Universidade Federal de Ouro Preto
Instituto de Ciencias Exatas e Biológicas
Departamento de Estatística
Bacharelado em Estatística

Aprendizado de Máquina Supervisionado: Classificação de músicas por gênero musical.

Ronan David Souza Abreu

Ouro Preto-MG
Março de 2023

Ronan David Souza Abreu

Aprendizado de Máquina Supervisionado: Classificação de músicas por gênero musical.

Monografia de Graduação apresentada ao Departamento de Estatística do Instituto de Ciências Exatas e Biológicas da Universidade Federal de Ouro Preto como requisito parcial para a obtenção do grau de bacharel em Estatística.

Orientador(a)

Dr. Tiago Martins Pereira

Universidade Federal de Ouro Preto – UFOP
Departamento de Estatística – DEEST

Ouro Preto-MG

Março de 2023

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
COLEGIADO DO CURSO DE ESTATISTICA

FOLHA DE APROVAÇÃO

Ronan David Souza Abreu

Aprendizado de máquina supervisionado: classificação de músicas por gênero musical

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

Aprovada em 24 de março de 2023

Membros da banca

Dr. Tiago Martins Pereira - Orientador (Universidade Federal de Ouro Preto)
Drª Diana Campos de Oliveira (Universidade Federal de Ouro Preto)
Dr. Marcelo Carlos Ribeiro (Universidade Federal de Ouro Preto

Professor Dr. Tiago Martins Pereira, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 24/03/2023

Documento assinado eletronicamente por **Marcelo Carlos Ribeiro**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 06/04/2023, às 11:13, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Diana Campos de Oliveira**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 06/04/2023, às 13:38, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Tiago Martins Pereira**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 10/04/2023, às 14:37, 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 **0498307** e o código CRC **4B5827AF**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: (31)3559-1312 - www.ufop.br

Dedico este trabalho a todos os que me ajudaram ao longo desta caminhada.

Agradecimentos

Primeiramente agradeço aos professores do departamento de Estatística, Matemática e Computação que por meio dos seus conhecimentos tornaram possível minha formação. Ao Professor Dr. Tiago Martins Pereira, por ter sido meu orientador e ter me apresentado as áreas de estatística multivariada e mineração de dados.

Aos meus amigos, (Vinicius Starlino, Luiz Cláudio Diniz, Rogéria Gerçóssimo, Jussara Pereira e João Santos) os quais fazem parte desta conquista sendo fundamentais nesta trajetória.

Agradeço também ao setor de citologia do laboratório de análises clínicas da escola de farmácia (Karla, Renata, Cristina, Mariana e Cláudia) que me acompanharam nesses anos enquanto cursava a graduação.

As máquinas me surpreendem muito frequentemente.

Alan Mathison Turing

Aprendizado de Máquina Supervisionado: Classificação de músicas por gênero musical.

Autor: Ronan David Souza Abreu
Orientador(a): Dr Tiago Martins Pereira

Resumo

Este trabalho apresenta um estudo sobre Aprendizado de Máquina Supervisionado aplicado à classificação de músicas por gênero musical. A metodologia consistiu na extração de características musicais relevantes de arquivos de áudio, seguida da aplicação de algoritmos de aprendizado de máquina para treinamento e teste de modelos de classificação. Usou-se um conjunto de dados de arquivos de áudio de diversos gêneros musicais disponibilizados pela base de dados GTZAN. Os resultados obtidos indicam que é possível alcançar uma acurácia significativa na classificação de músicas utilizando a abordagem proposta neste trabalho. O modelo poderá ser usado na organização de grandes catálogos de música, recomendação e análise de tendências.

Palavras-chave: Processamento de sinais, Aprendizado de máquina supervisionado, Classificação, Recuperação de informações musicais, GTZAN.

Supervised Machine Learning: Classification of songs by musical genre

Author: Ronan David Souza Abreu
Advisor: Dr Tiago Martins Pereira

Abstract

This work presents a study on Supervised Machine Learning applied to the classification of music by musical genre. The methodology consisted of extracting relevant musical features from audio files, followed by the application of machine learning algorithms for training and testing classification models. A dataset of audio files from various musical genres provided by the GTZAN database was used. The results obtained indicate that it is possible to achieve significant accuracy in music classification using the approach proposed in this work. The model could be used in organizing large music catalogs, recommendation, and trend analysis.

Keywords: Signal Processing, Supervised Machine Learning, Classification, Music Information Retrieval, GTZAN.

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

IFPI – International Federation Phonographic Industry

Sumário

1 Introdução

Em seu relatório publicado em 2022, IFPI , entidade que representa o mercado fonográfico global, informou que a arrecadação sobre o formato streaming representou 65% de toda receita global tornando-a a principal fonte de renda da música gravada, fato que se deve, em suma, pela ascensão de serviços como Spotify, Apple Music e Google Play em conjunto com a melhoria da qualidade de internet disponibilizada.

Dada a relevância desse tipo de consumo, alguns dados resumidos sobre a plataforma Spotify[1] ajudam a contextualizar o assunto deste estudo. Criada em 2007 na Suécia, ela é hoje detentora de impressionantes 4 bilhões de playlists, 90 milhões de faixas, sendo 60 mil delas adicionadas mensalmente na plataforma e com um público ativo de 422 milhões de usuários em 2022 tornando-se a líder neste segmento com 32% de participação no mercado.

Catalogar essa quantidade de novas faixas acrescentadas mensalmente, se torna uma tarefa muito difícil. Uma possível solução é desenvolver um classificador por gênero musical automático, que discrimine as músicas diretamente do arquivo de áudio utilizando técnicas de aprendizado de máquina. De acordo com Aucoutier e Pachet (2003), o gênero é uma variável valiosa para esta tarefa, o que permite a realização de agrupamento por associar certas características extraídas diretamente do arquivo digital, além de organizar o acervo e possibilitar a recomendação de novas canções para usuários que tem preferência por um determinado gênero.

O principal objetivo deste presente estudo visa, portanto, apresentar a tarefa de classificação de músicas em gêneros a partir da utilização de 9 algoritmos de aprendizado de máquina supervisionado, ilustrando sua aplicação com os dados da base GTZAN de onde serão extraídas características usadas no treinamento dos modelos. Além de comparar qual dos algoritmos utilizados possuem melhor performance para a tarefa, otimizar parâmetros do melhor modelo e apresentar os resultados.

[1] 20 Spotify Statistics 2023, disponível em https://toneisland.com/spotify-statistics/, acesso em 3/1/23.

2 Referencial Teórico

2.1 Aprendizado de Máquina

Murphy (2012, p. 4), define aprendizado de máquina como um conjunto de métodos que podem detectar automaticamente padrões nos dados e, em seguida, usar os padrões descobertos para prever dados futuros, ou para realizar outros tipos de tomada de decisão.

Podemos dividir o aprendizado de máquina em supervisionado, não supervisionado e aprendizado por reforço. O aprendizado supervisionado tem como objetivo o desenvolvimento de um modelo treinado a partir de uma base de dados D com total de N exemplos das entradas x_i, associadas às saídas y_i, onde os rótulos (y_i) ajudam o algoritmo a correlacionar as entradas x_i. D é o vetor de características D-dimensional contendo valores que podem ser provindos de dados de pixels de imagens, uma série temporal, texto, som, etc. Se y_i for uma variável numérica, tem-se o nome de regressão, se for categórica é chamada de classificação.

$$D = \{(x_i, y_i)_{i=1}^{N}\}$$

No aprendizado não supervisionado, não há associação das entradas x_i a uma variável y_i conhecida. A ideia é a descoberta de padrões naturais nas entradas x_i. Já o aprendizado por reforço, adota uma abordagem de recompensa. O algoritmo é recompensado quando acerta ou punido quando erra, dado um objetivo definido.

O aprendizado de máquina tem sido aplicado em uma variedade de campos incluindo reconhecimento de voz, tradução automática, detecção de spam e fraudes, recomendação de produtos e diagnóstico médico. Alguns exemplos conhecidos de sua aplicação incluem a assistente virtual Siri da Apple, o serviço de recomendação de filmes do Netflix e os sistemas de reconhecimento de voz do Google.

2.1.1 Classificação

Ainda de acordo com Murphy (2012, p. 4), o objetivo é aprender a relação das entradas x_i com as saídas y_i, sendo $y \in \{1, ..., C\}$ tal que C representa número de classes de modo que ao ser considerada como binária, com C = 2, ou seja $y \in \{0, 1\}$, ou multiclasse se C>2. Rótulos de classe que não são mutualmente exclusivos, são chamados de classificação multi-rótulo. A ideia desse tipo de tafefa é prever vários rótulos binários relacionados e gerados a partir de um modelo de saída múltipla. No caso deste trabalho quando nos referimos ao termo classificação, estamos falando de modelos multiclasse de saída única.

2.1.1.1 Classificação Probabilística

Nos casos ambíguos, se tem a necessidade de retornar a probabilidade de uma observação x de D, pertencer a uma classe y. A probabilidade sobre rótulos possíveis, dado um vetor de entrada x e conjunto de treinamento D é $P(y|x, D)$. O resultado é um vetor de comprimento C para os casos onde há multiclasse, ou um valor singular quando se trata de uma classificação binária, pois no caso binário temos:

$$P(y = 1|x, D) = P(y = 1|x, D) + P(y = 0|x, D)$$

Utiliza-se uma função para estimar a probabilidade de dado um novo x pertencer a uma classe y. Assume-se que $y = f(x)$ para um f desconhecido onde o objetivo do aprendizado é estimar a função f por um conjunto de treinamento e em seguida realizar previsões utilizando $\hat{y} = \hat{f}(x)$.

No caso multiclasse, temos um vetor C de probabilidades para cada uma das classes. A classe y ao qual a entrada x será atribuída é dada pela função:

$$\hat{y} = \hat{f}(x) = argmax P(y = y_i|x, D), y_i \in \{1, ..., y_n\}$$

Em que a nova observação x em D será atribuida a classe y_i com maior valor de probabilidade estimada.

2.1.1.2 Medidas de qualidade para modelos de classificação

Segundo Izbicki e Santos (2020, p. 138-139), matrizes de confusão são utilizadas a fim de obter o cálculo de medidas consideradas de avaliação de desempenho de classificadores. A Tabela 1 ilustra o caso binário que pode ser facilmente generalizado para a tarefa

multiclasse:

Tabela 1: Matriz de confusão

Valor Predito	Valor verdadeiro	
	Y=0	Y=1
Y=0	VN (verdadeiro negativo)	FN (falso negativo)
Y=1	FP (falso positivo)	VP (verdadeiro positivo)

Os quatro termos que aparecem na matriz VP, VN, FN, FP são:

- *Verdadeiros positivos (VP)*: É quando uma classe é corretamente atribuída, isto é : y =y ;
- *Verdadeiros negativos (VN)*: É quando uma observação não é classificada em uma determinada classe corretamente, isto é a observação de fato não pertence àquela classe.
- *Falsos positivos (FP)*: A observação é classificada como pertencente a uma determinada classe erroneamente. Isto é, a observação é atribuída a uma classe diferente da verdadeira.
- *Falsos negativos (FN)*: Neste caso a observação foi atribuida às classes a que não pertence.

A partir desta tabela, algumas medidas podem ser extraídas, tais como:

- **Sensibilidade ou Recall**

  É a razão entre o número de verdadeiros positivos divido pelo número de positivos estimados pelo modelo somados aos falsos negativos.

$$S = \frac{VP}{VP + FN}$$

- **Especificidade**

  É a razão entre o número de verdadeiros negativos divido pelo número de verdadeiros negativos estimados pelo modelo somados aos falsos positivos.

$$E = \frac{VN}{VN + FP}$$

- **Valor Preditivo Positivo ou Precision**

  A precisão é calculada dividindo os verdadeiros positivos pela soma entre verdadeiros positivos e falsos positivos.

  Esta razão é dada por:

$$VPP = \frac{VP}{VP + FP}$$

- **Valor Preditivo Negativo**

  Este cálculo difere do anterior ao ser consideradoa razão entre os verdadeiros negativos e a soma entre verdadeiros negativos e falsos negativos.

$$VPN = \frac{VN}{VN + FP}$$

- **Estatística F_1**

  É calculada tomando a média harmônica entre a Sensibilidade e o Valor preditivo positivo considerando o intervalo de variação 0,1.

$$\frac{2}{\frac{1}{S} + \frac{1}{VPP}}$$

- **Acurácia**

  É a razão entre as predições corretas pelo total. Esta será a medida de qualidade utilizada para avaliação do modelo proposto neste trabalho.

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$

  Neste trabalho, estamos focados na avaliação geral da capacidade do modelo em classificar corretamente a classe à qual uma observação pertence. Por isso, não estamos interessados em avaliar, no momento, a sensibilidade do modelo em identificar corretamente os verdadeiros positivos ou a especificidade em identificar corretamente os verdadeiros negativos.

  As medidas mais relevantes para serem usadas dependem da natureza do problema em questão. Por exemplo, se estivéssemos desenvolvendo um modelo para detectar

spam em e-mails, seria mais prejudicial classificar erroneamente um e-mail como spam do que classificá-lo como não-spam (especificidade). Já em um cenário de modelagem para uma doença grave, a prioridade seria a capacidade do modelo em identificar corretamente a presença da condição em pacientes (sensibilidade).

Embora existam diversas métricas que poderiam ser utilizadas, a acurácia será a medida de qualidade adotada neste trabalho para justificar a avaliação e tomada de decisão na seleção do melhor modelo.

2.2 Algoritmos

Nesta seção serão apresentados os algoritmos de classificação utilizados neste trabalho. Apesar de tratarmos do mesmo tipo de tarefa de aprendizado de máquina, diversas abordagens foram e são desenvolvidas a fim de aprimorar os classificadores.

2.2.1 Naive Bayes

Izbicki e Santos (2020, p. 164) explicam que uma abordagem para estimar $P(Y = c|x)$, é por meio do Teorema de Bayes:

$$P(Y = c|x) = \frac{f(x|Y = c)P(Y = c)}{\sum_{s \in C} f(x|Y = s)P(Y = s)}$$

É possível estimar $P(Y = c|x)$ por meio das probabilidades marginais P(Y = s) e condicionais $f(x|Y = s)$ para cada $s \in C$, sendo s uma das classes pertencentes ao vetor de classes C.

$$\begin{aligned} g(x) = argmax_{c \in C}\{\hat{P}(Y = c|x)\} &= argmax_{c \in C}\{\hat{f}(x|Y = c)\hat{P}(Y = c)\} \\ &= argmax_{c \in C}\{[\prod_{k=1}^{d} \hat{f}(x_k|Y = c)]\hat{P}(Y = c)\} \\ &= argmax_{c \in C}\{[\sum_{k=1}^{d} log(\hat{f}(x_k|Y = c)] + log\hat{P}(Y = c)\} \end{aligned}$$

O termo $P(Y = s)$ é estimado usando as proporções amostrais de cada classe, mas para que realmente haja o estimador é necessário supor alguma distribuição para as covariáveis. Nesse caso o método $s \in C$, $f(x|Y = s)$ pode ser fatorada como:

$$f(x|Y = s) = f((x_1, ..., x_d)|Y = s) = \prod_{j=1}^{d} f(x_j|Y = s)$$

Assumindo independência entre as variáves (daí o nome "Naives"do algoritmo) e que, cada uma seja pertencente ao vetor de características X tem distribuição normal que depende da classe:

$$X_j|Y = s \sim N(\mu_{k,s}, \sigma^2_{k,s}), \quad k = 1, ..., D.$$

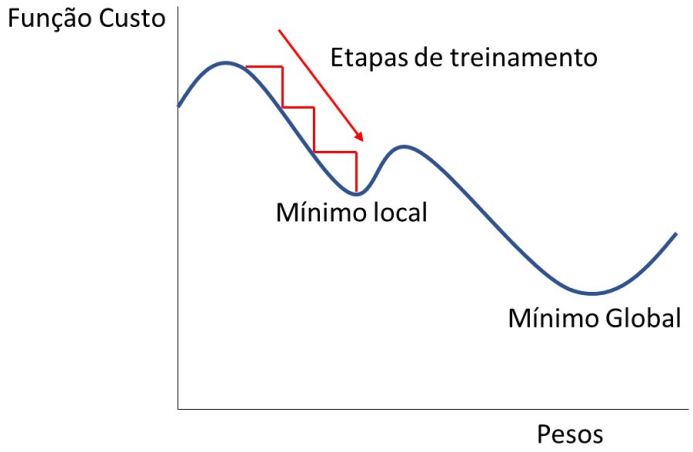
$$\hat{f}(x|Y=c) = \prod_{k=1}^{d} \hat{f}(x_k|Y=c) = \prod_{k=1}^{d} \frac{1}{\sqrt{2\pi\hat{\sigma}^2_{k,c}}} e^{-(\frac{x_k-\hat{\mu}_{k,c}}{2\sigma^2_{k,c}})}$$

Em que o estimador da classe é a probabilidade condicional da observação x dado a variável de classes Y, pertencer a classe c, seguindo uma distribuição normal com $\mu_{k,s}$ e $sigma^2_{k,s}$ onde k é o índice da observação ($k \in D$) e s a classe.

2.2.2 Stochastic Gradient Descent

Bottou (2010) defende algoritmos de gradiente estocástico a fim de minimizar a complexidade computacional de aprendizagem e superar esse gargalo frente a problemas de aprendizado de máquina de grande escala. O Stochastic Gradient Descent é um método de otimização aplicado a algoritmos de classificação linear com o objetivo de minimizar a função custo e encontrar o melhor vetor de pesos que otimize a taxa de aprendizado do classificador.

Figura 1: Algoritmo Gradiente Descendente Estocástico



Fonte: Adaptado de Bottou, 2010

O otimizador calcula o gradiente de uma única amostra aleatória para obter a estimativa do verdadeiro gradiente, fazendo uso de apenas alguns exemplos. Para grandes conjuntos de dados a abordagem economiza recursos computacionais, uma vez que o algoritmo teria que recalcular o gradiente a cada iteração. Também é menos propenso a ficar preso em mínimos globais já que adiciona uma certa quantidade de ruído.

A função a ser otimizada, conhecida como função custo tem a forma:

$$f(\theta_{t+1}) = \theta_t - \frac{1}{N}\sum_{i=1}^{N} f(\theta_t, z_i)$$

Onde $f(\theta_t, z_i) = -logP(y_i|x_i, \theta_t)$ é a função que mede o erro (perda) após a atualização. O índice $t+1$ indica que esse é o novo valor do parâmetro depois de uma iteração do algoritmo de otimização.

Na fórmula, a atualização do valor do parâmetro é dada pela diferença entre o valor atual do parâmetro (θ_t) e a média das funções de perda para cada amostra do conjunto de dados de treinamento ($\frac{1}{N}\sum_{i=1}^{N} f(\theta_t, z_i)$). Esse processo é repetido várias vezes até que o valor do parâmetro converja para um valor ótimo que minimiza a função de perda.

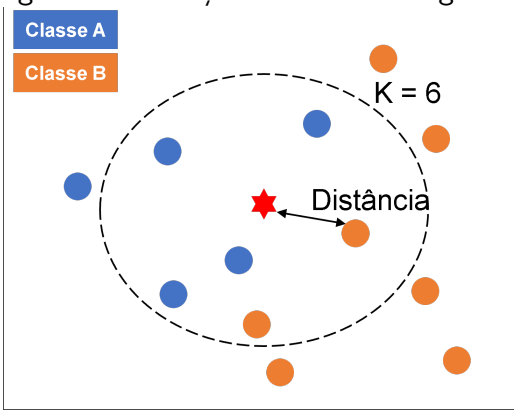
2.2.3 K-Nearest Neighbors

Izbicki e Santos (2020, p. 75) explicam que o método estima a função de regressão r(x) levando em consideração as covariáveis x com base nas respostas Y dos k-vizinhos mais próximos a x. Esta função pode ser reescrita como:

$$r(x) = \frac{1}{k}\sum_{i \in N_x} y_i$$

Acima, N_x representa o conjunto das k observações mais próximas de x, isto é, $N_x = \{i \in \{1, ..., n\} : d(x_i, x) \leq d_x^k\}$ e d_x^k é a distância do k-ésimo vizinho mais próximo de x usando como métrica uma média local no espaço das covariáveis. Existem diversos métodos de cálculo das distâncias entre valores de x, como por exemplo Euclidiana, Mahalanobis, Manhatam, Pan, dentre outras dezenas. O parâmetro k pode ser escolhido por validação cruzada, e a partir daí os valores de x serão agrupados de acordo com método de distância escolhido, além da possibilidade de escolha visualizando um gráfico do tipo silhueta.

Figura 2: Ilustração K-Nearest Neighbors



Fonte: Adaptado de Izbicki e Santos, 2020

2.2.4 Support Vector Machine

O objetivo do algoritmo é fornecer o melhor hiperplano que separe linearmente as classes. O hiperplano é o limite de decisão que discrimina uma classe da outra. Onde também estão presentes os vetores de suporte, que são traçados junto às observações mais próximas do hiperplano. Vetores de suporte contribuem para maximizar a margem a fim de posicionar da melhor forma o hiperplano.

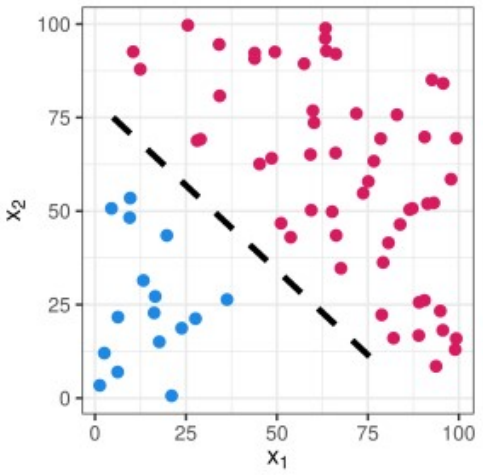
O classificador assume que Y = C(-1,1):

$$f(z) = \begin{cases} g(x) = -1, & \text{se } f(x) < 0 \\ g(x) = 1, & \text{se } f(x) \geq 0 \end{cases}$$

A função $f(z)$ é definida de forma a transformar $f(x)$ em uma saída binária. Se $f(x)$ é menor do que 0, atribui a classe -1 ao ponto x. Se $f(x)$ é maior ou igual a 0, atribui a classe 1 ao ponto x.

Já $g(x)$ é a função de ativação propriamente dita, responsável por atribuir uma classe binária a cada ponto no conjunto de dados de treinamento. A função de ativação é uma função degrau que mapeia qualquer valor negativo para -1 e qualquer valor positivo para 1.

Figura 3: Hiperplano de Separação Linear



Fonte: Adaptado de Izbicki e Santos, 2020

A Figura 3 ilustra o caso da separação linear, onde a reta tracejada representa o limear de separação, o hiperplano. Isso permite escrever uma função f(x) linear tal que f(x_i) < 0 se, e só se, y_i = -1. Assim, podemos escrever para todo i = 1,..., n:

$$y_i(\beta_0 + \beta_1 x_{i,1} + ... + \beta_d + \beta_d x_{i,d}) = y_i f(x_i) > 0$$

Num caso multiclasse, o objetivo será maximizar:

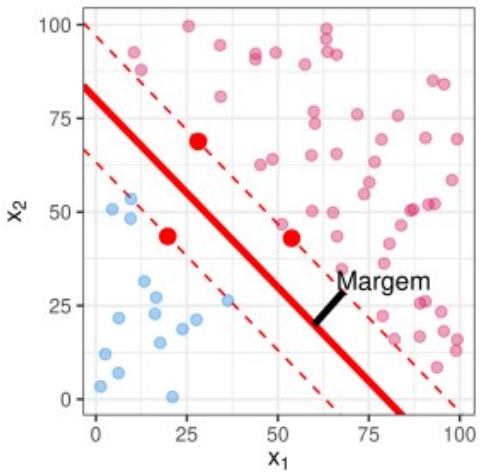
$$\beta = argmax_\beta M$$

Onde β são os coeficientes que retornam as maiores Margens de separação (M). Quanto maior essas margens, melhor é a separação entre as classes pelas partições dos hiperplanos. Essas margens são os limites inferiores e superiores equidistantes do hiperplano de separação.

A Figura 4, ilustra as margens de suporte, onde o hiperplano é alocado de forma equidistante da margem superior e inferior. A depender da complexidade de separação dos dados, faz-se o uso de algum método para resolver um problema não linear. Esses métodos, conhecidos como truque do kernel, funcionam como transformadores lineares elevando a dimensão dos dados a fim de torna-los linearmente separáveis.

Alguns kernels que podem ser usados são: Polinomial, Sigmoidal, Tangente Hiperbólico. A decisão de uso dependerá da estrutura dos dados.

Figura 4: Posição do Hiperplano Equidistante das Margens de Suporte



Fonte: Adaptado de Izbicki e Santos, 2020

2.2.5 Logistic Regression

Segundo Agresti (2010), há várias situações às quais a variável resposta possui varias categorias que se enquadram como nominais ou ordinais. No caso em que há respostas nominais, utiliza-se da fixação de uma determinada categoria de referência possibilitando a avaliação das mudanças dos efeitos das covariáveis à medida que essa categoria é comparada com as demais. Sejam considerados Y uma variável com 3 ou mais categorias nominais (r>2) e π_j a probabilidade de ocorrência da j-ésima categoria com j = 1,2,...,r e de forma que $\sum_{j=1}^{r} \pi_j = 1$. Considerando, assim, a r-ésima categoria como referencial, o modelo para este algoritmo pode ser expresso em termos de logitos como abaixo:

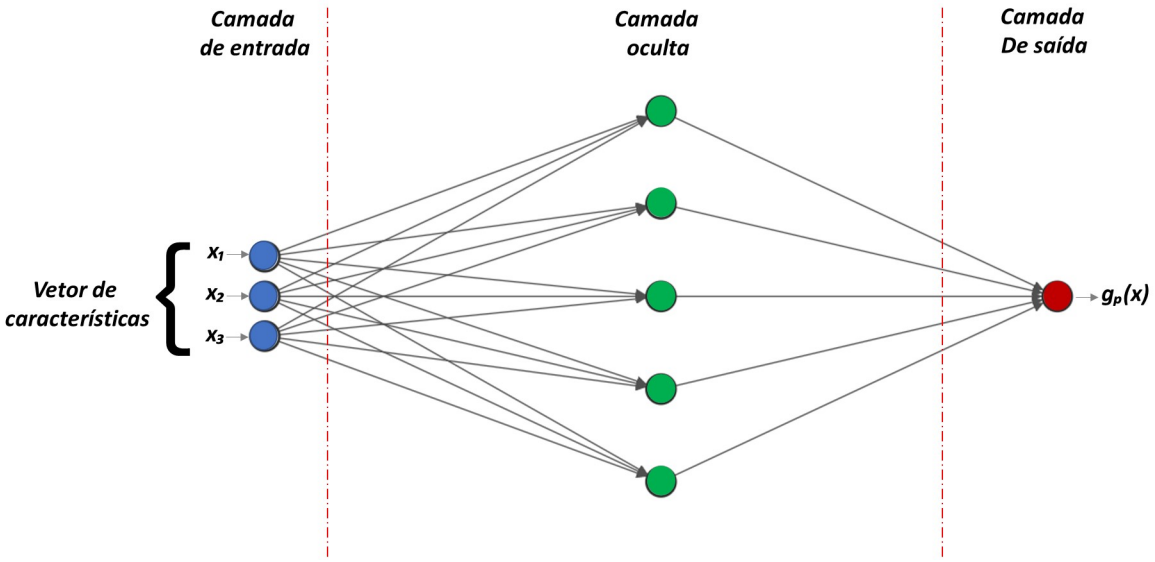
$$\frac{\pi_j(X)}{\pi_r(X)} = \frac{\pi_j(X)}{\pi_r(X)} log[P(Y \leq j)|x] = \alpha_j + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}, \quad j = 1, 2, ..., c-1.$$

Em que $\pi_j(X)$ é a probabilidade condicional da classe j dado o vetor de covariáveis X, e $\pi_r(X)$ é a probabilidade condicional da classe de referência (a c-ésima classe) dado o vetor de covariáveis X. O termo $log[P(Y \leq j)|x]$ é o log da probabilidade condicional de que a classe Y é menor ou igual a j dado o vetor de covariáveis X.Os demais termos são uma combinação linear das covariáveis x_{i1} a x_{ip}, com coeficientes β_1 a β_p e um intercepto α_j, para cada classe $j = 1, ..., c-1$.

2.2.6 Neural Nets

A figura 5 demonstra uma estrutura simples de uma rede neural com uma camada oculta. Os nós do lado esquerdo são as camadas de entrada, uma para cada covariável (x_1, x_2, x_3). A segunda camada é conhecida como camada oculta. As setas são os pesos, os nós são as transformações das variáveis da camada anterior, também conhecidos por neurônios.

Figura 5: Rede neural com uma camada oculta



Fonte: Adaptado de Izbicki e Santos, 2020

O processamento interno de um neurônio é ilutrado na Figura 6 em que β sãos os pesos onde a saída x_1^{l+1}, é dada por:

$$f(\beta_{0,j}^l + \sum_{i=1}^{3} \beta_{1,j}^l x_i^l)$$

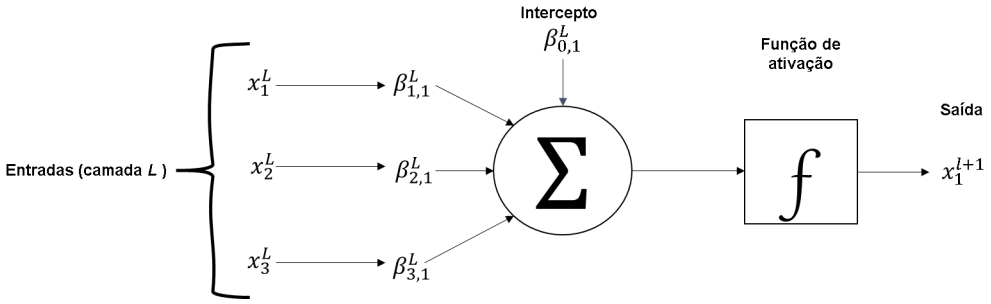
Em que a fórmula representa a combinação linear dos pesos ($\beta_{1,j}^l$) e entradas (x_i^l), adicionada ao viés da unidade ($\beta_{0,j}^l$), que é então passada pela função de ativação ($f()$) para obter a saída da unidade.

O termo $\beta_{0,j}^l$ é um dos parâmetros que são ajustados durante o treinamento da rede neural para minimizar a função de perda e melhorar sua capacidade de predição.

Algumas possibilidades para a a função de ativação são:

Identidade:

Figura 6: Processamento de Neurônios



Fonte: Adaptado de Izbicki e Santos, 2020

$$f(z) = z$$

Logística:

$$f(z) = \frac{1}{1 + e^{-z}}$$

Tangente hiperbólica:

$$f(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$$

ReLu (rectified linear):

$$f(z) = max\{0, z\}$$

ReLu (rectified linear):

$$f(z) = \begin{cases} 0.01z, & \text{se } z < 0 \\ z, & \text{se } z \geq 0 \end{cases}$$

A estrutura simples da rede neural apresentada pode ser generalizada para o uso de mais de uma função de ativação em redes com múltiplas camadas ocultas e outras estruturas complexas como retroalimentação e células LSTM.

2.2.7 Extreme Gradient Boosting

Os chamados modelos essemble são aqueles que permitem combinar modelos do mesmo tipo ou diferentes para fazer previsões. O modelo de árvore aleatória utiliza múltiplos modelos do mesmo tipo a partir de reamostragem da base de dados, também conhecida como bagging. Já os modelos do tipo boosting Chen e Guestrin (2016) são caracterizados por corrigir o aprendizado a cada nova árvore adicionada, dentro da sequência de modelos

criados.

O modelo é dado pela função:

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in \mathcal{F}$$

Em que $\phi(x_i)$ é a função que combina as previsões de todas as árvores para produzir uma previsão única.

A função f_k é escolhida a partir de um conjunto $\mathcal{F}$ de árvores possíveis. As f_k são árvores de decisão, cada uma delas treinada em um subconjunto aleatório dos dados de treinamento.

E a função que mede a qualidade da árvore gerada é:

$$obj* = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T$$

Em que o objetivo (obj) é minimizar o erro. O termo T é o número de arvores, G_j é a soma dos gradientes da função de erro, H_j a soma das derivadas de segunda ordem, λ é o parâmetro que controla o erro a fim de evitar sobreajuste e por fim γ controla a complexidade da árvore para que não haja árvores muito profundas que também podem levar ao sobreajuste. revisão geral.

Detalhadamente as funções G e H são respecivamente:

$$G = \sum_{i \in I} \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$$

$$H = \sum_{i \in I} \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$$

Ou seja, para calcular o índice de qualidade , basta somar o gradiente de primeira e segunda ordem em cada folha e, em seguida, inserir o resultado na fórmula de pontuação.

2.3 Som e Recuperação de Informações Musicais

Segundo Müller (2015, p. 19), uma música pode ser descrita de forma geral em termos físicos, como ondas acústicas transmitidas como oscilações de pressão no ar. Também podem ser descritas na forma de uma peça musical carregada de abstração, sentimentos e intenções. Quando citado, o termo áudio representa a transmissão, recepção e reprodução de sons que seres humanos são capazes de ouvir.

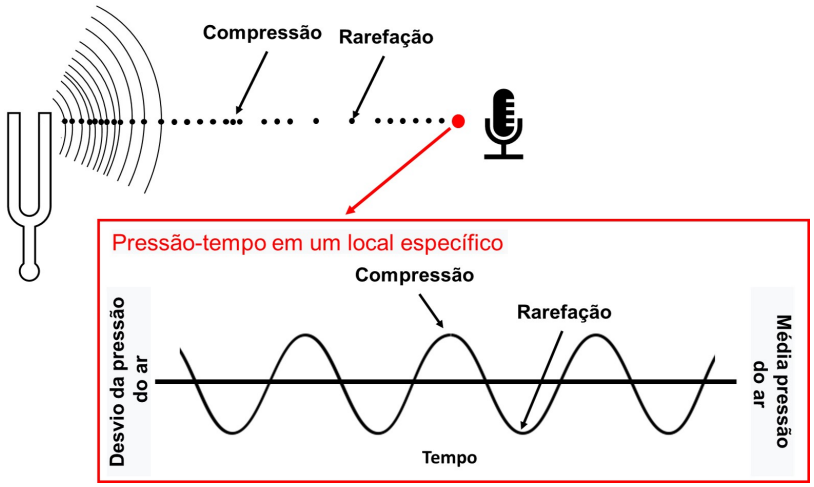
A análise de comparação de sinais de áudio é uma tarefa difícil, uma vez que há sobreposição de diferentes instrumentos e vozes. A percepção é carregada de critérios subjetivos resultantes do processamento complexo de um som no sistema auditivo humano, o que torna a frequência, intensidade e o timbre propriedades mais importantes de um áudio.

2.3.1 Som

O som é gerado por um objeto vibrante que causa deslocamentos e oscilações das moléculas de ar como uma onda com regiões de pressão e rarefação. Esta, por sua vez, pode ser gerada pelas cordas vocais, um instrumento musical ou um diapasão que faz com que sua pressão alternada viaje pelo ar até um ouvinte ou microfone.

A figura 7, exemplifica a propagação de uma onda longitudinal através do ar, desde sua fonte (um diapasão) até um microfone, além de também trazer a forma dessa onda, o desvio ao longo do tempo e a pressão média do ar em um local específico.

Figura 7: Propagação de Onda



Fonte: Adaptado de Müller (2015)

Visualmente, um sinal de áudio pode ser representado como uma forma de onda a qual, dada a repetição regular de pontos de alta e baixa pressão formam ciclos chamados de períodos. O tempo em que cada ciclo demora para se repetir é a média em Hertz (Hz).

Como exemplo, numa onda senoide onde a frequência é de 4 Hz, o período possui um quarto de segundo de duração. Dessa forma, quanto maior a frequência de uma onda, maior ela soa. A faixa audível para seres humanos é de 20 Hz e 20.000 Hz (20kHz).

2.3.2 Recuperação de Informações Musicais

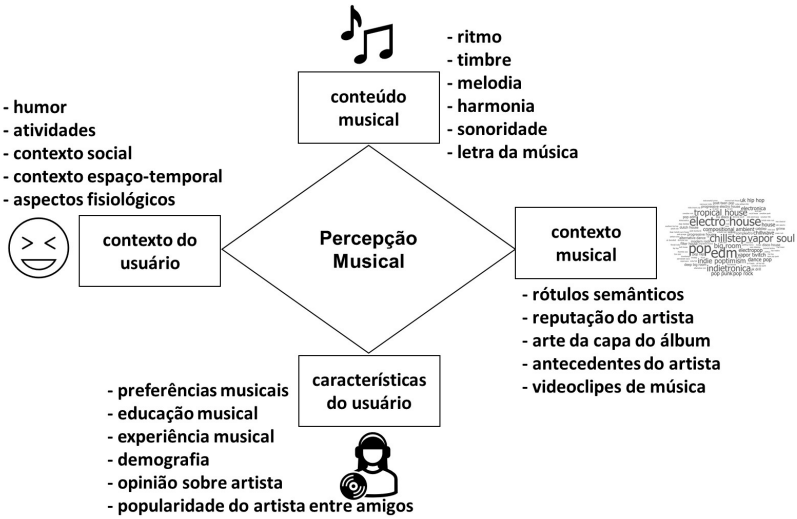
De acordo com M. Schedl et al (2014), Recuperação de Informações Musicais é um campo de estudo emergente focado em atender as necessidades dos usuários de música, abrangendo diferentes aspectos relacionados ao gerenciamento, facilidade de acesso e seu uso envolvendo, portanto, a extração e inferência de recursos de música, não somente do sinal de áudio em si, mas também de aspectos contextuais que podem ser coletados na internet.

Uma das tarefas é o reconhecimento de gêneros musicais, que envolve a extração de características de um determinado áudio e possibilita a atribuição do mesmo a uma classe. Além da classificação por gênero, diversas aplicações possíveis envolvem o áudio e a música, seja o reconhecimento de fala, reconhecimento de locutor, análise de áudio de vídeos, busca de músicas, classificação de áudio, identificação de artistas, reconhecimento de instrumentos e anotação musical. Na classificação de áudio, outros critérios podem ser

adotados para agrupar músicas, tais como sensação (emoção), identificação do artista, reconhecimento do instrumento ou anotação musical.

A Recuperação de Informações Musicais é, portanto, um campo interdisciplinar que reúne especialistas de diversas áreas para melhorar o gerenciamento, o acesso e o uso da música pelos usuários.

Figura 8: Percepção Musical



Fonte: Adaptado de Schedl et al (2014)

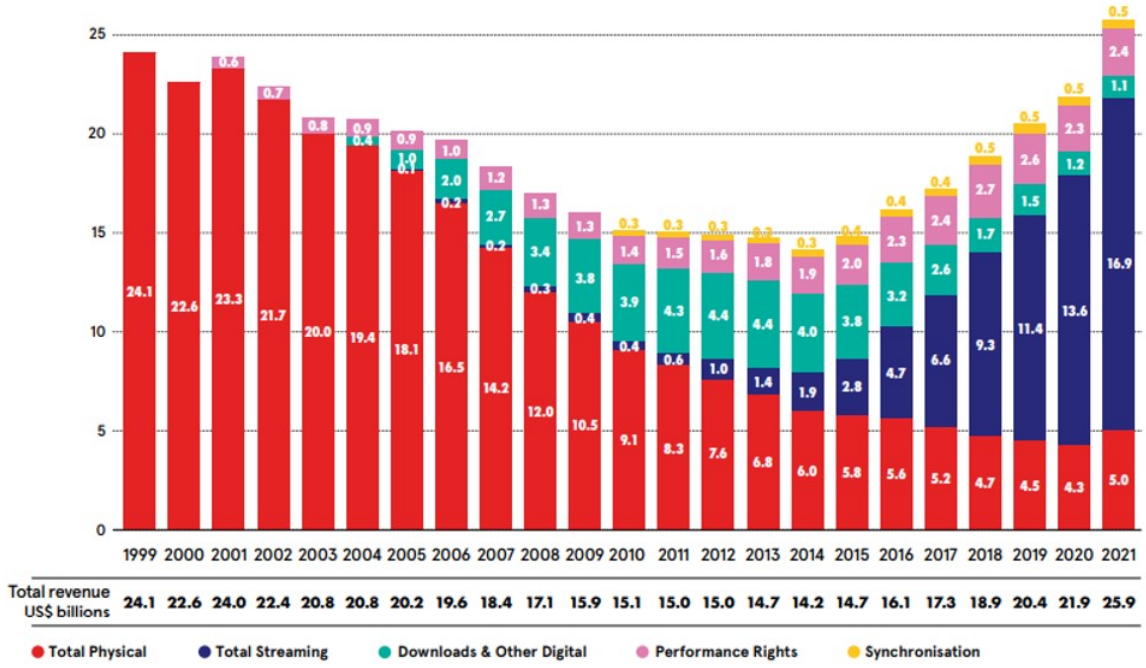
A figura 8 mostra alguns exemplos de como os aspectos da música podem ser categorizados em elementos musicais e informações contextuais, e como os aspectos do usuário podem ser categorizados em propriedades do usuário e contexto do usuário. Os aspectos do contexto do usuário são dinâmicos e mudam frequentemente, enquanto as propriedades do usuário mudam lentamente. Os aspectos da música podem ser modelados por conteúdo musical ou contexto musical, enquanto os rótulos semânticos podem descrever tanto o clima de uma peça musical quanto a emoção de um usuário. A recuperação de informações musicais pode combinar um ou mais elementos apresentados para desenvolver por exemplo, um robusto sistema de recomendação que combine outras técnicas e tarefas relacionadas a Recuperação de Informações Musicais.

2.4 Perfil de consumo de mídia música gravada.

A Figura 9, retirada do relatório global do IFPI, mostra que de 1999 até 2003 o consumo de música gravada era apenas por mídia física,isto é, disco de vinil, disco compacto a laser, pendrive, fitas K-7. No entando, em 2005 é observado o primeiro faturamento via streaming de música.

Streaming é uma tecnologia popular para o consumo de mídias de vídeo ou música que permite ao usuário acessar o arquivo online sem necessidade de baixar para seu periférico.

Figura 9: Receita Global de música gravada de 1999 a 2021



Fonte: IFPI, 2022

Com a melhora do serviço de internet no mundo e dada a facilidade e conveniência, podemos ver no gráfico uma profunda mudança no perfil de consumo de música gravada. Com um faturamento tímido quando comparado às demais fontes de renda em 2005, a escalada do streaming no gosto das pessoas fez com que em 2021 sua participação na receita saltasse de 0,5% para 65%. A mídía física por outro lado, sofreu uma redução de 99,5% para 19,3 na geração de caixa.

Essa nova forma de consumo de música trouxe oportunidade de novos negócios e também problemas como o de gerenciamento de grandes catálogos de canções, playlists, artistas e usuários conforme comentado anteriormente.

2.5 A Base de Dados GTZAN

O conjunto de dados público GTZAN é uma base amplamente utilizada em pesquisas que visam avaliar o desempenho de algoritmos de aprendizado de máquina aplicados ao reconhecimento de gênero musical. Seu autor, Tzanetakis e Cook (2002), afirma que embora a divisão da música em gêneros seja um tanto subjetiva e arbitrária, existem critérios que podem ser usados para classificar uma determinada peça musical em um gênero.

A base possui mil observações, e dez rótulos de gêneros (Rock, Pop, Reggae, Hip Hop, Metal, Jazz, Blues, Country, Clássica, Disco). As classes são balanceadas, isto é, a divisão entre gêneros e observações é exata. Os arquivos de música possuem 30 segundos de duração.

2.6 Extração de características

Müller (2015) descreve a importância da garantia de recursos musicais adequados para tornar dados de música comparáveis e acessíveis por meio de algoritmos. Esses recursos devem capturar informações relevantes enquanto suprem detalhes irrelevantes. Neste trabalho, utilizaremos algumas medidas que revelam as características da distribuição da energia do sinal de música em diferentes frequências. São apresentados Chroma Short-Time Fourier Transform (STFT), RMS (Root Mean Square), Spectral Centroid, Spectral bandwidth, Rolloff, Zero Crossing Rate, Harmony, Perceptive, Tempo, Mel-Frequency Cepstral Coefficients.

2.6.1 Chroma Short-Time Fourier Transform

A fórmula da Chroma Short-Time Fourier Transform é um método comum para extrair informações de tonalidade de um sinal de áudio. Ele usa uma transformada de Fourier de curta duração para calcular a energia em diferentes bandas de frequência e, em seguida, agrupa as informações em um vetor cromático que representa a distribuição a energia do sinal de áudio em diferentes frequências e tempos:

$$\mathcal{X}(m,k) = \sum_{n=0}^{N-1} x(n+mH)w(n)e^{(\frac{-2\pi ikn}{N})}$$

Em que $\mathcal{X}(m,k)$ é uma matriz que representa a distribuição de energia do sinal nas diferentes frequências em cada ponto no tempo. O termo $x(n+mH)$ é o sinal de áudio na posição $n+mH$ ponderada por $w(n)$ que geralmente é uma função suave com valor máximo no centro e decrescente em ambas as extremidades. Por fim o termo $e^{(\frac{-2\pi ikn}{N})}$ é a função exponencial complexa que representa a contribuição da frequência k na posição n.

Após o cálculo do espectograma é possível extrair informações Chroma:

$$C(n,c) := \sum_{\{p\in[0:127]:p mod 12=c\}} \mathcal{Y}_{LF}(n,p)$$

O valor do chroma $C(n,c)$ para uma janela n e uma dada classe de nota musical c, contém os 12 semitons de uma oitava somados da matriz $\mathcal{Y}_{LF}(n,p)$. Nessa matriz calcula-se uma transformação não-linear para a escala de frequência do piano, chamada de escala mel.

Os 12 semitons dentro de uma oitava referem-se às 12 notas musicais presentes na escala escala musical. Essas notas são: Dó, Dó/Réb, Ré, Ré/Mib, Mi, Fá, Fá/Solb, Sol, Sol/Láb, Lá, Lá/Sib, Si. Cada uma dessas notas tem um semitom de distância da nota anterior e da nota seguinte na escala, e juntas elas formam uma oitava.

2.6.2 Root Mean Square

É uma medida da amplitude de um sinal. No contexto do processamento de áudio, pode ser usado para calcular o volume médio de um sinal de áudio. Para calcular o Root Mean Square de um sinal, tomamos a raiz quadrada da média dos valores ao quadrado das amostras no sinal.

$$x_{rms} = \sqrt{\frac{(x_1^2 + x_2^2 + x_3^2 + ... + x_n^2)}{n}}$$

2.6.3 Spectral Centroid

Indica a média do espectro de frequência ponderada pela energia espectral. Ele fornece uma estimativa da frequência média do conteúdo espectral de um sinal.

$$C = \frac{\sum_1^N fM[f]}{\sum_1^N M[f]}$$

Em que C é o valor do Spectral Centroid, f é a frequência de um determinada janela

do espectro, $M[f]$ é a magnitude da janela de frequência f e N é o número total de janelas do espectro.

2.6.4 Spectral bandwidth

A largura de banda espectral de um sinal é definida como a diferença entre suas frequências mais altas e mais baixas. Envolve calcular a densidade espectral de potência do sinal e, em seguida, calcular a frequência na qual uma certa porcentagem da potência total do sinal está contida.

$$BW(p) := F_{tom}(p + 0.5) - F_{tom}(p - 0.5)$$

Em que $F_{tom}(p + 0.5)$ é a frequência da borda superior da janela p em escala de frequência de tom (uma escala não-linear que corresponde à maneira como os seres humanos percebem diferenças de frequência) e $F_{tom}(p - 0.5)$ é a frequência da borda inferior.

2.6.5 Rolloff

Mede a quantidade de energia espectral do sinal de áudio que está abaixo de85% da energia espectral do sinal está contida.

$$\sum_{i}^{R} M[f] = 0.85 \sum_{i}^{N} M[f]$$

Em que $M[f]$é a magnitude da janela de frequência f, N número total de janelas do espectro, R o índice da janela de frequência no qual a soma cumulativa é igual a 85% da energia espectral total.

2.6.6 Zero Crossing Rate

Taxa de cruzamento zero é uma medida do número de vezes que um sinal cruza o eixo horizontal ou o nível zero em um determinado intervalo de tempo.

$$Z = \frac{1}{T-1} \sum_{t=1}^{T-1} |\text{sgn}(x(t)) - \text{sgn}(x(t-1))|$$

Em que T é o comprimento da janela de análise e $x(t)$ é o sinal de áudio amostrado no tempo t. A função $sgn()$ retorna retorna 1 se o valor for positivo, -1 se for negativo e

0 caso contrário.

2.6.7 Harmony

A energia nos harmônicos se refere à quantidade de energia presente em cada harmônico de uma nota musical específica. Um harmônico é uma frequência que é um múltiplo inteiro da frequência fundamental de uma nota musical. Por exemplo, para a nota musical C, o primeiro harmônico é a própria nota C, o segundo harmônico é a nota G (que é o dobro da frequência de C) e assim por diante. Pode ser usada para identificar e classificar diferentes instrumentos musicais.

$$E = \sum_{i=1}^{N} (\frac{a_i^2}{2})$$

Em que N é o número de harmônicos e a_i é a amplitude do i-ésimo harmônico.

2.6.8 Perceptive

O peso perceptivo é uma forma de ajustar um espectrograma de potência para que a energia em cada banda de frequência corresponda melhor à resposta perceptiva do ouvido humano. Isso é importante porque o ouvido humano é menos sensível a certas frequências do que as outras, o que pode levar a uma análise incorreta do espectrograma de potência.

$$P = \sqrt{\frac{1}{N} \sum_{i=1}^{N} W_i (\frac{X_i}{T_i})^2}$$

Em que N é o número total de bandas de frequência, X_i é a magnitude da componente da janela i, T_i é o limiar de audibilidade i, que é uma medida de quanto ruído é necessário para mascarar a componente de frequência na banda, W_i é um fator de ponderação que reflete a sensibilidade do ouvido humano à frequência i.

2.6.9 Tempo

A quantidade de batidas por minuto é uma medida de tempo usada para descrever a velocidade de uma música ou ritmo.

É amplamente utilizado na indústria da música para classificar e categorizar as músicas

em diferentes gêneros e estilos. Por exemplo, uma música de jazz pode ter um Tempo mais baixo em comparação com uma música eletrônica de dança que geralmente tem um Tempo mais alto.

$$B = \frac{60}{\Delta}$$

onde Δ é o tempo médio entre duas batidas consecutivas, medido em segundos.

2.6.10 Mel-Frequency Cepstral Coefficients

Mel-Frequency Cepstral Coeficientes são características de áudio amplamente utilizadas em reconhecimento de voz e música, processamento de fala e reconhecimento do locutor. Esses recursos são sentidos na escala de frequência mel, que é uma escala não-linear que leva em relação à percepção não-linear do ouvido humano em relação às frequências.

$$c_m = \sum_{k=1}^{N} log(\frac{1}{N} \left| \sum_{n=0}^{N_1} x(n)h(n)e^{-j\frac{2\pi}{N}(n-1)(k-\frac{1}{2})} \right|^2)H_m(k)$$

onde:

O termo c_m é o coeficiente calculado para a m-ésima janela mel, N é o número de amostras no sinal de entrada, $x(n)$ é a amostra do sinal de entrada,$H(n)$ é uma janela aplicada no sinal de entrada (por exemplo, a janela de Hamming), k é o índice de frequência do espectro (de 1 a N), $H_m(k)$ é o filtros mel que transforma a escala de frequência linear em uma escala mel não linear.

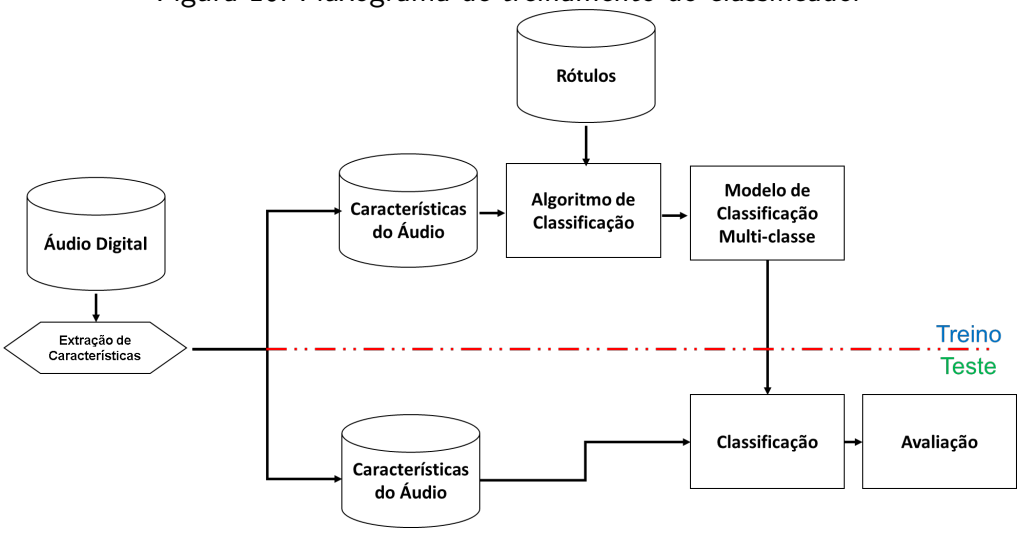
Em geral, o processo de cálculo envolve as seguintes etapas:

Aplicar uma janela (por exemplo, janela de Hamming) no sinal de entrada para reduzir os efeitos de vazamento espectral. Calcular a Transformada de Fourier de curta duração do sinal de entrada com sobreposição entre as janelas. Em seguida utilizar o filtros mel para transformar a escala linear da frequência em uma escala não linear da frequência mel. Tomar o logaritmo da energia em cada janela de filtro para tornar a escala mais parecida com a percepção humana de som. E por fim, calcular a Transformada Discreta de Cosseno para obter os coeficientes.

3 Metodologia

Após carregar os mil arquivos de áudio, utilizou-se a biblioteca Librosa[1], disponível na linguagem de programação Python para fazer o processamento e extração de características: contendo nome do arquivo, tamanho do vetor de áudio, Chroma Short-Time Fourier Transform, Root Mean Square, Spectral Centroid, Spectral Bandwidth, Rollof, Zero Crossing Rate, Harmony, Percetive, Tempo e os coeficientes Mel Frequency Cepstral.

Figura 10: Fluxograma de treinamento do classificador



Fonte: Adaptado de Schedl et al (2014)

Após extrair as características, a base de dados foi dividida em 70% para treino e 30% para teste, usando uma amostragem aleatória simples (Izbicki e Santos, 2020, p 14). Os dados foram apresentados a nove algoritmos de classificação: Extreme Gradient Boosting, Randon Forest, Stochastic Gradient Descent, Support Vector Machine, Logistic Regression, K-Nearest Neighbors, Naive Bayes, Decission Trees, Neural Nets. Como resultado, gerou-se nove modelos que foram testados com o restante dos dados para medir a acurácia

[1]Biblioteca para análise de áudio: Librosa, disponível em: https://librosa.org/doc/latest/index.html, acesso em 02/02/2023.

do acerto nas classificações.

O melhor classificador baseline, isto é, aquele cuja performance foi melhor sem nenhum ajuste ou otimização dos parâmetros foi escolhido para a próxima etapa conhecida por tuning.

O tuning nada mais é do que a tentativa de encontrar os valores ótimos dos parâmetros que levem a um balanço entre viés e variância (Izbicki e Santos, 2020, pag 137). Para tal , foi utilizado o otimizador Optuna[2], disponível na linguagem Python. O otimizador funciona de forma iterativa e utilizou-se 100 e 1000 iterações nos testes a fim de verificar se haveria melhora na aproximação dos valores ótimos dos parâmetros. Os parâmetros otimizados foram:

- *max depth*: refere-se à profundidade máxima de uma árvore;
- *learning rate*: tamanho do passo na atualização do aprendizado, a cada etapa pode-se obter novos coeficientes para os parâmetros;
- *n estimators*: quantidade de árvores construídas dentro do modelo;
- *min child weight*: valor limiar mínimo necessário para um novo particionamento em um nó ou folha;
- *min split loss*: redução mínima no resultado da função de erro para que seja realizada um novo particionamento em nó ou folha;
- *subsample*: proporção de subamostras realizadas nos dados antes da criação de cada árvore.
- *colsample bytree*: proporção de subamostras realizadas nos dados antes da criação de cada árvore.
- *reg alpha, reg lambda*: coeficientes de regularização do modelo, isto é, o quanto o modelo é penalizado em relação ao erro.
- *eval metric*: mlogloss, é a função de cálculo de erro, conhecida como erro logístico ou perda de entropia cruzada utilizada em modelos de classificação multi-classe:

$$L_{log}(y, p) = -(ylog(p) + (1 - y)log(1 - p))$$

[2]Framework para otimização de parâmetros Optuna, disponível em: https://optuna.org/, acesso em 02/02/2023.

Na tabela 2 estão os intervalos de varredura dos parâmetros pelo otimizador. São utilizadas faixas de valores mínimos e máximos:

Tabela 2: Intervalos de busca para o otimizador

Parâmetro	Intervalo
max depth	[1,9]
learning rate	[0.01,1]
n estimators	[50,500]
min child weight	[1,10]
min split loss	[0,1]
subsample	[0.01,1]
colsample bytree	[0.01,1]
reg alpha, reg lambda	[0,1]

Os parâmetros encontrados pelo otimizador são então utilizados na calibração do modelo final.

4 Resultados

4.0.1 Classicadores Baseline e Classificador otimizado

Após o treinamento, o classificador com menor desempenho obteve acurácia de 0.5, enquanto o melhor obteve 0.68, apresentando uma diferença de 36% de performance. Os resultados dos nove modelos baseline são apresentados na Tabela 3. Portando, utilizando a medida de acurácia como métrica de qualidade, o modelo Extreme Gradient Boosting foi utilizado na próxima etapa.

Tabela 3: Classificadores Baseline

Modelo	Acurácia
Extreme Gradient Boosting	0,68
Randon Forest	0,65
Stochastic Gradient Descent	0,61
Support Vector Machine	0,61
Logistic Regression	0,59
K-Nearest Neighbors	0,57
Naive Bayes	0,52
Decission Trees	0,51
Neural Nets	0,50

Na tabela 4, estão os valores sugeridos dentro da faixa de busca fornecida ao otimizador. Obteve-se acurácia de 0.7467, um aumento de 9,8% de performance quando comparado ao melhor modelo baseline utilizado na tarefa de otimização.

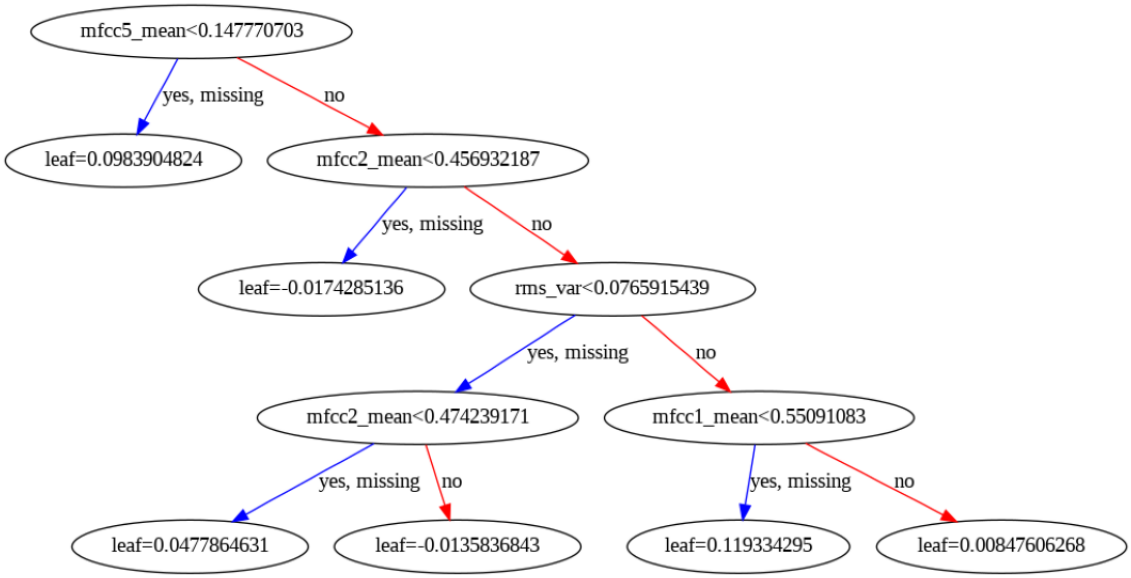
Tabela 4: Parâmetros otimizados

Parâmetro	Valor
max depth	4
learning rate	0.0313
n estimators	382
min child weight	1
min split loss	0.0006
subsample	0.5293
colsample bytree	0.4498
reg alpha	0.0008
reg lambda	0.0040

4.0.2 Modelo final

No caso do exemplo da Figura 11, as observações com valor maior que 0,147770703 na variável mfcc5 seguirão para o próximo nó, mfcc2, e assim por diante, até que sejam acomodadas em uma das folhas. Como o modelo é composto por 382 árvores, todas as observações são submetidas as regras de cada um dos classificadores e então os scores de cada folha é somado para que a observação seja atribuida a uma das 10 classes de gênero musical.

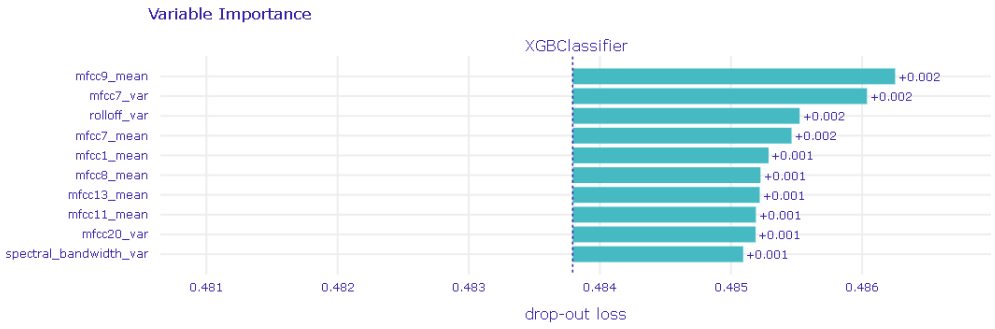
Figura 11: Representação gráfica da primeira de 382 árvores que compõem o modelo otimizado.



Fonte: Autor

A Figura 12 trás as 10 variáveis mais importantes, isto é, aquelas que possuem maior ganho de informação na construção do classificador. As variáveis estão ordenadas da maior para a menor em ganho de informação. A variável mfcc9 foi a que apresentou maior importância enquanto a variável spectral_bandwidth, na lista, foi a que apresentou menor ganho de informação.

Figura 12: Importância das Variáveis

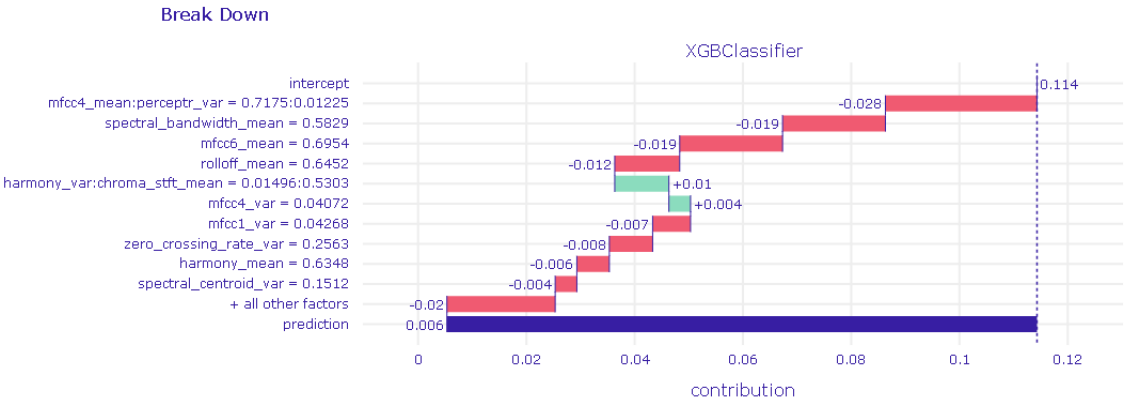


Fonte: Autor

Muitas vezes, explicar o comportamento de um modelo composto por vários classificadores pode ser uma tafera difícil. Na Figura 13 foi experimentada a fixação de uma observação e visualizado a contribuição das variáveis, de uma forma mais amigável para o score de previsão. Gráficos do tipo break down fornecem uma visualização do compor-

tamento do modelo com uma explicabilidade simples como a de uma regressão linear.

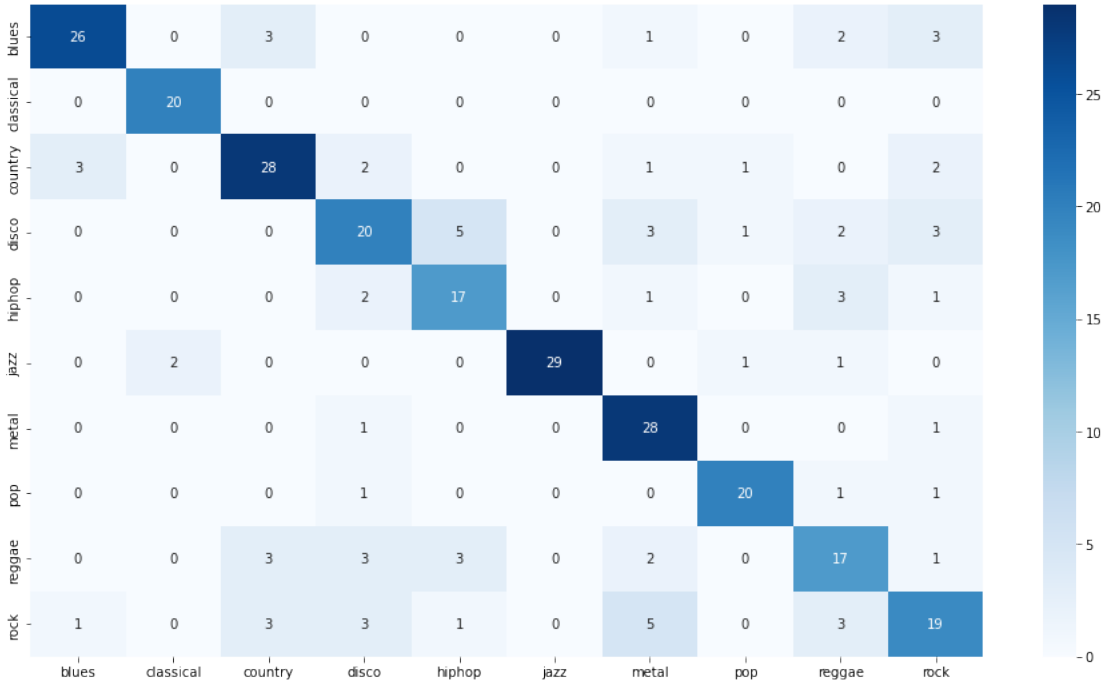
Figura 13: Comportamento do modelo para uma amostra de áudio de jazz.



Fonte: Autor

Para o exemplo foi escolhido uma amostra de jazz. Vemos que dado um intercepto 0.114, os estimadores das variáveis (mfcc6) ou de iterações de variáveis (mfcc4:perceptr) mostrados no eixo y, tem um efeito de acréscimos e decrécimos, sendo o score final utilizado para predição de 0.06. Se fixarmos outra amostra, teremos resultados diferentes dos efeitos das variáveis. Pode-se observar que as covariáveis são dispostas de forma decrescente no eixo y ordenadas pelo efeito na predição.

Figura 14: Matriz de confusão.



Fonte: Autor

Globalmente, podemos explicar a performance do modelo usando informações a partir da Figura 14. Nela contém a matriz de confusão dos valores reais versus valores preditos, a partir dos dados de teste. Foi apresentado que o modelo teve acurácia de 74,67%, próximo ao que um ser humano é capaz de reconhecer entre 75% a 80% (Schedl et all, 2014). Ou seja, se apresantarmos um dado conjunto de músicas uma pessoa em média, conseguirá distinguir os gêneros corretamente em aproximadamente 75-80% das vezes.

Quando olhamos para cada um dos rótulos, esse valor varia de 54,28% (rock) a 100% (clássico). Enquanto o reconhecimento das características funcionou bem para música clássica, jazz, metal e pop (superior a 80% de acerto), o desempenho médio para contry, hip-hop (entre 70% e 75%), os demais gêneros ficaram abaixo dos 70%.

5 Conclusão

Utilizando técnicas de processamento de sinais e recuperação de informação musical foi possível extrair características importantes de arquivos de áudio. Essa tarefa possibilitou o desenvolvimento de um classificador automático de gêneros musicais utilizando algoritmos de aprendizado de máquina.

A acurária foi uma boa medida de qualidade que possibilitou a comparação e escolha dos algoritmos testado neste trabalho.

O modelo pode ser aprimorado com outras técnicas não testadas, um estudo mais profundo de processamento de áudio e talvez uma base de dados com mais informações de metadados e marcações que poderiam contribuir para a construção de um classificador com melhor precisão.

A gestão de grandes portifólios de músicas são um grande desafio, e a organização dessas bibliotecas por gênero podem beneficiar em muito negócios de música como empresas de streaming, a própria área de recuperação de informações musicais e psicoacústica.

Referências

AGRESTI, A. **Analysis of ordinal categorical data**. [s.l.] Wiley, 2010.

AUCOUTURIER, J.-J.; PACHET, F. **Representing musical genre: A state of the art**. Journal of new music research, v. 32, n. 1, p. 8393, 2003.

BOTTOU, L. **Large-Scale Machine Learning with Stochastic Gradient Descent**. Siam Reviews, v. 60, n. 2, 2018.

CHEN, T.; GUESTRIN, C. **XGBoost: A Scalable Tree Boosting System**. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Anais...New York, NY, USA: ACM, 2016.

Global Music Report 2022, IFIP. Disponível em: <https://www.ifpi.org/resources/>. Acesso em: 3 mar. 2023.

IZBICKI, R.; SANTOS, T. M. **Aprendizado de máquina: uma abordagem estatística**, [livro eletrônico] – São Carlos, SP : 2020.

MÜLLER, M. **Fundamentals of music processing: Audio, analysis, algorithms, applications**. 1. ed. Basileia, Switzerland: Springer International Publishing, 2015.

MURPHY, K. P. **Machine Learning: A Probabilistic Perspective**. Londres, England: MIT Press, 2012.

SCHEDL, M.; GÓMEZ, E.; URBANO, J. **Music Information Retrieval: Recent Developments and Applications**. Foundations and Trends R in Information Retrieval, v. 8, n. 23, p. 127261, 2014.

TZANETAKIS E PERRY, G. George Tzanetakis e Perry Cook, **Automatic Musical Genre Classification Of Audio Signals**, ISMIR, [s.d.].