# Prediction of musical genres using machine learning techniques

**Cleyton Aparecido Dim**[1] **Luiz Alves**[2] **Pedro Sousa**[3]

[1]Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Universidade Federal do Pará - UFPA
Belém, Pará, Brasil

[2]Programa de Pós-Graduação em Desenvolvimento Regional e Meio Ambiente (PGDRA)
Universidade Federal de Rondônia - UNIR
Porto Velho, Rondônia, Brasil

[3]Programa de Pós-Graduação em Engenharia Elétrica (PPGEE))
Universidade Federal do Pará - UFPA
Belém, Pará, Brasil

cleytondim@ufpa.br, luizmeteoro@gmail.com, pedro.filho@icen.ufpa.br

***Abstract.*** *The Internet provides a huge amount of audio files dispersed by various services, often identifies and organizes these files by genre is fundamental to draw a user profile and provide a personalized service. This work applies to the single media method of music may be music, classes, music, music, jazz, blues, pop, rock and heavy metal. The proposed solution is supported in machine learning techniques, using a combination of base model (Random Matrix) in specific models for the Metal, Rock and Pop genres. The maximum accuracy achieved by the modeling was 80.64 %.*

***Resumo.*** *A Internet disponibiliza uma enorme quantidade de arquivos de áudio espalhados por inumeros serviços, muitas vezes identificar e organizar estes arquivos por gênero musical é fundamental para traçar um perfil de usuário e disponibilizar um serviço personalizado. Este trabalho pretende criar um método de classificação capaz de identificar 6 classes diferentes de gêneros musicais (música clássica, jazz, blues, pop, rock e heavy metal) de forma automática. A solução proposta apoia-se em técnicas de aprendizado de máquina, utilizando a combinação de um modelo de base (Random Forest) com modelos específicos para os gêneros Metal, Rock e Pop. A acurácia máxima alcançada pelo modelo combinado foi de 80,64%.*

## 1. Introdução

Aprendizado de Máquina é uma área de Inteligência Artificial cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado bem como a construção de sistemas capazes de adquirir conhecimento de forma automática [Monard and Baranauskas 2003]. É uma das tendências mais modernas da tecnologia atualmente, onde seus algoritmos são usados nas mais diversas áreas do conhecimento científico, em sua grande parte obtendo resultados satisfatórios.

Uma das aplicações dos algoritmos usados em Aprendizado de Máquina é para o reconhecimento de padrões, onde a máquina é treinada para, a partir de uma base de dados

de treinamento, identificar, classificar e reconhecer certos padrões desejáveis ao usuário, tais como formas geométricas, sons e imagens [Bergamini et al. ].

Para identificação e classificação de sons e ritmos musicais, os mais diversos algoritmos de Aprendizado de Máquina, tais como o Support Vector Machine (SVM), o k-Nearest Neighbors (KNN), as Redes Neurais Artificiais (RNA), entre outros foram muito utilizados e há vasta literatura a respeito de sua utilização mostrando o sucesso do uso destes algoritmos e a comparação de desempenho entre eles para a realização desta atividade.

[Borges Jr et al. 2014] utilizou SVM e RNA para identificar e classificar de forma automática 13 gêneros musicais diferentes, incluindo gêneros brasileiros como Sertanejo, MPB, Forró e Samba e concluiu que ambos algoritmos conseguiram atingir seu propósito, sendo que o SVM se mostrou 42% mais eficiente que a RNA.

[Moreira 2017] utilizou os algoritmos Random Forest, Bayes Net, AdaBoost, Bagging, SVM e KNN para classificar e identificar as 150 músicas mais populares de uma webradio divididas em 7 gêneros musicais (rock, jazz, pop, música clássica, MPB, heavy metal e samba) e comparou o desempenho entre eles para avaliar qual obteve melhor acurácia. O autor obteve 66,53% de acerto utilizando os algoritmos Bagging e Bayes Net e ressalta que como alguns gêneros musicais são derivados de outros gêneros, os algoritmos encontraram certa dificuldade em fazer tal diferenciação e estimula o uso de algoritmos como RNA e Algoritmos Genéticos (AG) para aprimorar esta diferenciação.

Já [Figueiredo 2017] ao invés de utilizar pequenos trechos musicais para identificação e classificação dos gêneros musicais como fizeram os autores acima mencionados, utilizou o algoritmo SVM para classificar os gêneros musicais baseando-se nos espectogramas em forma de imagem de cada música como base de dados de treinamento e reconhecimento. O autor obteve neste método uma acurácia superior a 70%, algo similar a percepção humana.

Como se pode observar, é possível treinar a máquina para identificar e classificar diversos gêneros musicais automaticamente, usando os diversos algoritmos existentes e nos mais variados métodos possíveis para isso. Ciente disso, este trabalho tem como objetivo analisar diferentes algoritmos para identificar e classificar de forma automática diversos trechos musicais de 6 gêneros diferentes e fazer uma comparação no desempenho de cada um deles. Este problema de classificação foi proposto em uma competição de machine learning utilizando a plataforma kaggle[1] como base.

## 2. Preparação dos dados

O conjunto de dados para treino, apresenta 12 495 vetores de características, compostos por 191 parâmetros quantitativos e um qualitativo dividido em seis categorias: música clássica, jazz, blues, pop, rock e heavy metal. Os primeiros 127 parâmetros são baseados no padrão MPEG-7 e os restantes descrevem coeficientes cepstrais e parâmetros dedicados relacionados ao tempo:

- parâmetro 1: Centróide Temporal,
- parâmetro 2: valor médio do Centróide Espectral,

[1] https://www.kaggle.com/c/music4ed

- parâmetro 3: variância do Centróide Espectral,
- parâmetros 4-37: valores médios do Envelope de Espectro de Áudio (ASE) em 34 faixas de freqüência
- parâmetro 38: valor médio ASE (média de todas as bandas de frequência)
- parâmetros 39-72: valores de variância ASE em 34 bandas de freqüência
- parâmetro 73: parâmetros de variância média do ASE
- parâmetros 74,75: Centróide do Espectro de Áudio - valores médios e de variância
- parâmetros 76,77: Espalhamento de Espectro de Áudio - valores médios e de variância
- parâmetros 78-101: valores médios da Medição da Planura Espectral (SFM) para 24 bandas de frequência
- parâmetro 102: valor médio SFM (média de todas as bandas de freqüência)
- parâmetros 103-126: valores de variância de medição de planura espectral (SPECM) para 24 bandas de frequência
- parâmetro 127: parâmetros de variância média do SFM
- parâmetros 128-147: 20 valores médios dos primeiros coeficientes de mel cepstral
- parâmetros 148-167: o mesmo que 128-147
- parâmetros 168-191: parâmetros dedicados no domínio do tempo baseados na análise da distribuição do envelope em relação ao valor rms

Foram testados diversos classificadores: Lazys, Árvores, Lineares, Vetores de Suporte, Funções e Modelos Probabilísticos. Dentre eles: SVM, KNN, Ensemble Selection, Rules, Random Subspace, Random Forest, NaiveBayes, Chirp, Linear e LogitBoost. Em um primeiro momento, utilizou-se os classificadores do software MatLab e depois passou-se a utilizar apenas o Weka. Todas as validações dos modelos foram feitas em Split de treino/teste 66%.

Inicialmente, para testar o sistema de submissão do kaggle, foi gerado um arquivo com classificação aleatória. Posteriormente, no Matlab, foram utilizados alguns classificadores sem realizar pré-processamento nos dados. Até então, o baseline da competição não havia sido atingido.

Foram realizados diversos teste com classificadores no weka, utilizando os dados sem pré-processamento, com padronização por desvio padrão e por normalização de reescala 0 a 1000, de modo que mantivesse distribuição numérica facilmente observável. A normalização por reescala aparesentou maior acurácia nos modelos de teste e na verificação do kaggle, então passou-se a utilizá-la em todos os modelos posteriores. Outra etapa impotante do pré-processamento, foi a remoção de dados duplicados, totalizando 615 registros repetidos. Verificou-se também que as acurácias nos modelos em geral, diminuíram com a remoção dos outliers.

## 3. Modelagem

Os modelos base utilizam dados normalizados por reescala, não possuem dados duplicados e mantêm os outliers. Entretanto, foram removidas 20 Features que estavam repetidas: PAR_MFCCV1 a PAR_MFCCV20. Restando testar a influência do balancemento das classes.

Foram realizados diversos testes com cortes aleatórios em instâncias de classes específicas, de modo a igualar o número de instâncias em cada classe. Este balanceamento ocasionou uma queda na acurácia. Em todos os testes, o melhor classificador

sempre demonstrou ser o Random Forest, apresentando boa acurácia no Kaggle, superior ao baseline.

Na tentativa de melhorar a acurácia, tentou-se então balancear os dados de outra forma, removendo dados de diferentes classes. Para investigar os possíveis candidatos à remoções, realizou-se uma validação cross-fold no classificador Random Forest, buscando uma maior precisão na matriz de confusão para identificar as classes com maiores taxas de falsos positivos. Identificou-se que foram as classes Classical, Jazz e Rock. A Figura 1 ilustra a matriz de confusão.

```
=== Confusion Matrix ===

    a    b    c    d    e    f   <-- classified as
 1630   47   18  127    7   11 |    a = Rock
   46 1426    4    6    1   17 |    b = Pop
   29   21  782    6    6   36 |    c = Metal
   16    0    1 2542  300    1 |    d = Jazz
    0    0    0  104 3175    1 |    e = Classical
    2    4    0    5    0 1509 |    f = Blues
```

**Figura 1. Matriz de Confusão. Fonte: Própria(2019)**

Após diversas tentativas, a melhor solução encontrada foi remover aleatóriamente algumas instâncias das classes Classical e Rock, bem como foram gerados dados novos para a classe Metal, que apresentava um número baixo de predições. Estes dados foram gerados de duas formas: Clonando as instâncias de Metal e fazendo um deslocamento "para baixo", e fazendo uma segunda clonagem com os novos valores tendo seus valores multiplicados por percentuais aleatórios não superiores a 150%. Esta solução resultou em acurácia de 76,83% no kaggle, com classificador Random Forest parametrizado com número de iterações 300.

O gráfico de distribuição em um dos parâmetros (PAR_ASS_V) e a distribuição numérica das classes, demonstra o rebalanceamento do modelo em contraste aos dados originais apenas normalizados:

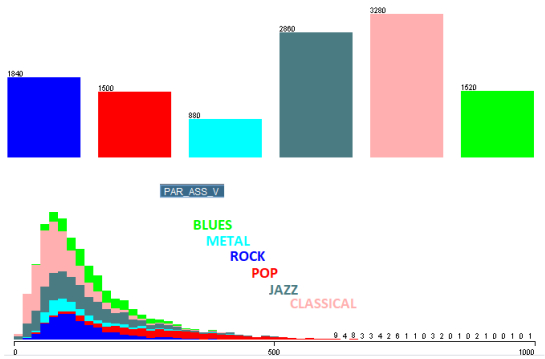


**Figura 2. Distribuição do parâmetro PAR_ASS_V sem balanceamento. Fonte: Própria (2019)**

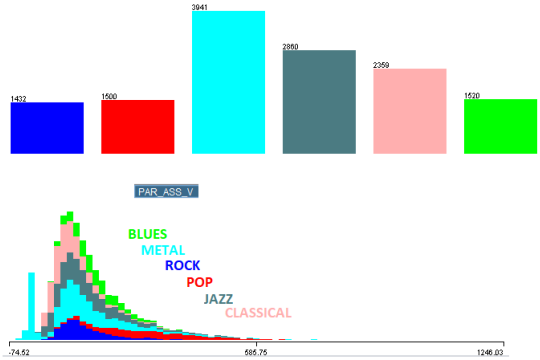


**Figura 3. Distribuição do parâmetro PAR_ASS_V com balanceamento. Fonte: Própria (2019)**

## 4. Avaliação dos modelos

Os diversos classificadores utilizados foram relacionados na Tablea 1, onde encontram-se as informações sobre remoção de outliers, métodos de feature selection e balanceamento:

| Classificador | % Modelo | % Kaggle | Featured | Outlied | Balanced |
|---|---|---|---|---|---|
| Random Forest 300 it | 93,94% | 76,83% | 20 eq | não | Fill Metal - Cut Rock e Classical |
| Random Forest 310 it | 93,94% | 76,83% | 20 eq | não | Fill Metal - Cut Rock e Classical |
| Random Forest 320 it | 93,97% | 76,70% | 20 eq | não | Fill Metal - Cut Rock e Classical |
| Random Forest 300 it | 94% | 76,38% | 20 eq | não | Fill Metal - Cut Rock e Classical |
| Random Forest | 94% | 75,74% | não | não | Fill Metal - Cut Rock e Classical |
| Random Forest 1800 it | 94% | 75,55% | não | não | não |
| Random Forest 3500 it | 94% | 75,49% | não | não | não |
| Random Forest 1500 it | 94% | 75,49% | não | não | não |
| Random Forest | 93,63% | 75,43% | não | não | Classical |
| Random Forest 1000 it | 94% | 75,43% | não | não | não |
| Random Forest | 93,58% | 75,23% | F-16 | não | não |
| Random Forest | 93,62% | 75,11% | não | não | Cut Classical 2 |
| Random Forest 500 it | 94% | 75,05% | não | não | não |
| Random Forest | 94,09% | 74,85% | não | sim | Classical e Jazz |
| Random Forest | 93,97% | 74,79% | não | sim | não |
| Random Forest | 94,28% | 74,21% | não | sim | Classical e Jazz 2 |
| Random Forest | 94,04% | 74,15% | não | sim | Classical |
| Random Forest | 94,41% | 74,02% | N50 | não | não |
| Random Forest | 93,56% | 73,83% | não | não | Cut Classical 3 |
| Random Forest | 93,28% | 73,58% | F-Test | não | não |
| Random Forest Out | 93% | 73,45% | não | sim | não |
| KNN 1 | 96,33% | 74,45% | não | sim | não |
| Random Forest | 94,02% | 73,32% | N-120 | não | não |
| BayesA1DE | 92,92% | 72,43% | não | sim | Classical e Jazz |
| Random Subspace | 90,97% | 72,30% | não | não | não |
| Random Forest | 93,83% | 72,30% | não | não | Classical e Jazz 2 |
| Random Forest F2 | 92% | 71,66% | sim | não | não |
| Random Forest | 96,72% | 71,60% | não | sim | não |
| Chirp | 84,41% | 71,09% | não | não | não |
| Random Forest | 93,26% | 70,89% | não | não | Cut Classical e Jazz |
| Ens. Sel. 1000it 1000bag | 89,00% | 69,87% | não | não | não |

| Classificador | % Modelo | % Kaggle | Featured | Outlied | Balanced |
|---|---|---|---|---|---|
| kNN 7 | 93,87% | 69,81% | não | não | Cut Classical e Jazz |
| Linear | 86,93% | 69,49% | não | não | não |
| Random Subspace | 91,62% | 69,43% | não | sim | Classical e Jazz 2 |
| KNN 1 | 96,32% | 68,60% | não | não | Classical |
| KNN 70 | 86,37% | 68,47% | não | não | não |
| KNN 25 | 90,50% | 68,47% | não | não | não |
| Random Forest | 94,57% | 67,96% | não | sim | não |
| KNN 100 | 84,92% | 67,90% | não | não | não |
| KNN 109 | 84,50% | 67,77% | não | não | não |
| Random Forest | 92,98% | 67,77% | não | não | Cut Classical |
| KNN 1 | 96,76% | 67,70% | não | sim | Classical e Jazz 2 |
| KNN 1 | 96,00% | 67,45% | 20 eq | não | Fill Metal - Cut Rock e Classical |
| Rules Part | 87,91% | 67,26% | não | não | não |
| KNN 1 - Manhattan | 97,06% | 66,43% | não | não | Peso Jazz |
| KNN 1 | 95,73% | 68,36% | F-Test | não | não |
| Naive Bayes | 73,16% | 63,68% | não | não | não |
| Ensemble Selection | 88,90% | 60,11% | não | sim | Classical e Jazz 2 |

Tabela 1: Comparação entre os classificadores utilizados

Pode-se observar, através da Tabela 1, que a acurácia na validação do modelo difere muito da acurácia obtida no kaggle. Uma hipótese seria que o conjunto de teste ou algumas classes pertencente a ele, compreendem faixas diferentes do conjunto de treino. A verificação da hipotese foi feita através de uma tabela contendo os dados de treino e teste, comparando essas distribuições nos visualizadores do weka.

A comparação indica que os dados do conjunto de teste ultrapassam os limites inferiores e superiores do conjunto de treino (com os dados normalizados para a escala de 0 a 1000). A Figura 4 ilustra o comportamento do parâmetro PAR_SFM_13.

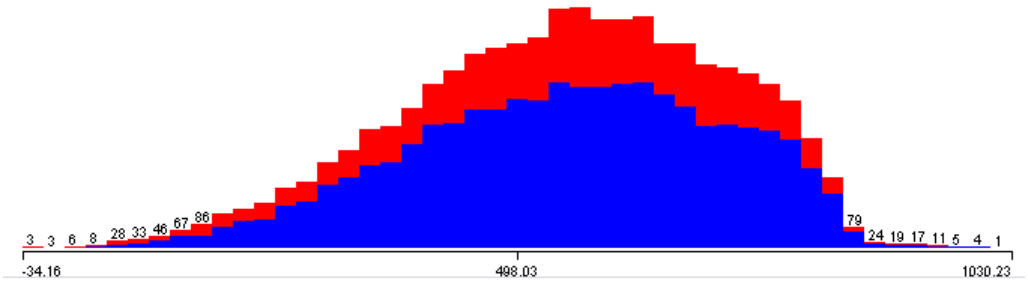


**Figura 4. Comparação dos dados de treino e teste para o parâmetro PAR_SFM_13. Fonte: Própria (2019)**

Buscando aumentar a acurácia dos modelos, aplicaram-se métodos de Feature Selection como RFE, Correlation, SubSets BestFirst, porém não foram obtidas melhorias

significativas na pontuação. Deduziu-se então que as features importantes para uma classe podem não ser importantes para uma outra, logo, a remoções de features podem aumentar a precisão de uma classe específica, mas diminuir em uma outra, não contribuindo para aumentar a acurácia geral do modelo.

Assim, pensou-se na seguinte solução: Utilizar como corpo base de predições a melhor submissão até o momento (Random Forest com 300 iterações em um conjunto de treino normalizado na escala 0 a 1000, com 20 Features repetidas removidas, sem remoção de outliers, preenchimento de dados da classe Metal e remoção de dados aleatórios das classes Rock e Classical). A este arquivo de predições, foram sobrescritas as predições de um modelo específico para as classes Metal, Rock e Pop, cada um com as features apropriadas para aquela classe.

A seleção destas features foi feita visualmente, pelo conjunto de treino original, observando em quais parâmetros aquela classe especificamente tinha os valores mais divergentes das demais classes, ou pelo menos divergentes de um grupo de classes. As classes Classical e Jazz tem uma distribuição muito semelhante de seus valores, sendo provavelmente a razão de, na matriz de confusão, instâncias de Jazz serem classificadas como Classical, e algumas destas como Jazz, optando-se então por não fazer seleção de features para elas. A classe Blues apresentou uma maior diferenciação das demais classes, considerando-se satisfatória suas predições. Nas imagens abaixo, exemplos de diferenciação de dados:

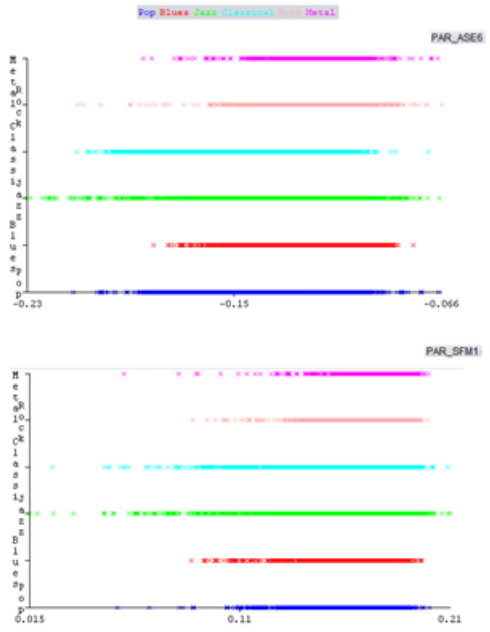


**Figura 5. Exemplo de Features utilizadas para a classe Metal. Fonte: Própria (2019)**

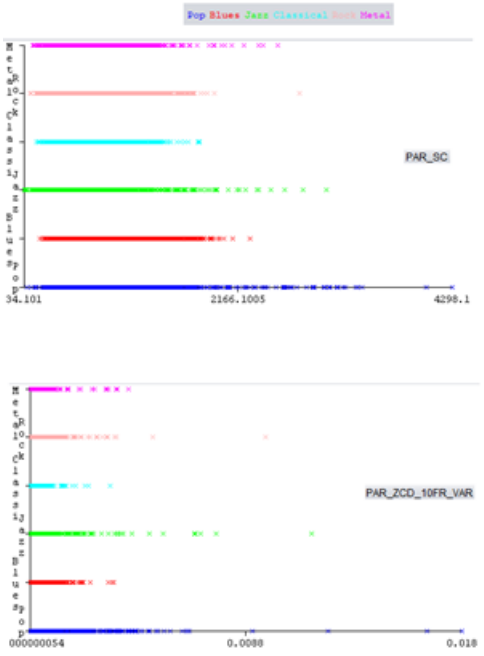


**Figura 6. Exemplo de Features utilizadas para a classe Pop. Fonte: Própria (2019)**

No arquivo base de predições foram substituidas as predições do modelo específico para a classe Metal, que teve o dobro do número anterior de predições. Ou seja, toda predição do gênero Metal no modelo específo para classe Metal, foi colocado

como Metal no arquivo base de predições. A submissão ao Kaggle indicou uma melhora expressiva na acurácia: 77,60%.

Em seguida, foi gerada a predição do modelo específico da classe Rock. Agora, ao inserir os dados da predição deste modelo no arquivo base, foi tido como regra que só seria inserida a predição caso a predição no arquivo base não fosse Metal. A acurácia manteve-se em 77,60%

Por fim, foi gerada a predição do modelo da classe Pop. Seguiu-se a regra anterior, não alterando as predições no arquivo base, caso fossem Metal ou Rock. A acurácia teve um salto para 79,38% (no modelo). Esta ultima predição foi a submetida à validação final no encerramento da competição.

Ao término da competição, com 139 submissoes, a equipe ficou em primeiro lugar, com a acurácia sendo 80,64% (no Kaggle). Abaixo, um gráfico temporal das submissões com suas respectivas acurácias parciais:

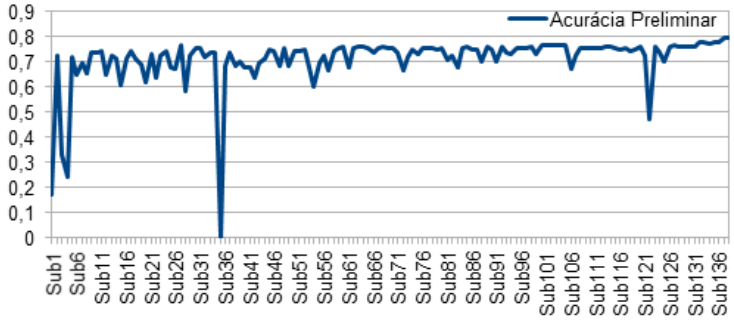


**Figura 7. gráfico das submissões e suas respectivas acurácias. Fonte: Própria (2019)**

## 5. Conclusão

Neste trabalho, foi apresentado um método para a classificação de gêneros musicais. Este método combina um modelo base e modelos especificos para os gêneros Metal, Rock e Pop. O modelo base utiliza Random Forest com 300 iterações, conjunto de treino normalizado (escala 0 a 1000), eliminação de Features repetidas, preenchimento de dados da classe Metal e remoção de dados aleatórios das classes Rock e Classical

Foram realizadas no total 139 submissões à plataforma Kaggle ao longo da competição, a acurácia máxima alcançada pelo modelo combinado foi de 80,64%.

## Referências

[Bergamini et al. ] Bergamini, C. M., Araujo, P. V., and Motter, G. Modelos de aprendizagem de máquina na classificação de caracteres manuscritos. *Synergismus scyentifica*, pages 338–348.

[Borges Jr et al. 2014] Borges Jr, E., Simas Filho, E. F., and Fernandes Jr, A. C. L. (2014). Classificadores de gêneros musicais usando máquinas de vetor de suporte e redes neurais. *In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 20, 2014,Belo Horizonte: SBA*, 1:1269–1276.

[Figueiredo 2017] Figueiredo, U. R. (2017). Uma abordagem visual para classificação de gêneros musicais utilizando pontos-chave de um espectograma. `http://www.swge.inf.br/CBA2014/anais/PDF/1569935375.pdf`. 70 p. Trabalho de Conclusão de Curso (Graduação) – Universidade Federal do Rio de Janeiro, Escola Politecnica, Curso de Engenharia Eletrônica e de Computação, Rio de Janeiro, 2017.

[Monard and Baranauskas 2003] Monard, M. and Baranauskas, J. (2003). Conceitos sobre aprendizado de máquina, chapter 4. *Volume*, 1:89–114.

[Moreira 2017] Moreira, P. S. C. (2017). Mineração de dados aplicada à classificação automática de gêneros musicais.