploratória 2**: *“Qual foi o motivo da maior queda em todo o período (2003-2024)?*

A maior queda ocorreu entre 2008 e 2009, com consequências até 2010, ano que teve uma taxa de fruto da crise financeira global de 2008-2009 que começou nos Estados Unidos e ficou conhecida como “Grande Recessão”. Essa crise começou no setor imobiliário norte-americano ao qual nos anos 2000 teve uma expansão muito grande no mercado, com preços de imóveis em rápida crescente e oferta excessiva de hipotecas de alto risco. Eventualmente, essa bolha imobiliária explodiu, levando à falência do banco Lehman Brothers, um banco tão importante que sua quebra levou os efeitos para fora dos EUA e a crise pôde ser sentida em todo mundo.

As consequências no setor agropecuário mundial foram devastadoras, principalmente pela queda da demanda global de *commodities* (matérias-primas) agrícolas, além da inflação de seus preços, somados a desconfiança com a contratação de crédito agrícola, visto que a crise começou com a contração excessiva de crédito, gerando uma desconfiança global nesse meio, dificultando o financiamento para insumos, equipamentos e expansão da produção.

Esse cenário afetou muito o estado de São Paulo, como podemos ver nos gráficos abaixo:

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Gráfico, Gráfico de cascata

Descrição gerada automaticamente

Por outro lado, importante notar que a média geral de crescimento do PIB em São Paulo foi a mais elevada em 2010, dado que a partir desse ano a economia nos outros setores começou a se recuperar e como os valores nos outros anos estavam numa queda muito grande, um mínimo crescimento que fosse apresentado acabaria subindo muito a curva do gráfico.

• **Pergunta Exploratória 3**: *“Existiram valores fora do esperado (outliers)? Qual o motivo?”*

Inicialmente, devemos analisar a distribuição geral:

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

Gráfico, Histograma

Descrição gerada automaticamente

Com base nesses gráficos, notamos que as taxas costumam variar entre -10% e +10%. Entretando, alguns valores se afastam um pouco do que é esperado, como -34,41% e + 34,48% em junho de 2017 e 2015, respectivamente, como podemos observar nas extremidades dos gráficos.

Dessa forma, vamos analisar cada um desses valores individualmente.

Gráfico, Gráfico de cascata

Descrição gerada automaticamente

Gráfico

Descrição gerada automaticamente

Em 2017, houve valores muito discrepantes em maio e junho em relação ao crescimento que o setor vinha tendo: -25% e -34,41%, respectivamente¨. Isso ocorreu por conta de condições climáticas desfavoráveis, principalmente com a escassez de chuvas, e essa seca afetou a produção agrícola, incluindo uma das principais culturas agrícolas de São Paulo, que vinha sendo uma das responsáveis pelo crescimento do setor agrícola mesmo durante a crise: a cana-de-açúcar. Com a queda nessa produção, esse setor ficou mais vulnerável às oscilações econômicas que o PIB do estado vinha sofrendo no ano de 2017.

Por outro lado, em 2015 teve uma alta considerável em março e junho, atingindo valores de +30% e +34% em relação ao ano anterior, respectivamente. Nesses meses, o crescimento é acentuado pela colheita de grãos, onde se intensifica essa colheita antes de começar o inverno, principalmente das culturas de soja e milho, sendo março o início e junho o fim, por isso uma variação maior nesses meses, além de junho costuma ser marcado pela época de abate de gado e variação nos preços da carne.

Logo, nota-se que esses dois meses costumam atrair valores *outliers.* Isso porque eles são os meses de rotação de culturas animais e vegetais e, portanto, geram uma oscilação no PIB agropecuário.

• **Pergunta Exploratória 4**: *“Qual foi o crescimento/decrescimento da participação do setor agropecuário no período pré-pandemia?”*

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Como verifica-se ao analisar o gráfico e pelos cálculos de variação percentual entre 2017 e 2018 e entre 2018 e 2019, nota-se que houve uma queda entre 2017 e 2018 de -9,63%, porém cresceu 6.83% entre 2018 e 2019, com uma média no período desses três anos de -1.40%. A queda inicial pode ser explicada pelo fato de ter existido um crescimento muito grande nos anos anteriores, e uma produção um pouco mais regular, e com quedas bruscas nos meses de março e junho (ver questão 3), ocasionaram uma queda anual percentual elevada em relação ao ano anterior, além de fatores como condições climáticas adversas (secas, como a do alto do Rio Negro) e a greve dos caminhoneiros do final de 2017 pelo preço dos combustíveis, custos operacionais e pela insatisfação com as condições de trabalho, consequentemente paralisando o transporte de insumos agrícolas para os portos e para as cadeias de produção.

Já no período de 2018 e 2019 houve a eleição do novo presidente, que investiu no setor agropecuário por meio do Plano Safra, destinando recursos para o custeio e comercialização de produtos agrícolas, e para investimentos, como a compra de maquinário.

• **Pergunta Exploratória 5**: *“Durante a pandemia, a economia global sofreu um colapso. Como ela afetou o setor agropecuário paulista?*

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Com a crise global encadeada pela pandemia, o setor agropecuário foi muito afetado principalmente entre 2020 e 2021, no início da pandemia, com uma queda de -7,81%. Entretando, a economia do setor se recuperou no final da pandemia entre 2021 e 2022 com um crescimento de 5,24%. Os motivos da queda foram maiores para os pequenos produtores pelas dificuldades de manutenção da dinâmica produtiva e comercial; como viajar para buscar e vender os produtos e impactos nos volumes de produção pela redução da equipe em serviço em nome do isolamento. Assim, a variação média percentual foi de -1,28%, representando uma queda menor que entre 2017 e 2019, o que pode ser explicado, principalmente, pelo investimento do novo governo.

Comparação de Desempenho das Árvores BST e AVL

Texto

Descrição gerada automaticamente

Considerando que a eficiência das operações em árvores binárias é O(h), sendo ‘h’ a altura da árvore, temos:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Inserção** | **Remoção** | **Busca** |
| **AVL para taxas** | 9 | 9 | 9 |
| **AVL para datas** | 8 | 8 | 8 |
| **BST para taxas** | 18 | 18 | 18 |
| **AVL para datas** | 87 | 87 | 87 |

Dados esses valores, podemos inferir que:

* A AVL para datas teve a melhor eficiência em operações, atingindo a altura mínima possível para árvores binárias de busca de 260 elementos, que como calculado é 8.
* A AVL para taxas teve uma altura que garante uma eficiência nas operações, com quase a altura mínima, mas com uma eficiência muito semelhante.
* A BST para datas teve a pior eficiência dentre as árvores geradas, dado que ela recebeu um vetor de entrada que estava originalmente ordenado, passou por um processo para gerar uma versão não ordenada, porém seguindo uma lógica não aleatória, o que acabou gerando a pior eficiência. Dito isso, vale mencionar que caso o vetor de entrada fosse passado em sua forma original, a BST para datas teria sua pior altura de 259.
* A BST de taxas foi a BST mais eficiente, com altura 18. Essa altura foi obtida considerando que os dados de inserção estavam desordenados.

Importante ressaltar que as árvores que trabalham com datas passavam por uma verificação a mais para diferenciar nós de mesmo ano pelo mês, mas o custo se mantém o mesmo.

Testes e Resultados

**Testes:**

* isTaxBST e isDateBST: Validando as árvores criadas respeitam as condições de uma BST
  + Validando também o acesso aos nós, verificando seus atributos e métodos.
  + Observação: Não validado se está balanceada por algum método específico, mas validado via método de obter altura, pois atingindo a altura mínima ou próxima disso, sabe-se que foi balanceado corretamente.
* getHeight(): Validando se é possível obter a altura das árvores
* getAverageFromYear(): Validando se é possível obter as médias anuais, passando um ano como chave. Além disso, valida-se a função de getTotalFromYear(), que faz a somatória das taxas de um ano.
* getMaxIncrease() e getMaxDecrease(): Validar se os dados foram inseridos corretamente ao confirmar que o valor mais à direita da raiz é o maior valor da árvore e o valor mais à esquerda, o menor valor.
* insertAsTax(), insertAsDate(): Validando se as árvores foram montadas corretamente, pela combinação dos métodos anteriores, verificando se respeitou as características de uma BST para cada caso.
* insertBalancedAsTax(), insertBalancedAsDate(): Validando se as árvores foram montadas corretamente, pela combinação dos métodos anteriores, verificando o balanceamento e as condições de uma BST também para cada caso.
* search(): Validando se a busca por valores está correta, ao realizar essa operação para descobrir os outliers.
* Implementações Map: Validando se é possível obter os valores desejados pela chave. Além disso, para TreeMap, testa-se o funcionamento da estrutura para ver se há de fato a ordenação natural dos valores.
* Leitura de arquivo correta: Validar se é possível obter os dados dos arquivos e usá-los nas estruturas.

**Entrada:**

Árvores:

Texto

Descrição gerada automaticamente

Implementações Map

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Arquivo: “08-2024\_tabelas\_PIB\_mensal\_Mensal04.csv”

Texto

Descrição gerada automaticamente

Saída:

* Texto

  Descrição gerada automaticamente

Conclusão

Concluímos que o setor agropecuário possui um comportamento de variação percentual muito diferente dos outros setores. Com efeito, ele estava em queda quando a média do PIB crescia, ou estava em alta quando a média geral diminuía. Conforme nossas pesquisas, isso se deve ao fato de possuir fatores que o afetavam por mais tempo ou isoladamente, como notamos durante a crise de 2008-2009, em que esse setor sofreu uma queda maior e demorou mais para se recuperar, além de percebermos que em 2017 apenas ele sofreu com as condições climáticas, enquanto a média geral estava voltando a subir. Por outro lado, alguns fatores eram mais benéficos para a agropecuária, como a desvalorização do real nos tempos da crise entre 2014 e 2017, quando ficou evidente que o setor prosperou mais que os outros, que estavam na maior taxa decrescente de todo o período.

Esses resultados nos ajudaram a inferir que esse setor possui um comportamento muito instável e que deve estar sempre sendo cuidadosamente analisado, pois sofre muitas oscilações, tanto positiva como negativamente. Assim, para cumprir com o objetivo 8 “Trabalho decente e crescimento econômico”, cujos objetivos visam sustentar o crescimento econômico por meio do desenvolvimento tecnológico e social dos países, entendemos que esse setor precisa estar em constante análise e evolução, pois ele pode ser um dos principais meios para se alcançar esse objetivo, ou acabar sendo um dos principais empecilhos para tanto.

Quanto as limitações, percebemos que os dados estavam sempre variando muito, não só pelas condições acima, mas também pelo tipo de dado analisado. Trata-se de variações percentuais comparando um ano ao anterior, o que pode gerar valores um tanto enganosos, dado que se um ano tiver uma crise e no outro já existir uma leve recuperação, com uma produção um pouco mais regular, os valores já cresciam muito, como se naquele ano a produção tivesse sido uma das melhores do período sendo que, realmente, ela só superou a do ano anterior.

Comparando as análises feitas em árvores, notamos que a AVL geralmente supera a BST, por ser uma BST só que balanceada, impedindo-a de crescer muito para um lado, buscando sempre se manter eficiente. Portanto, os custos acabavam se elevando muito para as BSTs.

Reflexões Finais

Enzo: “*Tive muitos problemas nos tratamentos de dados, porém superei isso com muitas pesquisas e desenvolvi conhecimento na linguagem R para análise de dados, o que me motivou a continuar os estudos na área.*”

Gabriel: “*Minha maior dificuldade foi ler o arquivo usando Java, o que me motivou a estudar mais sobre leituras de arquivo nessa linguagem, me permitindo aprender mais sobre os tipos de arquivo, e os tipos de dados dessa linguagem.*”

Lucas: “*Meus desafios foram para criar as estruturas das árvores e fazer suas operações se tornarem funcionais, o que me custou muito tempo corrigindo problemas de acessos indevidos à memória, em grande parte. Sendo assim, aprendi como lidar melhor com acesso à memória, tornando meus códigos mais eficientes e menos suscetíveis a erros*”.

Vitor: “*Aprendi a balancear as estruturas de AVL criadas e torná-las mais eficientes que as estruturas de BST, além de estudar um pouco mais e implementar as estruturas Map usadas, como HashMap e TreeMap. Meus maiores desafios foram para pensar em estruturas eficientes que melhorassem o desempenho do código.*”

Referências

[1] - OBJETIVO do Desenvolvimento Sustentável 8: Trabalho decente e crescimento econômico: Promover o crescimento econômico inclusivo e sustentável, o emprego pleno e produtivo e o trabalho digno para todos. Brasil: Nações Unidas Brasil, 2024. Disponível em: <https://brasil.un.org/pt-br/sdgs/8>. Acesso em: 25 nov. 2024.

[2] - O ANO em que o mundo quebrou: a causa e os reflexos da crise de 2008. Brasil: UOL, 4 jul. 2024. Disponível em: <https://economia.uol.com.br/noticias/redacao/2024/07/04/o-ano-em-que-o-mundo-quebrou-entenda-a-crise-financeira-de-2008.htm>. Acesso em: 25 nov. 2024.

[3] - PENA, Rodolfo F. Alves. "Crise financeira global"; *Brasil Escola*. Disponível em: <https://brasilescola.uol.com.br/geografia/crise-financeira-global.htm>. Acesso em 25 de novembro de 2024.

[4] - SECA atingiu 38 milhões de brasileiros em 2017 e cheias, 2 milhões: Estudo da ANA revela desigualdade na distribuição de recursos hídricos. Brasília: Brasil Escola, 20 dez. 2024. Disponível em: <https://agenciabrasil.ebc.com.br/economia/noticia/2018-12/seca-atingiu-38-milhoes-de-brasileiros-em-2017-e-cheias-2-milhoes#:~:text=Seca%20atingiu%2038%20milh%C3%B5es%20de%20brasileiros%20em%202017%20e%20cheias%2C%202%20milh%C3%B5es,-Estudo%20da%20ANA&text=No%20ano%20passado%2C%20quase%2038,afetados%20por%20cheias%20e%20inunda%C3%A7%C3%B5es>. Acesso em: 25 nov. 2024.

[5] - FALTA de chuva pode elevar em mais de 10% a conta de luz: Em 2017, as hidrelétricas geraram 21% menos do que o volume de energia que tinham direito de comercializar. [*S. l.*]: Veja Negócios, 15 jan. 2018. Disponível em: <https://veja.abril.com.br/economia/falta-de-chuva-pode-elevar-em-mais-de-10-a-conta-de-luz>. Acesso em: 25 nov. 2024.

[6] - LUIZ FERREIRA DE OLIVEIRA, Bernardo; GRIPP SIMÕES ALVES, Luna; LINS GÓES DE CARVALHO, Juliana; GARCIA DE OLIVEIRA, Daniel; SANTOS DA SILVA, Joecila. **A seca de 2018 no alto rio negro e os impactos a população**. [*S. l.*]: Brazilian Journal of Development, 19 ago. 2019. Disponível em: <https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/2759>. Acesso em: 25 nov. 2024.

[7] - PLANO Safra disponibiliza R$ 340,8 bilhões para o setor agropecuário: Serão destinados R$ 53,61 bilhões para financiamento de pequenos produtores pelo Programa Nacional de Fortalecimento da Agricultura Familiar (Pronaf), com juros de 5% e 6% ao ano. [*S. l.*]: Ministério da Agricultura e Pecuária, 29 jun. 2022. Disponível em: <https://www.gov.br/agricultura/pt-br/assuntos/noticias-2022/plano-safra-disponibiliza-r-340-8-bilhoes-para-o-setor-agropecuario>. Acesso em: 25 nov. 2024.

[8] - PLANO Safra 2022/2023 anuncia R$ 340,8 bilhões para a agropecuária: Volume é 36% maior do que o plano anterior. [*S. l.*]: Agência Brasil, 29 jun. 2022. Disponível em: <https://agenciabrasil.ebc.com.br/economia/noticia/2022-06/plano-safra-20222023-anuncia-r-3408-bilhoes-para-agropecuaria#:~:text=Sustentabilidade,monitoramento%20da%20umidade%20no%20solo>. Acesso em: 25 nov. 2024.

[9] - PLANO Safra: Governo prevê R$ 400 bilhões em créditos para médios e grandes produtores rurais: Também nesta quarta, Lula anunciou quase 85,7 bilhões de reais em incentivos para pequenos agricultores. [*S. l.*]: Carta Capital, 3 jul. 2024. Disponível em: <https://www.cartacapital.com.br/economia/plano-safra-governo-preve-r-400-bilhoes-em-creditos-para-medios-e-grandes-produtores-rurais/#:~:text=Do%20total%2C%20quase%20240%20bilh%C3%B5es,como%20a%20compra%20de%20maquin%C3%A1rio>. Acesso em: 25 nov. 2024.

[10] - ENTENDA a crise dos caminhoneiros: Combustível em alta e falta de fiscalização de novo piso dos fretees provocam insatisfação. [*S. l.*]: Folha de São Paulo, 2 set. 2018. Disponível em: <https://www1.folha.uol.com.br/mercado/2018/09/entenda-a-crise-dos-caminhoneiros.shtml>. Acesso em: 25 nov. 2024.

[11] - RODRIGUES, Natália. **Greve dos Caminhoneiros (2018)**. São Paulo , SP - Brasil: InfoEscola, 2018. Disponível em: <https://www.infoescola.com/historia/greve-dos-caminhoneiros-2018/>. Acesso em: 25 nov. 2024.

[12] - GREVE dos caminhoneiros: como se formou o nó que levou à paralisação: Protestos põem em questão política de preços da Petrobras e modelo de exploração do petróleo. Limitações logísticas do transporte no Brasil também ajudam a entender a crise dos combustíveis. São Paulo , SP - Brasil: El País, 2018. Disponível em: <https://brasil.elpais.com/brasil/2018/05/24/economia/1527177800_693499.html>

. Acesso em: 25 nov. 2024.

[13] - GREVE dos caminhoneiros: a cronologia dos 10 dias que pararam o Brasil. São Paulo , SP - Brasil: UOL, 30 maio 2018. Disponível em: <https://economia.uol.com.br/noticias/bbc/2018/05/30/greve-dos-caminhoneiros-a-cronologia-dos-10-dias-que-pararam-o-brasil.htm>. Acesso em: 25 nov. 2024.