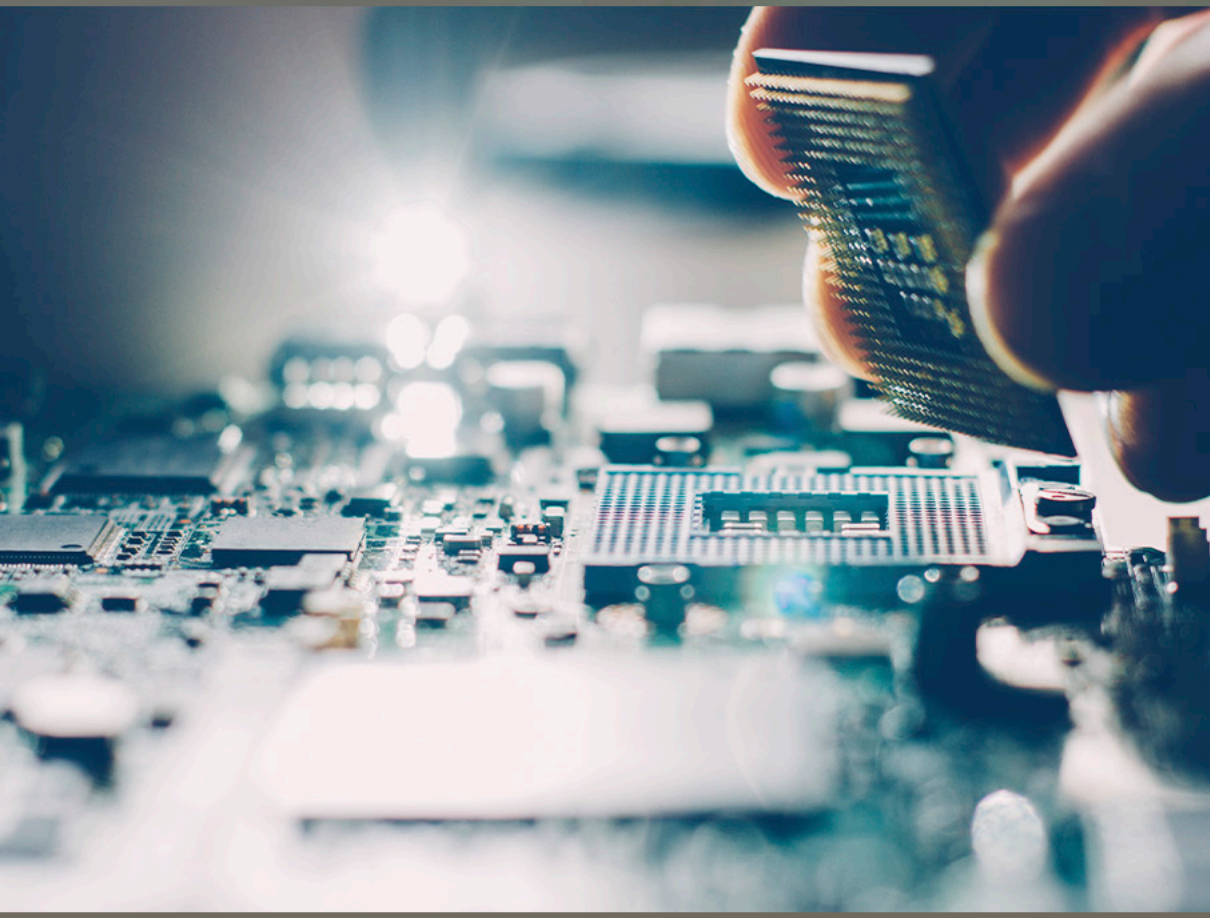
COLEÇÃO

DESAFIOS DAS ENGENHARIAS:

ENGENHARIA DE COMPUTAÇÃO 3



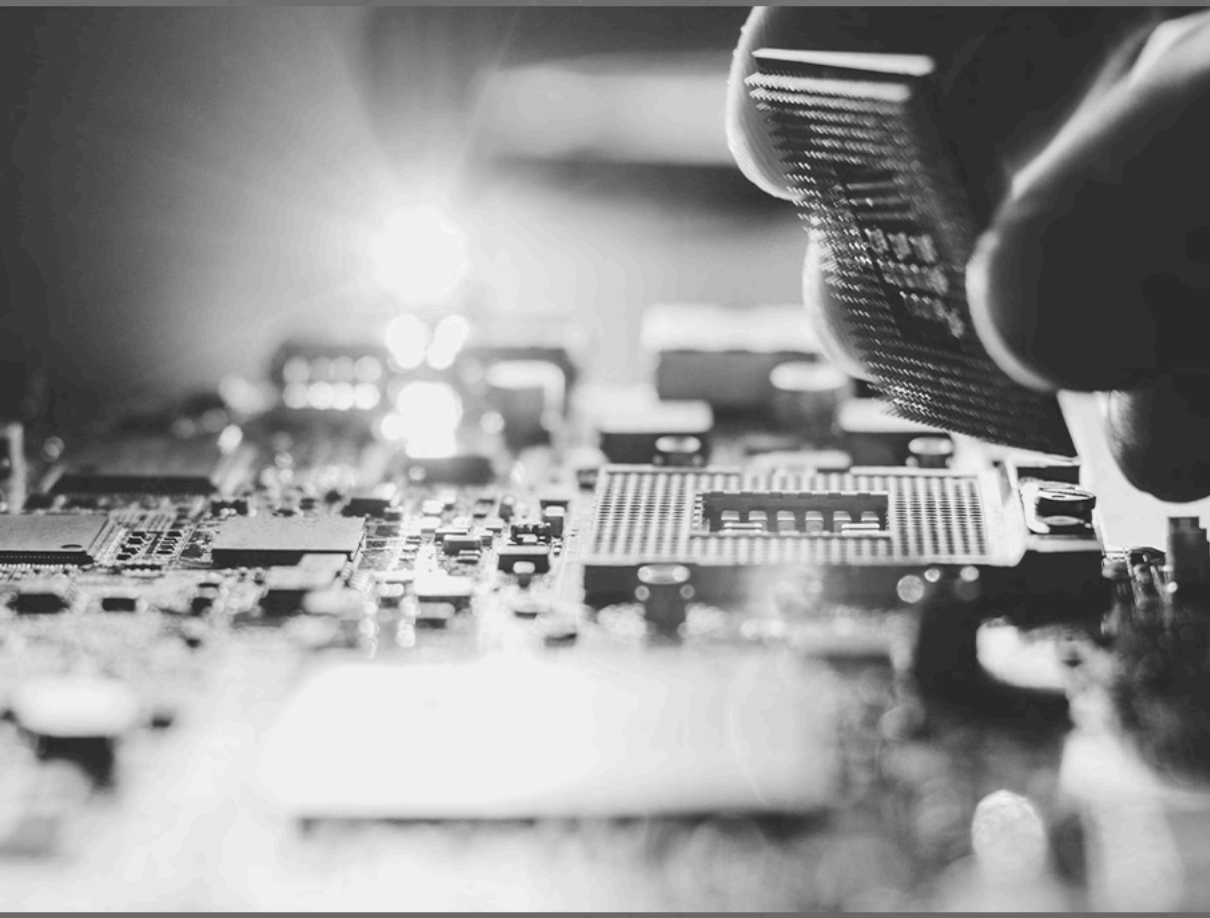
LILIAN COELHO DE FREITAS
(ORGANIZADORA)

Atena Editora
Ano 2021

COLEÇÃO

DESAFIOS DAS ENGENHARIAS:

ENGENHARIA DE COMPUTAÇÃO 3



LILIAN COELHO DE FREITAS
(ORGANIZADORA)

Atena Editora
Ano 2021

Editora chefe
Profª Drª Antonella Carvalho de Oliveira
Editora executiva
Natalia Oliveira
Assistente editorial
Flávia Roberta Barão
Bibliotecária
Janaina Ramos
Projeto gráfico
Camila Alves de Cremo
Daphynny Pamplona
Gabriel Motomu Teshima
Luiza Alves Batista
Natália Sandrini de Azevedo
Imagens da capa
iStock
Edição de arte
Luiza Alves Batista

Atena Editora
Ano 2021

Atena Editora
Ano 2021

Coleção desafios das engenharias: engenharia de computação 3

Diagramação: Daphynny Pamplona
Correção: Gabriel Motomu Teshima
Indexação: Amanda Kelly da Costa Veiga
Revisão: Os autores
Organizadora: Lilian Coelho de Freitas

Dados Internacionais de Catalogação na Publicação (CIP)

C691 Coleção desafios das engenharias: engenharia de computação 3 / Organizadora Lilian Coelho de Freitas. – Ponta Grossa - PR: Atena, 2021.

Formato: PDF
Requisitos de sistema: Adobe Acrobat Reader
Modo de acesso: World Wide Web
Inclui bibliografia
ISBN 978-65-5983-619-2
DOI: https://doi.org/10.22533/at.ed.192212911

1. Engenharia de computação. I. Freitas, Lilian Coelho de (Organizadora). II. Título.

CDD 621.39

Elaborado por Bibliotecária Janaina Ramos – CRB-8/9166

Atena Editora
Ponta Grossa – Paraná – Brasil
Telefone: +55 (42) 3323-5493
www.atenaeditora.com.br
contato@atenaeditora.com.br

Atena Editora
Ano 2021

Ciências Exatas e da Terra e Engenharias

DECLARAÇÃO DOS AUTORES

Os autores desta obra: 1. Atestam não possuir qualquer interesse comercial que constitua um conflito de interesses em relação ao artigo científico publicado; 2. Declaram que participaram ativamente da construção dos respectivos manuscritos, preferencialmente na: a) Concepção do estudo, e/ou aquisição de dados, e/ou análise e interpretação de dados; b) Elaboração do artigo ou revisão com vistas a tornar o material intelectualmente relevante; c) Aprovação final do manuscrito para submissão.; 3. Certificam que os artigos científicos publicados estão completamente isentos de dados e/ou resultados fraudulentos; 4. Confirmam a citação e a referência correta de todos os dados e de interpretações de dados de outras pesquisas; 5. Reconhecem terem informado todas as fontes de financiamento recebidas para a consecução da pesquisa; 6. Autorizam a edição da obra, que incluem os registros de ficha catalográfica, ISBN, DOI e demais indexadores, projeto visual e criação de capa, diagramação de miolo, assim como lançamento e divulgação da mesma conforme critérios da Atena Editora.

Atena Editora
Ano 2021

DECLARAÇÃO DA EDITORA

A Atena Editora declara, para os devidos fins de direito, que: 1. A presente publicação constitui apenas transferência temporária dos direitos autorais, direito sobre a publicação, inclusive não constitui responsabilidade solidária na criação dos manuscritos publicados, nos termos previstos na Lei sobre direitos autorais (Lei 9610/98), no art. 184 do Código Penal e no art. 927 do Código Civil; 2. Autoriza e incentiva os autores a assinarem contratos com repositórios institucionais, com fins exclusivos de divulgação da obra, desde que com o devido reconhecimento de autoria e edição e sem qualquer finalidade comercial; 3. Todos os e-book são *open access, desta forma* não os comercializa em seu site, sites parceiros, plataformas de *e-commerce,* ou qualquer outro meio virtual ou físico, portanto, está isenta de repasses de direitos autorais aos autores; 4. Todos os membros do conselho editorial são doutores e vinculados a instituições de ensino superior públicas, conforme recomendação da CAPES para obtenção do Qualis livro; 5. Não cede, comercializa ou autoriza a utilização dos nomes e e-mails dos autores, bem como nenhum outro dado dos mesmos, para qualquer finalidade que não o escopo da divulgação desta obra.

Atena Editora
Ano 2021

APRESENTAÇÃO

A Atena Editora tem a honra de presentear o público em geral com a série de *e-books* intitulada "*Coleção desafios das engenharias: Engenharia de computação*". Em seu terceiro volume, esta obra tem o objetivo de divulgar aplicações tecnológicas da Engenharia de Computação na resolução de problemas atuais, com o intuito de facilitar a difusão do conhecimento científico produzido em várias instituições de ensino e pesquisa do país.

Organizado em 20 capítulos, este volume apresenta temas como utilização de aprendizagem de máquina na avaliação de riscos de infeção por COVID-19; dispositivos automatizados para administração de remédios; comunicação científica apoiada por realidade aumentada; métodos de elementos finitos aplicados na análise de materiais para indústria aeronáutica; aplicações de processamento digital de imagens e de algoritmos genéticos; entre diversas outras aplicações da automação e do desenvolvimento de *software,* combinados para melhorar as atividades do nosso dia-a-dia.

Dessa forma, esta obra contribuirá para aprimoramento do conhecimento de seus leitores e servirá de base referencial para futuras investigações.

Os organizadores da Atena Editora, agradecem especialmente os autores dos diversos capítulos apresentados, parabenizam a dedicação e esforço de cada um, os quais viabilizaram a construção deste trabalho.

Boa leitura.

Lilian Coelho de Freitas

SUMÁRIO

CAPÍTULO 10

ANÁLISE DOS EFEITOS DA MÉTRICA DE DISTÂNCIA NA EXTRAÇÃO DE CONJUNTOS DE SIMILARIDADE

Data de aceite: 01/11/2021
Data de submissão: 06/08/2021

André Eduardo Alessi
Universidade Tecnológica Federal do Paraná
Pato Branco - Paraná
https://orcid.org/0000-0003-0268-9801

Bruno Duarte
Universidade Tecnológica Federal do Paraná
Pato Branco - Paraná
https://orcid.org/0000-0002-9750-6301

Ives Renê Venturini Pola
Universidade Tecnológica Federal do Paraná
Pato Branco - Paraná
https://orcid.org/0000-0001-7300-7535

Dalcimar Casanova
Universidade Tecnológica Federal do Paraná
Pato Branco - Paraná
https://orcid.org/0000-0002-1905-4602

Marco Antonio de Castro Barbosa
Universidade Tecnológica Federal do Paraná
Pato Branco - Paraná
https://orcid.org/0000-0001-9674-2348

RESUMO: O conjunto de similaridade é um conceito definido para tratar de forma natural dados complexos em sistemas de gerenciamento de banco de dados. Trata-se de um grupo de dados onde nenhum par de elementos são suficientemente similares entre si. O processo de extração de conjuntos de similaridade envolve vários procedimentos e variáveis, uma delas sendo a métrica de distância utilizada para comparar os dados. Neste artigo, foram feitos experimentos computacionais para extrair conjuntos de similaridade utilizando-se métricas de distância diferentes a fim de se fazer uma análise estatística para descobrir se a métrica de distância influencia o resultado da extração e qual métrica é mais indicada para cada caso, onde se concluiu que a escolha da métrica realmente influencia o resultado e, para situações onde se deseja extrair conjuntos de similaridade com menor tamanho possível é indicado a métrica cityblock e na situação contrária, indica-se a métrica Chebyshev.

PALAVRAS-CHAVE: Conjuntos de similaridade, grafos, dados complexos, métrica de distância, teste de hipótese.

ANALYSIS OF THE EFFECTS OF THE DISTANCE METRIC IN THE EXTRACTION OF SIMILARITY SETS

ABSTRACT: Similarity Sets are a concept defined to naturally manage complex data in database management systems. They are a group of data where no pair of elements are sufficiently similar between each other. The process to extract similarity sets contains many steps and variables, one of them being the distance metric used to compare the data. In this paper, computational experiments were made using different distance metrics with the objective to find, using statistical tests, if the distance metrics matters to the final result of the extraction and which metric is better for each case. It was concluded that the distance metrics indeed matter for the results, and for

situations where it's wanted to extract a similarity set with the lowest length it's better to use the cityblock metric, otherwise it's better to use the Chebyshev distance metric.
KEYWORDS: Similarity sets, graphs, complex data, distance metrics, hypothesis tests.

1 | INTRODUÇÃO

Dados complexos são cada vez mais utilizados e precisa-se armazená-los e manipulá-los com eficiência. O volume de imagens produzido por exames médicos cresceu exponencialmente, o que resulta em vários *Terabytes* de informação por dia em um hospital de médio porte, criando a necessidade de mecanismos mais eficientes para representação, armazenamento e busca destes dados (ZIGHED et al., 2009, p.113, 114).

Em muitos casos, a comparação de igualdade entre dados complexos não é útil, visto que a menor mudança resultaria em resposta negativa. Um exemplo a ser citado é o redimensionamento de uma imagem, cujo conteúdo continua igual ou semelhante, mas a estrutura dos dados acaba sendo diferente. É por isso que operações de igualdade não são muito úteis com dados complexos. Seria mais útil compará-los por *similaridade* (ZEZULA et al., 2006, p. 3).

Métricas de similaridade (e dissimilaridade) entre diferentes tipos de dados são pesquisadas há muito tempo. A similaridade entre *strings*, por exemplo, é abordada em diversos trabalhos, dentre os quais cabe citar o trabalho de Mukherjee (1989) com algoritmos sistólicos para determinar a similaridade entre duas *strings* como o maior comprimento de uma subsequência de caracteres de um dado par de *strings*. Entretanto, a métrica mais clássica é dada pela distância de Levenshtein. Nela, mede-se a dissimilaridade entre duas *strings* pelo número de operações de inserção, remoção e substituição de caracteres necessárias para torná-las idênticas (LEVENSHTEIN, 1965). O trabalho de Louza et al. (2019) apresenta dois algoritmos para computar métricas de similaridade baseadas na Transformação de Burrows-Wheeler (BWT). Benedetti et al. (2019) apresentam técnicas baseadas em análise de contexto e semântica para o cálculo de similaridade entre documentos de texto. Por fim, Liatsis et al. (2020) adaptaram características comumente utilizadas em visão computacional, como matriz de ocorrência simultânea e matriz de *run-length*, para calcular similaridade entre *strings*, onde as características propostas são puramente estatísticas e não são sensíveis ao idioma.

A busca por similaridade tem sido alvo de estudos e implementações em Sistemas de Gerenciamento de Banco de Dados (SGBDs). O trabalho de Bedo et al. (2018) propõe uma extensão baseada em similaridade para otimização de consultas, abordando otimizações físicas e lógicas. O trabalho de Barioni et al. (2009) possui grande relevância na pesquisa pela incorporação de similaridade na linguagem SQL, pois define predicados de consultas por similaridade, especifica ordem de precedência dos operadores de similaridade, e introduz o suporte para incorporação de consultas por similaridade em SQL. Em Kim (2020) realizou-

se a extensão de linguagens de consultas para permitir buscas por similaridade em SGBDs paralelizados.

Entretanto, apesar de todos estes esforços, o conceito matemático de conjuntos, base fundamental dos sistemas de banco de dados, não era considerado nas pesquisas por similaridade. De modo a respeitar esse fundamento matemático, deveria ser considerada a existência de elementos suficientemente similares em conjuntos de dados, de forma que eles não se repitam e os conceitos matemáticos de conjuntos, onde elementos iguais não se repetem, sejam respeitados.

O trabalho de Pola et al. (2015) introduz o conceito de *Conjuntos de Similaridade*, ou, em inglês, *Simsets*, um grupo de dados complexos onde elementos suficientemente similares não se repetem. Os autores, além de introduzirem os conceitos, terminologias e propriedades dos *Simsets*, propuseram também o algoritmo *Distinct* para extração de tais conjuntos de bases de dados, validando-o em bases reais e sintéticas. Ressalta-se que os conceitos apresentados em Pola et al. (2015) permitem a implementação natural nos SGBDs atuais.

Em trabalhos anteriores, estendeu-se a teoria dos *Simsets* para relações não simétricas. Foram propostos algoritmos para extração de conjuntos de similaridade assimétricos, baseados no método knn e em uma versão híbrida. Além disso, propôs-se uma variação do algoritmo apresentado em Pola et al. (2015), chamado *Asymmetric Distinct*, baseado no método guloso (ALESSI et al., 2021).

Este artigo apresenta a aplicação dos conceitos estabelecidos em Pola et al. (2015) e Alessi et al. (2021) em uma base de dados complexos real. Busca-se realizar a extração de *Simsets* da base de áudio GTZAN considerando quatro métricas de distância para o cálculo de similaridade: cityblock, euclideana, chebyshev e mahalanobis. Ao final dos experimentos, um tratamento estatístico é realizado nos resultados, para determinar se há diferença significativa no resultado pela escolha da métrica e, se sim, em qual contexto cada distância é melhor aplicada. Na Seção 2, será apresentado o processo passo a passo de extração de *Simsets.* Na Seção 3, serão explicadas as implementações computacionais realizadas para obter os resultados do artigo. Na Seção 4, os resultados obtidos serão apresentados e discutidos. Por fim, na Seção 5, o artigo será concluído.

2 | PROCESSO DE EXTRAÇÃO DE *SIMSETS*

O processo completo de extração dos conjuntos de similaridade é resumido na Figura 1:

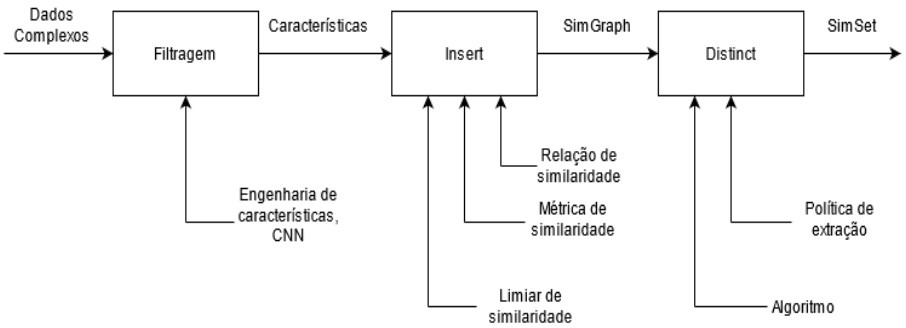


Figura 1 - Processo de extração dos conjuntos de similaridade.

Na primeira etapa, é realizado um procedimento de filtragem dos dados complexos, com a finalidade de levá-los para o espaço métrico. Isso normalmente é feito extraindo-se características, que podem variar de acordo com a natureza do dado. Isso pode ser feito utilizando-se engenharia de características ou um processo de convolução semelhante às Redes Neurais Convolucionais (CNN).

Em seguida, é feita a inserção dos elementos do conjunto de dados complexos original em um *grafo de similaridade* (*Simgraph*), um grafo direcionado onde cada nó representa um elemento do conjunto original e cada aresta representa uma relação de similaridade. A inserção no grafo de similaridade depende de três fatores:

1. **Relação de similaridade:** como os elementos serão comparados, e.g., por raio de abrangência, por k-vizinhos mais próximos (knn), um híbrido entre eles, etc.

2. **Métrica de similaridade:** como quantificar o quão similar os elementos são. De forma clássica isso pode ser feito com métricas de distância, porém outros métodos foram citados na introdução.

3. **Limiar de similaridade:** valor que define quando os elementos serão similares ou não, e.g., um raio de valor 1,5; os 3 vizinhos mais próximos, etc.

Para exemplificar o processo, considera-se a Figura 2. Na Figura 2(a) há um conjunto com 3 elementos, $\mathcal{P}0$, $\mathcal{P}1$, $\mathcal{P}2$, em um espaço métrico (no caso, o espaço euclidiano). Escolhe-se a relação de similaridade dos k-vizinhos mais próximos, com limiar de similaridade k = 1. A métrica utilizada é a distância euclidiana entre os elementos. Seguindo desta forma, na Figura 2(b), Figura 2(c) e Figura 2(d), respectivamente, o 1-vizinho mais próximo de é o de é e o de é Todas essas informações são inseridas no *Simgraph* da Figura 2(e).

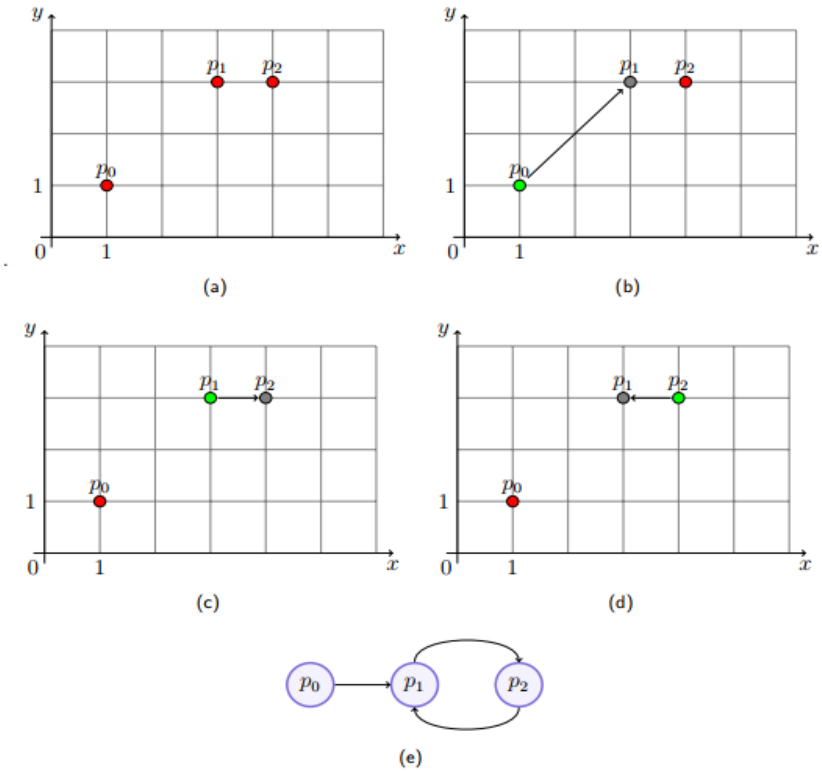


Figura 2 - Demonstração da operação Insert. Fonte: Alessi *et al.* (2021).

Por fim, o grafo de similaridade passa pelo processo *Distinct* para extração do conjunto de similaridade. Este conjunto contém apenas elementos distintos entre si, e é um conjunto dominante independente do grafo de similaridade. A forma como o Simset é extraído depende do Algoritmo utilizado e da política de extração. Em Alessi et al. (2021), por exemplo, foi definido o algoritmo *Asymmetric Distinct* que utiliza uma abordagem gulosa para extrair o Simset. Já a política de extração é uma condição adicional à extração. Existem diversos Simsets diferentes que podem ser extraídos de um Simgraph e essa condição visa reduzir essa quantidade, fazendo os conjuntos extraídos seguirem um padrão. Por exemplo, a política *Max* adiciona a condição de buscar uma extração onde os Simsets tenham a maior cardinalidade possível, e a política *min* busca o contrário, Simsets com a menor cardinalidade possível.

3 I DESENVOLVIMENTO

Todas as implementações computacionais deste trabalho foram realizadas na linguagem Python. Utilizou-se a base GTZAN[1], que consiste em 1000 arquivos de áudio com 30 segundos cada, divididos em 10 gêneros musicais com 100 músicas cada. Essa base é

1 <http://marsyas.info/downloads/datasets.html>

muito utilizada no ramo de processamento de áudio, em especial na classificação de músicas por gênero musical (TZANETAKIS; COOK, 2002).

Em processamento de áudio, o arquivo de áudio é dividido em *M* segmentos. Cada segmento é dividido em *N* janelas. Para cada janela é calculada uma sequência de características, chamadas *características a curto prazo*. Então, são calculadas estatísticas (normalmente a média e o desvio padrão) de cada sequência, resultando no vetor de *características a médio prazo* de cada segmento. Por fim, é feita uma média das características de cada segmento, gerando o vetor de *características a longo prazo*, representando o arquivo inteiro. Deve-se notar que o vetor final descarta a evolução temporal do áudio. Porém, essa forma de processamento tem atingido desempenho aceitável em vários problemas de processamento de áudio (GIANNAKOPOULOS; PIKRAKIS, 2014, p. 60 - 65). A Figura 3 contém um diagrama ilustrando o processamento descrito.

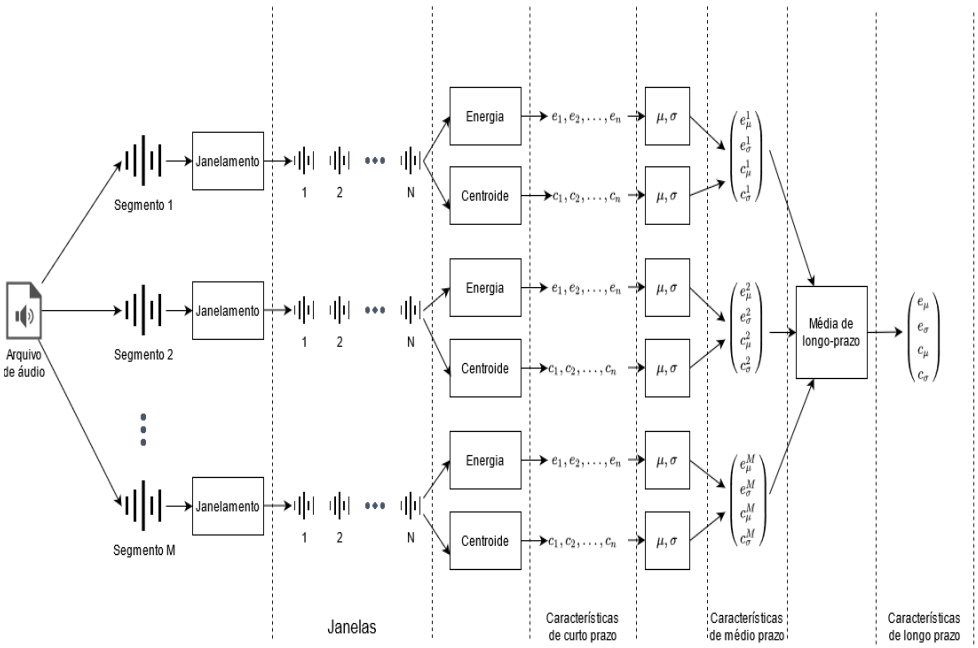


Figura 3 - Procedimento de extração de características de arquivos de áudio.

Realizou-se a extração das características de cada um dos mil arquivos de áudio da base GTZAN seguindo o procedimento da Figura 3, com o auxílio da biblioteca *pyAudioAnalysis* (GIANNAKOPOULOS, 2015). As características extraídas são mostradas no Quadro 1.

Como as características são extraídas a longo prazo para cada arquivo de áudio, para cada uma das 34 características obtém-se a sua média e desvio padrão, totalizando 68 descritores. Este processo de filtragem, seguindo o processo da Figura 1, resultou em características no espaço métrico distribuídas em uma tabela com 1000 linhas (para cada

arquivo de música) e 68 colunas (para cada descritor).

O próximo passo do processamento é a inserção no grafo de similaridade. São testadas três situações diferentes: relação de similaridade por raio de abrangência, por knn e por um híbrido entre raio de abrangência e knn. Em todas as situações é calculada uma matriz de distâncias entre os 1000 elementos que irão compor o grafo, utilizando 4 métricas de distância diferentes: cityblock, euclidiana, chebyshev e mahalanobis. Um mapa de calor de uma matriz de distância utilizando distância euclidiana pode ser visualizado na Figura 4. Percebe-se que a diagonal principal é composta por zeros, pois a distância de um elemento a si mesmo é zero, e a matriz é simétrica, pois a distância entre um elemento A e um elemento B é igual a distância do elemento B e o elemento A.

Índice	**Nome**	**Descrição**
1	Zero Crossing Rate	A taxa de mudança de sinal durante a duração de uma janela.
2	Energia	A soma dos quadrados dos valores do sinal, normalizada pelo comprimento da janela.
3	Entropia de energia	A entropia das energias normalizadas das subjanelas. Pode ser interpretada como a medida de mudanças abruptas.
4	Centróide espectral	O centro de gravidade do espectrograma.
5	Propagação espectral	O segundo momento central do espectrograma.
6	Entropia espectral	Entropia das energias espectrais normalizadas das subjanelas
7	Fluxo espectral	A diferença dos quadrados entre as magnitudes normalizadas de duas janelas consecutivas
8	Rolamento espectral	A frequência na qual abaixo dela está concentrada 90% da distribuição da magnitude do espectrograma.
9-21	MFCCs	Os Coeficientes Cepstrais da Frequência de Mel formam uma representação cepstral onde as frequências de banda não são lineares, mas sim distribuídas de acordo com a escala de Mel.
22-33	Vetor Chroma	Uma representação de 12 elementos da energia espectral.
34	Desvio do Chroma	O desvio padrão dos 12 elementos do Vetor Chroma

Quadro 1 - Características extraídas pela biblioteca pyAudioAnalysis. Fonte: Adaptado de Giannakopoulos (2015, p. 5).

Para cada relação de similaridade é definido um limiar de similaridade. No caso dos k-vizinhos mais próximos, é escolhido o limiar k = 3. Para o raio de abrangência, a diagonal principal e a parte triangular inferior da matriz de distâncias são desconsideradas. Em seguida, calcula-se a distribuição das distâncias e seleciona-se o primeiro quartil como limiar de similaridade. A distribuição de distâncias e o limiar selecionado são exemplificados na Figura 5, para a distância euclidiana. Para o método híbrido, seleciona-se k = 3 e o primeiro quartil da distribuição ao mesmo tempo.

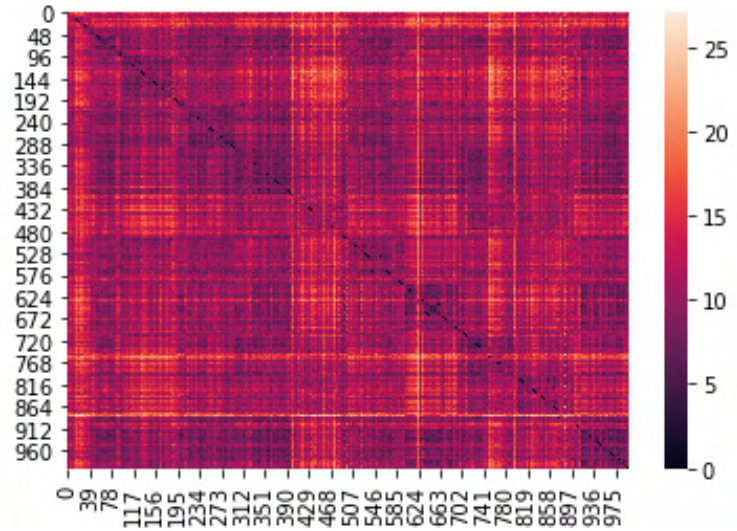


Figura 4 - Mapa de calor da matriz de distância calculada utilizando a distância euclidiana.

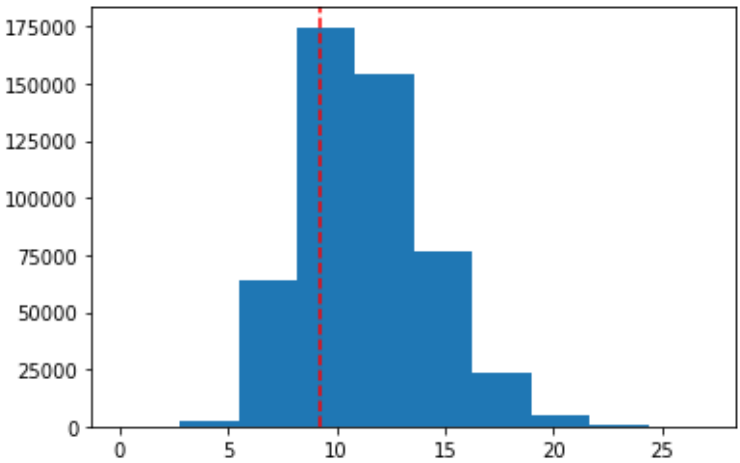


Figura 5 - Distribuição das distâncias utilizando distância euclidiana. A linha vermelha representa o primeiro quartil, que é utilizado como limiar de similaridade.

As implementações computacionais envolvendo grafos foram realizadas com o auxílio da biblioteca NetworkX (HAGBERG; SCHULT; SWART, 2008). Foram criados 24 grafos no total, pela combinação de 4 métricas de distância, 3 relações de similaridade e duas políticas de extração. Finalizando o processamento da Figura 1, cada *Simgraph* criado passa pelo algoritmo *Asymmetric Distinct* (ALESSI et al., 2021) tanto para a política *Max* quanto *min*. O algoritmo para cada política é executado 100 vezes e é gravado a cardinalidade do Simset extraído. Em seguida, são realizados os testes de hipótese ANOVA (CARSELLA; BERGER,

2018) e de Turkey (TURKEY, 1949) para responder às perguntas propostas na introdução.

4 I RESULTADOS E DISCUSSÕES

Ao executar o teste ANOVA com hipótese nula que não há diferença significativa entre a média da cardinalidade dos Simsets extraídos utilizando diferentes métricas de distância, resultou-se em um p-valor igual a 0, ou seja, recusa-se a hipótese nula e aceita-se a hipótese alternativa que há diferença entre as métricas utilizadas. Em sequência executa-se o teste de Turkey para cada par entre os 24 resultados da execução da extração. Para cada par, a hipótese nula é rejeitada. Isso permite concluir que a escolha da métrica de distância traz diferença significativa na cardinalidade do conjunto de similaridade extraído. Com esse resultado, pode-se analisar qual métrica de distância é melhor para cada caso. As Figuras de 6 a 11 mostram os *boxplots* das cardinalidades obtidas variando-se a métrica de distância. Percebe-se que, com exceção da Figura 9, de forma geral a cardinalidade do Simset extraído aumenta conforme a ordem cityblock - euclidiana - chebyshev - mahalanobis, com a última métrica apresentando resultados significativamente maiores. A princípio poderia-se afirmar que a distância cityblock é mais ideal para política *min* e a Mahalanobis para política *Max*, porém, considerando que esta última requer uma operação de inversão de matriz, ela é computacionalmente mais custosa do que as outras, então pode ser preferível a utilização da distância Chebyshev ao invés da Mahalanobis.

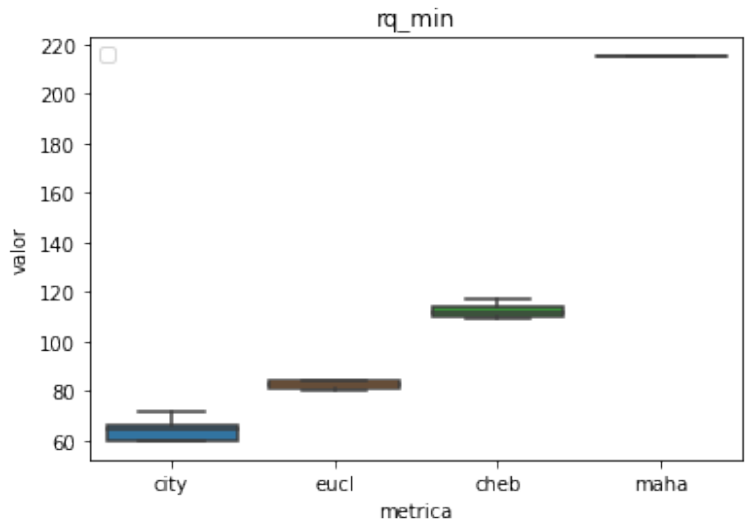


Figura 6 - Boxplot das cardinalidades dos Simsets extraídos para relação por raio de abrangência e política *min.*

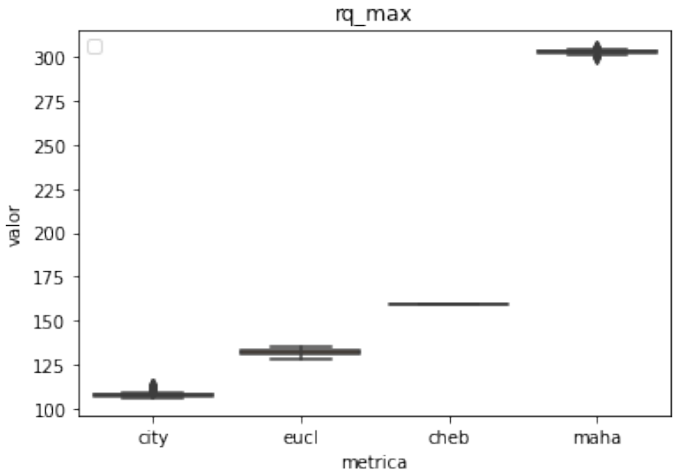


Figura 7 - Boxplot das cardinalidades dos Simsets extraídos para relação por raio de abrangência e política *Max*.

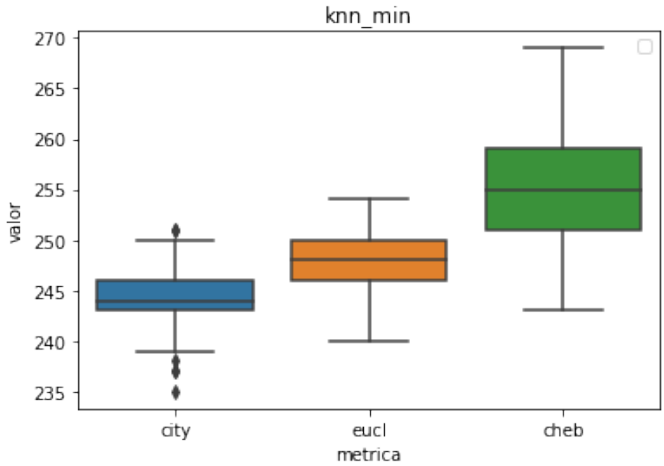


Figura 8 - Boxplot das cardinalidades dos Simsets extraídos para relação por k-vizinhos mais próximos e política *min*.

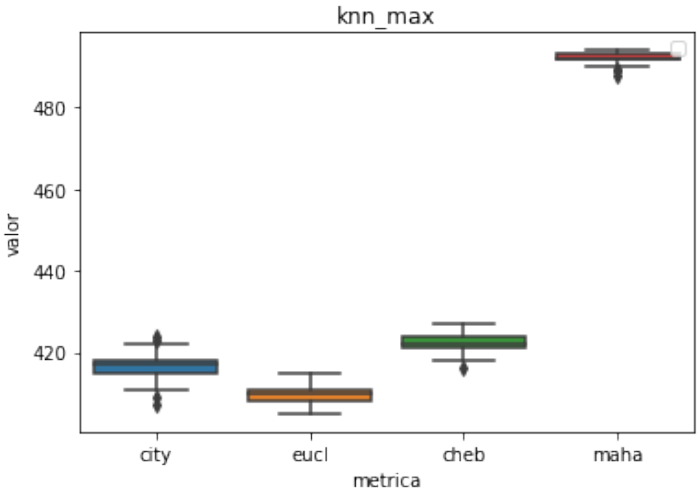


Figura 9 - Boxplot das cardinalidades dos Simsets extraídos para relação por k-vizinhos mais próximos e política *Max*.

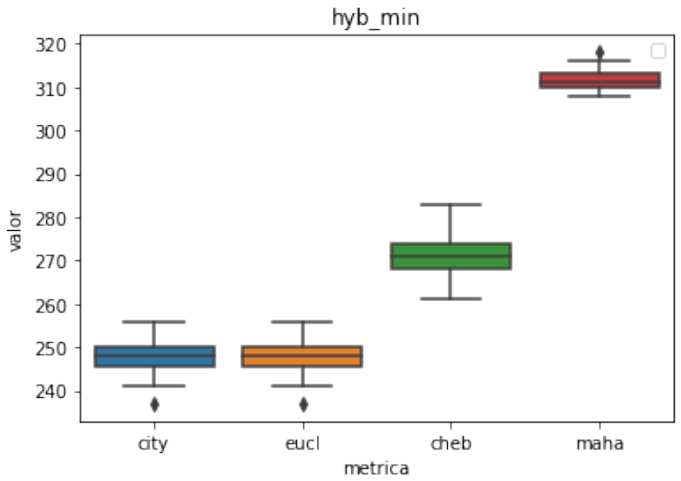


Figura 10 - Boxplot das cardinalidades dos Simsets extraídos para relação híbrida e política *min*.

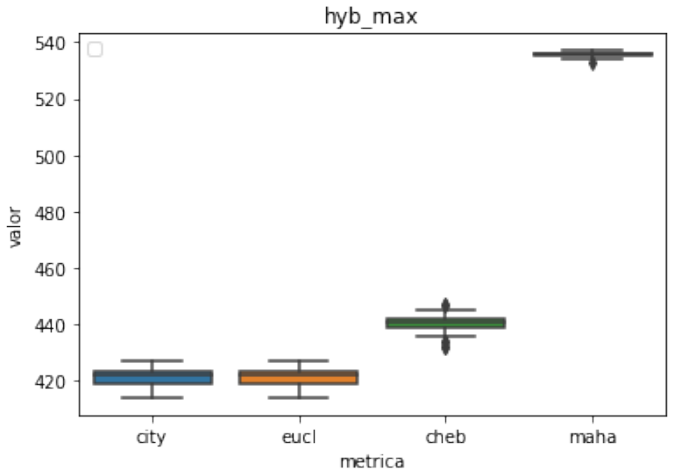


Figura 11 - Boxplot das cardinalidades dos Simsets extraídos para relação híbrida e política *Max*.

5 I CONCLUSÃO

O processo de extração de conjuntos de similaridade apresentado na Figura 1 está em constante aperfeiçoamento devido aos conceitos de Simset serem muito recentes e ainda terem pesquisas em andamento para determinar formas otimizadas de realizar as operações antes de implementar em um SGBD. Este trabalho buscou responder às perguntas: a métrica de distância influencia no resultado final da extração? Se sim, qual métrica é mais indicada para qual situação? Após realizar experimentos computacionais e testes estatísticos, chega-se à conclusão que sim, a métrica de distância importa e, para extrações com política *min*, onde deseja-se obter um Simset com menor cardinalidade possível, a distância cityblock é mais indicada, enquanto para a política *Max* indica-se a distância de Chebyshev.

REFERÊNCIAS

ALESSI, André E.; DUARTE, Bruno; POLA, Ives R.V.; BARBOSA, Marco A.C. Development of a novel similarity set extraction technique addressing non-symmetrical relations. **Brazilian Journal of Development,** v. 7, n. 8, p. 76163 - 76180.

BARIONI, M. et al. Seamlessly integrating similarity queries in SQL. **Software: Practice and Experience**, v. 39, p. 355 - 384, 2009.

BEDO, Marcos VN et al. The Merkurion approach for similarity searching optimization in Database Management Systems. **Data & Knowledge Engineering**, v. 113, p. 18-42, 2018.

BENEDETTI, F. et al. Computing inter-document similarity with context semantic analysis. **Information Systems,** v. 80, p. 136 - 147, 2019.

CASELLA, G.; BERGER, R. L. **Inferência Estatística**. São Paulo: Cengage Learning, 2018.

HAGBERG, A. A.; SCHULT, D. A.; SWART, P. J. Exploring network structure, dynamics, and function using networkx. In: **Proceedings of the 7th Python in Science Conference**. Pasadena, CA USA? [s.n.], 2008.

FINDLER, Nicholas V.; VAN LEEUWEN, Jan. A family of similarity measures between two strings. **IEEE transactions on pattern analysis and machine intelligence**, n. 1, p. 116-118, 1979.

GIANNAKOPOULOS, T. pyaudioanalysis: An open-source python library for audio signal analysis. **PloS one,** Public Library of Science, v. 10, n. 12, 2015.

GIANNAKOPOULOS, T.; PIKRAKIS, A. **Introduction to Audio Analysis: A MATLAB Approach**. UK: Elsevier, 2014.

KIM, Taewoo et al. Similarity query support in big data management systems. **Information Systems**, v. 88, p. 101455, 2020.

LEVENSHTEIN, V. I. Binary codes capable of correcting spurious insertions and deletions of ones. **Problems of Information Transmission**, Kluwer Academic Publishers, v. 1, p. 8 - 17, 1965.

LIATSIS, P. et al. Proposal and study of statistical features for string similarity computation and classification. **International Journal of Data Mining, Modelling and Management**, v. 12, p. 277, 2020.

LOUZA, Felipe A. et al. Algorithms to compute the burrows-wheeler similarity distribution. **Theoretical Computer Science**, v. 782, p. 145-156, 2019.

MUKHERJEE, Amar. Hardware algorithms for determining similarity between two strings. **IEEE Transactions on Computers**, v. 38, n. 4, p. 600-603, 1989.

POLA. I. et al. Similarity Sets: A new concept of sets to seamlessly handle similarity in database management systems. **Information Systems**, v. 52, p. 130 - 148, 2015.

TUKEY, J. W. Comparing Individual Means in the Analysis of Variance. **Biometrics**, vol. 5, no. 2, 1949, p. 99 – 114.

TZANETAKIS, G.; COOK, P. Musical genre classification of audio signals. **IEEE Transactions on Speech and Audio Processing**, v. 10, p. 293 - 302, 2002.

ZEZULA, P. et al. **Similarity Search - The Metric Space Approach.** New York: Springer, 2006.

ZIGHED, D. A. et al. **Mining Complex Data**. Heidelberg: Springer, 2009.

ÍNDICE REMISSIVO

P

R

S

T

U

V

W

COLEÇÃO

DESAFIOS DAS ENGENHARIAS:

ENGENHARIA DE COMPUTAÇÃO 3

www.atenaeditora.com.br
contato@atenaeditora.com.br
@atenaeditora
www.facebook.com/atenaeditora.com.br

Atena Editora
Ano 2021

COLEÇÃO

DESAFIOS DAS ENGENHARIAS:

ENGENHARIA DE COMPUTAÇÃO 3

www.atenaeditora.com.br
contato@atenaeditora.com.br
@atenaeditora
www.facebook.com/atenaeditora.com.br

Atena Editora
Ano 2021