



**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

Classificação de Gêneros Musicais com Extração Manual de Características

Pedro Henrique Mendes Rodrigues

TRABALHO DE CONCLUSÃO DE CURSO

ORIENTAÇÃO:

Dr. Rafael Frederico Alexandre

COORIENTAÇÃO:

Dr. Renato Vimieiro

**Outubro, 2024
João Monlevade–MG**

Pedro Henrique Mendes Rodrigues

Classificação de Gêneros Musicais com Extração Manual de Características

Orientador: Dr. Rafael Frederico Alexandre

Coorientador: Dr. Renato Vimieiro

Monografia apresentada ao curso de Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Outubro de 2024

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

R696c Rodrigues, Pedro Henrique Mendes.

Classificação de gêneros musicais com extração manual de características. [manuscrito] / Pedro Henrique Mendes Rodrigues. - 2024.
55 f.: il.: color., gráf., tab..

Orientador: Prof. Dr. Rafael Frederico Alexandre.

Coorientador: Prof. Dr. Renato Vimieiro.

Monografia (Bacharelado). Universidade Federal de Ouro Preto.
Instituto de Ciências Exatas e Aplicadas. Graduação em Engenharia de Computação .

1. Música. 2. Aprendizado do computador. 3. Identidade de gênero na música. 4. Sistemas de recuperação da informação - Música. I. Alexandre, Rafael Frederico. II. Vimieiro, Renato. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 004:78

Bibliotecário(a) Responsável: Sônia Marcelino - CRB6/2247



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTAÇÃO E SISTEMAS



FOLHA DE APROVAÇÃO

Pedro Henrique Mendes Rodrigues

Classificação de Gêneros Musicais com Extração Manual de Características

Monografia apresentada ao Curso de Engenharia da Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia da Computação

Aprovada em 21 de outubro de 2024

Membros da banca

Dr. Rafael Frederico Alexandre - Orientador - Universidade Federal de Ouro Preto
Dr. Renato Vimieiro - Coorientador - Universidade Federal de Minas Gerais
Me. Zilton José Maciel Cordeiro Junior - Pontifícia Universidade Católica de Minas Gerais

Rafael Frederico Alexandre, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 23/10/2024.



Documento assinado eletronicamente por **Rafael Frederico Alexandre, PROFESSOR DE MAGISTERIO SUPERIOR**, em 23/10/2024, às 10:46, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0799808** e o código CRC **9F490C17**.

Este trabalho é dedicado à minha família.

Agradecimentos

Agradeço aos ótimos professores que tive o prazer de aprender com durante toda a minha vida, que nunca mediram esforços em, não somente ensinar, mas também em apresentar oportunidades, possibilidades, apontar caminhos profissionais e educacionais que eu, muitas das vezes, não seria capaz de enxergar sozinho. Agradeço ao meu orientador, Prof. Dr. Rafael Alexandre, por ter sido presente na elaboração deste trabalho, compartilhando ideias, críticas e por sempre ter me motivado a continuar, independente da situação.

Agradeço à Universidade Federal de Ouro Preto pela qualidade de ensino e profissionalização oferecidos, e também à todas as pessoas que gerem o funcionamento do Instituto de Ciências Exatas e Aplicadas.

E principalmente, agradeço aos meus pais e irmã por sempre me apoiarem incondicionalmente, tornando momentos difíceis mais fáceis, e por terem colocado a educação como prioridade em nossas vidas.

*“Lança o barco contra o mar
Venha o vento que houver
E se virar, nada.”*

— Rubel,
in: O Velho e o Mar.

Resumo

Devido ao crescimento de compartilhamento de mídias digitais de músicas e surgimento de serviços de *streaming*, a área de Recuperação de Informações Musicais (MIR) é formentada pela indústria para desenvolver modelos de classificação de gêneros musicais. Inspirados pelo problema, o primeiro modelo e *dataset* são explorados desde o início dos anos 2000, utilizando técnicas clássicas de *machine learning* e extração manual de métricas. Atualmente, mais de vinte anos depois, devido ao avanço no campo de inteligência artificial presenciado nos últimos anos, a forma com que a classificação de gêneros é realizada sofreu mudanças significativas. Assim, a fim de comparar modelos modernos e clássicos com a utilização de extração tradicional das métricas, este trabalho, baseado na literatura, propõe uma metodologia de análise de trechos das músicas para classificação. Os resultados obtidos mostram que algoritmos tradicionais disputam em desempenho com redes neurais simples quando utilizadas métricas musicais. Outro resultado é a pequena melhora que ocorre na classificação para pequenos trechos da música, diferente do que é concluído em outro trabalho recente da literatura.

Palavras-chaves: classificação de gêneros musicais. recuperação de informações musicais. extração de características. aprendizado de máquina.

Abstract

Due to the growth of digital music media sharing and the emergence of streaming services, the Music Information Retrieval (MIR) area is encouraged by the industry to develop music genre classification models. Inspired by the problem, the first model and dataset have been explored since the early 2000s, using classic machine learning techniques and manual metric extraction. Currently, more than twenty years later, due to the advances in the field of artificial intelligence witnessed in recent years, the way in which genre classification is performed has undergone significant changes. Thus, in order to compare modern and classic models with the use of traditional metric extraction, this work, based on the literature, proposes a methodology for analyzing music excerpts for classification. The results obtained show that traditional algorithms compete in performance with simple neural networks when using musical metrics. Another result is the small improvement that occurs in the classification for small excerpts of the music, different from what is concluded in another recent work in the literature.

Key-words: music genre classification. music information retrieval. feature extraction. machine learning.

Lista de ilustrações

Figura 1	– Exemplificação da ideia da Transformada de Fourier, transformar um sinal em domínio do tempo, com sua intensidade em altura do som, para o domínio da frequência, que realça as principais frequências presentes no sinal. Imagem retirada de (PINTO, 2018).	19
Figura 2	– Espectrograma de trompete. Criado pela biblioteca <i>librosa</i> .	20
Figura 3	– Métricas <i>Spectral Centroid</i> e <i>Spectral Roll-Off</i> para um áudio de trompete sendo tocado. Exemplo retirado da documentação da biblioteca <i>librosa</i> .	23
Figura 4	– <i>Spectral Bandwidth</i> para um áudio de trompete sendo tocado, o primeiro gráfico são os valores obtidos e o segundo o <i>range</i> que este valor gera no centroide. Exemplo retirado da documentação da biblioteca <i>librosa</i> .	24
Figura 5	– Relação logarítmica entre o crescimento das escalas linear e mel.	25
Figura 6	– Processo para obtenção dos coeficientes MFCC.	26
Figura 7	– Utilizando o algoritmo de <i>chroma analysis</i> para obtenção das intensidades em cada frequência é possível destacar a importância de cada nota nos trechos da música. Exemplo retirado da documentação da biblioteca <i>librosa</i> .	27
Figura 8	– Esquema retirado de (DRIEDGER; MÜLLER; DISCH, 2014). (a) faixa de áudio original; (b) espectrograma original; (c) espectrogramas da harmonia, ruído e percussão; (d) faixas de áudio da harmonia, ruído e percussão.	28
Figura 9	– Função <i>sigmoid</i> em azul. Tracejado em vermelho no meio representa o <i>threshold</i> para classificação entre classes. Pontos amarelos são amostras.	30
Figura 10	– Exemplo de árvore de decisão. Imagem retirada do artigo em jeremyjordan.me.	31
Figura 11	– Classificação do KNN. Imagem retirada de artigo na Medium.	31
Figura 12	– Exemplo do funcionamento de uma SVM, as classes são divididas por hiperplano com margem maximizada. Imagem do artigo (BARBOSA et al., 2021).	32
Figura 13	– Modelo <i>Perceptron</i> , imagem de (ROSA; LUZ, 2012).	33
Figura 14	– Exemplo de arquitetura de uma rede neural densa.	33
Figura 15	– Exemplo de arquitetura de uma <i>convolutional neural network</i> . Exemplo retirado de adeshpande3.github.io.	34
Figura 16	– Operação de convolução em camadas de CNN. Exemplo retirado de Medium.	35

Figura 17 – Operação de <i>pooling</i> em camadas de CNN. Exemplo retirado de (AL-JAAFARI, 2018)	35
Figura 18 – Criação do <i>dataset</i> a partir da divisão das músicas em trechos, cada trecho recebe o gênero da música inteira. A divisão dos trechos entre treino e teste não separa trechos da mesma música em conjuntos diferentes.	41
Figura 19 – Criação das amostras de entrada da <i>convolutional neural network</i> , são criadas matrizes de músicas completas, onde cada linha é um trecho.	44
Figura 20 – Arquitetura proposta para CNN.	44
Figura 21 – Curvas de treinamento do modelo DNN. À esquerda o gráfico de erro e na direita a acurácia à medida que as épocas aumentam.	47
Figura 22 – Matriz de confusão dos gêneros dos SVMs.	48
Figura 23 – Curvas de treinamento do modelo CNN com trechos de 1 segundo. À esquerda o gráfico de erro e na direita a acurácia à medida que as épocas aumentam.	49
Figura 24 – Matriz de confusão dos gêneros das CNNs.	50

Lista de tabelas

Tabela 1	– Soluções propostas na literatura.	38
Tabela 2	– Características extraídas e suas dimensões.	41
Tabela 3	– Quantidade de amostras por intervalo temporal.	41
Tabela 4	– Hiperparâmetros dos modelos.	43
Tabela 5	– Médias dos resultados da classificação do primeiro experimento.	46
Tabela 6	– Médias dos resultados da classificação do segundo experimento.	48

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	15
1.2	Metodologia	15
1.3	Organização do trabalho	16
2	REVISÃO BIBLIOGRÁFICA	17
2.1	Recuperação de Informações Musicais	17
2.1.1	Conceitos Fundamentais	18
2.1.2	Extração de <i>Features</i>	19
2.1.2.1	<i>Features</i> de Textura de Timbre	21
2.1.2.2	<i>Mel-Frequency Cepstral Coefficients</i>	25
2.1.2.3	<i>Features</i> de Altura	26
2.1.2.4	<i>Features</i> de Ritmo	27
2.1.2.5	Decomposição Harmônica/Percussiva	27
2.2	Aprendizado de Máquina	29
2.2.1	Regressão Logística	29
2.2.2	Floresta Aleatória	30
2.2.3	K-ésimo Vizinho mais Próximo	31
2.2.4	Máquina de Vetores de Suporte	32
2.2.5	Redes Neurais Densas	32
2.2.6	Redes Neurais Convolucionais	34
2.3	Classificação de Gêneros	35
3	MODELAGEM	39
3.1	Pré-processamento dos dados	40
3.2	Classificação de Gêneros	41
3.2.1	Primeiro Experimento	42
3.2.2	Segundo Experimento	43
4	RESULTADOS	46
4.1	Primeiro Experimento	46
4.2	Segundo Experimento	48
5	CONCLUSÃO	51

REFERÊNCIAS 53

1 Introdução

A popularização ao acesso à Internet no final dos anos 90 e início dos anos 2000 impulsionou a digitalização e compartilhamento de conteúdos que previamente só poderiam ser obtidos por meio da mídia física. Além disto, a difusão de aplicativos que forneciam acesso a redes *peer-to-peer* (P2P) para compartilhamento de arquivos abriu ainda mais espaço para a disseminação destas mídias, que muitas vezes ultrapassavam os limites da lei por conta de direitos de *copyright*.

Tome como exemplo o sucesso do serviço de *streaming Napster*, fundado em 1999, especializado em compartilhamento de arquivos MP3 via redes P2P e que, em seu auge, obteve 80 milhões de usuários registrados¹, que foi encerrado em 2001 devido aos diversos processos judiciais envolvendo a distribuição ilegal de conteúdo das gravadoras detentoras dos direitos.

Diante do sucesso da digitalização musical, (PACHET; CAZALY, 2000), pesquisadores da *Sony*, evidenciam a importância da rotulagem dos gêneros musicais de forma a possibilitar que os serviços de distribuição possam catalogar seu conteúdo e suprir a necessidade dos usuários que buscam por conteúdos semelhantes aos que já consomem. Este trabalho ainda não propõe um processo automatizado para geração de metadados, porém apresenta uma taxonomia - relação hierárquica por meio de características em comum do gêneros -, para servir como base e formentar a pesquisa em classificação musical. Os autores definem 378 gêneros em uma estrutura hierarquizada e escalável.

De fato, atualmente percebemos que metadados se tornaram uma informação valiosa não só para serviços de *streaming*, mas para qualquer empresa. No ramo do áudio digital especificamente, notamos que serviços que se destacam no mercado, como o *Spotify* ou *Last.fm* por exemplo, não se limitam somente ao seu catálogo, mas também oferecem uma gama de serviços de análise dos metadados das músicas e usuários, proporcionando desde recomendações personalizadas até gráficos de comportamento e gostos de seus clientes.

Inspirados pelo problema, (TZANETAKIS; ESSL; COOK, 2001) propõe o primeiro modelo e *dataset* da área de classificação de gêneros musicais. Os autores utilizam algoritmos que atualmente são considerados tradicionais para classificar trechos de 30 segundos das músicas coletadas. O *dataset* GTZAN ainda é utilizado como *benchmark* em trabalhos da área.

Mais recentemente, desde trabalhos como, por exemplo, (LI; CHAN; CHUN, 2010) e (COSTA et al., 2011), onde o primeiro emprega redes neurais convolucionais e, o segundo, espectrogramas (visualizações gráficas de um áudio), novas metodologias e a forma como

¹ *Requiem for Napster*, 2002.

os dados musicais são tratados foram propostos. Esta mudança evidencia a diferença entre os novos processos com os tradicionais, que faziam muitas vezes a extração manual das métricas musicais, ou seja, utilizar conceitos físicos/matemáticos para extração de informações.

1.1 Objetivos

É notável a existência de poucos estudos na literatura que abordam a divisão das músicas em pequenos trechos e, os principais resultados obtidos são significativamente melhores que o estado da arte, mesmo utilizando técnicas mais simples em comparação aos conceitos mais avançados de modelagem e *deep learning* empregados normalmente. Portanto, o presente trabalho consiste em explorar o comportamento para classificação de gêneros musicais de algoritmos tradicionais e de redes neurais com a utilização de métricas extraídas manualmente dos trechos de música do *dataset* GTZAN, utilizando diferentes intervalos temporais, inspirado em trabalhos recentes da área, (SHARMA; UMAPATHY; KRISHNAN, 2020).

Além disto, o trabalho também pretende entender como as *convolutional neural networks* com filtros de uma dimensão lidam ao trabalhar com dados temporais do áudio para os diferentes intervalos considerados, isto contrasta com as recentes utilizações de imagens de espectrogramas para a realização da mesma tarefa, porém com filtros convencionais de duas dimensões. Assim, entender se dados numéricos pré-processados diante esta modelagem podem obter bons resultados faz parte desta pesquisa.

1.2 Metodologia

A fim de obter resultados que agregam informações à área, **(a)** são escolhidos algoritmos tradicionais e de *deep learning* para a tarefa de classificação com base na literatura, ou seja, ferramentas que já apareceram em trabalhos de classificação de gêneros e principalmente em trabalhos que já realizaram divisão do *dataset* em trechos.

Para utilizar as técnicas citadas acima, **(b)** são extraídas métricas musicais comuns de pesquisas semelhantes utilizando conceitos de análise de sinais e criando conjuntos de dados com trechos de 200 milissegundos, 1 e 3 segundos das músicas do GTZAN. **(c)** O processo de treinamento e validação é repetido 15 vezes para cada configuração, com três diferentes *seeds* da metodologia *k-fold cross validation*, que divide os dados em cinco partes (para $k=5$), preservando a frequência dos gêneros em cada divisão. Há também um cuidado na divisão de treino e teste, para impedir que trechos da mesma música estejam em diferentes conjuntos, o que enviesaria o modelo.

Com as tabelas de acurácia dos resultados e erro obtido entre os gêneros, **(d)** é

possível extrair conhecimento quanto as abordagens empregadas neste trabalho, em relação tanto aos intervalos temporais quanto à influência destes na tarefa, trazendo mais clareza à eficiência da classificação de pequenos trechos de música.

1.3 Organização do trabalho

O restante deste trabalho é organizado como se segue. O Capítulo 2 apresenta os principais conceitos da área de recuperação de informações musicais e aprendizado de máquina necessários para modelagem da tarefa proposta e, além disto, também destaca trabalhos relevantes historicamente de classificação de gêneros musicais. O Capítulo 3 discorre sobre a modelagem dos dois experimentos realizados. Por fim, 4 e 5 abordam os resultados coletados a partir da modelagem e conclusões gerais sobre o trabalho e o campo estudado, respectivamente.

2 Revisão bibliográfica

O trabalho desenvolvido foi motivado por diversas literaturas que são provenientes de diversos campos da computação e engenharia. No primeiro caso, é relevante destacar o campo de *Data Mining* (HAN; KAMBER; PEI, 2011), que se demonstra presente durante todo o fluxo de trabalho, trazendo ferramentas para pré-processamento dos dados, modelos de classificação, com técnicas de *Machine Learning* e análise de dados. Já as técnicas empregadas para gerar a base de dados final a partir de um *dataset* de músicas armazenadas como sinais digitais, estas são ferramentas da área de Sinais e Sistemas (OPPENHEIM; WILLSKY; NAWAB, 1996), que utilizam conceitos físicos e matemáticos para extrair atributos que representam o áudio no tempo e/ou frequência, produzindo assim dados tabulares ao invés do áudio bruto.

Este capítulo está organizado da seguinte maneira, a Subseção 2.1 explica os principais conceitos da área de recuperação de informações musicais, principalmente quanto a extração de métricas musicais. A Subseção 2.2 revisa os modelos de aprendizado de máquina utilizados neste trabalho, a fim de criar uma base comum para todos. Finalmente, 2.3 expõe trabalhos relevantes na área de classificação de gêneros musicais desde seu início.

2.1 Recuperação de Informações Musicais

No trabalho de (DOWNIE, 2003) sobre o campo de recuperação de informação musical (*MIR*), o autor fantasia sobre um mundo onde seria possível realizar uma pesquisa de músicas simplesmente cantarolando, obtendo, pelo buscador, vários arquivos MP3 de diferentes versões prontas para serem executadas. Certamente, o conceito de *query by humming*, ou seja, busca por cantarolado em tradução literal, citado neste mundo imaginário, não é um conceito novo, explorado por (GHAS et al., 1995). Porém, é extraordinário notar que vivemos nesta realidade, serviços como *hum to search*¹ da Google são de fácil acesso em qualquer *smartphone*, o que pode ser visto como resultado de pouco mais de 20 anos de pesquisa na área de recuperação de informação musical, juntamente ao desenvolvimento de tecnologias na inteligência artificial.

A área de MIR possui diversos ramos além da classificação de gêneros, na revisão sobre tendências na extração de métricas de áudio de (SHARMA; UMAPATHY; KRISHNAN, 2020), os autores destacam pesquisas como a segmentação de áudio e música, classificação de humor em músicas e classificação de discurso/música. Perceba que estas abordagens já são consideradas clássicas, muito exploradas na década dos anos 2000,

¹ <https://blog.google/products/search/hum-to-search/>

problemas mais recentes envolvem os avanços da inteligência artificial generativa e suas implicações sociais, o *MIREX*² destaca a geração de músicas, geração de música simbólica (que envolve teoria musical e partituras) e detecção de músicas criadas com *deep fake* como abordagens modernas da área.

Desta forma, para que seja possível construir modelos para a área, é necessário ter conhecimento de certas ferramentas de análise de sinais para extração de informações básicas. As próximas subseções discorrem sobre os conceitos fundamentais e extração de *features* musicais.

2.1.1 Conceitos Fundamentais

A análise de sinais musicais é normalmente realizada dentro de dois domínios principais: o *tempo* e a *frequência*. Por lidarmos desde o início de nossas vidas com o conceito de tempo, possuímos uma intuição melhor sobre este, diferente da frequência, que pode ser muitas vezes difícil de se compreender, no entanto, conseguimos coletar deste regime informações mais significativas.

Ao falarmos contra um microfone, o equipamento de captação está transformando a pressão do ar que emitimos em sinais elétricos - ao falarmos, estamos empurrando os gases com diferentes pressões, formatos e intensidades, gerando assim os diferentes sons que podemos reproduzir. Portanto, seria natural trabalhar com o áudio digital analisando a sua intensidade em função do tempo. E de fato, isto foi pesquisado, no entanto, a necessidade de se entender as contribuições dos diferentes valores de frequências no sinal trouxe à tona a outra perspectiva para o campo.

A transformada de Fourier (*TF*), é uma operação matemática que converte sinais no domínio do tempo para o domínio da frequência, o conceito é fundamental na área de sinais e sistemas e é bem definida para análises em tempo contínuo e discreto. Na computação, o algoritmo conhecido como *fast Fourier transform* (*FFT*), é a implementação da TF discreta otimizada, desenvolvida por (COOLEY; TUKEY, 1965), e a contribuição deste trabalho é extremamente relevante para a área de sinais, já que a implementação anterior do algoritmo possui complexidade de tempo de N^2 , enquanto a proposta de $N\log(N)$. A Figura 1 exemplifica a transformação de um sinal do domínio do tempo $s(t)$ para a sua representação no domínio da frequência, $S(\omega)$.

Esta ferramenta se baseia na ideia de que um sinal contínuo e periódico $s(t)$ pode ser aproximado por uma combinação linear de funções senoidais. Assim, $S(\omega)$ representa a intensidade dos valores de frequências utilizados para remontar o sinal original. Tendo em vista que os sinais precisam ser periódicos, torna-se evidente a limitação quanto à aplicação desta operação em sinais da vida real, principalmente os de músicas.

² [Music Information Retrieval Evaluation eXchange](#), principal competição da área de recuperação de informações musicais, que ocorre desde 2005.

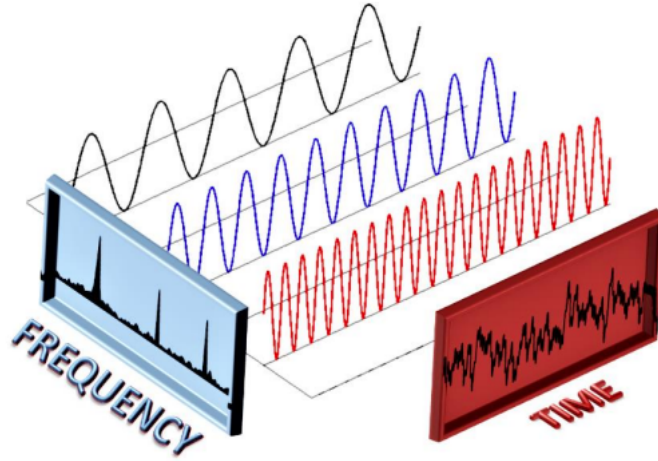


Figura 1 – Exemplificação da ideia da Transformada de Fourier, transformar um sinal em domínio do tempo, com sua intensidade em altura do som, para o domínio da frequência, que realça as principais frequências presentes no sinal. Imagem retirada de (PINTO, 2018).

Assim, de forma que seja possível analisar um sinal contínuo não periódico, foram propostas representações em tempo-frequência. Uma ferramenta crucial no desenvolvimento de trabalhos no campo de análise de áudios é a *short time Fourier transforms* (*STFT*). A ideia é tratar pequenas janelas do sinal de áudio como curtos sinais periódicos e aplicar em cada um destes a TF, obtendo as frequências significativas para os vários intervalos distribuídos no tempo. A formalização desta abstração é $STFT_x^w(t, f)$, onde w é o filtro aplicado no sinal $x(t)$ para obter a janela.

A utilização do *STFT* resulta numa matriz onde as linhas são os valores de frequência e as colunas o eixo temporal. Repare que o eixo temporal não possui o mesmo tamanho do sinal de áudio original, já que as janelas w englobam vários valores amostrados, geralmente numa cobertura entre 20ms e 100ms.

Por fim, uma das mais utilizadas representações de sinais em tempo-frequência são os *espectrogramas*, $SP_t^w(f, t)$, definido pela relação $SP_x^w(t, f) = |STFT_x^w(t, f)|^2$, ou seja, o quadrado do módulo do $STFT_x^w(t, f)$. A Figura 2 exemplifica a visualização de um espectrograma gerado a partir de um áudio de 5 segundos de um trompete sendo tocado, o gráfico possui três dimensões, o eixo vertical representa as frequências em escala logarítmica, em Hertz, o eixo horizontal o tempo, em segundos, e a cor representa o valor de $SP_t^w(f, t)$, a intensidade do sinal, em decibéis, onde o valor mais alto é o de referência, $+0dB$.

2.1.2 Extração de *Features*

São diversas as métricas utilizadas em trabalhos que constroem sistemas de detecção de padrões em músicas. Na realidade, muitas das características utilizadas na literatura

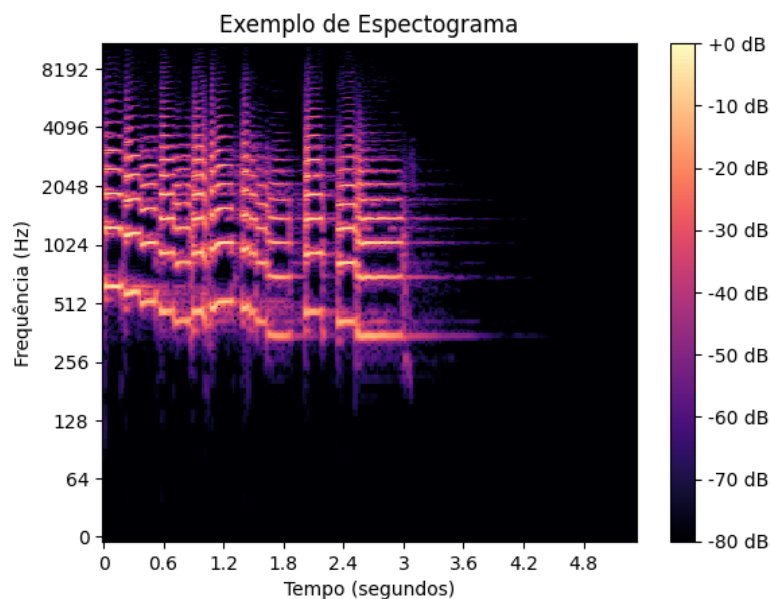


Figura 2 – Espectrograma de trompete. Criado pela biblioteca *librosa*.

clássica de classificação de gêneros datam de trabalhos do século passado, desde a análise de sinais de áudio somente no domínio do tempo ou frequência até métricas que de alguma forma se aproximam do contexto musical, como as da área de *ASR*, *automatic speech recognition*, como trabalhos de reconhecimento de fala, (HUNT; LENNIG; MERMELSTEIN, 1980), e de discriminação entre sinais de fala e música, (SCHEIRER; SLANEY, 1997).

As *features* extraídas são valores numéricos de trechos de áudio que têm como objetivo capturar padrões que ocorrem naquele pedaço da música. Desta forma, estes dados armazenam informações relevantes e compactas que podem servir como base para análise e processamento no futuro. Quando se trata de recuperação de informações de áudio, estes números coletados retratam comportamentos que podem ser muitas vezes subjetivos para o analista, sobretudo se este não estiver inserido no contexto de teoria musical e/ou análise de sinais, portanto se apoiar na literatura em busca de um bom conjunto de métricas é essencial, além de entender o comportamento extraído em cada caso.

Assim, pioneiros na área de classificação automatizada de gêneros musicais, (TZANETAKIS; COOK, 2002) empregam *features* dos trabalhos de *ASR*, citados anteriormente, para representar características de trechos das músicas de seu *dataset*, cunhado de *GTZAN*, juntando as iniciais do primeiro e segundo nome de um dos autores. Este trabalho explora diferentes modelos estatísticos *off-the-shelf* da época em combinação com diferentes conjuntos de métricas para classificar as músicas e comparar os desempenhos. Além disto, os autores separam suas *features* em três classes distintas. *Estrutura rítmica*, que diz respeito ao comportamento de ritmos, batidas, andamento da música, e por isto é considerado uma análise mais global que as demais métricas do áudio (ou pelo menos localmente mais

abrangente que as demais). *Conteúdo tonal*, referente à frequência de um trecho do som, que caracteriza as notas, acordes. E a terceira, *textura timbral*, que tenta descrever as características de timbre de um áudio, pode-se dizer que esta é a mais arduosa tarefa entre as três, já que não se sabe com exatidão quais fatores físicos implicam nos diferentes aspectos dos timbres.

Em sua extensa revisão bibliográfica sobre tendências em extração de *features* em sinais de áudio, (SHARMA; UMAPATHY; KRISHNAN, 2020) destaca outro tipo de classificação das métricas, contudo, desta vez, com foco no desenvolvimento histórico do campo. Os autores ressaltam quatro naturezas. *Métricas do domínio do tempo* foram desenvolvidas durante toda a década dos anos 1950 e estão associadas ao estudo do ritmo de um áudio, são as mais simples, porém utilizadas até hoje. *Métricas do domínio da frequência* fora uma evolução natural da análise de sinais, sendo os principais algoritmos elaborados desde o início dos anos 1950 até o fim da década de 1960. Em vista da análise destes domínios separadamente, eventualmente houve a necessidade de explorar as *Métricas do domínio tempo-frequência*, juntando ambas, a fim de entender o comportamento temporal da frequência e procurando evitar a perda de informação, tendo em vista que existem diversas complicações na transformação de um domínio para outro; estas métricas foram desenvolvidas desde os anos 1960 até o fim do século e aparentam descrever bem o timbre. Por fim, as chamadas *deep features*, métricas criadas por algoritmos de aprendizado profundo também vêm sendo exploradas desde o início dos anos 2000.

Nas seguintes subseções serão definidas as métricas utilizadas neste trabalho, seguindo a classificação de *features* de timbre, ritmo e tom do áudio. As definições a seguir, bem como outras métricas de áudio que vão além do contexto deste trabalho, podem ser aprofundados matematicamente e fisicamente em (KLAPURI; DAVY, 2006).

2.1.2.1 *Features* de Textura de Timbre

O timbre é normalmente explicado como a característica do som que permite que possamos distinguir e identificar as diferentes fontes emissoras. Existem outras propriedades do som que conseguimos quantificar com precisão, como a frequência, duração, ritmo; e se diferentes fontes, como uma guitarra e piano, por exemplo, produzirem sons com os mesmos valores destas características, ainda saberíamos distinguir qual é qual, isto é devido ao timbre único que cada fonte possui.

Na análise de (MCADAMS, 2013), o autor destaca dois marcantes aspectos deste fenômeno, que são conclusão de 40 anos de desenvolvimento do campo. O primeiro é que o timbre é dependente de vários atributos que envolvem a percepção acústica humana, tanto contínuos quanto discretos, alguns destes já são conhecidos e quantificados, porém ainda há o que ser explicado. Já o segundo se trata justamente do timbre ser o fator responsável por conseguirmos diferenciar emissores.

As *features* aqui apresentadas capturam algumas das características do timbre, o que é importante para descrever gêneros musicais: muitas vezes um conjunto de instrumentos ou tipos de vozes específicas são recorrentes em certos gêneros, portanto gerar métricas do comportamento psicoacústico do áudio é relevante para a criação de modelos de classificação.

Antes de tudo, é importante ressaltar que os valores de energia do espectrograma, aqui denotado por $M_t(n)$, lê-se magnitude da energia da frequência n em uma janela de análise t , são normalizados antes de serem realizados os cálculos.

A métrica conhecida como *Spectral Centroid*, SC , vem sido extensivamente utilizada na literatura de recuperação de informação musical por ser considerada uma *feature* que descreve muito bem a qualidade do timbre, demonstrando uma forte correlação com esta característica (SCHUBERT; WOLFE, 2006). Um dos aspectos do timbre diz respeito ao brilho deste som, conceito para categorizar como são os timbres, podemos, por exemplo, chamar de *claro* um timbre mais estridente, penetrante, com mais facilidade para propagar sons mais agudos, enquanto *escuro* seria o outro lado desta balança, um som mais aveludado, suave, mais grave; o *Spectral Centroid* consegue representar bem este comportamento.

A definição matemática desta métrica é descrita na Equação 2.1, onde, t são as janelas do *STFT* aplicado no sinal original, $n \in N$ são as frequências discretizadas e $M_t[n]$ é a magnitude da frequência n na janela de tempo t , assim são obtidos um conjunto de valores SC_t para um sinal completo. Nas literaturas, este conceito é normalmente comparado ao conceito físico de centro de massa de um objeto, que possui como objetivo encontrar o ponto naquele corpo onde podemos concentrar toda sua massa, ou seja, uma média ponderada onde cada ponto é pesado pela sua respectiva massa.

$$SC_t = \frac{\sum_{n=1}^N M_t[n] * n}{\sum_{n=1}^N M_t[n]} \quad (2.1)$$

Sabendo disto, o resultado da métrica são justamente os pontos médios, que são frequências, de acordo com a intensidade/altura do som para cada trecho analisado - geralmente são intervalos muito pequenos, na faixa de poucos milésimos de segundo. Observe a Figura 3, nela está plotado como um traço branco os valores da *feature* analisada para o áudio de cinco segundos de um trompete.

Outra métrica frequentemente utilizada é o *Spectral Roll-Off*, SR , que diz respeito à intensidade cumulativa das frequências, representando o comportamento da forma do espectro. Perceba que na Equação 2.2, a variável SR_t não se encontra isolada, mas sim no limite superior do somatório, desta maneira, facilitando a interpretação de que se busca a frequência SR_t tal que a porcentagem $0 \leq k \leq 1$ do somatório total das intensidades esteja concentrada neste intervalo de $[0, SR_t]$; normalmente o valor de k é escolhido como 0.85. Ainda na mesma Figura, 3, é possível compreender o valor da frequência de Roll-Off,

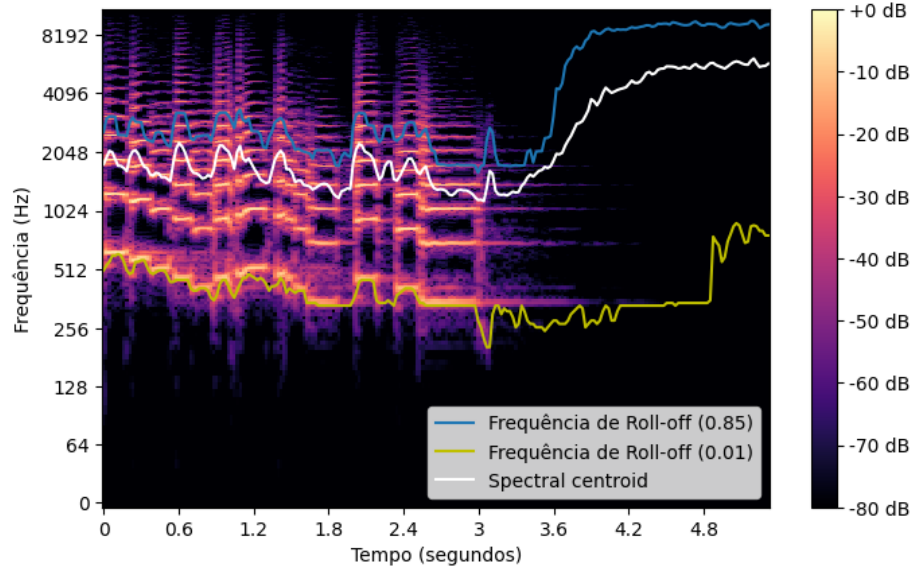


Figura 3 – Métricas *Spectral Centroid* e *Spectral Roll-Off* para um áudio de trompete sendo tocado. Exemplo retirado da documentação da biblioteca *librosa*.

perceba que as linhas azul e amarela, correspondentes aos valores de k igual à 0.85 e 0.01, respectivamente, acumulam a intensidade total desejada até certo valor de frequência, que é o valor que se busca encontrar, para cada janela t .

$$\sum_{n=1}^{SR_t} M_t[n] = k * \sum_{n=1}^N M_t[n] \quad (2.2)$$

O *Spectral Bandwidth*, SB , ou também chamado somente de *Bandwidth* ou Banda Larga Espectral, em português, possui uma relação muito íntima ao conceito de *Spectral Centroid*. Enquanto o último caracteriza o comportamento médio das frequências quanto às suas intensidades, deixando de lado possíveis concentrações ou dispersões dos dados, esta nova métrica procura preencher esta fresta, capturando então a variância de intensidade das frequências em função da distância à média, que neste caso é justamente o centroide espectral.

A fim de fornecer uma interpretação mais clara deste conceito, a Equação 2.3 define formalmente a *feature*, onde SC_t se refere ao valor do *Spectral Centroid* para a janela de análise em questão. Tendo em vista de que SC_t é a média ponderada da energia do espectro, o que está sendo feito é ponderar o erro quadrático, ou seja, a distância da frequência n até à média, com a energia contida nesta frequência, isto para todos os valores em N ; a fórmula captura a variância de intensidades nas frequências em comparação à média. Assim temos que, para espectrogramas com valores altos na intensidade para várias frequências no espectro, o valor da métrica de *bandwidth* é alto, enquanto para energias concentradas em volta de uma única frequência fundamental, por exemplo, este valor é baixo. A Figura 4 mostra um gráfico da propriedade.

$$SB_t = \sqrt{\sum_n^N M_t(n)(n - SC_t)^2} \quad (2.3)$$

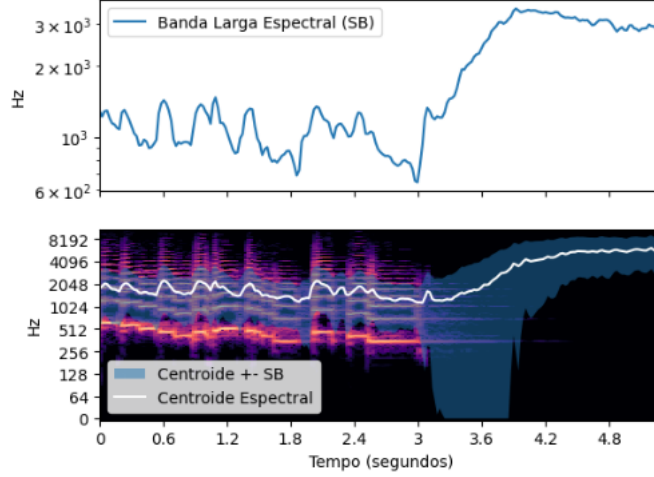


Figura 4 – *Spectral Bandwidth* para um áudio de trompete sendo tocado, o primeiro gráfico são os valores obtidos e o segundo o *range* que este valor gera no centroide. Exemplo retirado da documentação da biblioteca *librosa*.

A métrica conhecida como *Zero Crossing Rate* reflete a quantidade de vezes que um sinal de áudio cruza o eixo x em uma janela de análise t . A Equação 2.4 verifica, no domínio do tempo, se a amostra n possui sinal diferente da amostra anterior $n-1$, assim, sempre que a condição se satisfaz, soma mais um valor de mudança de sinal; ao final, o valor obtido pelo somatório é dividido em função do tamanho de amostras N , para obter um valor que representa a taxa de mudança ZCR_t . Diferente das *features* apresentadas anteriormente, o *Zero Crossing Rate* não é calculado no domínio da frequência, porém consegue descrever bem a quantidade de energia de frequências altas no sinal, trazendo indiretamente informação sobre o espectro e se relacionando fortemente com o centroide espectral, (KLAPURI; DAVY, 2006).

$$ZCR_t = \frac{1}{N} \sum_{n=1}^N |sign(x[n]) - sign(x[n-1])| \quad (2.4)$$

A métrica *root mean square* (*RMS*) pode ser calculada a partir do espectrograma de um áudio ou diretamente deste próprio sinal temporal. Esta *feature* é associada à percepção humana de volume, ou *loudness*. A Equação 2.5 demonstra o cálculo do valor dentro de uma janela t com N amostras.

$$RMS_t = \sqrt{\frac{1}{N} \sum_{n=1}^N x^2[n]} \quad (2.5)$$

2.1.2.2 Mel-Frequency Cepstral Coefficients

O conjunto de valores conhecidos como *mel-frequency cepstral coefficients*, *MFCCs*, são obtidos por meio de uma modelagem matemática baseada na percepção humana do áudio (LOGAN, 2000). A maneira como processamos sons ocorre de maneira logarítmica, a distinção entre ondas de 200Hz e 210Hz é muito mais fácil do que entre ondas de 800Hz e 810Hz, sendo que são exatamente os mesmos 10Hz de diferença entre estes pares de sinais. Um exemplo claro desta natureza humana é como categorizamos as notas musicais, o crescimento da frequência das notas entre diferentes oitavas é exponencial - mais detalhes sobre esta característica na subseção 2.1.2.3.

Em vista disto, foi desenvolvida uma escala alternativa da frequência, baseada na percepção humana aos níveis sonoros, que é mais sensível a baixas frequências. A escala *mel*, nome devido à palavra *melody*, é não linear e possui crescimento logarítmico, sendo formalmente definida de acordo com a Equação 2.6, onde k corresponde à escala linear da frequência que normalmente é utilizada. O gráfico da Figura 5 demonstra esta relação, veja que no início da curva, os valores entre as escalas são próximos, 1000 mels e 1kHz correspondem à mesma percepção sonora.

$$k_{mel} = 2595 \log \left[\frac{k}{700} + 1 \right] \quad (2.6)$$

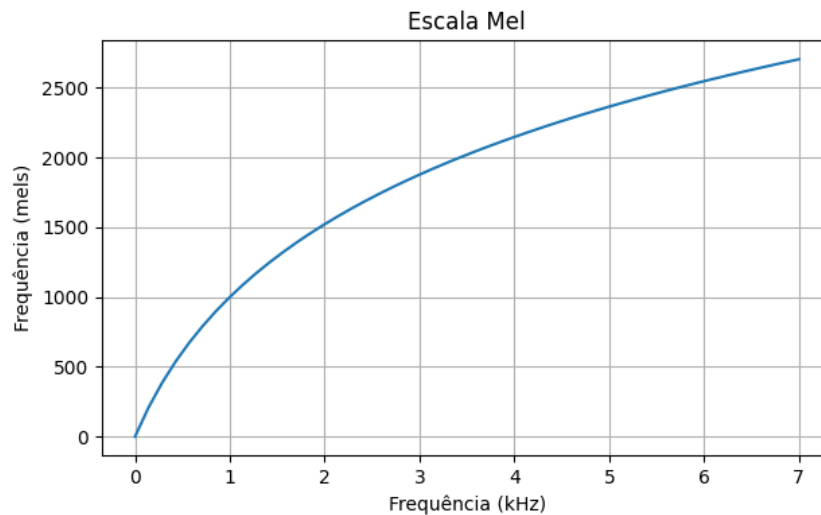


Figura 5 – Relação logarítmica entre o crescimento das escalas linear e mel.

Utilizando este conceito, é criado o *mel-scale filter-bank*, um conjunto de filtros triangulares de sinais linearmente espaçados no domínio da escala mel, utilizados para extrair coeficientes de um trecho de sinal que está sendo analisado já no domínio da frequência, realçando a qualidade nas partes consideradas de melhor percepção auditiva humana. Algumas etapas ainda são realizadas antes de voltar para o domínio do tempo, a fim de melhorar a qualidade dos valores obtidos e remover o correlacionamento do conjunto.

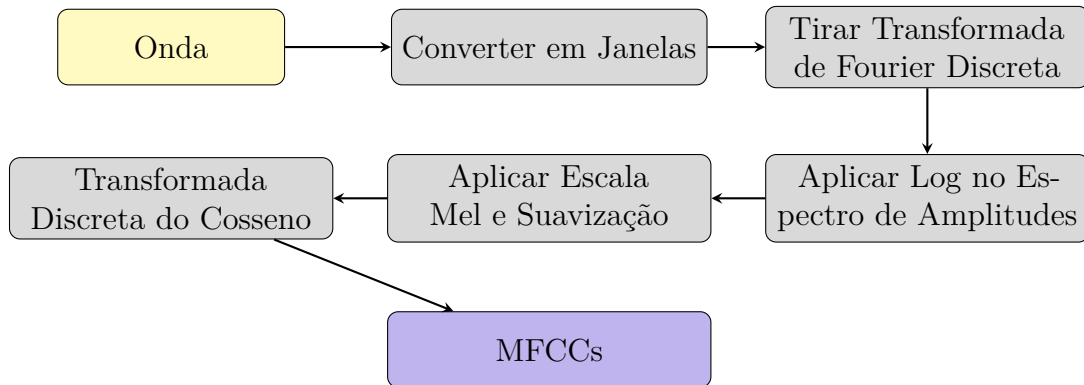


Figura 6 – Processo para obtenção dos coeficientes MFCC.

Ao final, n coeficientes são extraídos e fornecem informações úteis orientadas à percepção sobre frequência e tempo, observe a Figura 6.

2.1.2.3 Features de Altura

A característica de um som que diz respeito à percepção humana em relação às diferentes frequências é chamada de *altura*. A maneira como entendemos a música no ocidente do globo terrestre caracteriza 12 diferentes notas musicais (C, C#, D, ..., B), que são distribuídas em sequência formando uma oitava. Assim existem várias oitavas, formadas das mesmas notas, porém a frequência de cada nota dobra de uma oitava n para a oitava $n + 1$, justamente porque a percepção humana dos sons é de natureza logarítmica.

Então, para exemplificar, seja a nota dó, representada pela letra C, na primeira oitava, seu valor é de $C_1 = 32.7Hz$, já na próxima oitava terá valor de $C_2 = 65.41Hz$. De maneira prática, uma nota em dada oitava sempre reproduzirá a mesma frequência classificada na teoria, e diferentes instrumentos possuem diferentes limites de representação das notas, sendo uns capazes de reproduzir mais frequências que outros - inclusive a própria voz humana possui capacidade limitada de reproduzir as oitavas, o que é evidente, já que praticamente falamos de frequências.

Na literatura, as métricas referentes à altura das notas musicais, ou *pitch classes* em inglês, são comumente nomeadas de *chroma features*. Assim, utilizando algoritmos especializados na extração destas métricas, uma das informações que pode ser útil para classificação de gêneros é a energia presente em cada uma das notas, somando os valores destas em todas as oitavas que aparecem.

Diante disto, a fim de extrair a métrica aqui chamada de *chroma vector*, um vetor contendo valores normalizados das intensidades das 12 notas musicais dentro de uma janela de análise t , é preciso utilizar o algoritmo de STFT e analisar a intensidade nas notas nas diferentes oitavas. A Figura 7 mostra dois gráficos, o de cima é o espectrograma de um sinal de áudio, enquanto o de baixo é referente ao *chroma*, perceba que existe muito mais informação referente às notas ao se utilizar o algoritmo especializado nesta extração.

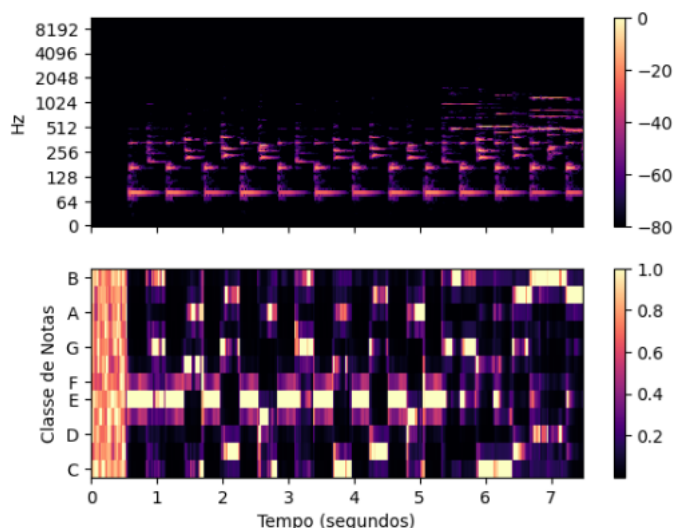


Figura 7 – Utilizando o algoritmo de *chroma analysis* para obtenção das intensidades em cada frequência é possível destacar a importância de cada nota nos trechos da música. Exemplo retirado da documentação da biblioteca *librosa*

A biblioteca *Librosa*, utilizada na execução deste trabalho, emprega algoritmos externos para a tarefa de *chroma feature extraction*³.

2.1.2.4 Features de Ritmo

O ritmo de um áudio está associado a padrões que se repetem ao longo do trecho de áudio analisado, sendo assim, as *features* que buscam representar este comportamento abrangem um escopo maior de análise do que a dos comportamentos anteriores; um padrão repetitivo pode ser um elemento percussivo ao longo de segundos na música, por exemplo, ou, não se limitando a instrumentos, também podem haver padrões recorrentes em som ambiente e discurso, como o canto de um pássaro ao fundo ou em poesias, (SHARMA; UMAPATHY; KRISHNAN, 2020).

A métrica de *tempo* descreve a quantidade de batidas por minuto, BPM, que uma música possui, trazendo informações sobre a velocidade em que ela é tocada. De maneira intuitiva, ao se ouvir uma música, é possível perceber a existência de uma relação entre a velocidade dela e seu gênero musical, dependência que é relevante para a tarefa de classificação.

2.1.2.5 Decomposição Harmônica/Percussiva

Músicas são constituídas pelos elementos harmônicos e percussivos presentes em sua composição. A *harmonia* da música se refere às combinações de acordes/notas sendo tocadas continuamente no tempo, gerando a melodia da música; este elemento é responsável por trazer profundidade à música, com guitarras, violinos, pianos, baixos, por exemplo,

³ Ellis, Daniel P.W. “Chroma feature analysis and synthesis” 2007/04/21

como instrumentos. Já a *percussão* está associada ao ritmo da música, o elemento que dita a velocidade e energia sentida pelo ouvinte, instrumentos deste tipo de componente são as baterias, tambores, pandeiros e outros.

No trabalho de (ONO et al., 2008), foi proposto uma maneira intuitiva de separar as componentes harmônicas e percussivas de uma faixa de áudio. Ao analisar o espectrograma de um trecho de música, é possível observar a existência de faixas mais intensas de energia horizontais e verticais, sendo a primeira correspondente à harmonia e a última à percussão. Assim, criando dois novos espectrogramas a partir do original, um removendo as partes verticais e no outro as horizontais, ao retornar para ao domínio do tempo, o resultado seriam faixas de áudio com as componentes separadas.

Observe a Figura 8, neste caso, é possível ver a transformação de uma faixa de áudio no domínio do tempo para o seu espectrograma, então a separação de três novos espectrogramas: a componente *harmônica*, o ruído e a componente *percussiva*, respectivamente; por fim, volta-se ao domínio do tempo para obter o áudio. Este esquema é proposto pelo trabalho de (DRIEDGER; MÜLLER; DISCH, 2014), que propõe um esquema de separação das componentes baseado na ideia apresentada anteriormente, porém adicionando uma variável relacionada ao ruído.

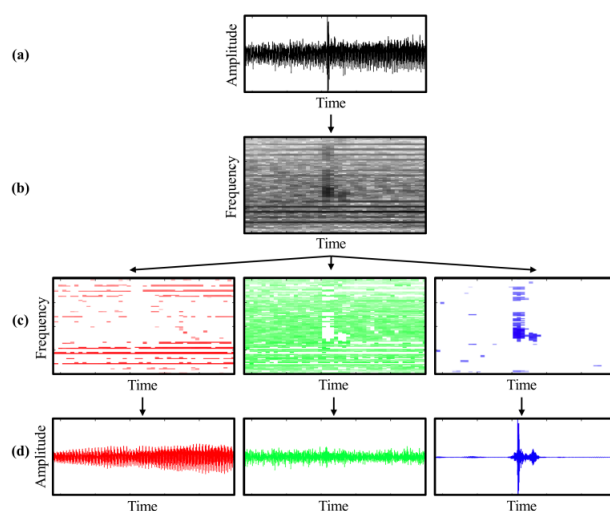


Figura 8 – Esquema retirado de (DRIEDGER; MÜLLER; DISCH, 2014). (a) faixa de áudio original; (b) espectrograma original; (c) espectrogramas da harmonia, ruído e percussão; (d) faixas de áudio da harmonia, ruído e percussão.

Com estas duas novas faixas de áudio em mãos, é possível tratá-las como a faixa original e realizar a extração de *features* de maneira arbitrária. No trabalho de (AGUIAR; COSTA; NANNI, 2016), por exemplo, a partir da separação das componentes, os autores trataram os espectrogramas e fizeram a extração, treinando modelos de predição de gêneros com estes dados. Como resultado, utilizando no mesmo conjunto de treinamento os espectrogramas originais com os outros dois extraídos, observou-se uma modesta melhora na acurácia do modelo.

2.2 Aprendizado de Máquina

Os algoritmos revisados nas próximas subseções pertencem à classe de modelos de aprendizado supervisionado, tipo de abordagem bem conhecida e difundida na área de inteligência artificial. Esta classe é caracterizada por ser treinada em uma base de dados que inclui necessariamente o valor correto do atributo que pretende ser predito/classificado para cada amostra, de forma que o modelo seja capaz de avaliar seu desempenho em cada iteração e ajustar seus parâmetros com base nos erros em comparação ao valor correto.

Ademais, existem duas tarefas triviais na área de aprendizado de máquina, a classificação e a predição de valores. A primeira ocorre quando o atributo em que o modelo é focado em aprender o comportamento é categórico, ou seja, possui uma quantidade limitada de valores que pode assumir, existem classes que cada amostra pode ser e, para cada classe, o algoritmo tenta encontrar a melhor configuração que classificará novas amostras em uma das possibilidades. Já para o segundo tipo de tarefa, a predição, o modelo deve entender o comportamento de um atributo contínuo, um número que pode assumir qualquer valor. Exemplos destas tarefas seriam, tentar classificar se uma pessoa está apta para receber um empréstimo com valores [SIM,NAO] ou prever um valor máximo $x \in \mathbb{R}^+$ que ela poderá receber.

Alguns dos modelos apresentados nesta seção realizam a classificação binária, ou seja, somente funcionam quando o *dataset* contém duas classes, sejam estas de fato grupos distintos que ocorrem no problema, e.g. [rock, samba], ou valores [VERDADEIRO,FALSO]. Então, como o problema deste trabalho possui 10 gêneros, a fim de utilizar estes algoritmos com limitação à classificação binária, será adotada a adaptação *One-Vs-Rest*, OvR, modelagem que permite que modelos binários classifiquem *datasets* multiclasse. Para tal, para um banco de dados com n gêneros, o modelo é treinado n vezes, onde, para cada iteração, o modelo verifica se a amostra é ou não a classe i . Ao final, o modelo com o valor de maior confiança é utilizado para definir a classificação.

Os algoritmos de aprendizado aqui apresentados, exceto o KNN, possuem parâmetros que à princípio são randômicos, porém com o passar das épocas de treinamento, o valor destes parâmetros são alterados de forma que o modelo classifique as amostras de maneira correta. A atualização destes valores é realizada baseada em uma função de custo f_c , que metrifica os erros obtidos das previsões em relação às *labels* originais, diminuir o valor de f_c enquanto se altera os parâmetros do modelo é um problema de otimização.

2.2.1 Regressão Logística

A *logistic regression*, ou regressão logística, é um algoritmo de classificação binária que utiliza a função *sigmoid* para obter a probabilidade de uma nova amostra pertencer à classe x ou y . A Figura 9 mostra uma curva *sigmoid* ajustada à um *dataset* de amostras

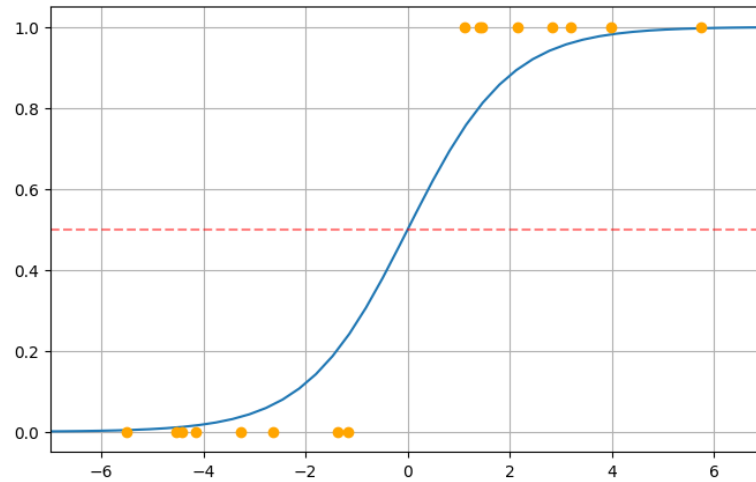


Figura 9 – Função *sigmoid* em azul. Tracejado em vermelho no meio representa o *threshold* para classificação entre classes. Pontos amarelos são amostras.

(pontos laranjas) que são divididas no plano no 0 e 1, de acordo com seu comportamento original. A linha tracejada em vermelho é o valor de *threshold* utilizado para definir se uma nova amostra, quando ajustada na curva de acordo com o atributo do eixo x, será classificada como 0 ou 1.

Dado um conjunto de atributos $X = (x_1, x_2, \dots, x_k)$ que definem uma amostra, esta será classificada utilizando a Equação 2.7, onde o valor de z é dado por 2.8, e os valores β são os parâmetros a serem ajustados no treinamento.

$$P(y = 1|X) = \text{sigmoid}(z) = \frac{1}{1 + e^{-z}} \quad (2.7)$$

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k \quad (2.8)$$

2.2.2 Floresta Aleatória

As *decision trees* são algoritmos que realizam seu treinamento dividindo o espaço amostral de acordo com a maneira com que os valores dos atributos dos dados se relacionam com as amostras a serem classificadas, por meio de uma métrica chamada de *Information Gain*, que utiliza os dados estatísticos e entropia entre diferentes nós da árvore para entender onde será a próxima divisão.

Observe a Figura 10, a árvore de decisão possui este nome justamente pela estrutura de árvore que ela gera a partir dos diferentes cortes realizados no espaço de amostras, onde os nós pai e internos da árvore definem as regras de divisão. As folhas da árvore dizem respeito à classe classificada caso uma nova amostra navegue pelos nós e chegue até esta folha. Perceba que o modelo evidencia as regras utilizadas para classificação, fazendo com que este sejam um ótimo algoritmo para explicação da importância das *features*.

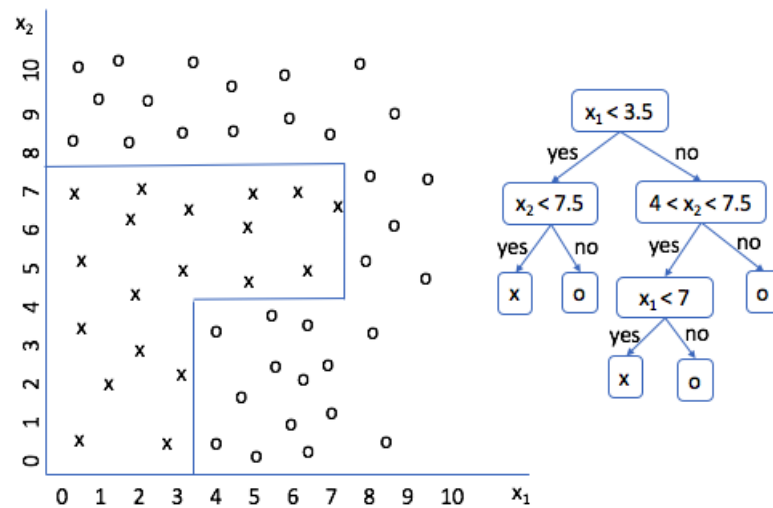


Figura 10 – Exemplo de árvore de decisão. Imagem retirada do artigo em jeremyjordan.me.

No caso do algoritmo de floresta aleatória, ou *random forest*, diversas árvores de decisão são instanciadas e diferentes subconjuntos do *dataset* são utilizados como treinamento para estas árvores, o que aumenta o desempenho do modelo consideravelmente. Uma votação é efetuada pelas diferentes árvores criadas ao final do processamento.

2.2.3 K-ésimo Vizinho mais Próximo

O algoritmo do k-ésimo vizinho mais próximo, do inglês *k-nearest neighbor* (KNN), pode ser considerado o algoritmo mais simples dos que foram utilizados neste estudo. Este modelo não passa por um processo de treinamento e, para classificar uma nova amostra, as classes dos k vizinhos mais próximos desta amostra ditam, por maioria, qual vai ser a classe da nova amostra. Na Figura 11 é possível ver este processo. A nova amostra, em vermelho, será classificada como classe B se $k=3$ e classe A se $k=6$.

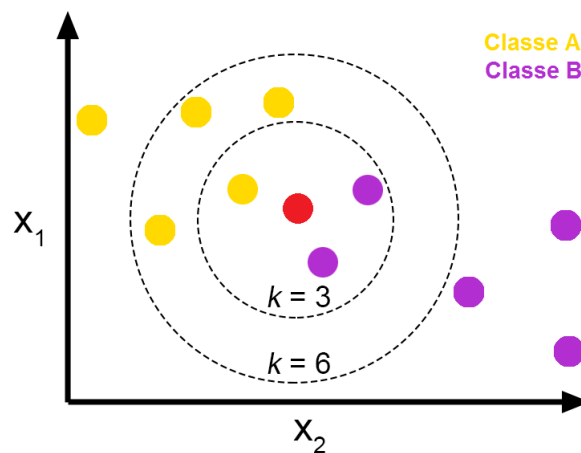


Figura 11 – Classificação do KNN. Imagem retirada de artigo na [Medium](https://www.medium.com).

2.2.4 Máquina de Vetores de Suporte

As *support vector machines* (SVM) foram introduzidas por (VAPNIK, 1995), técnica relativamente recente se considerarmos sua base matemática robusta. São soluções que, assim como as regressões logísticas, classificam dados binariamente, ou seja, neste caso é preciso aplicar o método OvR descrito no início da seção.

A ideia do algoritmo é dispor as amostras em um espaço n -dimensional, onde n é a quantidade de *features* que descrevem uma amostra, e então dividir as classes utilizando um hiperplano, nome dado ao plano para qualquer dimensão, ele possui dimensão de $n-1$.

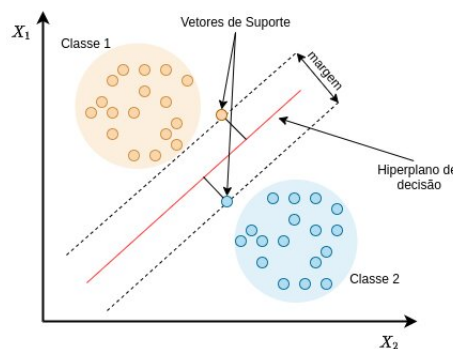


Figura 12 – Exemplo do funcionamento de uma SVM, as classes são divididas por hiperplano com margem maximizada. Imagem do artigo (BARBOSA et al., 2021)

A Figura 12 mostra a divisão das amostras da Classe 1 e Classe 2 utilizando o modelo. Perceba que o problema de otimização se dá no posicionamento do hiperplano, de forma a maximizar a distância deste com as amostras mais próximas de cada classe, ou seja, aumentar o valor de *margem* para a amostra no extremo mais próximo da divisão. Estas amostras mais próximas ao hiperplano possuem vetores normais à divisão, de forma que, geometricamente, o modelo ajuste a reta.

Em casos onde os dados não podem ser linearmente separáveis no espaço, utiliza o conceito de *kernels*, em que aplica a transformação vetorial do espaço, aumentando as dimensões para que a divisão seja possível. Um parâmetro importante é o valor de C , que dita a troca entre generalização e taxa de acertos do algoritmo, de forma que valores altos deste atributo faz com que o modelo seja mais penalizado em relação a erros, enquanto valores baixos permitem mais erros em troca de uma boa generalização pelo posicionamento do hiperplano.

São inúmeros os problemas que empregam como solução a utilização da SVM, é um modelo proeminente da computação com vários casos de sucesso comprovados.

2.2.5 Redes Neurais Densas

As redes neurais são modelos de aprendizagem criados a partir do neurônio artificial conhecido como *Perceptron*, ilustrado na Figura 13. Este algoritmo foi proposto no século

passado e é inspirado no funcionamento do neurônio biológico. Como é mostrado no modelo, um neurônio possui n entradas e cada uma destas é pesada pelos valores w_n , chamado de *bias*, o valor w_0 é somado à equação, que é inserida numa função de ativação, responsável por adicionar o elemento de não linearidade ao problema. A ideia é ajustar os valores de todos os pesos e o *bias*, de forma que a função de ativação seja algebricamente modificada a se encaixar nos dados de treinamento, fornecendo uma saída acurada para novos dados.

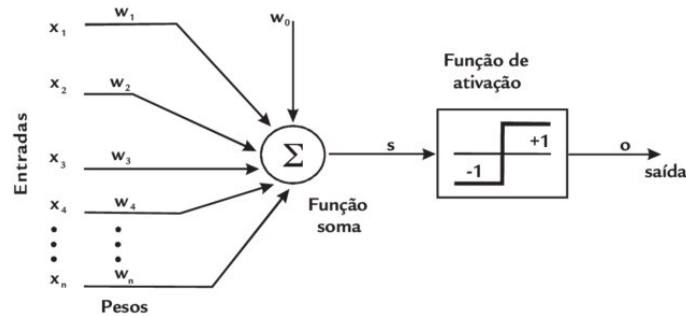


Figura 13 – Modelo *Perceptron*, imagem de (ROSA; LUZ, 2012).

As redes neurais são arquiteturas criadas juntando vários neurônios *Perceptron* uns aos outros, e nas redes densas, também chamadas de *multilayer perceptron*, existe conexão entre todos os nós de camadas vizinhas. Este é um algoritmo bem flexível, do ponto de vista de que é possível realizar processamento de classificação e regressão sobre qualquer input e output de dados, bastando somente configurar a arquitetura necessária para tal.

Na Figura 14 é possível observar as diferentes camadas do modelo, a primeira, em amarelo é a *input layer*, onde os dados são inseridos para treinamento ou classificação, as internas em azul são as *hidden layers*, onde o processamento mais complexo dos padrões dos dados é realizado e, em vermelho, as *output layers*, onde os resultados do algoritmo são encontrados.

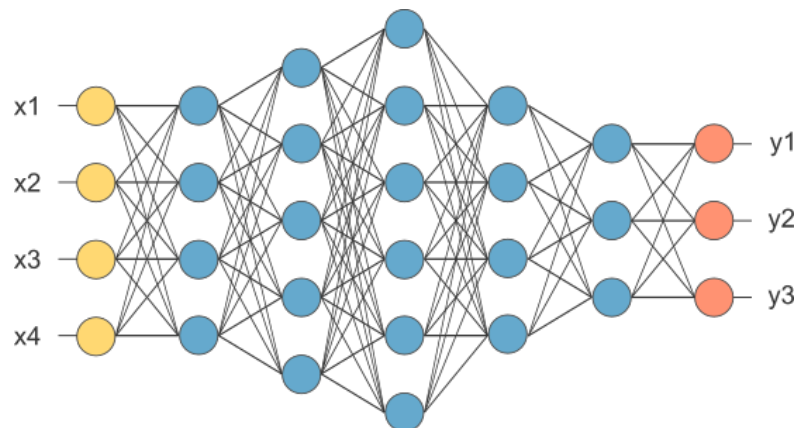


Figura 14 – Exemplo de arquitetura de uma rede neural densa.

2.2.6 Redes Neurais Convolucionais

Diferente das redes neurais densas, as redes neurais convolucionais são caracterizadas por possuírem camadas convolucionais em sua arquitetura. Observe a 15, as primeiras duas camadas são responsáveis por extrair *features* do comportamento da entrada, em seguida, uma rede neural densa continua o trabalho em cima dos valores extraídos pelas camadas convolucionais.

Esta arquitetura foi desenvolvida para trabalhar com dados bidimensionais e com forte relação entre valores vizinhos, como imagens. Neste trabalho foi utilizado a convolucional 1D, onde é feita a análise de relação entre vizinhos de maneira unidimensional.

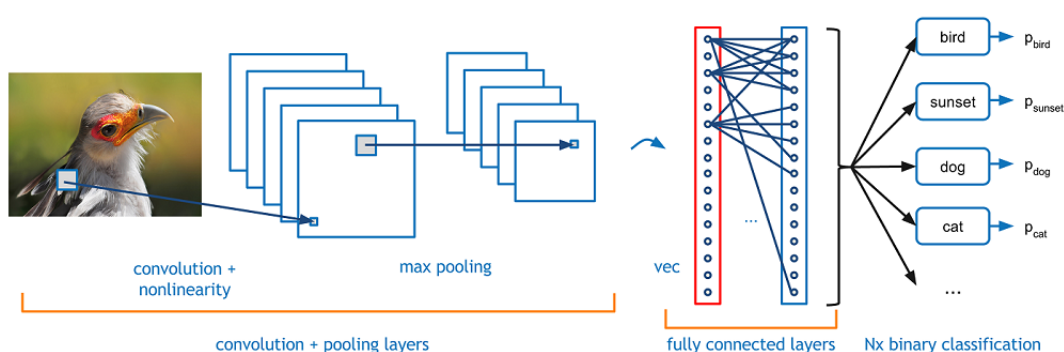


Figura 15 – Exemplo de arquitetura de uma *convolutional neural network*. Exemplo retirado de [adeshpande3.github.io](https://github.com/adeshpande3)

Perceba que existem duas operações principais nas camadas das *convolutional neural networks*, a convolução e o *pooling*. A operação de convolução é ilustrada na Figura 16, onde n filtros/kernels, valor definido na arquitetura, em cada iteração do treinamento convoluem sobre a saída da camada anterior, ou sobre o *input* da rede caso seja a primeira camada, convolucionar é o ato de multiplicar o peso de cada posição do filtro pelo respectivo valor na matriz original e somar todos os valores obtidos, o resultado desta operação é uma matriz menor, caso não seja realizado o *padding*, ou seja, acrescentar zeros ao redor da entrada para que a saída possua as mesmas dimensões. A atualização dos pesos/*bias* que acontece nas redes neurais densas, também acontece sobre os valores em cada filtro, de forma que estes aprendam a extrair padrões bidimensionais importantes da entrada.

Na Figura 17, as duas principais operações de *pooling* são demonstradas. Esta ferramenta serve como uma camada de simplificação dos dados, de forma a diminuir a dimensão dos dados de entrada. O *Max Pooling* é uma função que recebe um conjunto de valores e retorna somente o maior destes, o *Average Pooling* tira a média dos valores. Perceba que o tamanho do *kernel* destas operações são arbitrários e quanto maior for este valor, menor será a saída da camada.

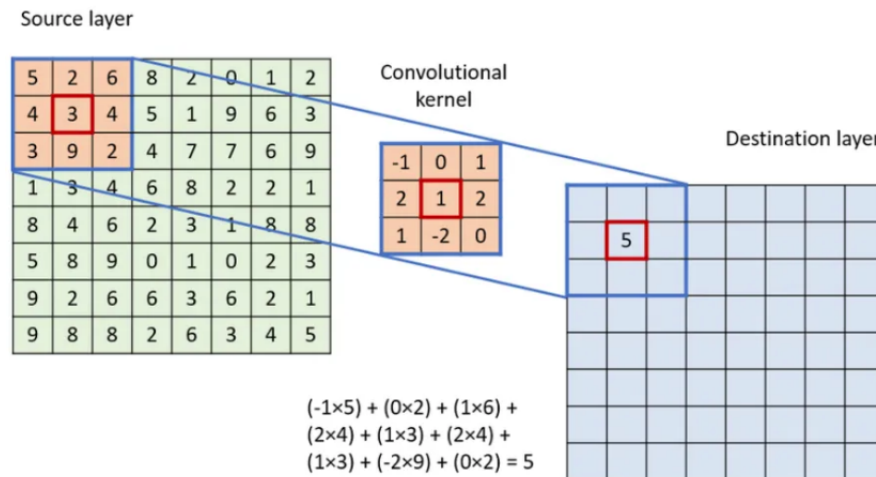


Figura 16 – Operação de convolução em camadas de CNN. Exemplo retirado de [Medium](#)

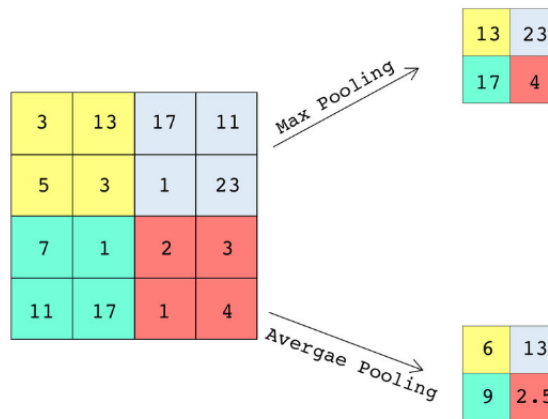


Figura 17 – Operação de *pooling* em camadas de CNN. Exemplo retirado de ([ALJAAFARI, 2018](#))

2.3 Classificação de Gêneros

Com o fim de classificar gêneros musicais, as pesquisas no campo diversificaram o uso e comparação de modelos entre diferentes algoritmos de aprendizado de máquina, tanto aqueles considerados tradicionais quanto os mais atuais, principalmente os de aprendizado profundo - estes sendo mais utilizados na última década.

Ainda sobre o trabalho de ([TZANETAKIS; COOK, 2002](#)), ponto de partida da classificação musical, os autores propõem o uso de três diferentes classificadores para realização da tarefa, *gaussian*, GS, *gaussian mixture model*, GMM, e *k-nearest neighbor*, KNN; juntamente com um vetor de *features* composto por métricas de timbre, incluindo os *MFCCs*, altura e ritmo. O melhor resultado alcançado foi com o GMM, com acurácia de 61%, com número de componentes setado para 3,4 ou 5 (todos alcançaram os mesmos resultados), este é considerado um ótimo resultado para a época. Os demais modelos, o KNN (com 3 vizinhos) e GS também demonstraram bons resultados, com 60% e 59% de acurácia, respectivamente.

Ainda é importante ressaltar que, neste estudo, os modelos foram treinados com base nas métricas referentes a todo o trecho de 30 segundos de cada música, o que os autores chamam de *whole-file features*. Em contraponto, também é gerada uma base de dados com as *real-time features*, onde as métricas de timbre são calculadas em trechos menores dos arquivos de áudio. Neste caso, foi utilizado somente o *GS* com resultado de 44% de taxa de acerto.

Outra conclusão pertinente do trabalho é a importância das *features* relacionadas ao timbre, utilizar somente o conjunto de métricas de altura ou ritmo em comparação com o de timbre faz com que a tarefa de classificação desempenhe de maneira significativamente menos efetiva, ou seja, o espectro fornece uma boa contribuição para a tarefa.

Com o foco na construção de *features* musicais, (LI; OGIHARA; LI, 2003) desenvolvem uma nova métrica para descrição do comportamento musical, as *Daubechies Wavelet Coefficient Histograms*. Este trabalho é feito imediatamente depois do lançamento do GTZAN e a adição desta métrica à análise do *dataset* com as *features* utilizadas no artigo original resulta em significativa melhora na tarefa de classificação para diferentes algoritmos e, o melhor resultado obtido foi a classificação correta de 78,5% das amostras com o modelo de *support vector machines*.

Em (LI; CHAN; CHUN, 2010), os autores propõe uma nova abordagem para realizar a tarefa, utilizando as *Convolutional Neural Networks* (CNNs), que na época já eram conhecidas por obterem ótimos resultados na área de computação visual, porém ainda não haviam sido introduzidas neste problema da área de recuperação de informação musical. A fim de extrair *features* dos áudios utilizando as camadas convolucionais, de *input* para a rede foi proposto uma matriz de valores de 13 MFCCs por 190 *frames* seguidos (13x190), ou seja, a ideia seria extrair comportamentos de cada gênero musical a partir dos valores temporais dos 13 MFCCs extraídos para cada *frame*.

A rede proposta foi capaz de capturar comportamentos locais de cada gênero musical de trechos de 127ms, 541ms e 2.2 segundos respectivamente, nas três camadas convolucionais. Desta maneira, após o treinamento, os filtros responsáveis pelo processamento de dados introduzidos na rede foram utilizados como *deep features* em outro classificadores, já que aprenderam comportamentos de cada gênero nesta etapa. Devido à limitação encontrada de que, se aumentado a quantidade de gêneros rotulados, também aumenta a taxa de erro, a solução encontrada foi treinar quatro redes com subconjuntos dos gêneros e utilizar as métricas criadas como *input* em classificadores baseados em árvore, criando um sistema de votação. O resultado alcançado no *dataset* GTZAN foi de 84% de acurácia e mostra que a utilização de dados tabulares temporais em conjunto à CNNs é promissor se bem modelado.

Em (COSTA et al., 2011), uma nova abordagem foi introduzida à área, a possibilidade de se utilizar algoritmos de detecção de padrões em imagens de espectrogramas, 2.1.1,

para coletar métricas por meio da extração de texturas, visto que estas são consideradas representações visuais do áudio. Associando os padrões extraídos das imagens a um SVM, no contexto do *dataset* LMD, 67,2% de acurácia foi atingida, mostrando que a abordagem é promissora.

Esta abordagem de utilizar imagens de espectrogramas é explorada no contexto do GTZAN no trabalho de (MEHTA et al., 2021), onde os autores utilizam a transferência de conhecimento entre redes neurais para aproveitar modelos pré-treinados já bem estabelecidos na literatura. Com a rede *Resnet34*, os autores obtiveram o melhor resultado do estudo com 79%, porém notaram que certos modelos favoreciam a classificação de gêneros específicos.

Por fim, no trabalho de (NDOU; AJOODHA; JADHAV, 2021), os autores modelam e comparam o desempenho de modelos de *machine learning* tradicionais e de aprendizado profundo utilizando o *dataset* GTZAN, com métricas pré-processadas nas músicas completas, com 30 segundos de duração, e também em relação à trechos menores de 3 segundos cada. De maneira semelhante com o que já foi mostrado aqui, são extraídas as *features* referentes aos contextos de timbre, ritmo, altura e também de *chroma*. E três conjuntos de *features* são criados para serem utilizados separadamente nos classificadores e a tarefa de *feature selection* é aplicada em todos os casos, o que diminui o tamanho destes conjuntos, preservando um *ranking* de valores considerados mais relevantes para a tarefa.

Este trabalho emprega os algoritmos *KNN*, regressão logística, *multilayer perceptron* (MLP), *random forest* e *SVM* como os considerados tradicionais na literatura e uma *convolutional neural network* (CNN) relacionada à tarefa de aprendizado profundo. É válido ressaltar os melhores resultados para cada conjunto de *features* proposto. Para os dois primeiros conjuntos, no contexto da música inteira, entre os algoritmos tradicionais utilizados, o melhor foi a regressão logística, com acurácia de 81.00% e o *SVM*, com 80.80% respectivamente. Já no terceiro caso, onde as *features* são referentes aos trechos de três segundos e também há a utilização da *CNN*, o melhor resultado foi obtido pelo *SVM* novamente, com 92.69% de taxa de acerto.

Utilizando processos de extração manual de *features* considerados normais de acordo com a literatura, este trabalho alcançou resultados consideravelmente melhores que de demais trabalhos que empregam técnicas mais avançadas, como foi exposto anteriormente. Assim, estes resultados e abordagem serão questionados no presente estudo.

A Tabela 1 exibe os estudos descritos nesta subseção.

Trabalho	<i>Dataset</i>	Representação dos Dados	Modelos Utilizados	Acurácia
(TZANETAKIS; COOK, 2002)	GTZAN	<i>Conjunto de features</i>	Gaussian Mixture Model	61,0%
(LI; OGIHARA; LI, 2003)	GTZAN	<i>Conjunto de features</i>	SVM CNN	78,5%
(LI; CHAN; CHUN, 2010)	GTZAN	<i>Conjunto de features</i> <i>Deep Features</i>	Votação de classificadores em árvore	84,0%
(COSTA et al., 2011)	LMD	Espectrogramas Extração de Texturas	SVM	67,2%
(MEHTA et al., 2021)	GTZAN	Espectrogramas	CNN	79,0%
(NDOU; AJOODHA; JADHAV, 2021)	GTZAN (3seg)	<i>Conjunto de features</i>	SVM	92,69%

Tabela 1 – Soluções propostas na literatura.

3 Modelagem

O presente capítulo descreve a abordagem utilizada para realizar a tarefa de classificação de gêneros musicais utilizando abordagens tanto de algoritmos tradicionais de *machine learning* quanto os de aprendizagem profunda, aplicados no *dataset* GTZAN¹, (TZANETAKIS; COOK, 2002). Neste último, os autores coletaram trechos de músicas a partir de rádios, discos e MP3's encontrados na *Internet* no início dos anos 2000, disponibilizando assim 100 trechos de 30 segundos para cada um dos 10 gêneros: *classical*, *country*, *disco*, *hiphop*, *jazz*, *rock*, *blues*, *reggae*, *pop* e *metal*. Cada um dos arquivos foi armazenado com uma frequência de amostragem de 22050Hz, 16 bits de qualidade e com um único canal de som, ou seja, arquivos de áudio ditos monofônicos. O *dataset* contém 8 horas e 20 minutos de conteúdo e, apesar de controvérsias, (STURM, 2013), por conta de possíveis erros na rotulagem dos gêneros e duplicação de dados, é um *benchmark* bem estabelecido desde seu lançamento, sendo utilizado em pesquisas até hoje, como foi exposto no capítulo anterior, na subseção 2.3.

A gama de estudos publicados na área de classificação musical que utilizam o *dataset* GTZAN é abundante, datando desde os anos 2000, portanto, com o intuito de explorar o *dataset* de um ponto de vista abordado recentemente em um dos estudos da literatura que obteve bons resultados, as músicas são divididas em trechos de três segundos e, de maneira que novas informações possam ser agregadas ao contexto de divisão temporal dos dados, nesta pesquisa, foi decidido gerar as *features* musicais também em intervalos de 1 segundo e 200 milésimos de segundo, a fim de analisar o resultado para trechos menores.

A escolha de realizar a abordagem do problema via algoritmos tradicionais e de *deep learning* se deu por colocar em contraste as duas abordagens, destacando seus aspectos positivos, problemáticas, tratamentos de dados e, principalmente, os resultados obtidos e se estes podem ser comparados de maneira justa. Outro aspecto importante desta decisão é baseado na literatura, pois, como foi visto na revisão, os últimos anos de pesquisa em classificação de gêneros musicais é abundante no emprego de modelos de redes neurais profundas, justamente por esta tecnologia conseguir obter ótimos resultados para a tarefa.

As próximas seções tratam das etapas de pré-processamento dos dados, 3.1, discutindo principalmente sobre a extração e armazenamento das *features*, e em seguida a parte da modelagem da classificação em si, 3.2, que aborda sobre os modelos escolhidos para classificar os dados, bem como seus hiperparâmetros e fluxo de dados.

¹ Normalmente disponibilizado para *download* em <http://marsyas.info>, no entanto, no momento da escrita desta pesquisa, o site se encontra fora do ar. É possível baixar o *dataset* pelo [Kaggle](#).

3.1 Pré-processamento dos dados

A etapa de pré-processamento de dados é importante por preparar as informações que serão ingeridas em modelos de inteligência artificial. O desenvolvimento de bons modelos de *machine learning* se dá não só por uma boa arquitetura do modelo, mas principalmente pelo processamento dos quais os dados de treinamento passaram, sendo esta uma etapa crucial para definir o sucesso da tarefa; perceba que, se as métricas extraídas dos dados de análise não descreverem suficientemente bem o comportamento a ser modelado, todo o esforço investido neste trabalho será perdido simplesmente pelas *features* não cobrirem bem o problema.

Para este trabalho foram extraídas as *features* descritas na Tabela 2, já discutidas na seção 2.1.2 do capítulo anterior. A primeira coluna da tabela se refere à característica extraída da faixa de áudio, a segunda coluna quanto ao dado estatístico coletado, sendo estes a média ou desvio padrão de cada *feature* dentro do intervalo de análise, e na terceira coluna a quantidade de dimensões que cada características traz ao *dataset*, no caso dos MFCCs, foram extraídos os 20 primeiros, não há uma fórmula exata que define a quantidade específica de coeficientes de *mel-frequency cepstral* utilizados em tarefas de análise de áudio, mas as literaturas geralmente utilizam entre 12 e 20 valores. Outro fator importante a se destacar é que também foram calculadas as variações de MFCC em função do valor do trecho anterior (Δ MFCCs), bem como o valor da variação da variação calculada ($\Delta\Delta$ MFCCs), não se espera uma contribuição importante destas *features* para análise de trechos longos de áudio, mas sim para trechos menores, como o trecho de 200 milissegundos. As métricas escolhidas para o trabalho são baseadas no trabalho de (NDOU; AJOODHA; JADHAV, 2021), a fim de realizar uma comparação final dos resultados dos trechos.

A Tabela 3 destaca os três diferentes *datasets* criados a partir da divisão das músicas do GTZAN em diferentes intervalos temporais, a informação da quantidade de amostras por conjunto de dados também se encontra na tabela. Perceba que é preciso ter um certo cuidado ao se treinar os classificadores utilizando estes dados, na separação entre as amostras de treino e teste, perceba que trechos longos de 30 segundos de uma mesma música foram decompostos em trechos menores, portanto, permitir que trechos da mesma música estejam tanto em treino quanto em teste pode enviesar a classificação do modelo, exibindo resultados equivocados; as músicas que devem ser separadas entre treino e teste e seus respectivos trechos devem respeitar a divisão. A Figura 18 exemplifica o processo.

Todo processamento de dados apresentado neste estudo foi realizado utilizando bibliotecas do *Python*. As features de áudio aqui citadas foram extraídas utilizando a biblioteca *Librosa*, (MCFEE et al., 2015), desenvolvida para facilitar a recuperação de informações musicais e hoje também é muito utilizada para análise de áudios em geral devido à sua abundância de ferramentas.

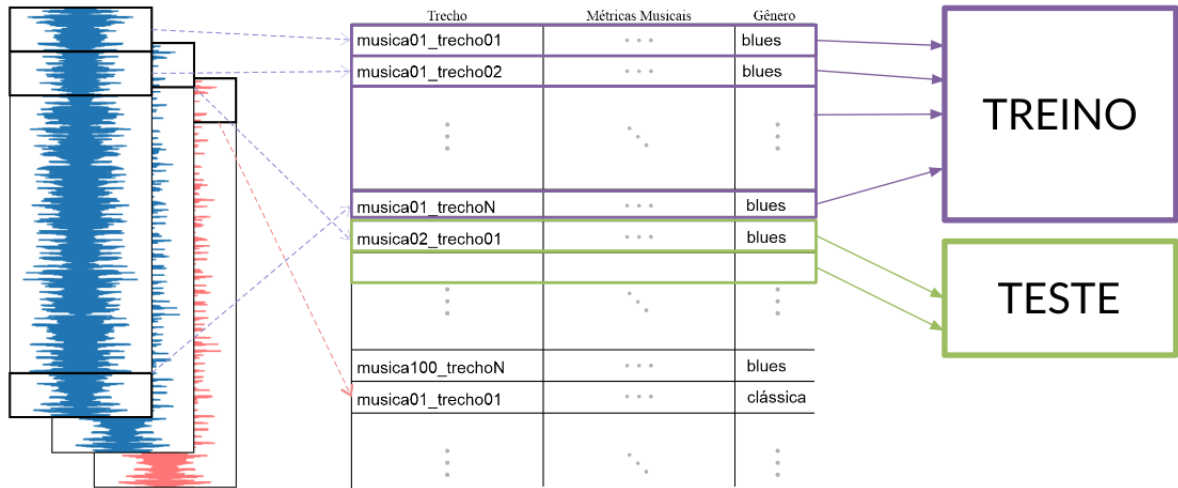


Figura 18 – Criação do *dataset* a partir da divisão das músicas em trechos, cada trecho recebe o gênero da música inteira. A divisão dos trechos entre treino e teste não separa trechos da mesma música em conjuntos diferentes.

Feature	Representação	Dimensão
Chroma	Média + DP	2
Root Mean Square	Média + DP	2
Spectral Centroid	Média + DP	2
Spectral Bandwidth	Média + DP	2
Spectral Rolloff	Média + DP	2
Zero Crossing Rate	Média + DP	2
Harmonia	Média + DP	2
Percussão	Média + DP	2
MFCCs	Média + DP	40
Δ MFCCs	Média + DP	40
$\Delta\Delta$ MFCCs	Média + DP	40
Tempo	Média	1

Tabela 2 – Características extraídas e suas dimensões.

Intervalo Temporal	Nº amostras
3 segundos	10000
1 segundos	30000
200 milissegundos	150000

Tabela 3 – Quantidade de amostras por intervalo temporal.

3.2 Classificação de Gêneros

No que tange a principal tarefa deste estudo, a classificação de gêneros musicais, a seguir serão descritos os dois experimentos propostos neste trabalho, utilizando os algoritmos tradicionais e abordagens com *deep learning*, respectivamente nas subseções 3.2.1 e 3.2.2.

Perceba que pela natureza dos *datasets* extraídos na subseção anterior, 3.1, os algoritmos construídos serão modelos de aprendizado supervisionado, ou seja, serão treinados em conjuntos de dados já rotulados, cada trecho extraído possui uma *label* que é o gênero da música original.

Para cada modelo treinado nos experimentos descritos abaixo, foi utilizada a técnica

de *k-fold cross validation*, para validar os resultados. Esta abordagem separa o *dataset* em k parte, $k = 5$ neste trabalho, e realiza k iterações do algoritmo, fazendo com que o subconjunto de teste seja diferente em cada uma das iterações, ou seja, cada subconjunto é pelo menos uma vez utilizado para validação. Além disto, as *seeds* de separação do *dataset* foram mudadas para cada modelagem, resultando em 15 reprocessamentos.

Ademais, como já foi ressaltado na seção de pré-processamento dos dados, trechos das mesmas músicas nunca pertencem ao mesmo tempo aos *datasets* de treino e de teste, esta garantia faz com que os resultados gerados sejam corretos e com o menor enviesamento possível.

3.2.1 Primeiro Experimento

Na primeira rodada de testes, a ideia é entender como os diferentes algoritmos se comportam na classificação de gêneros nos *datasets* de trechos de 3 segundos e 1 segundo de duração das músicas. Para tal, os algoritmos SVM, *Random Forest*, KNN, *Logistic Regression* e uma DNN *fully-connected* são utilizados e os valores obtidos de acurácia e tempo de treinamento são comparados.

A fim de melhorar o desempenho de cada um dos algoritmos propostos para análise, uma metodologia de *hyperparameters tuning* foi proposta. Cada um dos modelos, além da forma como eles intrinsecamente funcionam, também possuem parâmetros onde o usuário pode modificar e fazer com que o treinamento ocorra de maneira ligeiramente diferente, o que pode ter implicações consideráveis nos resultados finais, estes valores são conhecidos como hiperparâmetros. Assim, otimizar estas configurações de cada modelo é a tarefa conhecida como o *tuning*, ou *optimization*. Neste trabalho, é utilizada a técnica conhecida como *grid search* para encontrar um bom conjunto de valores para os algoritmos tradicionais; com esta abordagem, para cada um dos hiperparâmetros é definida uma lista de valores que este pode assumir, então é realizado um processamento exaustivo de todas as combinações, de forma a encontrar a melhor acurácia entre todas.

A Tabela 4 mostra as configurações finais para cada modelo. Foram utilizadas as implementações disponíveis na biblioteca do *Scikit-learn* para utilizar os algoritmos tradicionais e o *Tensorflow/Keras* para utilizar modelos de *deep learning*, como o último modelo da Tabela. No caso deste último, foi proposto um esqueleto de rede com três camadas internas e então diferentes quantidades de nós e *dropout* para cada camada foram listados como possibilidade, então, aplicando o *grid search*, entre 216 combinações de redes, o resultado do algoritmo foi uma rede com resultados semelhantes ao de um modelo somente com uma camada interna de 100 neurônios e *dropout* de 0.2, então foi utilizada esta última para simplificação. Os demais parâmetros de cada modelo que não foram especificados são os valores *default* das bibliotecas citadas.

Algoritmo	Hiperparâmetros
SVM	kernel='rbf', C=50, gamma='scale'
Random Forest	n_estimators=200, max_depth=20, max_features='sqrt'
KNN	n_neighbors=10
Logistic Regression	C=10, class_weight='balanced', penalty='l2'
DNN	Input(137) Dense(100), ReLU(), Dropout(0.2) Dense(10), Sigmoid()

Tabela 4 – Hiperparâmetros dos modelos.

Por fim, antes de aplicar os dados nos algoritmos de classificação para realizar o processo de treinamento, é importante aplicar um método de normalização dos dados para que o intervalo de valores fique entre um intervalo específico, normalmente entre 0 e 1, de forma a evitar que a alta magnitude de valores do *dataset* perturbe os cálculos estatísticos dos modelos. A Equação 3.1 exemplifica o método conhecido como escalador *MinMax*, que utiliza os valores máximo e mínimo de cada *feature* para normalizar toda a coluna. Esta operação é feita no conjunto de treino e depois aplicada no conjunto de teste, a fim de diminuir ao máximo a influência das músicas de teste.

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3.1)$$

3.2.2 Segundo Experimento

O segundo experimento tem o intuito de entender qual a influência do tamanho dos trechos temporais na captura dos comportamentos dos diferentes gêneros nas músicas pelas redes neurais convolucionais utilizando dados tabulares já que, como foi exposto por (LI; CHAN; CHUN, 2010) - artigo revisado na subseção 2.3 do capítulo anterior-, são modelos capazes de processar métricas temporais e realizar a tarefa de classificação utilizando esta abordagem.

Utilizando os mesmos *datasets* de métricas extraídas em intervalos de 200 milissegundos, 1 segundo e 3 segundos, são montadas matrizes das várias amostras da mesma música, de forma que os 30 segundos totais estejam totalmente descritos dentro desta matriz, que será empregada como *input* de uma CNN 1D, para analisar o comportamento temporal de cada *feature* ao longo destes 30 segundos. A Figura 19 ilustra a criação das amostras de *input* para a CNN.

O modelo escolhido para realizar o processamento está descrito na Figura 20. Desta vez, o processo de *grid search* não foi realizado. Após a realização de vários testes empíricos, chegou-se à conclusão de que a modelagem aqui apresentada seria adequada para a realização da tarefa.

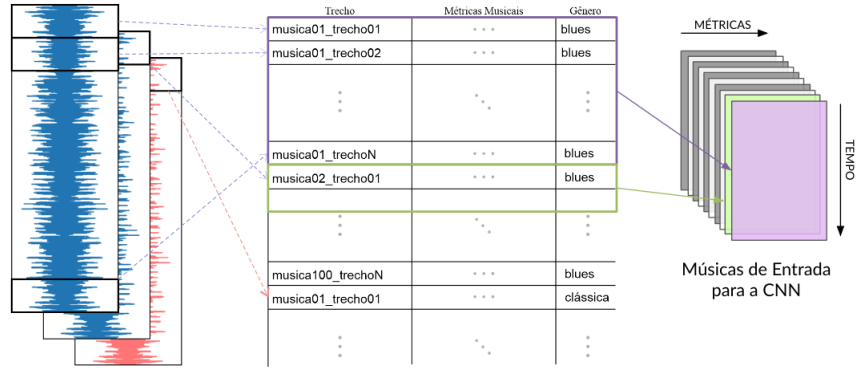


Figura 19 – Criação das amostras de entrada da *convolutional neural network*, são criadas matrizes de músicas completas, onde cada linha é um trecho.

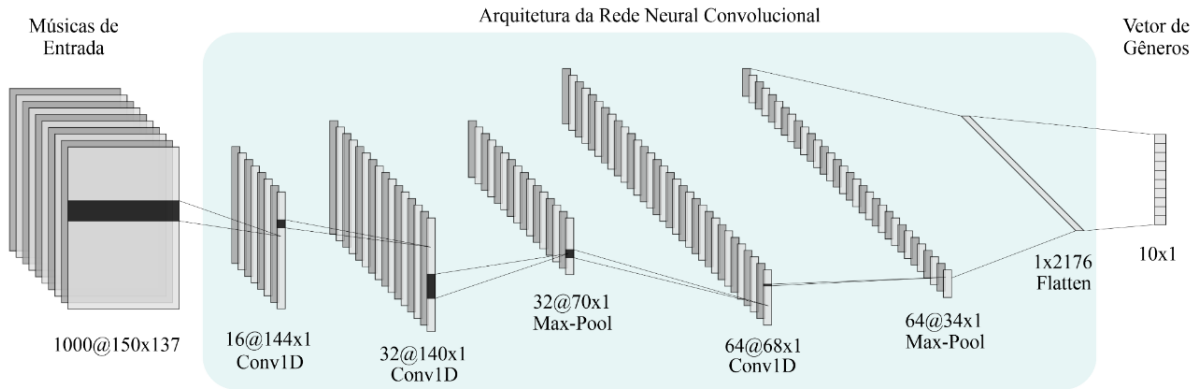


Figura 20 – Arquitetura proposta para CNN.

A arquitetura é composta por três camadas convolucionais no começo para extração de *features*, seguida de uma camada densa totalmente conectada de 10 neurônios. Uma camada convolucional mais comum é a 2D, no entanto, para este trabalho, foi escolhida a 1D, de forma que os vetores temporais, uma coluna, de cada métrica extraída seja tratada como um agregador de comportamentos temporais, então as camadas convolucionais deveriam ser capazes de extrair comportamentos destes vetores. As camadas mais iniciais analisam uma janela de tempo maior e tratam de características mais básicas do áudio, enquanto as mais internas analisam trechos cada vez menores e mais complexos.

Diferente do experimento anterior, neste caso, os dados foram tratados utilizando o *StandardScaler*, do *scikit-learn*, que ao invés de fazer o processo de normalização dos dados, efetua a padronização. Isto é diferente por alterar a distribuição estatística do *dataset*, fazendo com que a média μ se torne 0 e o desvio padrão σ igual a 1. A Equação 3.2 é a ferramenta estatística utilizada neste caso. A mudança de escalador foi necessária pois os modelos de CNN não se adaptaram bem aos dados normalizados, mas houve boa adaptação à padronização.

$$x_z = \frac{x - \mu}{\sigma} \quad (3.2)$$

Todos os modelos de redes neurais profundas neste trabalho utilizaram também o método de *Early Stopping*, de forma que o processo de treinamento é previamente interrompido quando, neste caso, o erro no *dataset* de validação começa a crescer.

4 Resultados

4.1 Primeiro Experimento

Os resultados do experimento são apresentados na Tabela 5. A partir destes dados, é possível notar que o SVM obteve um desempenho melhor que os demais modelos tanto na análise de trechos de um segundo de duração quanto dos trechos de três segundos, onde alcançou a média, nas 15 execuções, de 71.97% e 73.48%, respectivamente. Este resultado é coerente do ponto de vista de que SVMs sempre se destacaram no desempenho da tarefa de classificação de gêneros musicais, como foi mostrado na revisão bibliográfica. Outro detalhe a se ressaltar é o algoritmo ter alcançado um bom tempo de treinamento para as amostras de 3 segundos, no entanto, não há um crescimento linear do tempo de execução do algoritmo quando tratamos de amostras de 1 segundo, onde o *dataset* é três vezes maior; o algoritmo pode enfrentar dificuldades no treinamento à medida que as amostras aumentam.

Como segundo melhor resultado em ambas as abordagens, o *Random Forest* obteve taxas de acerto de 66.80% e 69.92% trabalhando com os intervalos de 1 segundo e 3 segundos. Por outro lado, apresentou também os maiores tempos de processamento nas duas fases. Os piores desempenhos foram perfomados pelo KNN, porém com tempo irrisório de processamento, graças à sua natureza estatística mais simples, o que pode ser o suficiente para escolher entre um modelo ou outro a depender do tamanho do *dataset* e custo-benefício na troca entre tempo e performance, este obteve resultado consideravelmente pior no intervalo de 1 segundo, porém, para 3 segundos, possui desempenho próximo ao *Logistic Regression* e *DNN*.

Estes dois últimos dividem o meio da tabela e não apresentam nenhum resultado marcante. A Figura 21 apresenta as curvas do treinamento da rede neural totalmente conectada utilizada neste experimento. É importante ressaltar que os resultados aqui obtidos para a DNN são específicos para a arquitetura proposta, visto que a área de redes

	1 segundo			3 segundos		
	Acurácia	DP	Tempo(s)	Acurácia	DP	Tempo(s)
SVM	0.7197	0.017	39.41	0.7348	0.0178	4.23
Random Forest	0.6680	0.015	93.15	0.6992	0.0149	24.86
Logistic Regression	0.6160	0.020	2.49	0.6645	0.0240	0.80
DNN	0.6315	0.025	49.46	0.6518	0.0264	21.47
KNN	0.5861	0.017	0.00	0.6459	0.0139	0.00

Tabela 5 – Médias dos resultados da classificação do primeiro experimento.

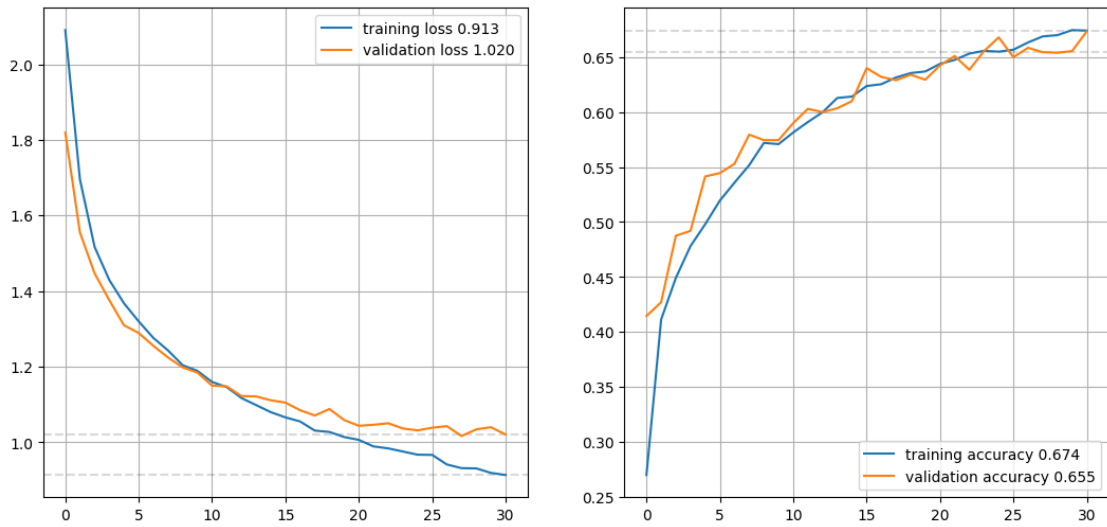


Figura 21 – Curvas de treinamento do modelo DNN. À esquerda o gráfico de erro e na direita a acurácia à medida que as épocas aumentam.

neurais é fortemente orientada à modelagem.

Para os *datasets* de métricas de 1 segundo e de 3 segundos, a acurácia obtida pelos modelos possui um valor médio de 64.43% e 67.92%, respectivamente. Portanto, pode-se dizer que a existência de mais amostras na análise no contexto do *dataset* GTZAN não implica necessariamente numa melhora de desempenho dos modelos de *machine learning*.

Na Figura 22 é possível entender a confusão dos melhores algoritmos ao prever os gêneros neste experimento. As linhas se referem aos valores reais das amostras de teste, enquanto as colunas são os gêneros que o algoritmo predizeu, no encontro destas duas é possível interpretar a frequência de como um gênero na linha i foi interpretado como cada um dos gênero das colunas, ou seja, a soma dos valores de uma linha é igual a 1. Duas paletas diferentes de cores foram propostas, a da diagonal principal para destacar a acurácia específica para cada gênero e a em vermelho para o resto, para destacar os valores mais altos de erro na predição.

O comportamento geral da predição é bem semelhante nas duas matrizes, apesar de valores ligeiramente diferentes, os pontos de destaque são os mesmos nas duas figuras. Os pares (*rock*, *country*), (*rock*, *disco*) e (*reggae*, *hiphop*) apresentam valores consideráveis de confusão na classificação. Diante disto, é intuitivo pensar em pontos em comum entre os gêneros, o *rock* e *country* possuem a presença marcante de instrumentos de corda e um ritmo dançante, também há muito canto e voz envolvida e isto remonta inclusive o surgimento do *rock*, que possui raízes no *country*. O *disco* e *rock* possuem uma interseção conhecida como o *dance-rock*, subgênero que origina ainda da diminuição de melodia no *rock* e o aumento de ritmo e batidas, comportamento influenciado pelo *disco*, este subgênero ganhou popularidade a partir dos anos 80 e uma presença maior deste estilo no *dataset* pode justificar a dificuldade de classificação. Os gêneros *hip hop*, atualmente conhecido

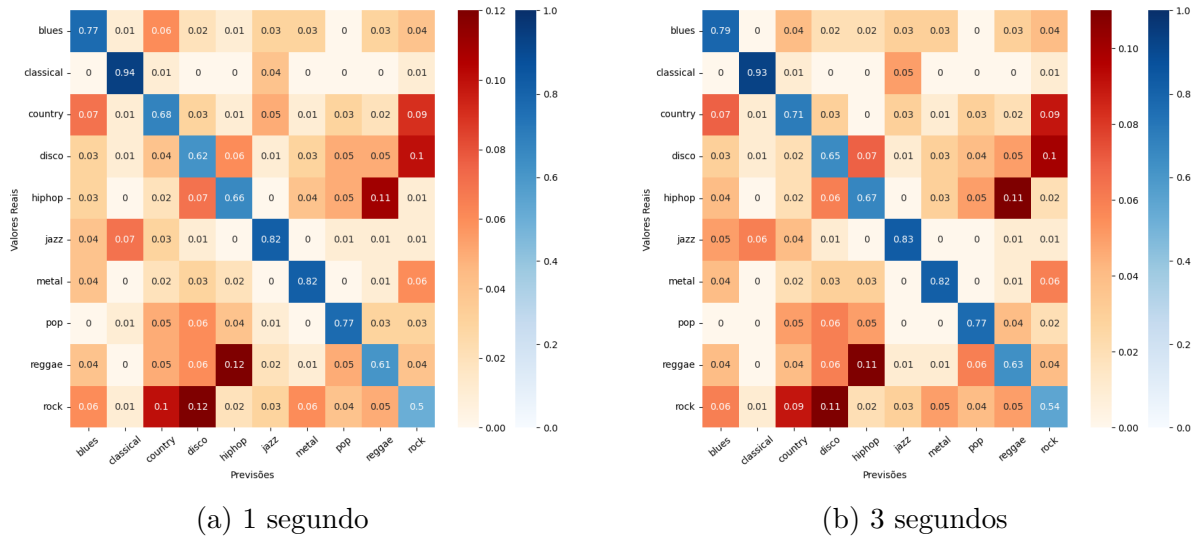


Figura 22 – Matriz de confusão dos gêneros dos SVMs.

Intervalo Temporal	Acurácia	DP	Tempo(s)
3 segundos	0.7227	0.0277	9.12
1 segundo	0.7310	0.0364	11.33
200 milissegundos	0.6953	0.0601	19.92

Tabela 6 – Médias dos resultados da classificação do segundo experimento.

como *rap*, e *reggae* possuem ambos elementos de voz sobre batidas, com uma presença marcante de poesia, inclusive, assim como o exemplo do *rock* e *country*, o surgimento do *rap* americano remonta a artistas originalmente do eixo do *reggae*.

Os gêneros musicais melhor classificados foram a música clássica, com 94%, *jazz* e metal com 82% de taxa de acerto ambos. A música erudita possui aspectos muito singulares, são poucos os ritmos que não possuem canto, também há uma diversidade de presença de vários instrumentos, dos mais diferentes tipos. O gênero não foi classificado de maneira errada com quase nenhum outro gênero, somente com o *jazz*, isto pode ser devido a ambos apresentarem uma grande gama de instrumentos musicais em suas composições. Bem como, o *jazz* apresentou interferências com o *blues* e *country*. O metal teve uma confusão um pouco mais distribuída, porém é marcante os gêneros *rock* e *blues* nesta análise.

4.2 Segundo Experimento

A Tabela 6 apresenta os resultados colhidos no segundo experimento do trabalho, com utilização da *convolutional neural network*. Diferente do que foi hipotetizado no capítulo de modelos, 3, o melhor resultado obtido foi no *dataset* de 1 segundo, que alcançou a taxa de acerto de 73.10%, seguido das divisões de 3 segundos e 200 milissegundos, com 72.27% e 69.53% respectivamente.

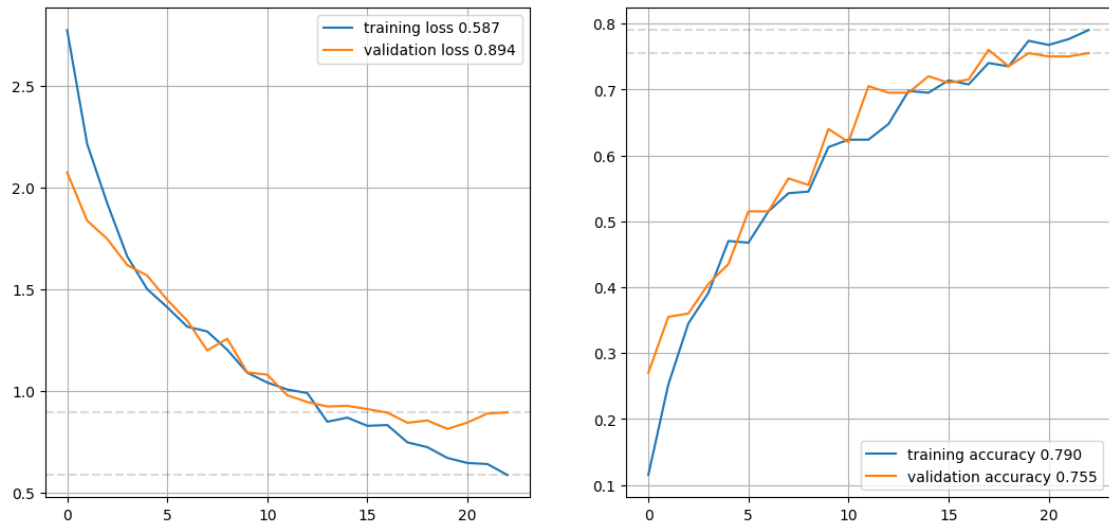


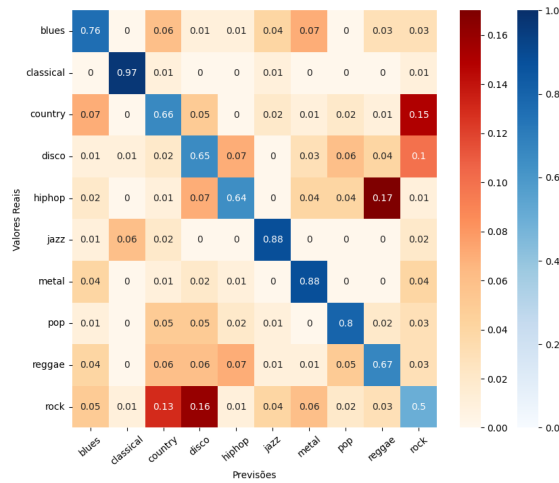
Figura 23 – Curvas de treinamento do modelo CNN com trechos de 1 segundo. À esquerda o gráfico de erro e na direita a acurácia à medida que as épocas aumentam.

A Figura 23 apresenta as curvas do processo de treinamento da CNN com os trechos de 1 segundo de duração. A melhor acurácia obtida é próxima à melhor obtida no primeiro experimento realizado, no entanto, existe uma diferença essencial entre estes dois experimentos, no primeiro, a classificação que está sendo realizada é a de trechos de músicas, de 1 segundo e 3 segundos, enquanto neste segundo a classificação é de músicas inteiras, por mais que elas estejam agrupadas pelos seus trechos. Portanto, são ideias diferentes, por mais que ambas sejam válidas, no contexto do GTZAN, esta segunda abordagem é mais apropriada, pois as *labels* são atribuídas às músicas, não aos trechos - o que foi feito no primeiro caso é atribuir o mesmo gênero da música a todos os seus trechos, o que pode ser prepotente.

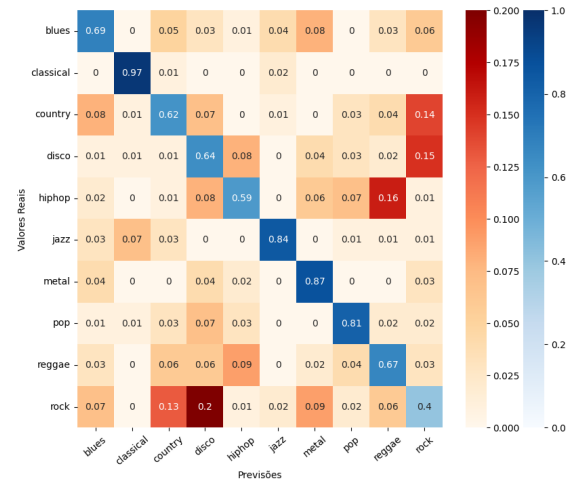
A confusão dos gêneros, apresentado na Figura 24, possui o mesmo comportamento para a matriz de confusão obtida do SVM na subseção anterior para os trechos de 1 e 3 segundos, no entanto, as relações entre alguns gêneros mudam consideravelmente para os trechos de 200 milissegundos.

Note que a perda de desempenho do modelo treinado no *dataset* de 1 segundo para o de 200 milissegundos não é uniforme na acurácia para os gêneros. Na matriz 24d é claro este comportamento na diagonal principal, enquanto gêneros como a música clássica, o *pop*, *reggae* e *rock* tiveram pouca mudança, seja esta para melhor ou pior, os gêneros *country*, *disco* e *jazz* perderam fortemente a qualidade na classificação, cerca de 12% a menos. Portanto, o modelo teve 3.57% a menos de acurácia média e isto não se reflete em cada um dos gêneros.

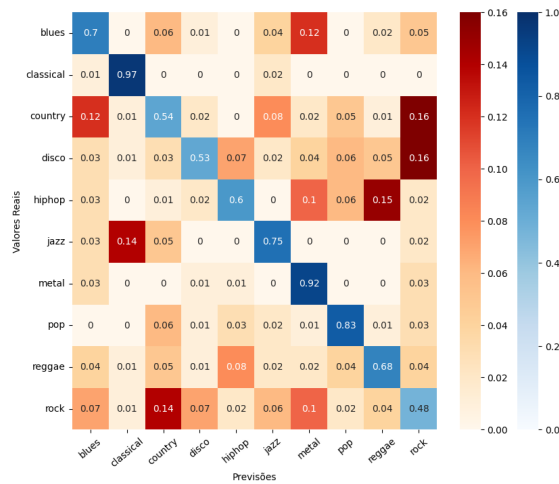
No caso do *disco*, percebe-se que em toda a coluna houve uma diminuição nos valores em comparação com a matriz de confusão de 1 segundo, isto é bom para valores fora da diagonal principal, pois há uma confusão menor entre gêneros e, de fato, houve diminuição



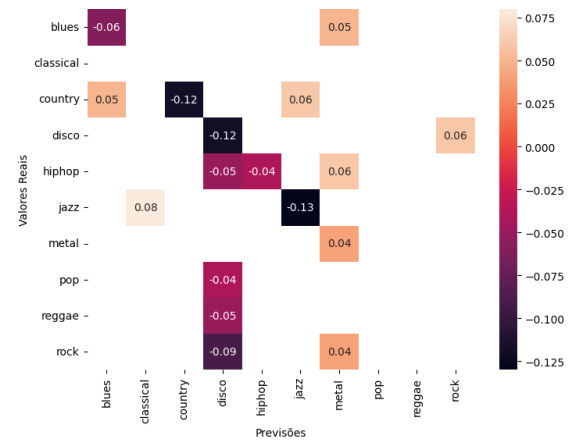
(a) 1 segundo.



(b) 3 segundos.



(c) 200 milissegundos.



(d) Diferença entre 200ms e 1seg, valores entre -3% e 3% foram omitidos.

Figura 24 – Matriz de confusão dos gêneros das CNNs.

em amostras de *rock* sendo previstas como *disco*, porém houve um aumento do caso contrário, onde amostras de *rock* são classificadas como *disco*. Outra relação onde houve perda de qualidade no classificador é no caso do *jazz*, onde suas amostras passaram a ser classificadas como música clássica. Esta última não apresentou mudança de comportamento, demonstrando ser um gênero consistente para classificadores.

5 Conclusão

O presente trabalho disserta sobre a área de classificação de gêneros musicais, pertencente ao campo conhecido como *MIR*, *music information retrieval*, que se dedica a obter informações sobre os metadados de músicas por meio de técnicas de inteligência artificial e análise. A exploração de técnicas para realizar a tarefa passou por mudanças desde os mais recentes avanços em *deep learning*, fazendo com que estes e os algoritmos mais tradicionais fossem abruptamente contrastados.

A extração manual de *features* é uma etapa essencial para a utilização de algoritmos tradicionais e possui uma densa gama de pré-requisitos na área de análise de sinais, fazendo com que o emprego desta abordagem seja muito complexa em comparação com as técnicas mais recentes de extração de *deep features* com redes neurais como as *convolutional neural networks*. Assim, por um lado - extração manual - há uma facilidade no entendimento dos significados de cada *feature*, enquanto por outro, facilidade e precisão na extração dos comportamentos.

Como foi exposto, os algoritmos tradicionais com o conjunto de *features* coletadas desempenham melhor em intervalos de 1 segundo contra redes neurais simples e, os resultados obtidos neste experimento não vão de acordo com os obtidos por (NDOU; AJOODHA; JADHAV, 2021), que encontrou valores muito altos para a divisão do *dataset* em trechos de 3 segundos. Por mais que os experimentos não sejam exatamente os mesmos, aqui foi utilizado um superconjunto de métricas em relação ao do trabalho citado e os algoritmos são os mesmos, com poucas variações nos hiperparâmetros. Se hipotetiza que neste trabalho utilizado como referência, os autores não separaram os trechos da música corretamente entre teste e treino, deixando pedaços da mesma música em ambos os conjuntos, o que pode ter enviesado os modelos deste experimento.

No segundo experimento, as CNNs utilizadas com uma modelagem não convencional (camadas convolucionais de uma dimensão e treinamento com dados tabulares) obtiveram resultados consideráveis, porém um fator de se destacar é o melhor desempenho em intervalos de 1 segundo do que em trechos menores de 200 milissegundos, quem possuem uma quantidade maior de dados temporais a serem analisados pela rede.

Por fim, algumas ressalvas são importantes. O tempo para gerar e a complexidade das *features* utilizadas na classificação favorecem a escolha de um modelo de *deep learning* que realiza a extração enquanto realiza o treinamento, por mais que neste caso seja preciso extrair o espectrograma.

O GTZAN possui poucas amostras para gêneros que já podem estar desatualizados devido à mudança no comportamento de consumo de músicas. Atualmente centenas de

gêneros são comercialmente utilizados nas plataformas de *streaming* e existe uma grande interseção entre estes. Outro ponto a ser destacado é a definição de um gênero para trechos menores de áudio, pois uma mesma música pode apresentar comportamentos diferentes ao longo de sua composição. Por mais que o GTZAN ainda seja empregado em trabalhos recentes, estudos com espectrograma tendem a adotar outros *datasets*, como como o LMD, *Latin Music Database*, (SILLA; KOERICH; KAESTNER, 2008), outro *benchmark* com mais amostras.

Referências

- AGUIAR, R.; COSTA, Y.; NANNI, L. Music genre recognition using spectrograms with harmonic-percussive sound separation. In: . [S.l.: s.n.], 2016. p. 1–7. Citado na página 28.
- ALJAAFARI, N. *Ichthyoplankton Classification Tool using Generative Adversarial Networks and Transfer Learning*. Tese (Doutorado), 02 2018. Citado 2 vezes nas páginas 10 e 35.
- BARBOSA, G. et al. Segurança em redes 5g: Oportunidades e desafios em detecção de anomalias e previsão de tráfego baseadas em aprendizado de máquina. In: _____. [S.l.: s.n.], 2021. p. 145–189. ISBN 9786587003658. Citado 2 vezes nas páginas 9 e 32.
- COOLEY, J.; TUKEY, J. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, v. 19, n. 90, p. 297–301, 1965. Citado na página 18.
- COSTA, Y. et al. Music genre recognition using spectrograms. In: . [S.l.: s.n.], 2011. p. 1 – 4. Citado 3 vezes nas páginas 14, 36 e 38.
- DOWNIE, J. S. Music information retrieval. *Annual Review of Information Science and Technology*, v. 37, n. 1, p. 295–340, 2003. Disponível em: <<https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/aris.1440370108>>. Citado na página 17.
- DRIEDGER, J.; MÜLLER, M.; DISCH, S. Extending harmonic-percussive separation of audio signals. In: *Proceedings of the 15th International Conference on Music Information Retrieval (ISMIR)*. Taipei, Taiwan: [s.n.], 2014. p. 611–616. Citado 2 vezes nas páginas 9 e 28.
- GHIAS, A. et al. Query by humming: musical information retrieval in an audio database. In: *Proceedings of the Third ACM International Conference on Multimedia*. New York, NY, USA: Association for Computing Machinery, 1995. (MULTIMEDIA '95), p. 231–236. ISBN 0897917510. Disponível em: <<https://doi.org/10.1145/217279.215273>>. Citado na página 17.
- HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. 3rd. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers, 2011. ISBN 0123814790, 9780123814791. Citado na página 17.
- HUNT, M.; LENNIG, M.; MERMELSTEIN, P. Experiments in syllable-based recognition of continuous speech. In: *ICASSP '80. IEEE International Conference on Acoustics, Speech, and Signal Processing*. [S.l.: s.n.], 1980. v. 5, p. 880–883. Citado na página 20.
- KLAPURI, A.; DAVY, M. *Signal Processing Methods for Music Transcription*. Springer, 2006. 456 p. [Http://www.springer.com/sgw/cda/frontpage/0,11855,5-102-22-107940371-0,00.html?changeHeader=true](http://www.springer.com/sgw/cda/frontpage/0,11855,5-102-22-107940371-0,00.html?changeHeader=true). Disponível em: <<https://inria.hal.science/inria-00120226>>. Citado 2 vezes nas páginas 21 e 24.
- LI, T.; CHAN, A.; CHUN, A. Automatic musical pattern feature extraction using convolutional neural network. *Lecture Notes in Engineering and Computer Science*, v. 2180, 03 2010. Citado 4 vezes nas páginas 14, 36, 38 e 43.

- LI, T.; OGIHARA, M.; LI, Q. A comparative study on content-based music genre classification. In: . [S.l.: s.n.], 2003. p. 282–289. Citado 2 vezes nas páginas 36 e 38.
- LOGAN, B. Mel frequency cepstral coefficients for music modeling. *Proc. 1st Int. Symposium Music Information Retrieval*, 11 2000. Citado na página 25.
- MCADAMS, S. Musical timbre perception. In: _____. [S.l.: s.n.], 2013. p. 35–67. ISBN 9780123814609. Citado na página 21.
- MCFEE, B. et al. librosa: Audio and music signal analysis in python. In: . [S.l.: s.n.], 2015. p. 18–24. Citado na página 40.
- MEHTA, J. et al. Music genre classification using transfer learning on log-based mel spectrogram. In: *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*. [S.l.: s.n.], 2021. p. 1101–1107. Citado 2 vezes nas páginas 37 e 38.
- NDOU, N.; AJOODHA, R.; JADHAV, A. Music genre classification: A review of deep-learning and traditional machine-learning approaches. In: *2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*. [S.l.: s.n.], 2021. p. 1–6. Citado 4 vezes nas páginas 37, 38, 40 e 51.
- ONO, N. et al. Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram. 01 2008. Citado na página 28.
- OPPENHEIM, A. V.; WILLSKY, A. S.; NAWAB, S. H. *Signals & systems (2nd ed.)*. USA: Prentice-Hall, Inc., 1996. ISBN 0138147574. Citado na página 17.
- PACHET, F.; CAZALY, D. A taxonomy of musical genres. In: . [S.l.: s.n.], 2000. p. 1238–1245. Citado na página 14.
- PINTO, A. A. *Processamento de Sinais Sonoros para Identificação de Notas Musicais Utilizando o Raspberry Pi*. Monografia (Graduação em Engenharia Mecatrônica) — Centro Federal de Educação Tecnológica de Minas Gerais, Divinópolis, 2018. Citado 2 vezes nas páginas 9 e 19.
- ROSA, G.; LUZ, J. da. Mix grinding simulation by artificial neural network. *Rem: Revista Escola de Minas*, v. 65, p. 247–256, 06 2012. Citado 2 vezes nas páginas 9 e 33.
- SCHEIRER, E.; SLANEY, M. Construction and evaluation of a robust multifeature speech/music discriminator. In: *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*. [S.l.: s.n.], 1997. v. 2, p. 1331–1334 vol.2. Citado na página 20.
- SCHUBERT, E.; WOLFE, J. Does timbral brightness scale with frequency and spectral centroid? *Acta acustica united with acustica*, European Acoustics Association, v. 92, n. 5, p. 820–825, 2006. Citado na página 22.
- SHARMA, G.; UMAPATHY, K.; KRISHNAN, S. Trends in audio signal feature extraction methods. *Applied Acoustics*, v. 158, p. 107020, 2020. ISSN 0003-682X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0003682X19308795>>. Citado 4 vezes nas páginas 15, 17, 21 e 27.
- SILLA, C.; KOERICH, A.; KAESTNER, C. The latin music database. In: . [S.l.: s.n.], 2008. p. 451–456. Citado na página 52.

STURM, B. L. The GTZAN dataset: Its contents, its faults, their effects on evaluation, and its future use. *CoRR*, abs/1306.1461, 2013. Disponível em: <http://arxiv.org/abs/1306.1461>. Citado na página 39.

TZANETAKIS, G.; COOK, P. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, v. 10, n. 5, p. 293–302, 2002. Citado 4 vezes nas páginas 20, 35, 38 e 39.

TZANETAKIS, G.; ESSL, G.; COOK, P. Automatic musical genre classification of audio signals. 12 2001. Citado na página 14.

VAPNIK, V. N. *The nature of statistical learning theory*. [S.l.]: Springer-Verlag New York, Inc., 1995. ISBN 0-387-94559-8. Citado na página 32.