Reconhecimento de Gêneros Musicais Usando Modelos De Aprendizado De Máquina

Paulo Victor M. R. Huguenin de Lima

***Resumo*— Especialistas musicais vêm tentando há muito tempo entender o som, o que diferencia uma música da outra e como visualiza-lo. Neste trabalho deseja-se estudar e analisar a aplicação de algoritmos de aprendizado de máquina para classificação de gêneros musicais dado um conjunto de dados com diversas músicas de variados gêneros. O objetivo é extrair os atributos dos sinais de áudio como entradas para que diversos algoritmos de aprendizado de máquina (Regressão Logística, KNN, Árvores de Decisão, Árvores Aleatórias, AdaBoosting, SVM, entre outros) possam fazer a classificação por gênero musical de cada sinal. Serão avaliados e comparados a acurácia e f1-score de cada modelo testado. Técnicas de seleção de atributos serão utilizadas para diminuir a complexidade do modelo e novamente serão avaliados os desempenhos dos modelos.**

***Palavras-Chave*— Aprendizado de Máquina, Reconhecimento de Padrões, Gêneros Musicais, Classificação**

I. Introdução

Quando ouvimos uma música podemos perceber se o som que chega aos ouvidos é MPB ou rock, sertanejo ou axé. As vezes é um instrumento diferente, uma cadência mais rápida ou até mesmo não conseguimos explicar o porquê sabemos o gênero daquela música apenas por ouvir. Normalmente os gêneros musicais são rotulados por humanos especialistas em musicas. Como o número de regras geradas manualmente aumenta, pode produzir interações inesperadas e efeitos. O processo de classificação de especialistas é implementado sem seguir uma taxonomia universal e esta rotulagem processo de indexação de áudio está sujeito a erros. A percepção da música depende de uma variedade de aspectos pessoais, aspectos culturais e emocionais. Portanto, resultados da classificação por gênero podem ser complexas e a as fronteiras entre os gêneros são difusas [1] [2].

Mas será que uma máquina pode identificar o gênero musical de uma música? A classificação do gênero musical pode ser dada pelas estruturas da música e como ela é analisada e percebida pelos humanos. Pode-se separar a música em diversas categorias, como por exemplo ritmo, estilos e formação cultural. Os estilos são o que chamamos de gêneros musicais. Gêneros musicais são rótulos categóricos criados por especialistas humanos e usados para categorizar, descrever e até comparar músicas, álbuns, ou autores no vasto universo da música [3]. Existe um número de descrições perceptivas de nível superior ou de música, como instrumentação, gênero, humor e artista. O gênero musical é um dos principais descritores de nível superior e encapsula informações semânticas de determinada peça musical. Gêneros diferentes diferem uns dos outros em seu tom conteúdo, instrumentação, estrutura rítmica e timbre características da música[4]. No entanto, não existe nenhum acordo completo em sua definição e limites de distinção estritos entre gêneros. Devido a isso há um grande desafio na escolha de quais características são relevantes para uma máquina poder classificar uma música em gênero de forma automática. Para isso deseja-se extrair recursos de áudio de baixo nível para classificação de alto nível. Além disso, por ser tratar de um problema multi classe, ou seja, há diversos gêneros e sub-gêneros em que uma música pode ser classificada, ocorre uma competição em encontrar o melhor modelo com menor erro de classificação possível.

Neste trabalho, iremos projetar modelos capazes de classificar sinais de áudio em gênero musical fazendo uso de diversos algoritmos de aprendizado de máquina. Primeiramente, será apresentado o conjunto de dados GTZAN no qual foi utilizado para a classificação e apresenta um acervo com 1000 músicas divididas igualmente em dez gêneros. Em seguida iremos extrair os atributos dos sinais de áudio e usando conjunto de classificadores será feito um estudo comparativo para observar quais classificadores usados neste trabalho obtiveram um melhor desempenho. Esta etapa será subdividida em duas. A primeira subetapa é feita uma classificação com um conjunto de dados inicial sem nenhum tipo de seleção atributos aplicados a modelos ditos como padrão, ou seja, sem nenhum tipo de ajuste dos hiper-parâmetros. A segunda subetapa será feita uma seleção de atributos e uma busca pelos melhores hiper-parâmetros de cada modelo. Por fim iremos avaliar a acurácia e matriz de confusão dos melhores modelos.

II. Conjunto de dados

O conjunto de dados escolhido para classificação de gêneros musicais é o GTZAN, no qual é o conjunto de dados público mais usado para avaliação em pesquisa de escuta de máquina para reconhecimento de gênero musical. Os arquivos foram coletados em 2000-2001 de uma variedade de fontes, incluindo CDs pessoais, rádio, gravações de microfone, a fim de representar uma variedade de condições de gravação[5]. Nele consiste 1000 sinais de áudio separados por gênero. Trata-se de um conjunto balanceado onde para todos os gêneros há uma quantidade de 100 músicas cada. Dentre os gêneros inclusos no conjunto temos: blues, música clássica, country, disco, hip-hop, jazz, metal, pop, reggae e rock. Todas elas com duração de 30 segundos cada. Temos assim dez gêneros, tornando-se um problema de multi classes. Pela figura 1 temos uma ilustração de um sinal áudio do gênero rock no tempo.

Paulo Victor M.R. Huguenin de Lima Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro, Rio de Janeiro-RJ, Brasil, E-mails: pv.huguenin@gmail.com

TABELA I
CONJUNTO DE DADOS COM EXTRAÇÃO DE ATRIBUTOS

Dado	**Descrição**
Entradas	
Chroma_CQT	Espectrograma Constante-Q
Chroma_STFT	Espectrograma STFT
Centroide	Centroide do Espectro
Fluxo	Fluxo Espectral
RMSE	Erro Médio Quadrático
Zero Crossing	Zero Crossing do Espectro
Contraste	Contraste Espectral
Banda	Largura de Banda
Flatness	Flatness Espectral
Rolloff	Rolloff
BPM	Batimento Por Minuto
MFCC_1	Coeficientes Cepstrais Mel 1
...	
MFCC_20	Coeficientes Cepstrais Mel 20
Saída	
Rótulo	Gênero Musical de 0 a 9

Olhando essa faixa não conseguimos identificar de qual gênero ela pertence. Teríamos que ouvir essa faixa para podermos classificá-la. Como a máquina não consegue ouvir, devemos extrair informações ou características desse sinal de áudio que façam que com um treinamento supervisionado seja possível classificar uma música por gênero.

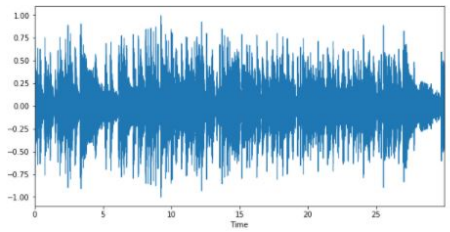

Fig. 1. Faixa de rock amostrada ao longo do tempo

III. EXTRAÇÃO DE ATRIBUTOS

Os recursos de áudio podem ser divididos principalmente em dois níveis, como nível alto e nível baixo, de acordo com a perspectiva da música compreensão [6]. Os rótulos de nível superior fornecem informações sobre como os ouvintes interpretam e entendem a música usando diferentes gêneros, humores, instrumentos, etc. Características de baixo nível também podem ser categorizadas em de curto e longo prazo com base em sua escala de tempo e frequência [7]. Portanto, é necessário extrair alguns recursos que descrevem a onda de áudio usando um formato compacto representação. Normalmente um sinal de áudio é analisado no domínio do tempo, da frequência e no tempo-frequência e é com base nisso que foram extraídas os seguintes atributos.

1) **Centroide Espectral**
   Comumente associado com a medida da forma ou brilho de um som calculando a média ponderada frequência de cada período de tempo. O centroide espectral é definido como o "centro de gravidade"de uma STFT usando a frequência da transformada de Fourier e informações de magnitude.
2) **Fluxo Espectral**
   Fluxo espectral é uma medida de quão rapidamente o espectro de potência de um sinal está mudando, calculado comparando o espectro de potência de um quadro com o espectro de potência do quadro anterior.[8] Mais precisamente, é geralmente calculado como a norma 2 (também conhecida como distância euclidiana) entre os dois espectros normalizados. O fluxo espectral pode ser usado para determinar o timbre de um sinal de áudio, ou na detecção de início, entre outras coisas.
3) **RMSE**
   Calcula a raiz do erro médio quadrático de cada quadro
4) **Zero Crossing**
   Mede o ruído do som calculando o número de vezes que a forma de onda de áudio cruza o eixo zero por unidade de tempo. Um cruzamento zero ocorre quando o áudio adjacente e as amostras têm sinais diferentes
5) **Contraste**
   É definido como a magnitude da diferença entre os picos e vales do espectro
6) **Largura de banda**
   A largura de banda é o intervalo de frequência de um sinal, normalmente calculada como a diferença entre as frequências superiores e inferiores em uma banda contínua de frequências.
7) **Flatness Espectral**
   É uma medida usada no processamento de sinal digital para caracterizar um espectro de áudio. O nivelamento espectral é normalmente medido em decibéis e fornece uma maneira de quantificar o quão semelhante é um som, em oposição a ser semelhante ao ruído[9].
8) **Roll-off**
   Roll-off é a inclinação de uma função de transferência. É comum medir o roll-off em função da frequência logarítmica. O ponto de rolloff espectral é definido como a frequência limite onde 85% da distribuição de energia no espectro está abaixo deste ponto.
9) **BPM** Batimentos por minuto é uma medida de pulsação rítmica. Também está ligada a velocidade da música. Pela figura 2 podemos ver observar qual é o alcance e bpm médio para cada gênero musical.
10) **Chroma CQT** O cromograma está intimamente relacionado com as doze classes de notas diferentes. Os recursos baseados em croma, também chamados de "perfis de classe de afinação", são uma ferramenta poderosa para analisar música cujos tons podem ser categorizados de forma significativa (geralmente em doze categorias) e cuja afinação se aproxima da escala de temperamento igual. Uma propriedade principal dos recursos de croma é que eles capturam características harmônicas e melódicas da música, ao mesmo tempo que são robustos a mudanças no timbre e na instrumentação.
    a transformação de Q constante, simplesmente conhecida como CQT, transforma uma série de dados no domínio da

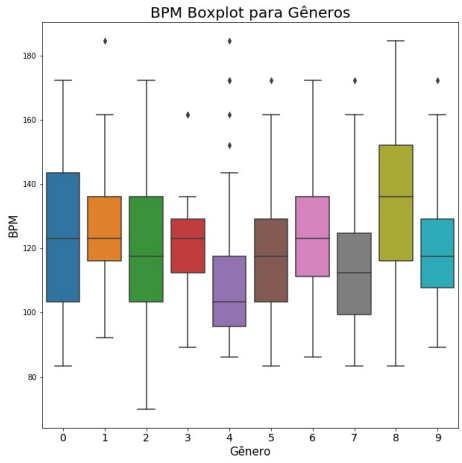

Fig. 2. bpm médio para cada gênero musical

frequência. Ela está relacionada à transformada de Fourier e muito intimamente relacionada à transformada wavelet de Morlet. Semelhante à escala de mel, a transformada de Q constante usa um eixo de frequência com espaçamento logarítmico.

11) **Chroma STFT**
Assim como a Chroma CQT a STFT está relacionada a transformada de Fourier usada para determinar a frequência senoidal e o conteúdo de fase das seções locais de um sinal à medida que muda ao longo do tempo. [1] Na prática, o procedimento para calcular STFT é dividir um sinal de tempo mais longo em segmentos mais curtos de igual comprimento e, em seguida, calcular a transformada de Fourier separadamente em cada segmento mais curto. Isso revela o espectro de Fourier em cada segmento mais curto.

12) **MFCC**
Coeficientes cepstrais de Frequências Mel (MFCCs) são compactos, de curta duração descritores do conjunto de recursos de áudio do envelope espectral e normalmente calculado para segmentos de áudio de 10-100ms. MFCC são um dos conjuntos mais populares de recursos usados no padrão de reconhecimento. MFCC foi originalmente desenvolvido para automático sistemas de reconhecimento de voz, ultimamente têm sido usados com sucesso em várias tarefas de recuperação de informação musical [19]. Embora este conjunto de recursos seja baseado na percepção humana análise, mas após características calculadas, pode não ser compreendido como a percepção humana de ritmo, altura, etc.

Pela tabela I podemos observar o conjunto de dados com os atributos extraídos mencionados acima. Uma observação importante é que para cada atributo temos um vetor de valores que ocorrem ao longo do trecho musical. Utiliza-los como entrada de um classificador torna o modelo muito mais complexo e talvez não tão relevante ou difícil para que o modelo reconheça esses padrões. O que é feito normalmente é calcular a média, desvio padrão, máximo, mínimo, eviesamento e kurtosis. Não necessariamente é preciso usar todos esses, vale uma análise para determinar os mais relevantes e fazer uma combinação entre eles. Neste trabalho vamos usar primeiramente todas essas métricas mencionadas. Com isso o conjunto de dados completo possui 169 entradas e 1000 amostras.

IV. Engenharia de Atributos

É de suma importância para um projeto de reconhecimento de padrões conhecer o que os seus dados representam, com isso, normalmente é realizado uma análise prévia observando como é feita a distribuição dos dados, o que representa cada parâmetro fisicamente, se há alguma correlação entre os dados e caso possua dados categóricos é importante visualizar a quantidade de informação para cada categoria, sabendo por exemplo que um conjunto de dados desbalanceado é prejudicial à acurácia. A Figura 3 ilustra a distribuição dos dados

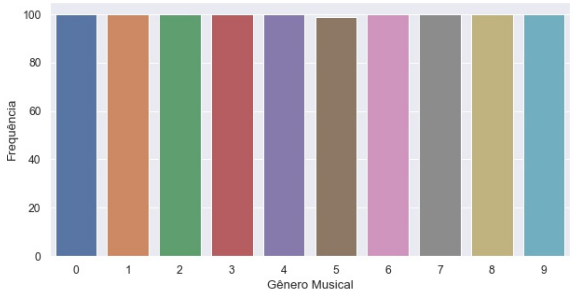

Fig. 3. Histograma das 9 classificações referentes aos gêneros musicais

para cada gênero musical. A partir desta análise já podemos concluir que o conjunto de dados usados neste trabalho é balanceado pois para cada gênero musical há a mesma quantidade de amostras. Outro ponto importante na análise de dados é quanto a dados faltante e quais são as maneiras de lidar com eles. Se forem dados contínuos é possível inserir a média, repetir valores, excluir linhas, interpolar e até mesmo utilizar aprendizado de máquina para preencher esses dados. O Dataset deste trabalho não apresentou ausência de dados. Outra análise de dados que normalmente precisa ser feita quando queremos um desempenho melhor do nosso modelo é ver se o conjunto de dados precisa ser normalizado ou verificar se o conjunto de dados possuem parâmetros redundantes, o que atrapalha o classificador pois, com mais entradas maior é a complexidade da rede. A Figura 4 representa uma boa forma de visualizar os dados. Através dele podemos verificar se o conjunto pode ter ”outliers”ou se há alguma relação linear entre os dados. Nesta figura está representada apenas a média dos atributos

A seleção de recursos é uma das etapas mais importantes do aprendizado de máquina. É o processo de restringir um subconjunto de recursos a serem usados na modelagem preditiva sem perder as informações totais. Às vezes, a seleção de recursos é confundida com a redução de dimensionalidade. Ambos os métodos tendem a reduzir o número de recursos no conjunto de dados, mas de uma maneira diferente. A redução

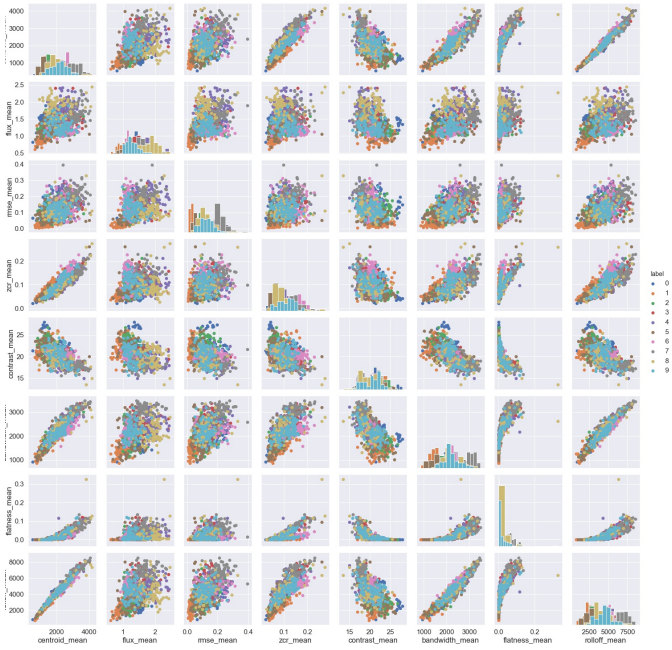

Fig. 4. Pairplot da média dos atributos

da dimensionalidade reduz o número de recursos, criando novos recursos como combinações de recursos existentes. Todos os recursos são combinados para criar alguns recursos exclusivos. A seleção de recursos, por outro lado, funciona eliminando os recursos irrelevantes e mantendo apenas os relevantes. Aqui estão as principais vantagens da seleção de recursos:Ele melhora o desempenho do modelo; quando você tem recursos irrelevantes em seus dados, esses recursos agem como um ruído, o que faz com que os modelos de aprendizado de máquina funcionem mal. Isso leva a modelos de aprendizado de máquina mais rápidos. Isso evita overfitting, o que aumenta a generalização do modelo.

A. Normalização

Uma das principais razões de normalizar os dados são porque as variáveis que são medidas em escalas diferentes não contribuem igualmente para o ajuste do modelo e função aprendida do modelo e podem acabar criando um viés. Assim, para lidar com este problema potencial, a padronização em termos de recursos $(\mu = 0, \sigma = 1)$ é normalmente usada antes do ajuste do modelo. A normalização torna o treinamento menos sensível à escala de recursos, para que possamos resolver melhor os coeficientes. Os pesos estimados serão atualizados de forma semelhante, em vez de em taxas diferentes durante o processo de construção. Isso lhe dará resultados mais precisos quando os dados forem primeiro normalizados Além disso, se usarmos qualquer algoritmo neste conjunto de dados antes de normalizarmos, seria difícil (potencialmente impossível) convergir os vetores devido aos problemas de escala. A fórmula da normalização considerando $(\mu = 0, \sigma = 1)$ pode ser dada como:

$$z = \frac{x - \mu}{\sigma}$$

onde μ é a média e σ o desvio padrão do conjunto de dados.

B. Seleção dos K melhores

A seleção univariada de recursos funciona selecionando os melhores recursos usando testes estatísticos univariados, como qui-quadrado ou ”F-value”. Ele examina cada recurso individualmente para determinar a força da relação do recurso com a variável de resposta. Esse método remove todos, exceto o número especificado de recursos de maior pontuação. O”F-value”é simplesmente uma razão de duas variâncias. variância entre médias de amostra sobre a variância dentro das amostras. Ela será usada para determinar os K melhores atributos em cada classificador.

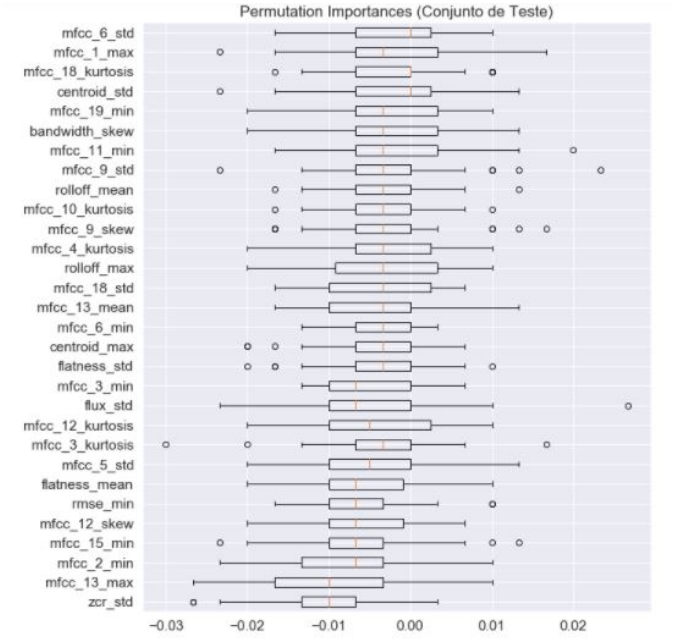

Fig. 5. Atributos mais relevantes usando a técnica *Pemutation Importance*

C. Análise da Componente Principal(PCA)

A análise de componentes principais (PCA) é um pré-processamento de dados e teoria da redução de dimensão comumente usada em estatísticas multivariadas. Quando o número de variáveis é muito grande, a complexidade do processamento aumenta. A principal capacidade do PCA é que ele pode representar mais informações com menos variáveis. Quando o coeficiente de correlação entre as variáveis iniciais não são 0, poderia ser explicado que lá é uma certa sobreposição entre essas variáveis. Para todas as variáveis inicias, o PCA pode excluir relações repetidas de variáveis e criar novas variáveis o mínimo possível. Estas novas as variáveis não estão relacionadas entre si, mas as informações originais podem ser mantidas o máximo possível. Transformando o espaço vetorial composto de amostras de entrada, o PCA pode fazer a direção do maior erro como o vetor base do novo espaço linear. Na verdade, o PCA pode representar mais informações com relativamente menos fatores. Geralmente simplifica o complexidade do problema e reduz o consumo

de recursos de hardware. A fórmula de cálculo do PCA é descrita como:

$$\mathbf{Z}_i = \sum_{n=1}^{p} \mu_{in} \mathbf{X}_{in}$$

onde p vetor aleatório dimensional $\mathbf{X_t}$ representa a variável original. t representa o momento t. Uma transformação linear é realizada para $\mathbf{X}$, e a variável original $\mathbf{X}$ é alterada para novas variáveis $\mathbf{Z}$. Ao escolher o coeficiente apropriado μ, os fatores de $\mathbf{Z}$ não podem ser correlacionados. As principais informações estão concentradas nos primeiros componentes de $\mathbf{Z}$. As primeiras componentes podem ser usadas para representar toda informação. O cálculo dos coeficientes geralmente é feito por uma matriz de coeficiente de correlação, que é numericamente equivalente ao autovetor do coeficiente de correlação matriz. A figura 6 ilustra referente as duas componentes principais em como os dados estão distribuídos. Pode-se observar que para 2 componentes principais há sobreposição dos dados, dificultando a classificação.

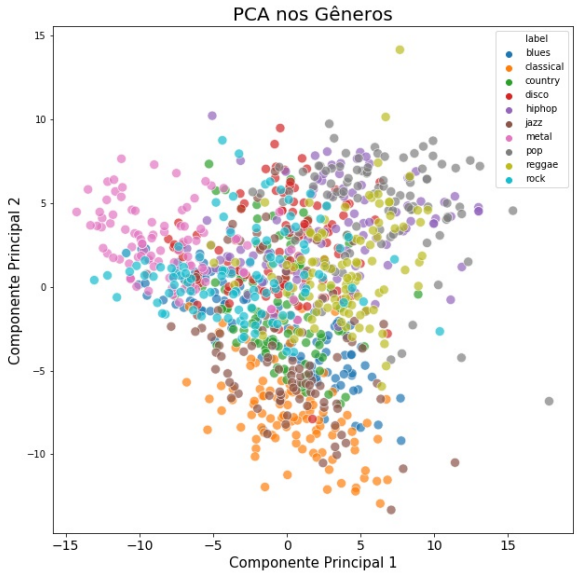

Fig. 6. PCA para 2 componentes principais separados pelo gênero musical

V. Experimentos

Neste trabalho foram usados 8 algoritmos de aprendizado de máquina para realizar classificação. Dentre eles temos: Regressão Logística; KNN; Árvores de Decisão; Árvores Aleatórias; AdaBoost; GradientBoost; XGBoosting e SVM. A ideia de usar esses algoritmos é para fins comparativos. Devemos avaliar qual modelo apresenta melhores resultados ao lidarem com este problema multi classe. O projeto será dividido em duas etapas: A primeira etapa fazer uso do conjunto de dados completo e apenas realizando a normalização iremos dividir o conjunto de dados em treino e teste com divisão de 70% para o conjunto de treino e 30% para o conjunto de teste. Após essa etapa iremos realizar o treinamento do modelo usando os classificadores mencionados acima em seu esquema padrão, onde não há a preocupação em achar os melhores hiper-parâmetros de cada modelo. Além disso, nesta primeira etapa iremos avaliar a importância de cada atributo relacionado a cada modelo.

A segunda etapa consiste em repetir o processo da primeira etapa, só que agora iremos realizar uma seleção de atributos analisando a correlação entre alguns atributos, a seleção dos K melhores atributos com base no "F-Value"e a redução da dimensão usando o PCA. Além disso, A escolha do melhor modelo depende de inúmeras variáveis e a grande dificuldade do treinamento é justamente essa. Definir os melhores hiper-parâmetros que resultam no modelo com maior acurácia. Há técnicas que conseguem testar diversos modelos e salvar o modelo que obteve melhor resultado(acurácia). Um dos métodos mais comuns de procura por hiper-parâmetros é a busca por grade, que é simplesmente uma busca exaustiva por meio de um subconjunto especificado manualmente do espaço de hiperparâmetros de um algoritmo de aprendizagem. Um algoritmo de pesquisa de grade deve ser guiado por alguma métrica de desempenho, normalmente medida por validação cruzada no conjunto de treinamento. A validação cruzada é uma técnica para avaliar a capacidade de generalização de um modelo, a partir de um conjunto de dados. Foi usada neste experimento o método *k-fold*. Este método consiste em dividir o conjunto total de dados em k subconjuntos mutuamente exclusivos do mesmo tamanho e, a partir daí, um subconjunto é utilizado para teste e os k-1 restantes são utilizados para estimação dos parâmetros, fazendo-se o cálculo da acurácia do modelo. Este processo é realizado k vezes alternando de forma circular o subconjunto de teste, obtendo assim uma medida mais confiável sobre a capacidade do modelo de representar o processo gerador dos dados. Para todos os experimentos foram utilizadas 5 pastas com 3 repetições. Todo o desenvolvimento do projeto foi escrito em Python e todos os classificadores utilizados neste trabalho foi usando a biblioteca SKLearn. A tabela II apresenta as configurações dos classificadores fazendo a busca em grade dos respectivos hiper-parâmetros. Como avaliação dos modelos apresentados será avaliada a acurácia de cada modelo no conjunto de treinamento e teste.

VI. Resultados

Nesta seção serão apresentados os resultados referentes a cada etapa descrita anteriormente. Será feita uma análise e comparação da acurácia dos modelos, do desempenho do modelo na fase de treinamento e teste. Iremos avaliar se há problemas de viés e variância e a complexidade dos modelos.

A. Etapa 1

A Tabela IV apresenta a acurácia dos conjuntos de treino e teste de cada modelo e o f1-score do conjunto de teste. Podemos observar que os classificadores que obtiveram melhores desempenhos foram a Regressão Logística, Árvores Aleatórias, GradBoost, XGBoost e SVM. Todos na faixa de 70% de acurácia e um destaque para o SVM que obteve acuráci da 78% no conjunto de teste. Nesta primeira análise

TABELA II
CONFIGURAÇÃO DOS CLASSIFICADORES COM OS AJUSTES DOS HÍPER-PARÂMETROS

Modelo	Hiper-parâmetros	Valores
Reg. Logística	K Melhores	[150,120,80,60,30]
	Penalidade	[l1, l2]
	C	logspace(-4,4,20)
	Solver	[liblinear,lbfgs]
KNN	K Melhores	[150,120,80,60,30]
	K Vizinhos	[3,5,7,10]
	p	[1,2]
	Pesos	[uniforme, distância]
Árv. Decisão	K Melhores	[150,120,80,60,30]
	Critério	[gini, entropy]
Árv. Aleat.	K Melhores	[150,120,80,60,30]
	Estimadores	[100,500,1000,4000]
	Critério	[gini, entropy]
AdaBoost	K Melhores	[150,120,80,60,30]
	Estimadores	[50,100,500]
	Tx. de Aprend.	[0.1,0.4,0.7,1, 1.3, 1.6,1.9]
GradBoost	K Melhores	[150,120,80,60,30]
	Estimadores	[50,100,500]
	Tx. de Aprend.	[0.1,0.4,0.7,1, 1.3, 1.6,1.9]
XGBoost	K Melhores	[150,120,80,60,30]
	Gamma	[0.5,1,5]
SVM	K Melhores	[150,120,80,60,30]
	C	[1,10,100,1000]
	Gamma	[1,0.1,0.001,0.0001]
	Kernel	['linear','rbf', 'poly','sigmoid']

podemos concluir que os classificadores AdaBoost e Árvore de Decisão não obtiveram um bom desempenho para classificação de gênero musical dado esse conjunto de dados. A Figura 7 ilustra o resultado do modelo SVM em forma de matriz de confusão, onde a diagonal principal da matriz representa os verdadeiros positivos e é onde desejamos que a concentração da nossa previsão esteja, pois ela está ligada diretamente a acurácia do modelo. Acima da diagonal principal estão os falsos positivos enquanto abaixo da diagonal estão os falsos negativos.

É possível notar também que a maioria dos classificadores obtiveram acurácia próxima ou igual a 100% no conjunto de treino. Esses resultados podem indicar que está havendo um *overfitting* e isso pode ser um dos motivos de não obter acurácia maiores no conjunto de teste. Como tentativa de reduzir esse *overfitting* utilizou-se técnicas para reduzir esse problema na segunda etapa do projeto

B. Etapa 2

A Tabela III apresenta as configurações dos modelos escolhidos usando o método de grade. Pode-se observar que o algoritmo escolheu normalmente 60 melhores atributos para fazer a classificação reduzindo assim a complexidade do modelo. Já a Tabela V ilustra a acurácia dos modelos após a seleção de atributos e o PCA e usando os modelos com

TABELA III
CONFIGURAÇÃO DOS CLASSIFICADORES COM OS HIPER-PARÂMETROS ESCOLHIDOS

Modelo	Hiper-parâmetros	Valores
Reg. Logística	K Melhores	60
	Penalidade	l2
	C	0.61
	Solver	lbfgs
KNN	K Melhores	80
	K Vizinhos	7
	p	1
	Pesos	distância
Árv. Decisão	K Melhores	60
	Critério	entropy
Árv. Aleat.	K Melhores	80
	Estimadores	4000
	Critério	gini
AdaBoost	K Melhores	60
	Estimadores	500
	Tx. de Aprend.	1.9
GradBoost	K Melhores	30
	Estimadores	500
	Tx. de Aprend.	0.5
XGBoost	K Melhores	60
	Gamma	0.5
SVM	K Melhores	60
	C	100
	Gamma	0.001
	Kernel	rbf

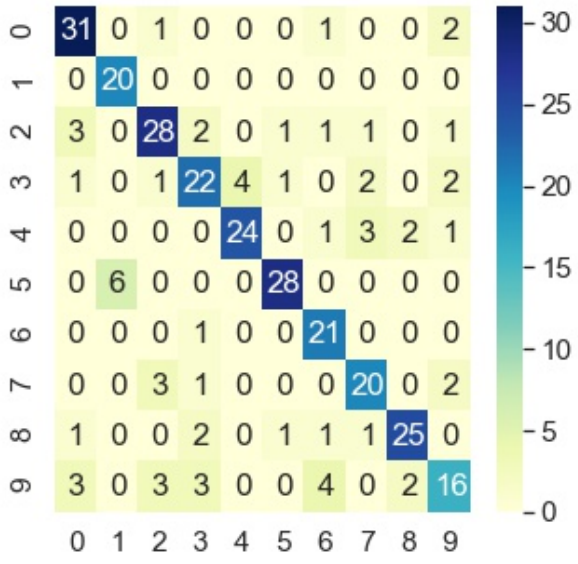

Fig. 7. Matriz de Confusão para o classificador SVM padrão da etapa 1

TABELA IV
ACURÁCIA DOS MODELOS PADRÕES SEM TRATAMENTO DOS DADOS

Modelo	Acurácia Treino	Acurácia Teste	f1-score Teste
Reg. Log	100%	73,6%	73,6%
KNN	80%	69,3%	68,9%
Arv. Decisão	100%	50,3%	49,4%
Árv. Aleatória	100%	71%	70,6%
AdaBoost	20%	18,3%	10,76%
GradBoost	100%	74%	73,7%
XGBoost	100%	73,3%	72,73%
SVM	95%	78,3%	78%

os parâmetros ajustados. Podemos observar que para todos os modelos houve um aumento na acurácia no conjunto de teste, com destaque para regressão logística e SVM que obtiveram maiores acurácias agora já em torno de 80%. Outro detalhe interessante é na acurácia no conjunto de treino da regressão logística que diminui enquanto a do teste aumentou 7%. Tal observação pode indicar que obtemos um modelo mais geral com menos *overfitting*. Outra observação que pode-se fazer é que houve um desempenho pior para os algoritmos GradBoost e XGBoost. Isso pode ter ocorrido devido a escolha dos hiper-parâmetros usados para fazer a busca. As Figuras 8 e 9 ilustram a matriz de confusão dos modelos de regressão logística e SVM, na qual obtiveram melhores desempenhos.

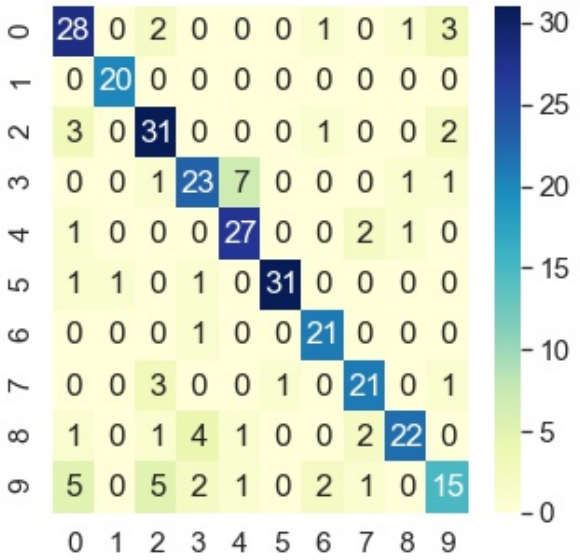

Fig. 8. Matriz de Confusão para o classificador SVM padrão da etapa 2

VII. CONCLUSÕES

Neste trabalho, estudou-se um projeto de aprendizado de máquina com o objetivo de realizar classificações de gêneros musicais dado a base de dados GTZAN com um acervo

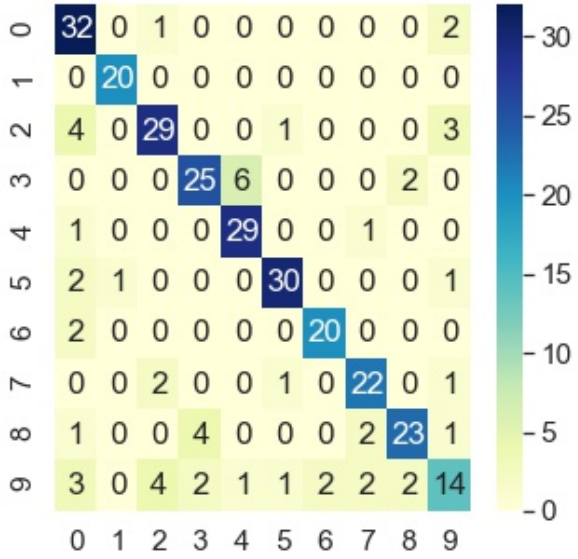

Fig. 9. Matriz de Confusão para o classificador de Regressão Logística padrão da etapa 2

TABELA V
ACURÁCIA DOS MODELOS AJUSTADOS COM SELEÇÃO DE ATRIBUTOS

Modelo	Acurácia Treino	Acurácia Teste	f1-score Teste
Reg. Log	92,4%	81,3%	80,9%
KNN	100%	72,6%	72,5%
Arv. Decisão	100%	45,5%	44,7%
Árv. Aleatória	100%	71,7%	71,7%
AdaBoost	72,4%	50,64%	50,64%
GradBoost	100%	64,5%	64,5%
XGBoost	100%	64,6%	64,6%
SVM	95%	79,3%	79,3%

com mais de 1000 músicas. Coube neste trabalho extrair os atributos deste Dateset e formar um conjunto de dados que podem ser usados pelos classificadores. Vimos que foi possível obter uma acurácia de em torno de 75% nos modelos de regressão logística e SVM, enquanto que os Ensembles usados não obtiveram um desempenho tão satisfatório. Já quando realizamos a seleção de atributos usando os K melhores atributos, PCA e o ajuste dos hiper-parâmetros dos modelos observamos que a busca em grade escolheu 60 atributos e houve uma melhora na acurácia dos modelos, principalmente a regressão logística e o SVM. Com isso, vimos a importância de realizar a engenharia de atributos em conjunto de dados antes de usá-los nos modelos de reconhecimento de padrões. Deseja-se em trabalhos futuros explorar melhor a otimização dos hiper-parâmetros e utilizar outras técnicas, buscar outras ferramentas capazes e reduzir os problemas de viés e variância e melhorar o desempenho dos modelos.

REFERÊNCIAS

[1] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content: a survey," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 133–141, 2006.

[2] Y. Chathuranga and K. Jayaratne, "Automatic music genre classification of audio signals with machine learning approaches," *GSTF Journal on Computing (JoC)*, vol. 3, no. 2, 2014.

[3] D. Jang, M. Jin, and C. D. Yoo, "Music genre classification using novel features and a weighted voting method," in *2008 IEEE International Conference on Multimedia and Expo*. IEEE, 2008, pp. 1377–1380.

[4] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing*, vol. 10, no. 5, pp. 293–302, 2002.

[5] B. L. Sturm, "The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use," *arXiv preprint arXiv:1306.1461*, 2013.

[6] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "A survey of audio-based music classification and annotation," *IEEE transactions on multimedia*, vol. 13, no. 2, pp. 303–319, 2010.

[7] D. Chathuranga and L. Jayaratne, "Musical genre classification using ensemble of classifiers," in *2012 Fourth International Conference on Computational Intelligence, Modelling and Simulation*. IEEE, 2012, pp. 237–242.

[8] D. Giannoulis, M. Massberg, and J. D. Reiss, "Parameter automation in a dynamic range compressor," *Journal of the Audio Engineering Society*, vol. 61, no. 10, pp. 716–726, 2013.

[9] S. Dubnov, "Generalization of spectral flatness measure for non-gaussian linear processes," *IEEE Signal Processing Letters*, vol. 11, no. 8, pp. 698–701, 2004.