



**Universidade de Brasília
Departamento de Estatística**

**SPOTIFY
AUTOMATIZAÇÃO DE PLAYLISTS**

João Victor Melo Corrêa Mendes

Projeto apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

**Brasília
2023**

João Victor Melo Corrêa Mendes

**SPOTIFY
AUTOMATIZAÇÃO DE PLAYLISTS**

Orientador(a): Prof. Eduardo Monteiro de Castro Gomes

Projeto apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

**Brasília
2023**

Agradecimentos

Primeiramente, gostaria de agradecer aos meus pais, Luciana Melo e João Luiz Mendes, e ao meu irmão, Felipe Melo, por todo apoio familiar que me deram. O incentivo e a educação que me passaram, foram essenciais para o meu amadurecimento e evolução dentro da vida acadêmica.

Aos amigos que a universidade me deu, Antônio Caio Cavalcante, Guilherme Silva, João Pedro Pupe e Lucas Menezes. Eles estiveram ao meu lado durante toda a jornada dentro do curso. Sou grato por todos os ensinamentos e trocas de ideias, as quais foram fundamentais para eu conseguir adquirir cada vez mais conhecimento. Obrigado também pelo suporte nos momentos difíceis, tanto pessoal quanto acadêmico.

Aos meus colegas da disciplina de Laboratório de Estatística 1 2.2022. Como o tema desse trabalho também foi abordado na matéria, as trocas de ideias e de conhecimentos dentro da sala de aula foram essenciais para o desenvolvimento do projeto. Obrigado por toda a ajuda que recebi.

À minha namorada, Julia Rossi Gonçalves, que sempre esteve ao meu lado durante a minha formação acadêmica, principalmente na reta final do curso. Sou muito grato por toda a ajuda moral e mental que recebi. Ela foi fundamental para que eu conseguisse concluir o trabalho da maneira mais saudável possível.

Aos meus professores e orientadores James Matos Sampaio e Eduardo Monteiro, que, acreditaram em mim no decorrer de todo o projeto. Obrigado por todos os ensinamentos e aprendizados durante essa caminhada.

Por fim, porém não menos importante, quero agradecer a todo o corpo docente do Departamento de Estatística da UnB. Desde quando eu entrei na universidade, no segundo semestre de 2018, sempre presenciei aulas incríveis com os melhores professores que eu poderia ter. Para sempre vou os ter em minha memória. Obrigado por tudo.

Muito obrigado a todos que participaram, de alguma maneira, da minha formação em Bacharelado em Estatística.

Resumo

O presente trabalho tem como objetivo identificar e analisar práticas de automatização de playlists no Spotify, bem como organizar e classificar as músicas de forma precisa, e criar um sistema de recomendação aos usuários. Para isso, foi utilizado um conjunto de dados oriundo da página Kwordb.net, o qual contém as músicas mais ouvidas do Spotify entre 2014 e 2022. Para a realização de todas as análises no estudo e criação das playlists, foram empregadas as técnicas estatísticas *K-Nearest Neighbors* (KNN) e Análise de Cluster, juntamente com uma análise de componentes principais (PCA). Além disso, para a apresentação dos resultados dos sistemas de recomendação musical, foi criada uma interface utilizando o Streamlit, a fim do usuário obter uma melhor visualização, e ainda, ser bastante interativo. Os resultados obtidos das playlists criadas, foram bastantes coesos e coerentes. Toda a implementação computacional foi construída utilizando as linguagens de programação Python e R.

Palavras-chaves: Cluster; K-Nearest Neighbors; Músicas; Playlist; Spotify; Streamlit; Usuário.

Abstract

The aim of this paper is to identify and analyze practices for automating playlists on Spotify, as well as organizing and classifying songs accurately, and creating a recommendation system for users. To do this, we used a dataset from the Kwordb.net website, which contains the most listened to songs on Spotify between 2014 and 2022. To carry out all the analyses in the study and create the playlists, the statistical techniques K-Nearest Neighbors (KNN) and Cluster Analysis were used, along with a principal component analysis (PCA). In addition, for the presentation of the results of the music recommendation systems, an interface was created using Streamlit, in order for the user to obtain a better visualization, and also to be very interactive. The entire computer implementation was built using the Python and R programming languages.

Keywords: Cluster; K-Nearest Neighbors; Music; Playlist; Spotify; Streamlit; User.

Lista de Tabelas

1	Medidas de Resumo	19
2	Notas Musicais	25
3	Assinatura de Compasso	26
4	Top 10 Músicas Populares	27
5	Correlações	29
6	Playlist 1	31
7	Playlist 2	32
8	Playlist 3	33

Lista de Figuras

1	Boxplots Dançabilidade, Energia e Discurso	20
2	Boxplots Acústico, Vivacidade e Valência	21
3	Boxplots Popularidade, Sonoridade, Tempo e Instrumentalidade	22
4	Notas Musicais	24
5	Time Signature	25
6	Top 10 músicas mais populares	27
7	Matriz de Correlação	28
8	Matriz de Correlação	29
9	Cluster Playlist 1 - Rock	35
10	Cluster Playlist 2 - HipHop	36
11	Cluster Playlist 3 - Pop	37
12	Streamlit - Introdução	38
13	Streamlit - KNN	39
14	Streamlit - Playlist	40

Sumário

1 INTRODUÇÃO	8
2 REFERENCIAL TEÓRICO	10
2.1 Análise de Cluster	10
2.2 K-Nearest Neighbors	11
2.3 Streamlit	12
2.4 Análise de Componentes Principais	13
2.5 Variável Dummy	13
3 METODOLOGIA	14
3.1 Tipo e descrição geral da pesquisa	14
3.2 Conjunto de dados	14
3.3 Instrumentos e procedimentos de coleta dos dados da pesquisa	16
3.4 Procedimento de análise dos dados	16
3.4.1 Análise Descritiva	16
3.4.2 K-Nearest Neighbors	16
3.4.3 Análise de Componentes Principais	17
3.4.4 Clusterização	17
3.5 Apresentação da Interface (Streamlit)	18
4 RESULTADOS	19
4.1 Análises Descritivas	19
4.2 Boxplots	20
4.2.1 Dançabilidade, Energia e Discurso	20
4.2.2 Acústico, Vivacidade e Valência	21
4.2.3 Popularidade, Sonoridade, Tempo e Instrumentalidade	22
4.3 Gráficos	24
4.3.1 Notas Musicais	24
4.3.2 Compasso	25

4.3.3 Popularidade	27
4.4 Matriz de Correlação.	28
4.5 K-Nearest Neighbors	30
4.6 Clusterização	34
4.6.1 K-means	34
4.7 Streamlit	37
5 CONCLUSÃO	41
REFERÊNCIAS	42
APÊNDICE	43

1 INTRODUÇÃO

O Spotify é um dos serviços de streaming de música e de podcast mais utilizados atualmente desde 2008, quando foi lançado por Daniel Ek e Martin Lorentzon (FLEISCHER; SNICKARS, 2017 apud MOSCHETTA; VIEIRA, 2018). A empresa foi fundada no ano de 2006, na Suécia, na cidade de Estocolmo. Segundo Daniel Ek, a ideia de criar o Spotify foi tentar minimizar o consumo da música de forma ilícita no mundo, uma vez que a indústria musical sofria bastante com a pirataria há algumas décadas (CANALTECH, 2017).

O aplicativo de streaming conta com duas opções de serviço: a versão paga e a versão gratuita. A versão paga, a qual é chamada de Spotify Premium, permite ao usuário ouvir músicas no modo offline, além de não transmitir nenhum tipo de publicidade e/ou propaganda durante o uso, o que não ocorre na versão gratuita. Atualmente, os planos do Spotify Premium são divididos em quatro categorias, e conforme o número de contas que cada plano comporta, o valor da assinatura aumenta. Os planos são: Individual e Universitário (uma conta apenas), Duo (duas contas) e Família (ate seis contas).

Atualmente, mais de 430 milhões de contas estão ativas no Spotify (CANALTECH, 2017), sendo quase que a metade (44,19%) de assinantes Premium. Além disso, dentro da plataforma, o usuário pode encontrar playlists já prontas ou criar suas próprias playlists, seja por gênero musical, artista, humor de momento (“mood”) ou apenas por estilos parecidos. Outrossim, é possível também que o usuário tenha uma biblioteca com as suas “músicas curtidas”. Nessa seção, ficaram armazenadas as músicas as quais o usuário mais se identificou e teve uma maior afinidade. Quando uma música recebe um like, ou seja, quando é curtida, o download da música é realizado automaticamente e o usuário poderá escutá-la no modo offline, caso a conta seja Premium.

Porém, como a quantidade de músicas disponíveis dentro do aplicativo atualmente é muito alta, a organização e classificação musical produzida manualmente por um usuário, a fim de criar suas próprias playlists, tende a ser uma tarefa bastante complicada (MOREIRA; TSUNODA, 2021).

Diante do conhecimento acerca da história do Spotify e da dificuldade apresentada ao usuário em relação a criação manual de suas próprias listas de reprodução, quais práticas e processos podem ser utilizados para a criação e automatização de playlists dentro da plataforma, a fim de otimizar o tempo dos criadores e facilitar a busca de músicas e artistas durante o uso do serviço?

O estudo a seguir tem como objetivos identificar e analisar práticas de automatização de playlists dentro do serviço de streaming Spotify, com o intuito de otimizar o tempo dos usuários e ainda, organizar e classificar as músicas, de acordo com características semelhantes. A automatização de playlists será realizada por métodos de Machine Learning, utilizando a técnica estatística KNN seguida de uma análise de componentes principais e clusterização. Além disso, a apresentação de toda a automatização do projeto será apresentada de maneira totalmente interativa, através de uma plataforma de interfaces, o Streamlit. Desse modo, essa pesquisa também tem como utilidade demonstrar, tanto para fins acadêmicos quanto para fins pessoais, que é possível otimizar o tempo de criação de uma playlist, e ainda sim ter um resultado eficaz.

No presente trabalho, foi realizada uma análise e pesquisa aprofundada sobre a automatização de playlists no Spotify, visando ampliar a compreensão acerca dos desafios dessa área de estudo. O referencial teórico, essencial para fundamentar nossa pesquisa, abordará os principais conceitos, teorias e estudos relacionados ao tema, proporcionando uma base sólida para a análise e interpretação dos dados. Os resultados obtidos serão apresentados de maneira clara e objetiva, promovendo uma reflexão sobre suas implicações. Por fim, a conclusão abordará as principais descobertas, contribuições para a área e possíveis direções para futuras pesquisas, consolidando assim a relevância e o impacto do estudo exibido.

2 REFERENCIAL TEÓRICO

2.1 Análise de Cluster

A principal ideia da análise de Cluster é classificar os objetos (variáveis) em grupos, de maneira que os itens do mesmo grupo sejam bastante semelhantes. E por outro lado, os grupos sejam diferentes em sua composição (KASSAMBARA, 2017 apud BATTISTI, 2019). Além disso, a agregação dos objetos é baseada na proximidade entre eles. A medida de proximidade mais comum utilizada nessa análise é a distância Euclidiana, que é calculada por meio da fórmula:

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

O método de clusterização utilizado foi o K-means. Este método classifica os objetos (ou variáveis) em diversos grupos, de maneira que a variação entre os clusters seja minimizada pela soma dos quadrados das distâncias Euclidianas entre os objetos e seus centroides (KASSAMBARA, 2017).

$$W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2 \quad (2)$$

Portanto, x_i é o ponto que pertence ao cluster C_k , e μ_k é a média do valor atribuído ao cluster. Cada x_i é inserido em um cluster, de forma que a soma dos quadrados da distância de x_i em relação a μ_k é mínima (KASSAMBARA, 2017).

A variação intra-cluster é definida por:

$$\sum_{k=1}^k W(C_k) = \sum_{k=1}^k \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

O processo de clusterização pelo método de K-means, por padrão, segue as seguintes etapas:

- 1) determinação prévia do número de clusters;
- 2) seleção aleatória dos objetos para determinar os valores centrais;
- 3) fixar as observações pela distância Euclidiana em relação aos centróides;
- 4) calcular a nova média dos valores dentro do cluster que foi definido;

5) minimizar a soma dos quadrados intra-cluster.

2.2 K-Nearest Neighbors

O algoritmo K-Nearest Neighbors (KNN) é um método estatístico não paramétrico de Machine Learning de agrupamento e classificação de objetos em diferentes categorias, o qual se baseia em suas características e na proximidade dos exemplos de treinamento para tomar decisões (TIMOTEO, 2023). Ele é considerado um algoritmo simples, mas pode ser bastante poderoso. A ideia central do KNN é classificar um ponto de dados com base na classe mais comum entre os seus k vizinhos mais próximos no espaço de características. Ou seja, o ponto que está mais perto de outros pontos de uma determinada categoria é provável que ele pertença a mesma.

O “K” em KNN significa o número de vizinhos que são considerados para a tomada de decisão. A título de exemplo, se o K é igual a 5, o cálculo irá verificar os cinco vizinhos mais próximos e conceder ao exemplo a classe mais frequente entre esses vizinhos (TIMOTEO, 2023). A escolha do valor de K é um elemento de suma importância para o desempenho do algoritmo.

Uma das vantagens da técnica KNN é que não é necessário realizar uma fase de treinamento muito complexa, uma vez que o algoritmo utiliza diretamente os exemplos de treino para as tomadas de decisão. Além disso, por ser um método não paramétrico, pode se ajustar a diferentes tipos de dados, situações e problemas (TIMOTEO, 2023).

Para a implementação do KNN é preciso definir a medida de distância utilizada para calcular a proximidade entre os pontos. Normalmente, a mais utilizada é a distância Euclidiana ($\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$), porém outras métricas também podem ser usadas dependendo da situação e do problema (TIMOTEO, 2023). Além disso, é fulcral normalizar os dados antes da implementação do algoritmo, a fim de evitar que as características com escalas distintas atrapalhem no cálculo da distância.

Segundo Belmonte (2023), o processo da implementação do algoritmo segue os seguintes passos:

- 1. Estruturação dos dados:** Nessa etapa, precedendo a aplicação do método, é necessário a preparação e divisão dos dados em conjunto de teste e treinamento. Esse processo também engloba a normalização das características das informações.

- 2. Cálculo da distância:** Após realizar a estruturação dos dados, é necessário calcular a distância entre o ponto desconhecido e os pontos conhecidos dentro do conjunto

de treinamento. As distâncias mais utilizadas são a Euclidiana ou a de Manhattan.

3. Seleção dos vizinhos mais próximos: No 3º passo, será preciso selecionar os K vizinhos mais próximos do ponto desconhecido. Os vizinhos serão utilizados para determinar a classe do ponto incógnito.

Portanto, o método K-Nearest Neighbors (KNN) se destaca como uma rica ferramenta no âmbito do Aprendizado de Máquina. Sua eficácia reside na oferta de uma abordagem flexível e intuitiva para lidar com problemas de classificação. Ao identificar os K vizinhos mais próximos a um ponto desconhecido, o KNN realiza a classificação desse ponto com base na maioria dos vizinhos. Essa técnica revela-se versátil, encontrando aplicabilidade em diversos contextos, desde o reconhecimento de padrões até a recomendação de produtos. Compreendendo plenamente o funcionamento e o potencial do algoritmo KNN, é viável integrá-lo de maneira efetiva em projetos de Machine Learning (BELMONTE, 2023).

2.3 Streamlit

O Streamlit é uma biblioteca de código aberto em Python projetada para facilitar a realização rápida de aplicativos web interativos. Os comandos simplificam significativamente o processo de desenvolvimento, permitindo que os desenvolvedores transformem facilmente scripts em aplicativos web funcionais (AWARI, 2023).

O principal objetivo do Streamlit é permitir que os cientistas de dados e desenvolvedores construam interfaces interativas para o usuário com apenas algumas linhas de código. Além disso, oferece um amplo conjunto de widgets (elementos de interação), como botões, gráficos e tabelas, que podem ser facilmente incorporados no aplicativo. Outrossim, a atualização automática dos componentes torna a visualização de dados mais dinâmica e interativa, valorizando as análises exploratórias e as apresentações de dados (AWARI, 2023).

Além de simples, o Streamlit é bastante flexível. Os usuários podem integrar facilmente bibliotecas populares, como Pandas, por exemplo, para aprimorar as capacidades de visualização e análise (AWARI, 2023). Além disso, o framework é extensível, permitindo a criação de componentes personalizados para atender a necessidades e ideias específicas de aplicativos.

Por mais que a aplicação da biblioteca esteja voltada para a ciência de dados, o Streamlit é adequado para uma variedade de casos de uso, incluindo painéis interativos,

demonstrações de produtos e de aplicativos.

Por conseguinte, o Streamlit é uma ótima ferramenta para criar aplicativos web interativos em Python. A biblioteca simplifica o processo de desenvolvimento, permitindo que os provedores foquem na criação de boas experiências para seus usuários, sem se preocupar excessivamente com a complexidade técnica.

2.4 Análise de Componentes Principais

A Análise de Componentes Principais (PCA) é uma técnica estatística utilizada para simplificar e reduzir a dimensionalidade em conjuntos de dados complexos. Além disso, esse método ajuda a reduzir a quantidade de variáveis em um conjunto de dados, mantendo a maior parte da sua variabilidade. Isso é útil especialmente quando lidamos com conjuntos de dados com muitas características, o que pode levar a problemas de dimensionalidade.

2.5 Variável Dummy

Uma variável dummy, também conhecida como variável indicadora ou variável binária, é uma variável que assume apenas dois valores distintos, geralmente 0 ou 1 (Falso ou Verdadeiro). Ela é utilizada para representar categorias qualitativas ou atributos categóricos em modelos estatísticos ou de aprendizado de máquina, onde variáveis categóricas não podem ser diretamente utilizadas.

3 METODOLOGIA

Neste capítulo, serão apresentadas as técnicas e metodologias utilizadas a fim de operacionalizar a pesquisa. O conteúdo está dividido entre as seguintes partes: tipo e descrição geral da pesquisa, conjunto de dados, instrumentos da coleta de dados da pesquisa, procedimento de coleta de dados da pesquisa, análise dos dados coletados e apresentação da interface.

3.1 Tipo e descrição geral da pesquisa

O seguinte estudo consiste em uma pesquisa científica aplicada, uma vez que, segundo Gerhardt e Silveira (2009) é possível obter conhecimentos para uma aplicação prática, a fim de chegar a uma solução de uma problema específico.

Além disso, para complementar a pesquisa, os dados os quais serão coletados são secundários, visto que os mesmos são oriundos de fontes já disponíveis.

3.2 Conjunto de dados

O banco de dados utilizado no estudo para demonstração, bem como a realização de análises e resultados, foi obtido a partir das músicas disponíveis na página Kwordb.net, cujo o link do site está disponível nas referências ao final do projeto (KWORB, 2022). Essa página contém informações das músicas mais ouvidas (total de streams) no Spotify, entre o período de 10/08/2014 à 22/05/2023. No total são 9224 músicas presentes, porém, neste trabalho, foram utilizadas 8625. As variáveis presentes nessa base de dados são: artista, título e total de streams.

Após obter o banco de dados, as variáveis citadas acima foram mescladas com as variáveis que caracterizam as músicas, como por exemplo, ID, gênero, popularidade, energia, tempo, dançabilidade, valência, entre outras.

É importante ressaltar a dificuldade encontrada para gerar o banco de dados. Desde a parte da importação até a fase final de tratamento, cada detalhe e ajuste foi de suma importância. Ao final de todo o tratamento da base, as informações estavam devidamente organizadas e prontas para serem analisadas. Além disso, foram utilizadas as linguagens de programação R e Python para a obtenção dos dados.

Com isso, as variáveis presentes no conjunto de dados são as seguintes:

• ID: contém o id da música dentro da plataforma de streaming Spotify. Quando copiado no navegador do usuário, o mesmo é redirecionado para o aplicativo e poderá ouvir a faixa.

• Artista: nome do artista principal ou da banda de cada música.

• Música: nome da música.

• Gêneros: principais gêneros acerca dos artistas ou da banda. É considerável ressaltar que os gêneros contidos nessa variável estão associados com os artistas/bandas e não com a música. (Ao longo do trabalho essa variável foi dividida em “genero_principal”, que contém o principal gênero do artista/banda)

• Álbum: nome do álbum em que cada música está inserida.

• Key: nota musical.

• Duração: duração de cada faixa em minutos.

• Popularidade: indica o quão popular uma música é. Assume valores de 0 a 100. Quanto mais próximo de 100, mais popular será essa faixa.

• Tempo: batidas por minuto (BPM) de cada faixa.

• Time_Signature: assinatura de tempo ou compasso. Indica a organização rítmica de uma música. Os valores estão entre 1 e 5. O número representa uma batida por compasso.

• Sonoridade: é o volume total de uma faixa em decibéis (dB).

As variáveis abaixo apresentam valores entre 0 e 1.

• Acústico: identifica a presença de elementos acústicos em uma música. É representada por valores entre 0 e 1. Quanto mais perto de 1, mais acústica será a música.

• Discurso: detecta dentro da música, a proporção de palavras que são ditas na faixa. Valores acima de 0.66 descrevem faixas que provavelmente são feitas inteiramente de falas, geralmente são os casos de podcasts. Valores entre 0.33 e 0.66 descrevem faixas que podem conter música e fala. Valores abaixo de 0.33 provavelmente representam músicas e outras faixas não relacionadas à fala;

• Vivacidade: representa a detecção de audiência e público nas músicas. Valores acima de 0.8 indicam que a música é ao vivo;

• Instrumentalidade: indica a presença de instrumentos presentes na música. Músicas com baixa instrumentalidade, possuem predominantemente vocais.

- Energia: indica o nível e o percentual de intensidade e atividade da música. Quanto mais alto o valor da energia de uma música, mais rápida, animada e barulhenta será;

- Dançabilidade: mostra o quanto uma música é “dançável”, de acordo com o ritmo, estabilidade e quantidade de batidas;

- Valência: expressa a positividade contida na música. Faixas mais alegres e eufóricas terão valores mais altos do que músicas tristes e raivosas;

3.3 Instrumentos e procedimentos de coleta dos dados da pesquisa

Primeiramente, é utilizada a linguagem de programação Python para obter as informações contidas na página Kwordb.net. Em seguida, ainda operando pelo Python, foram acrescentadas as variáveis de métrica que caracterizam as faixas, pois o banco de dados não possuía essas observações, as quais serão necessárias para realizar as análises descritivas e de cluster e criar as playlists.

Ademais, é utilizado também o software estatístico RStudio para estruturar e organizar os dados da melhor forma possível. No R, foi instalado e carregado o pacote “Spotifyr”, o qual funciona como um wrapper da Web API do Spotify, ou seja, tem a finalidade de fornecer informações sobre músicas, artistas e usuários no Spotify. (ESTATMG, 2021). O Python também oferece o mesmo pacote com a mesma funcionalidade, chamado “Spotipy”.

3.4 Procedimento de análise dos dados

3.4.1 Análise Descritiva

Foi realizada uma análise descritiva e exploratória das variáveis que caracterizam as músicas, bem como uma tabela com as medidas de resumo, boxplots e gráficos.

3.4.2 K-Nearest Neighbors

Com o conjunto de dados já estruturado, preparado e carregado, a técnica estatística KNN é aplicada, a fim de agrupar as músicas semelhantes e gerar um sistema de

recomendação (playlist).

A linguagem de programação escolhida para a implementação de todo o algoritmo foi o Python. Com isso, as variáveis do banco de dados utilizadas para realizar o modelo foram: gêneros, key, popularidade, acústico, discurso, vivacidade, instrumentalidade, energia, tempo, time_signature, sonoridade, dançabilidade e valência.

Primeiramente, foi necessário criar um novo conjunto de dados e transformar as variáveis “gêneros” e “key” em variáveis dummies, uma vez que estas são variáveis categóricas. As demais variáveis, as quais caracterizam as faixas, são numéricas e não foi necessário modificá-las.

Com as variáveis transformadas e inseridas no novo banco, os dados foram normalizados para que todas as informações fiquem na mesma escala e não dominem o cálculo da distância. O pacote utilizado para a padronização foi o *StandardScaler* oriundo da biblioteca *Sklearn*.

Na aplicação do método, a fim de calcular a distância entre a música selecionada e o seus vizinhos, a técnica utilizada foi a padrão do algoritmo. Além disso, a seleção dos vizinhos mais próximos foi igual a 30. Ou seja, a playlist gerada ao final da implementação da técnica terá 30 músicas (1 escolhida e 29 semelhantes).

Após a implementação da metodologia e já realizado o ajuste final, o modelo retorna o número da posição no banco de dados inicial de cada música semelhante selecionada. Para obter a playlist, foi utilizado o index dessas músicas para concatenar com as informações restantes (nome da música, artista, id, variáveis de métrica, etc).

3.4.3 Análise de Componentes Principais

Antes de realizar a clusterização, foi implementado nos dados padronizados a técnica de análises de componentes principais, no intuito de reduzir o número de variáveis explicativas e obter um melhor resultado.

3.4.4 Clusterização

Em seguida, foi realizada uma análise de Cluster para classificar e separar as músicas dentro da playlist criada, e ainda, verificar se existe uma semelhança ainda maior entre elas. O método de clusterização que será utilizado é o K-means.

3.5 Apresentação da Interface (Streamlit)

Para a apresentação do aplicativo criado para fornecer as playlists, foi utilizada a biblioteca Streamlit no Python.

Primeiramente, foram carregados os pacotes necessários utilizados no script. Além disso, foram criados alguns containers iniciais para intitular o aplicativo. Ou seja, o site e o aplicativo receberam um nome. Um container no Streamlit é utilizado para agrupar códigos de modo que o esboço gerado fique acoplado em uma área específica da interface.

Em seguida, novos containers foram criados para inserir todo o código utilizado na implementação da técnica K-Nearest Neighbors. Dentro desses novos containers contém funções as quais tem o papel de inserir alguns “widgets”, como por exemplo, uma caixa de seleção ou uma entrada de texto. Os widgets são aplicados também para obter uma maior interação do usuário com a interface.

4 RESULTADOS

Neste capítulo, serão apresentados os resultados que foram adquiridos através do conjunto obtido na metodologia acima, bem como análises descritivas, KNN e análise de cluster. É importante ressaltar que as variáveis analisadas a seguir são oriundas da página Kworb.net. Esse banco foi utilizado para realizar as análises e também para a apresentação do aplicativo criado.

4.1 Análises Descritivas

Primeiramente, realizamos um estudo descritivo das variáveis que caracterizam as músicas, as quais foram citadas no item 3.3, no conjunto de dados do projeto. Portanto, pela Tabela 1 apresentada abaixo, é possível observar os resultados das medidas de resumo dessas variáveis.

Tabela 1: Medidas de Resumo

	Variável	Mínimo	Q1	Mediana	Média	Q3	Máximo	DP
1	Dançabilidade	0.07	0.56	0.68	0.67	0.78	0.99	0.15
2	Energia	0.00	0.53	0.66	0.64	0.77	0.99	0.18
3	Discurso	0.02	0.04	0.07	0.13	0.17	0.97	0.12
4	Acústico	0.00	0.04	0.13	0.23	0.35	0.99	0.25
5	Vivacidade	0.02	0.10	0.13	0.18	0.23	0.99	0.14
6	Valência	0.03	0.31	0.47	0.48	0.65	0.98	0.23
7	Popularidade	0.00	52.00	60.00	59.57	68.00	100.00	12.19
8	Sonoridade	-34.48	-7.19	-6.09	-6.52	-4.77	1.51	2.70
9	Tempo	40.32	98.05	121.00	122.09	141.98	216.33	25.18
10	Instrumentalidade	0.00	0.00	0.00	0.17	0.00	0.96	0.09

Conforme a Tabela 1, a respeito dos resultados das medidas de resumo, é válido destacar a variável Popularidade que apresentou o valor de 52 para o primeiro quartil e uma média de 59.57, o que significa que as músicas estão bem distribuídas. Além disso, a variável Discurso apresentou um valor de máximo (0.97) bem distante do 3º quartil (0.17). Esse fator significa que as músicas presentes no banco de dados possuem baixa proporção de palavras que são ditas nas faixas. Esse mesmo fator ocorre nas variáveis Vivacidade e Acústico, uma vez que os valores máximos apresentados são ambos de 0.99, enquanto os 3ºs quartis são de 0.18 e 0.35, respectivamente. Ou seja, a maioria das faixas presentes no conjunto de dados, não são ao vivo com a presença de público e também não

possuem muita acústica. Todas as variáveis, com exceção da Sonoridade, possuem uma alta discrepância entre os valores de mínimo e máximo.

Ademais, nas subseções a seguir, será possível analisar as variáveis individualmente de uma forma mais precisa através dos Boxplots.

4.2 Boxplots

4.2.1 Dançabilidade, Energia e Discurso

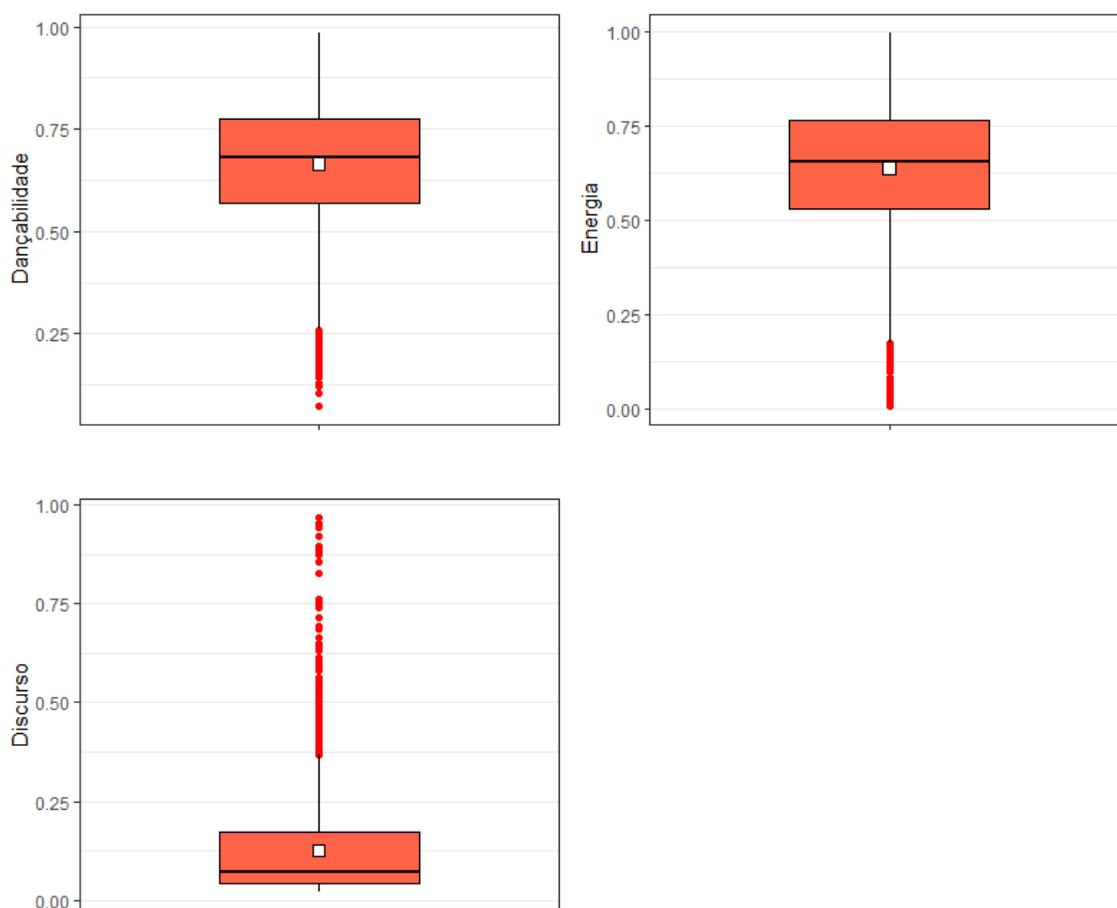


Figura 1: Boxplots Dançabilidade, Energia e Discurso

Analisando os boxplots das variáveis Dançabilidade e Energia, os quais são bem semelhantes, pode-se perceber uma pequena dispersão entre os dados, uma vez que os valores do 1^o e 3^o quartil estão próximos. Além disso, apresentam alguns outliers com

valores baixos. E ainda, ambas as medianas se encontram bem equilibradas entre os quartis.

Entretanto, o boxplot da variável Discurso, apresenta valores bem baixos e possui dados assimétricos positivos, uma vez que a linha da mediana está mais próxima do 1º quartil. Os outliers apresentados nessa variável são de valores positivos. Ou seja, como visto na tabela das medidas de resumo, músicas presentes no banco de dados possuem baixa proporção de palavras que são ditas nas faixas.

4.2.2 Acústico, Vivacidade e Valência

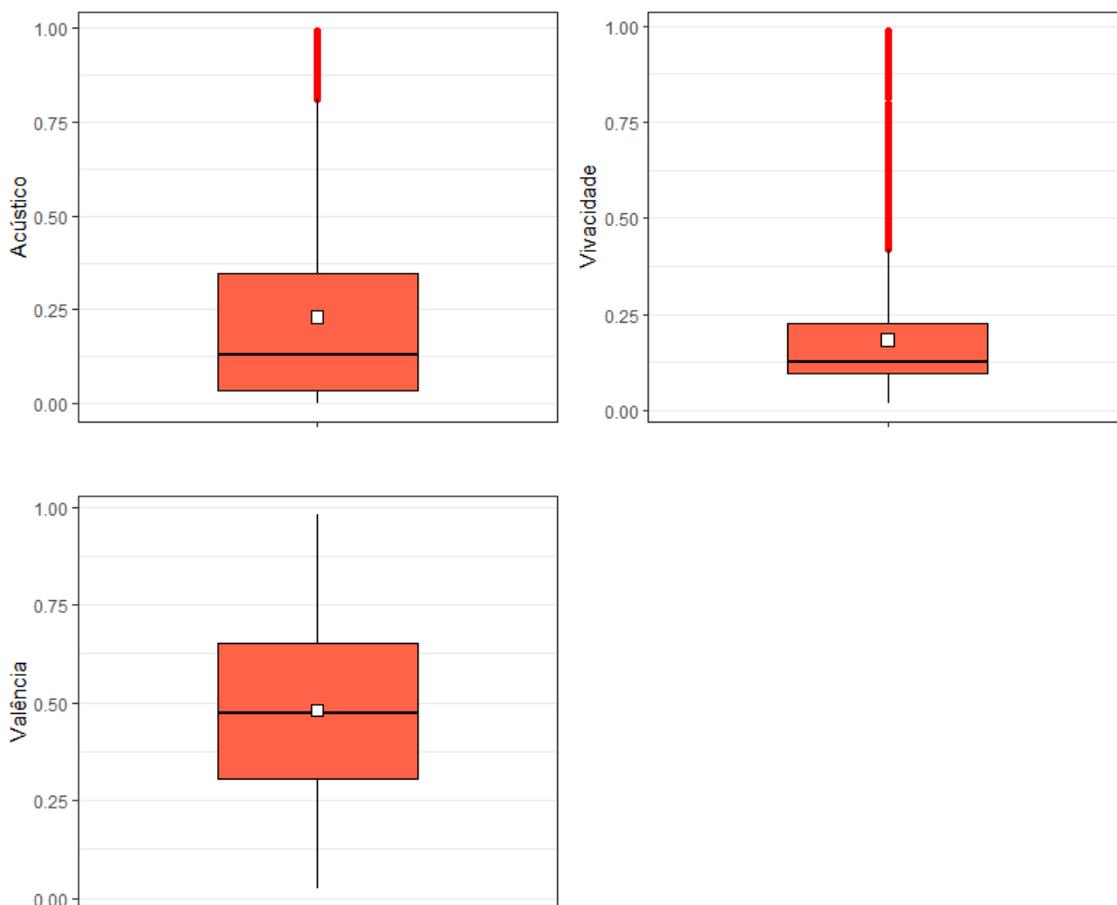


Figura 2: Boxplots Acústico, Vivacidade e Valência

Observando o boxplot da variável Acústico, os dados apresentados estão bem dispersos e assimétricos positivos. A linha mediana se encontra mais próxima do 1º quartil.

No entanto, no boxplot da variável Vivacidade, é importante observar que foram apresentados muitos outliers com valores acima do 3º quartil. Além disso, a linha media está mais próxima do 1º quartil, o que significa que os dados são assimétricos positivos.

Por fim, analisando o boxplot da variável Valência, os dados apresentados estão bastante equilibrados e simétricos. A linha mediana, juntamente com a média, se encontram bem ao centro da caixa, que indica a presença de normalidade. Além disso, não apresenta outliers.

4.2.3 Popularidade, Sonoridade, Tempo e Instrumentalidade

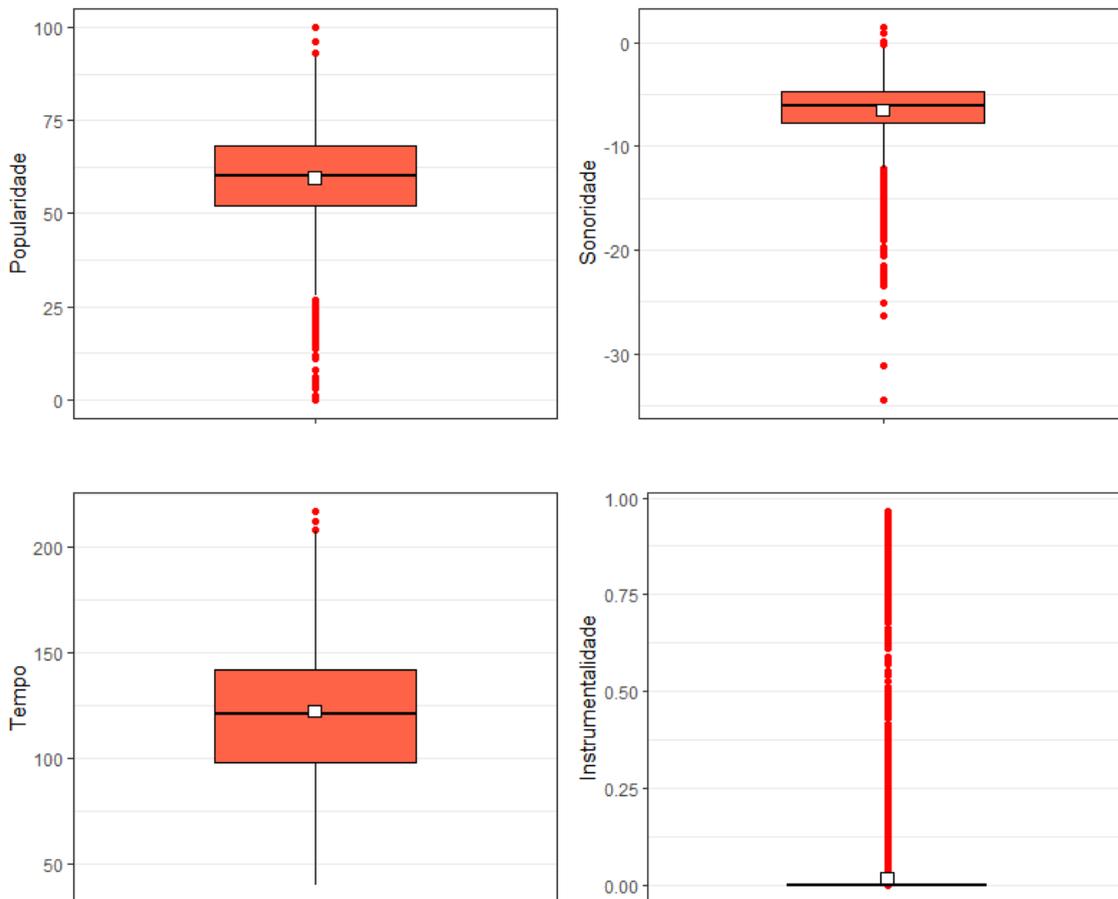


Figura 3: Boxplots Popularidade, Sonoridade, Tempo e Instrumentalidade

Verificando o boxplot da variável Popularidade, os dados se encontram pouco dispersos e equilibrados. Há presença de outliers com valores abaixo do 1^o quartil e acima do 3^o quartil. A mediana e a média apresentam valores bem semelhantes.

Acerca do boxplot da variável Sonoridade, a maioria dos outliers apresentados possuem valores negativos, ou seja, as músicas presentes no banco de dados são menos barulhentas.

Analisando o boxplot da variável Tempo, é notório que os dados estão bem simétricos e equilibrados, com média e mediana bem ao centro da caixa. Apresentou apenas 3 outliers com valores perto do máximo.

Por fim, tendo em vista o boxplot da variável Instrumentalidade, os dados apresentados possuem o valor 0 em quase que 100% das músicas. Ou seja, a dispersão entre os dados é quase nula.

4.3 Gráficos

4.3.1 Notas Musicais

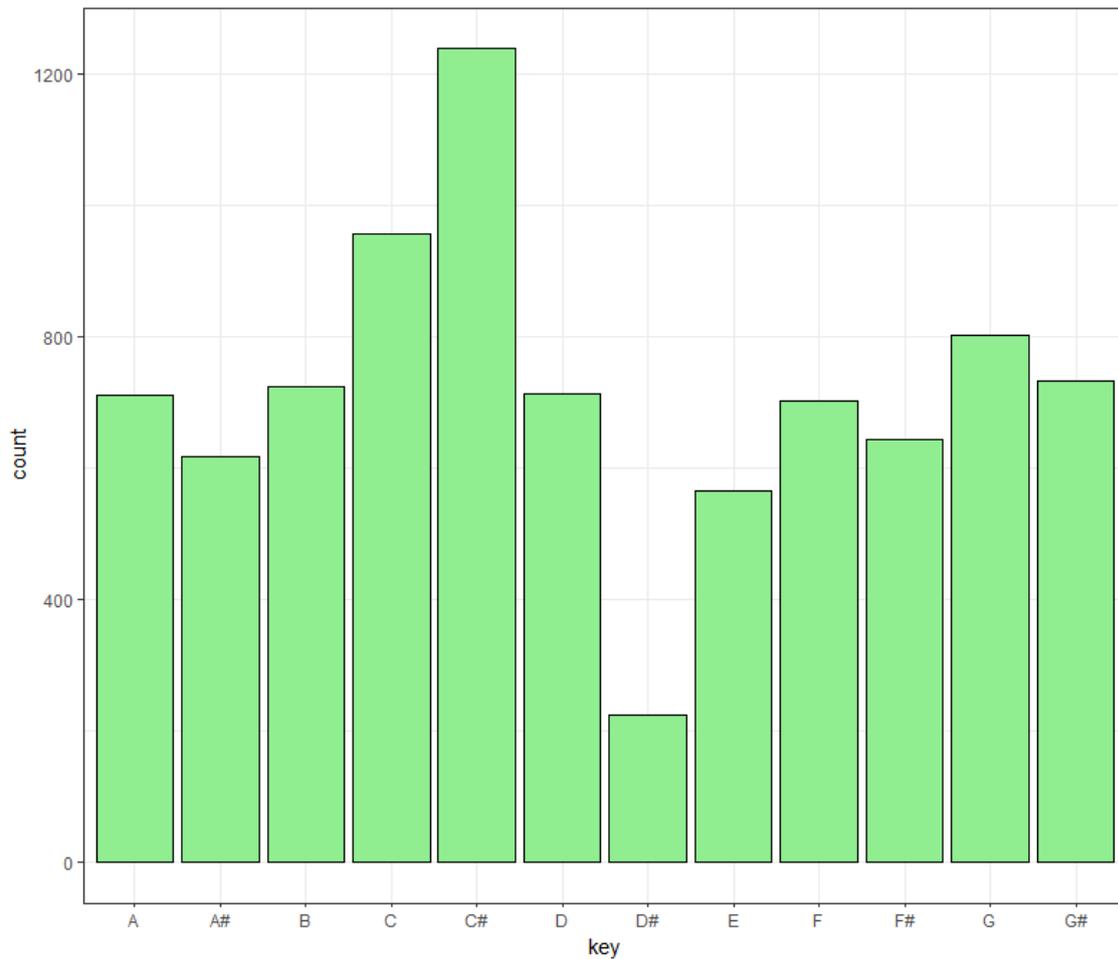


Figura 4: Notas Musicais

Observando a Figura 4 acima e a Tabela 2 abaixo, pode-se observar que a maioria das músicas presentes no conjunto de dados possuem o tom da nota musical C# ou “Dó sustenido”, com 1238 faixas. Já o tom da nota musical D# ou “Ré sustenido foi observada” em menor quantidade, com apenas 225 canções.

Tabela 2: Notas Musicais

Notas	Quantidade
A	710
A#	618
B	723
C	956
C#	1238
D	713
D#	225
E	565
F	701
F#	643
G	801
G#	732

4.3.2 Compasso

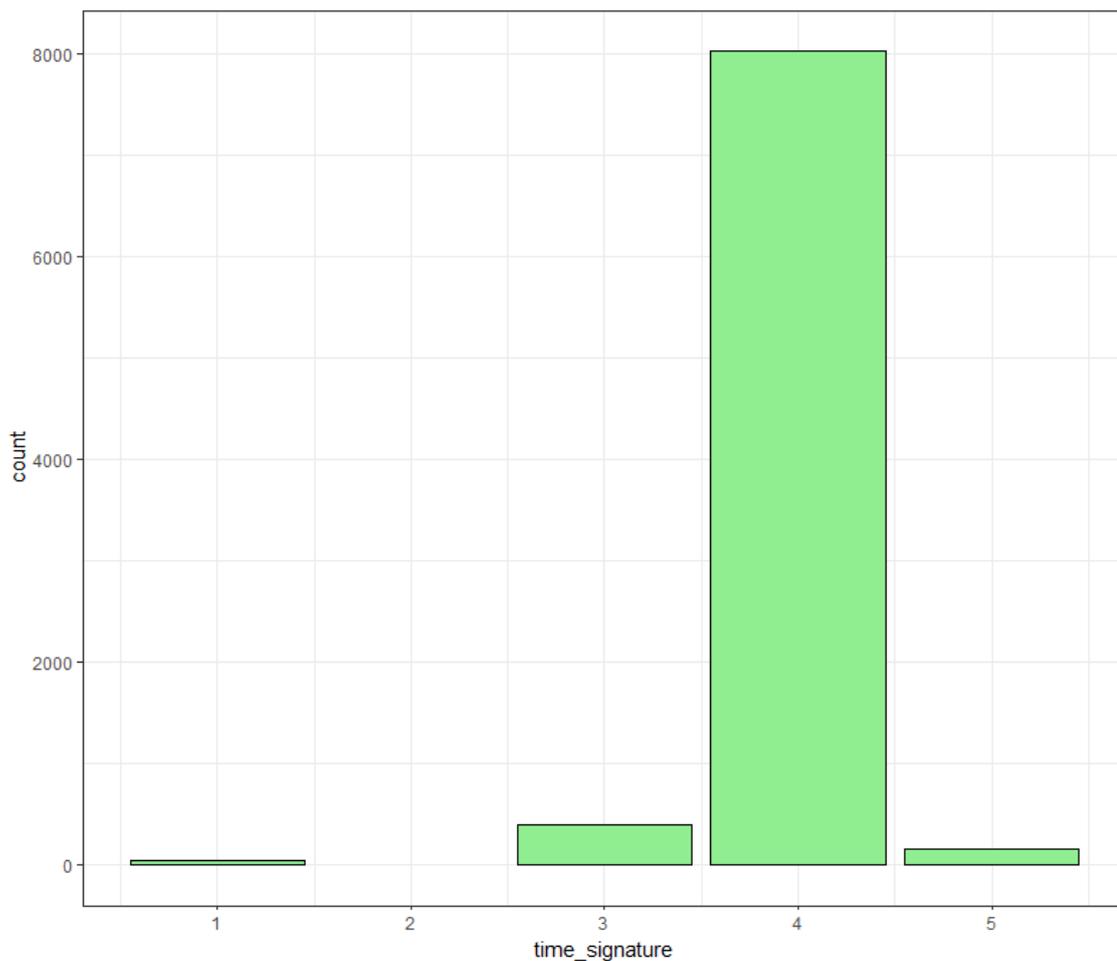


Figura 5: Time Signature

Tabela 3: Assinatura de Compasso

Compasso	Quantidade
1	44
2	0
3	399
4	8026
5	156

Analisando a Figura 5 e a Tabela 3, é notório que quase todas as músicas apresentaram a assinatura de compasso com o valor 4, com mais de 8000 faixas. Ou seja, possuem 4 batidas ou pulsos por compasso. Essas músicas, geralmente, estão presentes no gêneros musicais do Rock e Pop. Por outro lado, nenhuma música do conjunto de dados apresentou a assinatura de compasso com o valor 2, as quais estão presentes no Samba, usualmente.

4.3.3 Popularidade

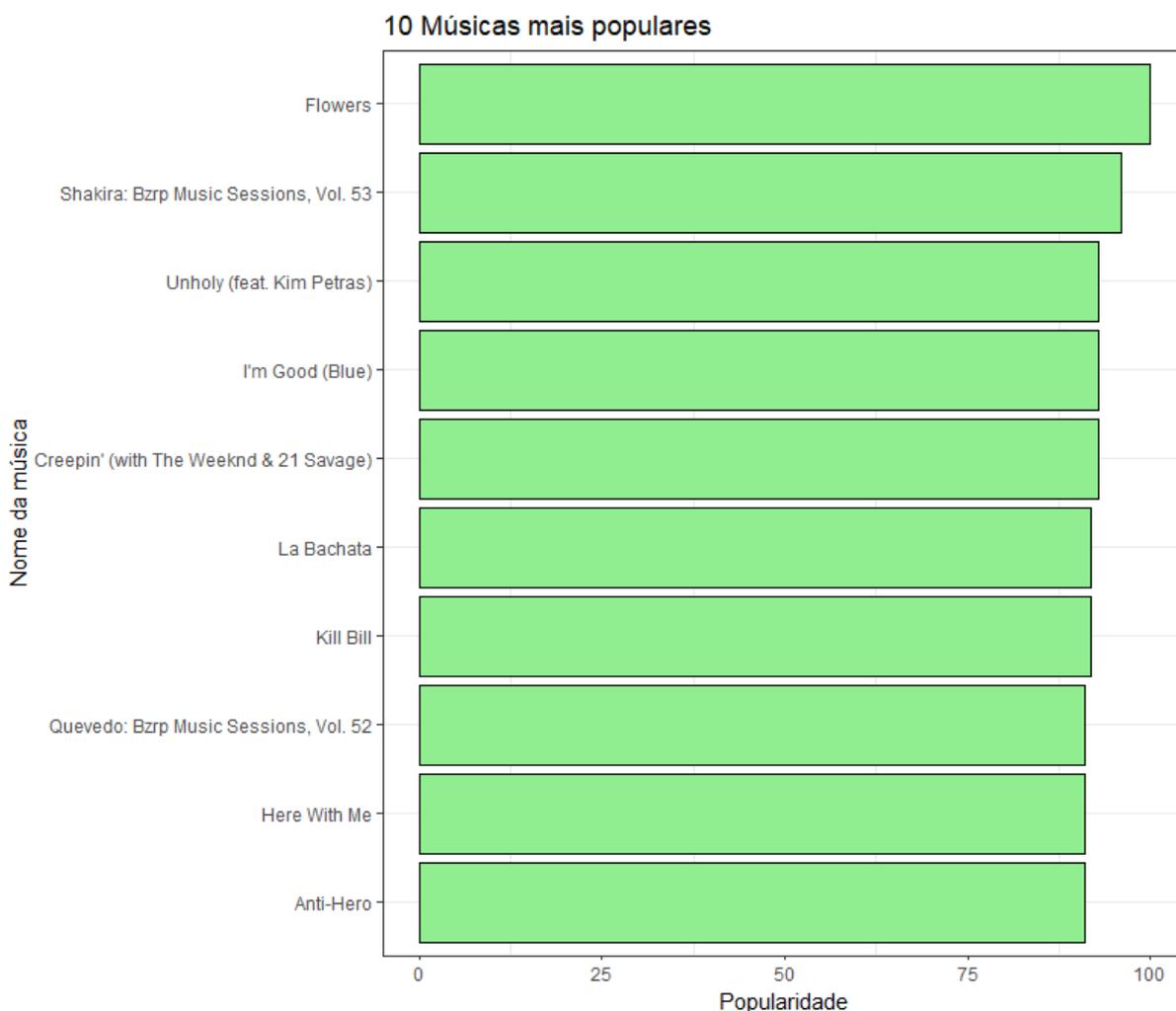


Figura 6: Top 10 músicas mais populares

Tabela 4: Top 10 Músicas Populares

	Nome da Música	Artista	Popularidade
1	Flowers	Miley Cyrus	100.00
2	Shakira: Bzrp Music Sessions, Vol. 53	Bizarrap	96.00
3	Unholy (feat. Kim Petras)	Sam Smith	93.00
4	I'm Good (Blue)	David Guetta	93.00
5	Creepin' (with The Weeknd & 21 Savage)	Metro Boomin	93.00
6	La Bachata	Manuel Turizo	92.00
7	Kill Bill	SZA	92.00
8	Quevedo: Bzrp Music Sessions, Vol. 52	Bizarrap	91.00
9	Anti-Hero	Taylor Swift	91.00
10	Here With Me	d4vd	91.00

De acordo com a Figura 6 e a Tabela 4, as músicas mais populares presentes no banco de dados apresentaram valores acima de 90. Apenas uma faixa possui popularidade máxima (100), a canção Flowers da artista norte-americana Miley Cyrus.

4.4 Matriz de Correlação

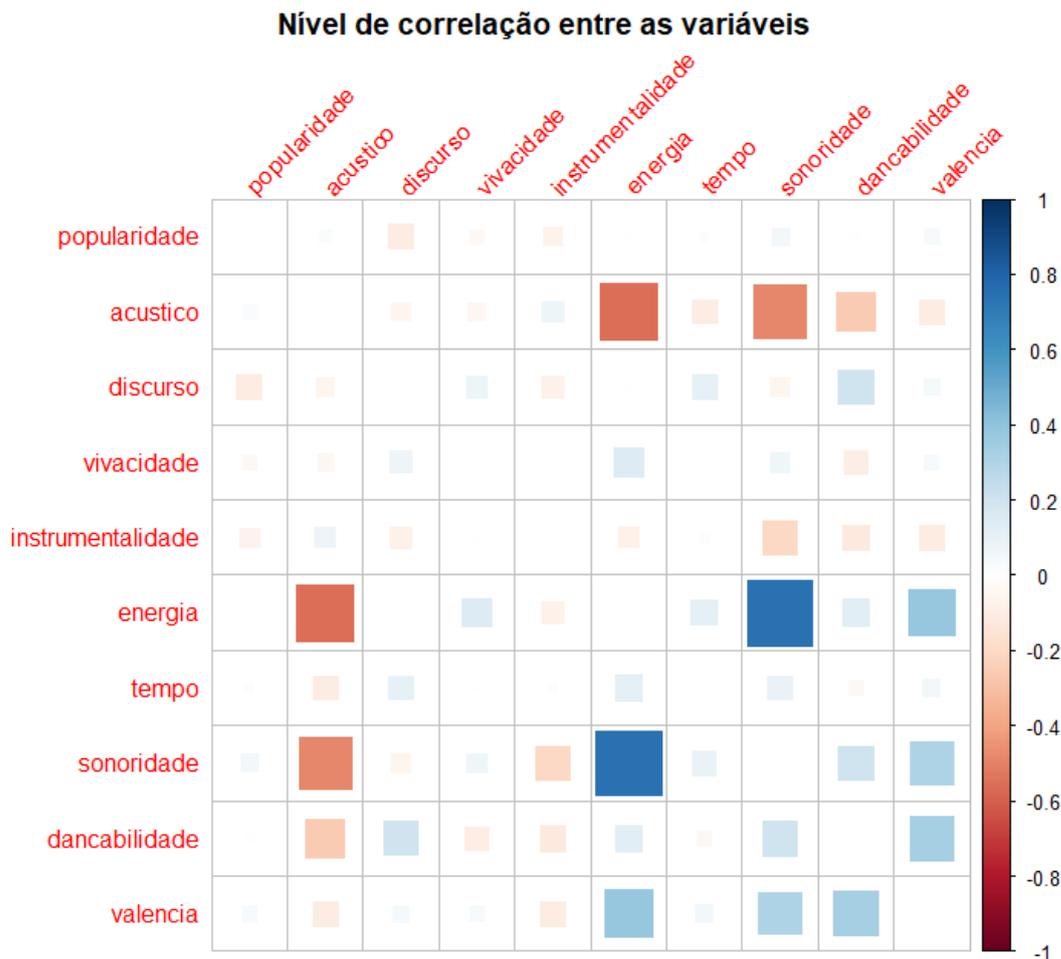


Figura 7: Matriz de Correlação

Analisando a Figura 7, acerca da matriz de correlação entre as variáveis, é possível observar que as variáveis que possuem maior associação positiva e forte, são entre “Energia” e “Sonoridade”. Além disso, as variáveis “Valência” e “Energia” também possuem uma correlação positiva, porém moderada. Por outro lado, outras associações nítidas, no entanto negativas e também moderadas, estão presentes nas variáveis “Energia” e

“Acústico” e “Sonoridade” e “Acústico”.

Logo abaixo, na Figura 8 e Tabela 5, estão detalhadas as correlações entre todas as variáveis do conjunto de dados

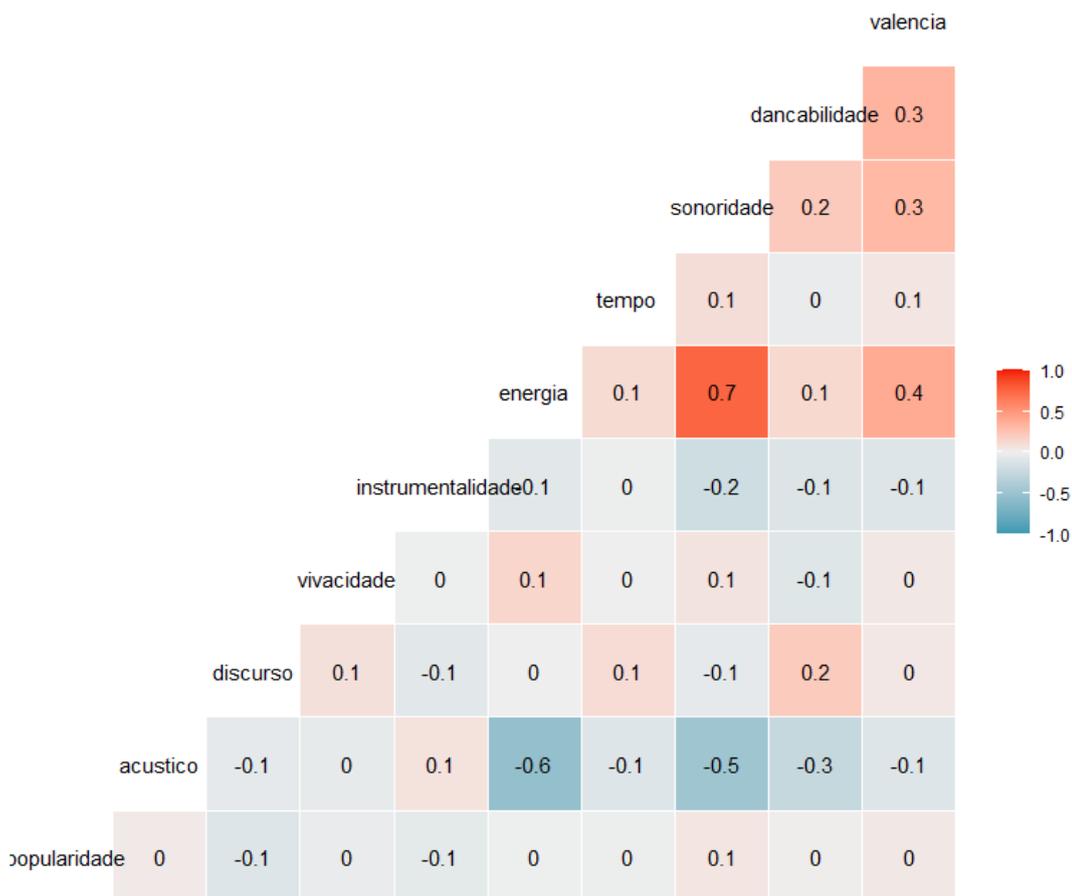


Figura 8: Matriz de Correlação

Tabela 5: Correlações

Variáveis	Correlação
Energia e Sonoridade	0.75
Energia e Valência	0.38
Energia e Acústico	-0.56
Acústico e Sonoridade	-0.49

Pode-se observar que a maior correlação positiva foi de 0.75, e a menor correlação negativa foi de -0.56. Ou seja, as variáveis Energia e Sonoridade são diretamente proporcionais, quanto maior a energia de uma música maior será a sonoridade. No entanto, as variáveis Energia e Acústico são inversamente proporcionais, quanto maior a energia de uma faixa menor será o acústico da mesma.

4.5 K-Nearest Neighbors

Nesta seção, serão demonstradas três playlists criadas utilizando a técnica estatística K-Nearest Neighbors (KNN). As três músicas inseridas no modelo para gerar o sistema de recomendação, foram selecionadas de forma aleatória. Além disso, foi preciso que cada faixa contivesse um gênero diferente das outras.

A primeira playlist criada teve a música “#icanteven” selecionada, da banda de Rock, The Neighbourhood.

Playlist 1

Tabela 6: Playlist 1

	artista	musica	genero_principal	key
1	The Neighbourhood	#icanteven	Rock	Fs
2	David Bowie	Blackstar	Rock	Fs
3	Soundgarden	Black Hole Sun	Rock	Fs
4	Imagine Dragons	Zero	Rock	Fs
5	Bring Me The Horizon	Doomed	Rock	Fs
6	Bastille	Joy	Rock	Fs
7	Twenty One Pilots	Fairly Local	Rock	Fs
8	Bastille	World Gone Mad	Rock	Fs
9	Metallica	Enter Sandman	Rock	Fs
10	The Beatles	Help! - Remastered 2009	Rock	Fs
11	BØRNS	Electric Love	Rock	Fs
12	Echosmith	Bright	Rock	Fs
13	AC/DC	Highway to Hell	Rock	Fs
14	Guns N' Roses	Sweet Child O' Mine	Rock	Fs
15	Twenty One Pilots	Smithereens	Rock	Fs
16	The Beatles	Day Tripper - Remastered 2015	Rock	Fs
17	Tom Petty and the Heartbreakers	Refugee	Rock	Fs
18	Twenty One Pilots	Formidable	Rock	Fs
19	Fall Out Boy	The Kids Aren't Alright	Rock	Fs
20	Panic! At The Disco	The Greatest Show	Rock	Fs
21	Twenty One Pilots	Ride	Rock	Fs
22	The Script	Rain	Rock	Fs
23	The Killers	The Man	Rock	Fs
24	Bring Me The Horizon	mother tongue	Rock	Fs
25	Twenty One Pilots	Levitate	Rock	Fs
26	Fall Out Boy	HOLD ME TIGHT OR DON'T	Rock	Fs
27	a-ha	Take on Me	Rock	Fs
28	Imagine Dragons	Monday	Rock	Fs
29	Cornelia Jakobs	Hold Me Closer	Pop	Fs
30	Ritt Momney	Put Your Records On	Pop	Fs

Nota-se que na playlist acima as músicas semelhantes estão bastante condizentes com a faixa escolhida, uma vez que a grande maioria pertence ao mesmo gênero (Rock), com exceção de apenas duas. Além disso, todas as melodias da playlist estão na mesma

tonalidade musical (Fá sustenido).

Já a segunda playlist criada teve a música “#PROUDCATOWNERREMIX” selecionada, do XXXTENTACION. O principal gênero desse artista é o Hip Hop.

Playlist 2

Tabela 7: Playlist 2

	artista	musica	genero_principal	key
1	XXXTENTACION	#PROUDCATOWNERREMIX	HipHop	Cs
2	Lil Baby	Global	HipHop	Cs
3	DaBaby	PRACTICE	HipHop	Cs
4	Apache 207	Bläulich	HipHop	Cs
5	Stormzy	Audacity (feat. Headie One)	HipHop	Cs
6	Future	Where Ya At (feat. Drake)	HipHop	Cs
7	Lil Baby	Perfect Timing	HipHop	Cs
8	21 Savage	break da law	HipHop	Cs
9	21 Savage	asmr	HipHop	Cs
10	Lil Yachty	OOPS	HipHop	Cs
11	Lil Baby	I Am (Lil Baby & Gunna)	HipHop	Cs
12	Lil Yachty	Peek A Boo	HipHop	Cs
13	DaBaby	iPHONE (with Nicki Minaj)	HipHop	Cs
14	Russ	The Flute Song	HipHop	Cs
15	Lil Baby	Detox	HipHop	Cs
16	KC Rebell	GEHT NICH GIBS NICH	HipHop	Cs
17	Future	Ridin Strikers	HipHop	Cs
18	Farid Bang	INTERNATIONAL GANGSTAS	HipHop	Cs
19	Future	Realer N Realer	HipHop	Cs
20	Lil Baby	Heatin Up (feat. Gunna)	HipHop	Cs
21	21 Savage	good day	HipHop	Cs
22	Takeoff	Martian - Intro	HipHop	Cs
23	Playboi Carti	Beno!	HipHop	Cs
24	Future	200	HipHop	Cs
25	Future	Real Baby Pluto	HipHop	Cs
26	Lil Baby	Get Ugly	HipHop	Cs
27	Lil Yachty	GET MONEY BROS.	HipHop	Cs
28	Lil Pump	Crazy	HipHop	Cs
29	21 Savage	Disrespectful	HipHop	Cs
30	DaBaby	TALK ABOUT IT	HipHop	Cs

Observa-se que na playlist acima as músicas vizinhas estão bastante condicentes com a faixa escolhida, uma vez que todas as canções pertencem ao mesmo gênero (Hip Hop), e ainda, todas as melodias da lista estão na mesma tonalidade musical (Dó sustenido).

Por fim, a terceira playlist criada teve a música “#SELFIE” selecionada, do grupo pop The Chainsmokers.

Playlist 3

Tabela 8: Playlist 3

	artista	musica	genero_principal	key
1	The Chainsmokers	#SELFIE	Pop	C
2	Jesse & Joy	3 A.M.	Pop	C
3	Megan Thee Stallion	Hot Girl Summer	Pop	C
4	Lady Gaga	Free Woman	Pop	C
5	Tory Lanez	Jerry Sprunger (with T-Pain)	Pop	C
6	Machine Gun Kelly	RAP DEVIL	Pop	C
7	Alok	Never Let Me Go	Pop	C
8	Little Mix	Is Your Love Enough?	Pop	C
9	Axwell /\ Ingrosso	I Love You	Pop	C
10	Pusha T	Rock N Roll	Pop	C
11	Sarah Connor	Ring Out The Bells	Pop	C
12	benny blanco	Graduation (with Juice WRLD)	Pop	C
13	David Guetta	Bang My Head (feat. Sia & Fetty Wap)	Pop	C
14	Martin Solveig	Places	Pop	C
15	Charli XCX	Break the Rules	Pop	C
16	Katy Perry	Smile	Pop	C
17	Rauw Alejandro	Que Le De	Pop	C
18	Bebe Rexha	No Broken Hearts (feat. Nicki Minaj)	Pop	C
19	One Direction	What a Feeling	Pop	C
20	Fetty Wap	RGF Island	Pop	C
21	Ricky Martin	La Mordidita (feat. Yotuel)	Pop	C
22	Pitbull	Fun (feat. Chris Brown)	Pop	C
23	Luis Fonsi	Échame La Culpa	Pop	C
24	Matoma	Old Thing Back	Pop	C
25	Gambi	POPOPOP	Pop	C
26	Jason Derulo	Goodbye	Pop	C
27	Jessie J	Masterpiece	Pop	C
28	Luísa Sonza	BRABA	Pop	C
29	Lady Gaga	Plastic Doll	Pop	C
30	Anne-Marie	Alarm	Pop	C

Identifica-se que na playlist acima as músicas vizinhas estão bastante ajustadas com a faixa escolhida, uma vez que todas as canções pertencem ao mesmo gênero (Pop), e ainda, as melodias da lista estão na mesma tonalidade musical (Dó).

4.6 Clusterização

Após a realização das análises descritivas e implementação da técnica KNN, é realizada uma análise de Cluster para inserir as músicas presentes nas playlists criada em grupos diferentes (novas playlists), de acordo com suas características e semelhanças, como foi dito na seção anterior.

As variáveis utilizadas no processo da análise de componentes principais foram as mesmas utilizadas na aplicação da técnica KNN, citadas na metodologia. Após realizar a análise, verificou-se que com apenas quatro componentes da PCA seria possível explicar quase 80% da variância dos dados. Com isso, ao invés de utilizar dez variáveis na etapa de clusterização, foram necessárias apenas quatro. As variáveis que caracterizam as músicas empregadas foram as que mais tinha correlação entre si, são elas: “acústico”, “sonoridade”, “energia” e “valência”.

4.6.1 K-means

O método de clusterização utilizado foi o K-means. É importante ressaltar que a normalização das variáveis já foi realizada antes da implementação do KNN, ou seja, não foi necessário efetuar outra padronização dos dados. O número de clusters escolhido foi igual a 3.

Logo abaixo será possível observar os três gráficos criados com as músicas das playlists geradas pelo KNN separadas em três grupos distintos.

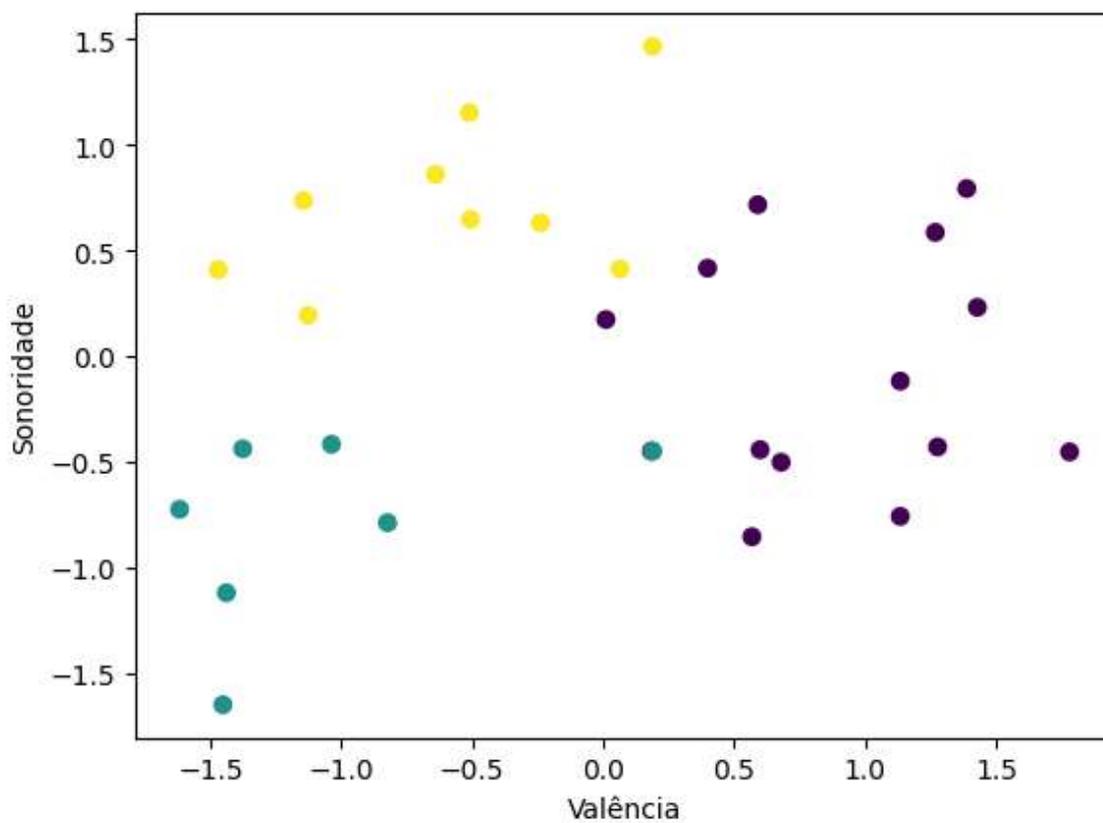


Figura 9: Cluster Playlist 1 - Rock

No gráfico acima pode-se notar que as variáveis que melhor conseguiu separar as músicas da primeira playlist de maneira mais acertiva foram “valência” e “sonoridade”.

O grupo em amarelo concentrou as músicas com baixas valências e sonoridades elevadas, enquanto o grupo em azul claro possui sonoridades baixas e um outlier. Já a categoria representada pela cor roxa possui valências elevadas e sonoridades moderadas.

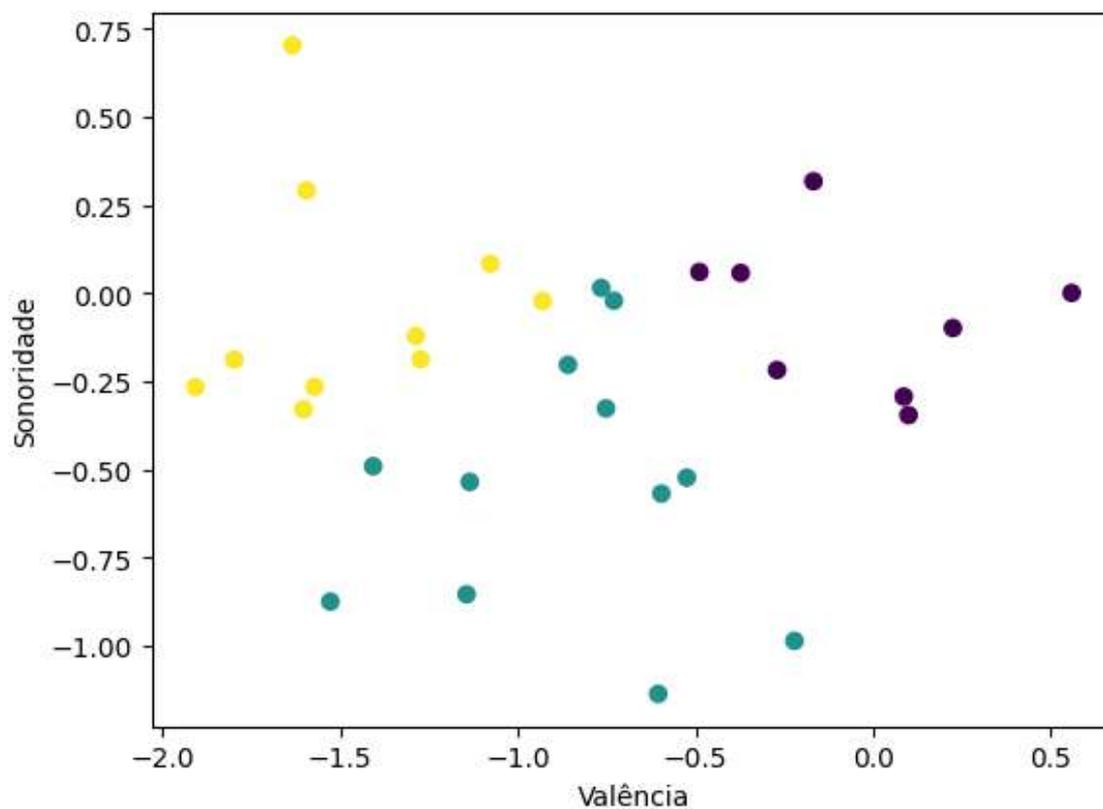


Figura 10: Cluster Playlist 2 - HipHop

Assim como no gráfico da primeira playlist, o gráfico da playlist 2 também contou com as variáveis “valência” e “sonoridade” como sendo as quais definiram da melhor maneira a separação dos clusters.

O grupo em amarelo concentrou as músicas com valências moderadas e sonoridades elevadas, além de possuir um outlier, enquanto o grupo roxo possui valências elevadas. Já a categoria representada pela cor azul claro possui valências moderadas e sonoridades variando entre baixa e moderada.

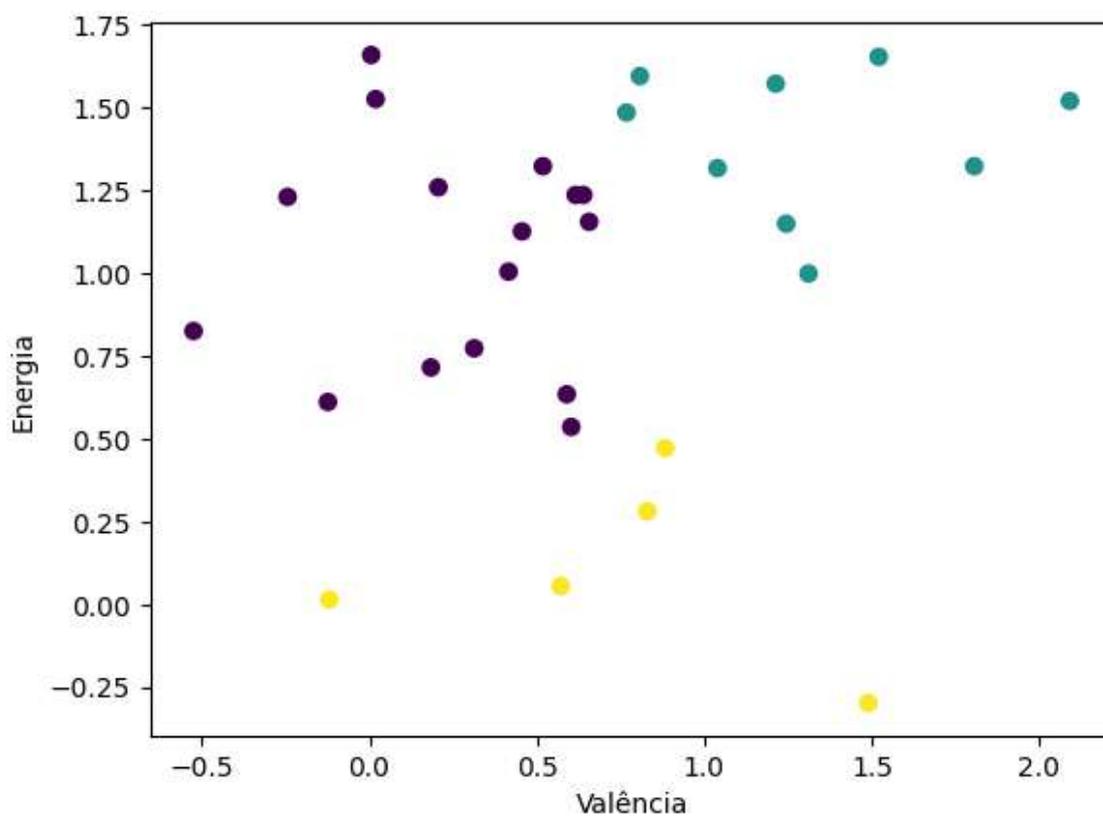


Figura 11: Cluster Playlist 3 - Pop

Já para a terceira playlist, observando o gráfico acima, as variáveis que melhor definiram a separação dos clusters foram “energia” e “valência”.

Os grupos em azul claro e roxo concentraram as músicas com energias elevadas, no entanto, possuem valências variando entre baixa (grupo roxo) e elevada (grupo azul). Todavia, o grupo em amarelo agrupou as faixas com menor energia e valências baixas e moderadas.

4.7 Streamlit

Nesta seção, será demonstrado como manusear a interface criada com o Streamlit, a fim de gerar um sistema de recomendação a partir de uma música escolhida previamente.

🔗 Lista de músicas - Spotify 🎧

🔗 Spotipy + Streamlit

Informações e características das músicas mais ouvidas do Spotify entre os anos de 2014 e 2022

Selecione um gênero musical

Rock ×

You selected:

```

[
  0 : "Rock"
]

```

	id	artista	musica	generos
311	11ilikXxC6NP0Ma8vMD27x	Vundabar	Alien Blues	boston indie, ir
331	7d339Lvm6sTy8HIId4HIzfY	The Stone Roses	All for One	alternative roc
380	6KqiVIOLFdZJVijVAPym1p	The Beatles	All You Need Is Love - Remastered 20	beatlesque, bri
427	3D4Xkcir5YEhk8Xidkd1dL	Fall Out Boy	American Beauty/American Psycho	emo, modern r
428	5RmlP2OglmIOvsOxrE43Xw	MKTO	American Dream	dance pop, pop
429	7MRyJPksH3G2cXHN8UKYzP	Tom Petty and the Hea	American Girl	album rock, cli

Figura 12: Streamlit - Introdução

Primeiramente, é necessário que o usuário escolha um gênero que tenha preferência, mais afinidade ou até mesmo interesse para montar uma playlist. É importante ressaltar que apenas um gênero poderá ser escolhido.

Por padrão do aplicativo, o gênero musical já selecionado é “Rock”. Caso o usuário deseje alterar, é necessário excluir o que já estava selecionado e selecionar outro.

457	0UuNO0yYvsMPlyLF7RtQlg	Rammstein	Angst	alternative me
461	5vdp5UmvTsnMEMESIF2Ym7	Queen	Another One Bites The Dust - Remas	classic rock, gl
462	2Fn4gZI3MAeWTiv7cSVxZQ	Bastille	Another Place	metropolis,

Na tabela acima contém as informações de todas as 8165 músicas presentes no banco de dados. Está ordenada em ordem alfabética pelo nome das músicas.

Para a realização da montagem das playlists, é necessário que o usuário escolha uma música da tabela acima.

Selecione e insira o ID uma faixa

0xSfELjbi6QwmpRlDu4Nrx

O ID selecionado foi: 0xSfELjbi6QwmpRlDu4Nrx

PLAYLIST

A playlist abaixo contém as 30 músicas mais semelhantes do banco de dados, de acordo com a música que foi inserida no campo acima. A junção foi realizada utilizando a técnica estatística K-nearest Neighbors

id	artista	musica	generos
----	---------	--------	---------

Figura 13: Streamlit - KNN

Em seguida, para iniciar a geração da playlist, é exigido que o usuário selecione e insira o ID de uma faixa escolhida do gênero que foi definido anteriormente. Para isso, é necessário apenas que o consumidor copie e cole o ID da música no campo designado, e em seguida, pressione enter. Após essas etapas serem concluídas, uma mensagem automática será exibida com o ID selecionado.

PLAYLIST

A playlist abaixo contém as 30 músicas mais semelhantes do banco de dados, de acordo com a música que foi inserida no campo acima. A junção foi realizada utilizando a técnica estatística K-nearest Neighbors

	id	artista	musica	generos
0	0xSfELjbi6QwmpRlDu4Nrx	The Neighbourhood	#icanteven (feat. French Montana)	modern alterna
1,602	30XpOUqxPaFDO9rLnn1Jvb	Catey Shaw	Cuddle Up	indie pop
1,933	2AHnmlkyZbnxqSA66B7jK3	ABBA	Don't Shut Me Down	europop, swedi:
446	0wzPHTgOtFIDhk2rD3Au4x	The Weeknd	Angel	canadian contei
2,343	2RZo2VN0dTHOyPFjVXqxZI	sangiovanni	farfalle	italian pop
2,729	59bY27SZQ4r25o02vqOhgC	Juice WRLD	Girl Of My Dreams (with SUGA of BTS)	chicago rap, me
8,000	6Kfoo60npYPdvNxMPMiDpX	Post Malone	Wrapped Around Your Finger	dfw rap, melodi
3,803	2zcPV7NeGyhSZJK0WjETx7	Years & Years	Karma	dance pop, ednr
7,328	4Ft0391aLVNtCPPseh98lp	Sebastian Yatra	Traicionera	colombian pop,
520	5VIZcrKXPd9wL4hqE4YN8z	Wiktoria	As I Lay Me Down	europop, swedi:

Figura 14: Streamlit - Playlist

Por fim, a playlist gerada é apresentada em forma de uma tabela contendo todas as informações musicas das 30 faixas.

5 CONCLUSÃO

Ao longo deste trabalho, foram exploradas com sucesso duas metodologias poderosas, K-Nearest Neighbors (KNN) e Análise de Cluster, a fim de realizar uma automação criativa de playlists no Spotify. Ambas as técnicas implementadas de maneira conjunta, mostraram-se altamente eficientes na organização de um sistema de recomendação de músicas, proporcionando uma experiência musical mais envolvente e específica para os usuários.

A utilização do método KNN apresentou-se particularmente precisa na identificação de padrões parecidos entre diferentes faixas musicais. Ao considerar a proximidade de uma música com as outras, foi possível criar playlists coerentes e nítidas em relação as semelhanças, refletindo gostos musicais específicos de maneira mais definida.

A Análise de Cluster, por sua vez, permitiu agrupar músicas com base em características similares, proporcionando uma abordagem complementar ao KNN. Essa metodologia foi especialmente útil para identificar afinidades musicais mais amplas, levando à recomendações mais abrangentes.

Além das técnicas selecionadas, a escolha do Streamlit como interface para a apresentação dos resultados se mostrou acertada. A facilidade de uso e a capacidade de criar visualizações interativas contribuíram significativamente para a compreensão de todo o desfecho gerado pelos algoritmos. A integração com as playlists criadas no Spotify proporcionou uma experiência intuitiva e agradável ao usuário.

Em suma, a combinação eficaz do KNN, Análise de Cluster e Streamlit não apenas automatizou a criação de playlists, como também elevou a qualidade dessa automação, garantindo que as recomendações fossem verdadeiramente personalizadas e alinhadas com as preferências musicais escolhidas. Este trabalho destaca a eficiência dessas técnicas apresentadas, bem como a importância de uma interface de usuário acessível na implementação bem-sucedida de soluções de recomendação de música. Outrossim, à medida que a tecnologia evolui na era da personalização, a aplicação dessas metodologias na indústria da música promete transformar a maneira como os ouvintes descobrem e desfrutam de novas faixas musicais e artistas.

Referências

- AWARI. *Streamlit: A Biblioteca Python que Revoluciona o Desenvolvimento de Aplicações Web*. 2023. Last accessed 20 November 2023. Disponível em: <https://awari.com.br/streamlit-python/>.
- BATTISTI, F. M. d. S. S. I. D. E. *Software R: curso avançado*. 2019. Last accessed 9 July 2023. Disponível em: <https://smolski.github.io/livroavancado/analise-de-clusters.html#k-means>.
- BELMONTE, P. *Machine Learning KNN: Introdução ao Algoritmo K-Nearest Neighbors no Machine Learning*. 2023. Last accessed 22 November 2023. Disponível em: https://awari.com.br/machine-learning-knn-introducao-ao-algoritmo-k-nearest-neighbors-no-machine-learning/?utm_source=blog&utm_campaign=projeto+blog&utm_medium=Machine%20Learning%20KNN:%20Introdu%C3%A7%C3%A3o%20ao%20Algoritmo%20K-Nearest%20Neighbors%20no%20Machine%20Learning.
- CANALTECH. *Spotify*. 2017. Last accessed 12 November 2022. Disponível em: <https://canaltech.com.br/empresa/spotify-ltd/>.
- ESTATMG. *Como Utilizar Dados do Spotify*. 2021. Last accessed 21 December 2022. Disponível em: <https://estatmg.com.br/2021/09/24/como-utilizar-dados-do-spotify/>.
- FLEISCHER, R.; SNICKARS, P. Discovering spotify-a thematic introduction. *Culture Unbound*, v. 9, n. 2, p. 130–145, 2017.
- GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de pesquisa*. [S.l.]: Plageder, 2009.
- KASSAMBARA, A. *Practical guide to cluster analysis in R: Unsupervised machine learning*. [S.l.]: Sthda, 2017. v. 1.
- KWORB. *Spotify Daily Charts Totals - Global*. 2022. Last accessed 22 May 2023. Disponível em: https://kworb.net/spotify/country/global_daily_totals.html.
- MOREIRA, P. S. da C.; TSUNODA, D. F. Proposta de base de dados para classificação automática de gêneros musicais brasileiros. *Revista Brasileira de Educação em Ciência da Informação*, v. 8, 2021.
- MOSCHETTA, P. H.; VIEIRA, J. Música na era do streaming: curadoria e descoberta musical no spotify. *Sociologias, SciELO Brasil*, v. 20, p. 258–292, 2018.
- TIMOTEO, A. *Machine Learning KNN: Introdução ao Algoritmo K-Nearest Neighbors no Machine Learning*. 2023. Last accessed 22 November 2023. Disponível em: [https://awari.com.br/machine-learning-knn-introducao-ao-algoritmo-k-nearest-neighbors-no-machine-learning-2/#:~:text=O%20algoritmo%20KNN-,O%20algoritmo%20K%2DNearest%20Neighbors%20\(KNN\)%20%C3%A9%20uma%20t%C3%A9cnica,usado%20para%20classifica%C3%A7%C3%A3o%20e%20regress%C3%A3o.](https://awari.com.br/machine-learning-knn-introducao-ao-algoritmo-k-nearest-neighbors-no-machine-learning-2/#:~:text=O%20algoritmo%20KNN-,O%20algoritmo%20K%2DNearest%20Neighbors%20(KNN)%20%C3%A9%20uma%20t%C3%A9cnica,usado%20para%20classifica%C3%A7%C3%A3o%20e%20regress%C3%A3o.)

APÊNDICE

Toda a codificação implementada no presente trabalho pode ser encontrada na página do Github a seguir.

`<https://github.com/joaovmeloc/TCC>`