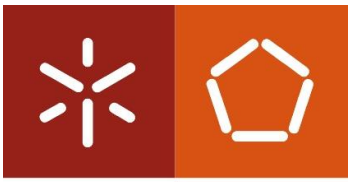


Universidade do Minho

Departamento de Informática

Mineração de Dados

Trabalho Prático

António Silva (PG47033)
Benjamim Costa (PG50258)
Diogo Sambento(PG49999)

16 de junho de 2023

Conteúdo

Capítulo 1

Introdução

1.1 Contextualização

Este trabalho foi desenvolvido no âmbito da Unidade Curricular de Mineração de Dados, do perfil de Engenharia de Conhecimento, do 2º semestre do 1º ano do Mestrado em Engenharia Informática da Universidade do Minho.

O objetivo deste trabalho é implementar uma aplicação e/ou estudo que envolva recolha, processamento, análise e mineração de dados publicamente acessíveis com o objetivo de **extrair conhecimento útil e não óbvio** para os utilizadores, integrando mais do que uma fonte de dados.

O código e dados utilizados estão disponibilizados no repositório do grupo.

1.2 Motivação e objetivos

A música é uma forma de arte e expressão que pode influenciar o humor, as emoções e as preferências das pessoas. Com a popularidade dos serviços de streaming de música, como o Spotify, os utilizadores têm acesso a uma grande variedade de músicas de diferentes géneros, artistas e épocas. No entanto, encontrar músicas que se adequem ao gosto e ao estado de espírito de cada um pode ser uma tarefa difícil, por isso é importante desenvolver sistemas de recomendação de músicas que possam oferecer sugestões personalizadas e satisfatórias aos utilizadores.

Assim sendo, com este trabalho o grupo pretende ver se é possível fazer boas recomendações de músicas através da análise dos seus atributos e sem ter em conta os géneros das mesmas. Atributos estes que são características quantitativas ou qualitativas como o ritmo, o tom, a energia, a acústica, a dançabilidade, a valência, etc. Esses atributos podem ser obtidos através de algoritmos de processamento de áudio ou de fontes externas, como a API do Spotify, que fornece metadados e análises de áudio para milhões de músicas. A análise dos atributos das músicas pode ajudar a entender melhor as semelhanças e as diferenças entre as mesmas, assim como as preferências e os padrões dos utilizadores.

Capítulo 2

Trabalho relacionado

Recomendadores de músicas não são novidade na industria musical. Um bom sistema de recomendações deve ser capaz de detetar automaticamente as preferências do utilizador e, desta forma, indicar algo semelhante para este ouvir.

Os serviços de *streaming* já possuem estes sistemas incorporados e são altamente sofisticados de forma a dar ao utilizador a melhor experiência possível. Uma técnica bastante popular é a filtragem colaborativa, que consiste na deteção de padrões de audição e gostos semelhantes. Outro método é a filtragem baseada no conteúdo, que se baseia na leitura dos atributos dos itens recomendados e cria uma espécie de perfil genérico de utilizador que se deve interessar por temas semelhantes. Por vezes são usados múltiplos métodos em conjunto formando um sistema híbrido. No presente trabalho é tentada uma abordagem idêntica à segunda.

Capítulo 3

Fontes de dados

3.1 Acesso aos dados

Como fontes de dados para este trabalho decidimos usar um dataset com 114000 músicas pertencentes a 114 géneros diferentes (1000 músicas por género) e ainda a API do spotify para obter músicas adicionais de playlists de diferentes géneros.

O dataset foi obtido através da plataforma Kaggle, estando por isso disponível publicamente e possui as seguintes features:

- **track_id :** String que representa o id único da música (constituído por letras e números);
- **track_artist:** String que representa o artista da música;
- **album_name:** String que representa o álbum a que a música pertence;
- **track_name:** String que representa o nome da música;
- **popularity:** Inteiro que representa a popularidade da música (quanto maior mais popular a música é);
- **duration_ms:** Inteiro que representa a duração da música em milissegundos;
- **explicit:** String que informa se a música contem linguagem explícita ou não, apenas tem como valores "FALSO" ou "VERDADEIRO";
- **danceability:** Float entre 0.0 e 1.0 que descreve a adequação da música para dançar com base numa combinação de elementos musicais, incluindo "tempo", a estabilidade do ritmo, a força da batida e a regularidade geral (quanto maior mais dançável a música é);
- **energy:** Float entre 0.0 e 1.0 que representa uma medida perceptual de intensidade. Normalmente, as faixas energéticas são rápidas e barulhentas, sendo as características que contribuem para este atributo a gama dinâmica, a percepção do volume, o timbre, a entropia geral, etc;

- **key:** Inteiro com valores de -1 a 11 que representa o tom em que a música se encontra;
- **loudness:** Float que varia entre -60 e 0 que representa o volume geral da música em decibéis (dB). Os valores são calculados como média em toda a música e são úteis para comparar a loudness relativa das músicas;
- **mode:** Inteiro que é 0 ou 1 que indica a modalidade da música, que pode ser maior (1) ou menor (0);
- **speechiness:** Float entre 0.0 e 1.0 que deteta a presença de palavras faladas numa música. Quanto mais exclusivamente semelhante à fala for a gravação (por exemplo talk show, livro áudio, poesia), mais próximo de 1.0 será o valor do atributo. Valores superiores a 0.66 descrevem músicas que são provavelmente constituídas apenas por palavras faladas. Valores entre 0.33 e 0.66 descrevem músicas que podem conter tanto música como fala (como música rap) e valores abaixo de 0.33 representam muito provavelmente música que não é de fala;
- **acousticness:** Float entre 0.0 e 1.0 que é uma medida de confiança para determinar se a música é acústica (quanto mais próximo de 1.0 mais provável é que a música é acústica);
- **instrumentalness:** Float entre 0.0 e 1.0 que prevê se uma faixa não contém vocais (quanto mais próximo o valor estiver de 1.0, maior a probabilidade de a faixa não conter conteúdo vocal). De notar que os sons "Ooh" e "aah" são tratados como instrumentais e as músicas de rap ou de palavras faladas são claramente "vocais". Assim, os valores acima de 0.5 destinam-se a representar músicas instrumentais, mas a confiança é maior à medida que o valor se aproxima de 1.0;
- **liveness:** Float entre 0.0 e 1.0 que deteta a presença de público na gravação, sendo que valores mais elevados representam uma maior probabilidade de a música ter sido tocada ao vivo;
- **valence:** Float entre 0.0 e 1.0 que descreve a positividade musical transmitida por uma música. Músicas com valência alta soam mais positivas (felizes, alegres, eufóricas), enquanto que músicas com valência baixa soam mais negativas (tristes, deprimidas, zangadas);
- **tempo:** Float que representa o "tempo" geral estimado da música em batidas por minuto (BPM). Na terminologia musical, o "tempo" é a velocidade ou ritmo de uma determinada música e deriva diretamente da duração média das batidas;
- **time_signature:** Inteiro que varia entre 3 e 7 e que é uma convenção notacional para especificar quantas batidas existem em cada medição;
- **track_genre:** String que representa o género musical de cada música.

Através da API do spotify retiramos informações de várias músicas, analisando playlists de vários géneros. De todas as informações presentes no dataset descrito acima, as músicas provenientes da API só não possuem **album_name**, **explicit** e **popularity**. Para além de todas as outras características vistas anteriormente, estas músicas têm ainda as seguintes features:

- **type:** String que descreve o tipo do objeto, neste caso *audio_features*;
- **id:** Outro campo que contém o id da música dado pelo spotify;
- **uri:** String que representa o URI da música dado pelo spotify;
- **track_href:** String que representa link que redireciona para a página Web que fornece todos os pormenores da música;
- **analysis_url:** String que representa um URL para aceder à análise de áudio completa da música, é necessário um token de acesso para aceder a estes dados.

Assim sendo, após serem armazenadas todas as informações das músicas provenientes das duas fontes de dados, juntámos os dados todos num só *dataframe* e procedemos ao processamento dos dados.

3.2 Processamento dos dados

Quanto ao processamento de dados foram inicialmente eliminadas todas as colunas cujos valores não eram numéricos como **track_artist**, **track_id**, **track_name**, etc. Para além destas, foram também removidas as colunas numéricas **duration_ms** e **popularity** visto que o grupo achou que não acrescentam informação relevante para o modelo. Após removermos todas estas colunas, foi analisada a correlação das features restantes de forma a podermos analisar se fazia sentido mantermos todas elas. Tal como podemos ver na figura 3.1, a feature **acousticness** não está a adicionar informação relevante ao modelo visto que está altamente correlacionada com várias outras features e por isso decidimos retirar também esta coluna.

Figura 3.1: Correlação das features.

Após removermos todas as colunas em que foi concluído que não era adicionada informação nova, analisamos também se estavam presentes valores nulos e decidimos "filtrar"as músicas que tinham o mesmo nome e artista mas que vinham com id´s diferentes, de forma a prevenir a presença da mesma música mais que uma vez nos dados do modelo. Por fim, foram normalizados os valores de **loudness**, **tempo**, **key** e **time_signature** de forma a não ter um impacto muito elevado sobre as outras variáveis, visto que os seus valores têm uma escala maior que as restantes features.

3.3 Análise dos dados

Após todos os dados serem tratados, ficamos com uma coleção de músicas que serão mais tarde usadas para treinar o modelo e que contêm as seguintes features **todas com valores entre 0.0 e 1.0**:

- **danceability**;
- **energy**;
- **key**;
- **loudness**;

- **mode**;
- **speechiness**;
- **instrumentalness**;
- **liveness**;
- **valence**;
- **tempo**;
- **time_signature**.

Podemos analisar na figura 3.2 um exemplo destes dados tratados.

danceability	energy	key	loudness	mode	speechiness	instrumentalness	liveness	valence	tempo	time_signature
0.733	0.794	0.454545455	0.688949202	1	0.0307	0.0367	0.33	0.931	0.6088375	0.8
0.64	0.663	0	0.742525591	1	0.0374	0.00806	0.152	0.663	0.545543448	0.8
0.395	0.843	0.727272727	0.807492574	1	0.0374	0	0.0404	0.481	0.461938925	0.8
0.31	0.7	0.818181818	0.781804971	1	0.047	0.00965	0.0828	0.763	0.774066039	0.8
0.412	0.902	0.818181818	0.65524758	1	0.405	0.131	0.405	0.422	0.368735927	0.8
0.446	0.952	0.545454545	0.789434317	1	0.0523	0.0857	0.112	0.624	0.514862022	0.8

Figura 3.2: Exemplo dos dados tratados.

Capítulo 4

Obtenção dos resultados

Após possuirmos todos os dados relevantes e tratados da forma correta num único *dataframe*, treinamos um modelo de aprendizagem automática com estes dados usando o algoritmo *K_Means*. Depois de o modelo estar treinado, a ideia é fornecer como input uma determinada música e através deste modelo ver a qual cluster essa música fornecida pertence após ser feito predict com a mesma. Depois acedemos às músicas pertencentes a esse cluster e fornecêmo-las como output, sendo por isso importante que o modelo consiga associar a música fornecida a um cluster onde estejam presentes músicas do mesmo estilo, sendo isso analisado no próximo capítulo.

Capítulo 5

Análise e discussão dos resultados

Para melhor análise dos resultados obtidos, o programa desenvolvido tem como *output* uma página *HTML* com *media players* para a música fornecida e as músicas recomendadas de forma a facilitar a audição destas (Exemplo na figura 5.1 da página 12).

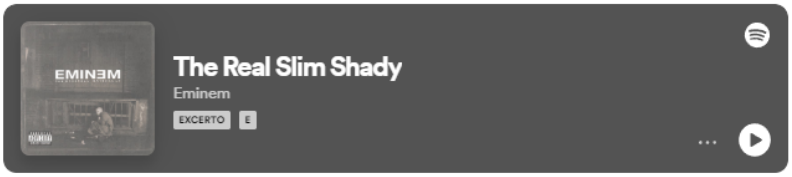
Analisando os resultados a partir da audição das músicas, de forma geral, o sistema não gera boas recomendações. Por vezes, fornece boas recomendações, principalmente quando fornecemos musicas clássicas como musica de procura.

Embora possa parecer tentador utilizar as propriedades das músicas da *API* do *Spotify* para criar um sistema de recomendação de músicas, é crucial reconhecer as desvantagens de o fazer. Embora medidas como o ritmo, a energia, a capacidade de dançar e outras sejam úteis para compreender as características de uma canção, são insuficientes para captar os pormenores e estilos musicais preferidos. Esta análise baseada em atributos é uma simplificação excessiva do gosto musical. A música é uma forma de expressão profundamente pessoal e emocional que transcende as qualidades quantificáveis. Além disso , a música é fortemente influenciada por factores culturais, históricos e sociais.

É também observado que os atributos das músicas não conseguem, por si só, englobar a vasta diversidade de géneros e subgéneros musicais bem como as suas características únicas associadas a cada um deles.

Podemos então concluir que, face às razões ditadas em cima, é preciso recorrer a um sistema híbrido para construir um recomendador musical capaz de fornecer a melhor experiência possível ao utilizador.

Se gostas desta música:



Então experimenta estas: (a ordem é irrelevante)

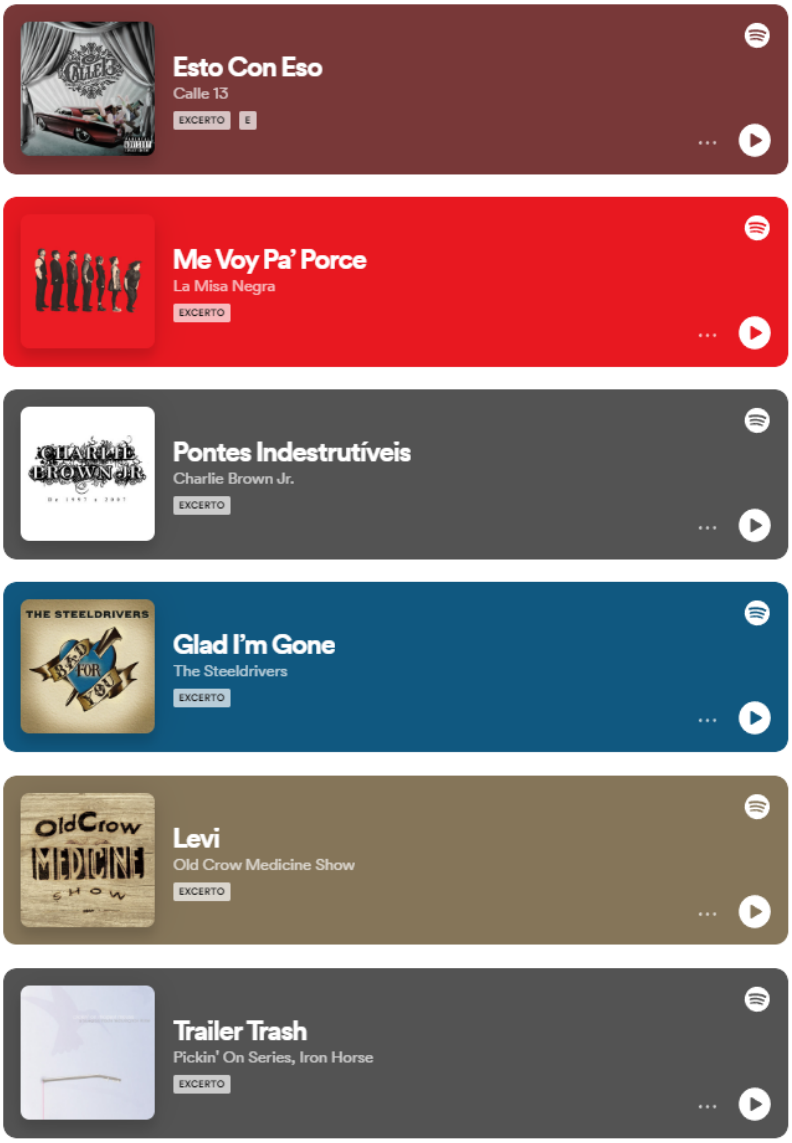


Figura 5.1: Exemplo de página gerada

Capítulo 6

Conclusões e trabalho futuro

Ao desenvolver este projeto, foi possível consolidar os conhecimentos adquiridos ao longo do semestre sobre a área de mineração de dados. Através da análise dos atributos musicais e a aplicação de um algoritmo de *clustering* não supervisionado, foi possível implementar um sistema de recomendações musicais que, por vezes, faz boas recomendações.

Apesar do grupo não ter conseguido implementar a utilização de um algoritmo de *KNN* como se tinha proposto para a última fase do projeto, com a realização do trabalho o grupo acredita ter alcançado, de forma geral, as metas pretendidas.