Qual é a música?
Uma abordagem de Machine Learning em Classificação de música por Gênero

Bruno Peixoto*, Beatriz Vasconselos†, Rodolfo Lindemute*, Thaís Pietro†
Departamento de Estatística do IMECC, UNICAMP
Campinas, São Paulo, Brasil
Email: *b116297@ime.unicamp.br, †b160291@ime.unicamp.br, ‡r116297@ime.unicamp.br, §t15737@ime.unicamp.br

***Resumo*—O artigo apresenta uma abordagem de Machine Learning para classificar músicas por gênero. Dado um fragmento musical o objetivo é classificá-lo quanto ao seu gênero através de técnicas de Machine Learning visando atingir a maior acurácia. Para tanto, inicialmente, foram aplicadas técnicas de redução de dimensão por PCA e LDA. Para classificação foram aplicados os modelos de KNN, Floresta Aleatória e XGBoost. Observando a acurácia de cada modelo, a maioria dos ajustes obtiveram performances similares, porém, o melhor ajuste foi com método XGBoost, sem aplicação de métodos de redução de dimensão nos dados.**

Palavras Chave - Machine Learning, PCA, XGBoost, Música, Classificação

I. Introdução

A classificação de músicas quanto ao seu gênero tem potencial de aplicação em diversos serviços de streaming, como no Spotify e no Youtube, por exemplo, essa classificação é utilizada na recomendação e gestão de playlists. O que possibilita a aplicação de técnicas de Machine Learning na classificação de áudio são parametrizações como as tratadas no trabalho de Kostek (2001) [1], que levam em conta características como timbre, ritmo,frequências sonoras, magnitude de sinal, entre outros. Isso oferece clareza e eficiência na descrição de dados baseado em quantidades compactas de informação.

Neste trabalho utilizou-se o banco de dados Music Genres (Kotesk et. al, 2011) [2] disponível na plataforma TunedIT. O banco contém fragmentos de músicas que foram classificados quanto ao gênero e parametrizados. O objetivo é classificar os gêneros musicais de cada fragmento, através técnicas de Machine learning, visando atingir a maior acurácia.

II. Trabalhos Relacionados

Com o banco de dados foi identificado um artigo relacionado ao tema em estudo (Kostek et. al, 2011) [1]. No qual propõe classificar automaticamente os sons de instrumentos musicais. O artigo tem como objetivo encontrar recursos sonoros significativos para instrumentos musicais e remover a redundância do sinal musical por meio de experimentos de classificação que foram realizados com redes neurais possibilitando uma discussão sobre a eficiência do processo de extração de características e suas limitações.

III. Banco de Dados

O conjunto de dados foi construído a partir de músicas de 60 intérpretes, de 15 a 20 músicas para cada intérprete. As músicas foram classificadas em seis gêneros: música clássica, jazz, blues, pop, rock e heavy metal e em seguida particionadas em 20 fragmentos. Cada segmento foi parametrizado segundo as técnicas de Kotesk (2001) [1]. Os dados foram divididos pelos autores em bancos de treino e teste com 12495 e 10296 observações respectivamente. Em ambos 191 variáveis preditoras (parâmetros), todas contínuas, e a classe (gênero musical).

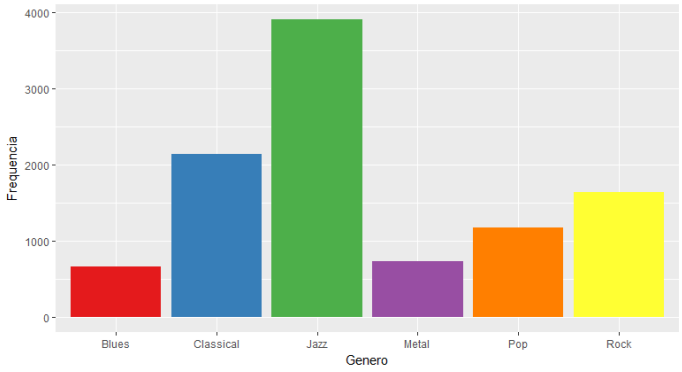

Figura 1. Frequência de observações por gênero.

Devido a grande quantidade de variáveis, uma visualização possível do banco de dados é feita através de suas duas primeiras componentes principais, que explicam cerca de 30% da variabilidade total dos dados, como pode-se observar na Figura 2.

Apesar da confusão entre os gêneros, pode-se fazer uma separação visual entre eles, nota-se uma partição com Rock e Metal e outra com Clássica e Jazz. Blues se sobrepõe sobre quase todos os gêneros. Enquanto Pop tem valores bem isolados.

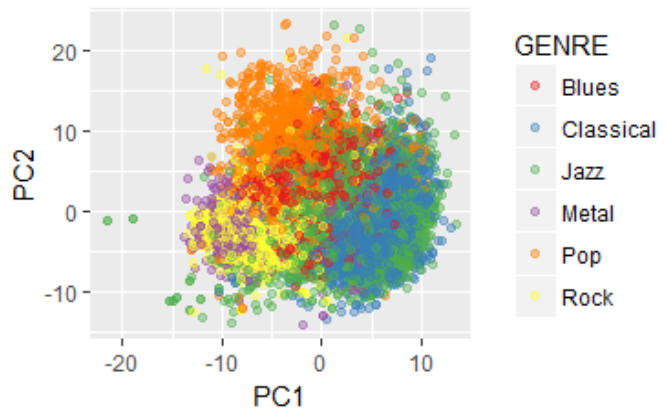

Figura 2. Primeira e segunda componentes principais, por gênero musical, nos dados de treino.

IV. Metodologia

A. Análise de Componentes Principais (PCA)

As componentes principais, são obtidas através de uma transformação linear nas variáveis, de forma que a primeira componente tenha a maior variância possível e as subsequentes tenham a maior variância possível desde que não correlacionadas com as anteriores. Assim o conjunto resultante é ortogonal e concentra a variabilidade dos dados nas primeiras componentes, podendo ser utilizado para redução de dimensão.

Aplicou-se a análise de componentes principais no conjunto de dados, após padronizá-los. Devido ao tamanho do banco, resolveu-se criar dois novos bancos: Um com 22 componentes principais, no qual todas componentes explicam mais de 1% da variabilidade do modelo. Outro com 73 componentes, na qual a variabilidade conjunta explicada é cerca de 95%.

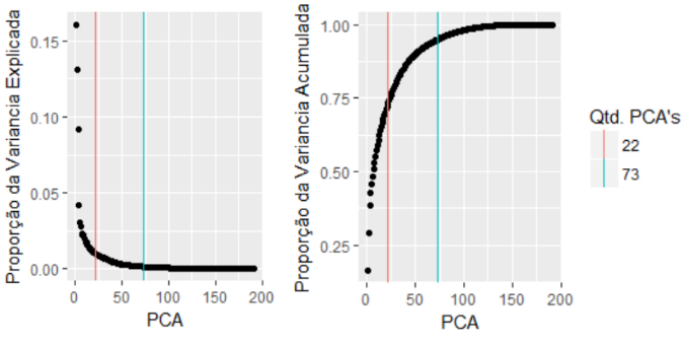

Figura 3. Proporção da Variância Explicada e Proporção da Variância Explicada Acumulada.

B. Análise Discriminante Linear (LDA)

A análise de discriminante é uma técnica que visa encontrar combinações lineares das variáveis preditoras que melhor discriminam os grupos de observações. Isto é, dado um conjunto de observações que podem ser agrupadas em partições a análise de discriminante busca encontrar vetores que minimizam a probabilidade de má classificação.

Esta técnica pode ser usada tanto em problemas de classificação quanto como redução de dimensão, que foi o caso do deste projeto.Para sua aplicação foi necessário utilizar PCA, devido a problemas computacionais.

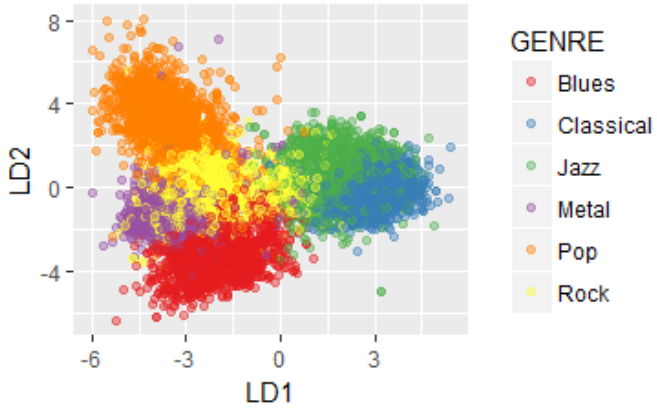

Figura 4. Primeiro e segundo vetores da Análise Discriminante Linear.

C. K Vizinhos mais Próximos (KNN)

O "K-Vizinhos mais próximos" ou KNN, é um método de classificação não-paramétrico que consiste em classificar uma nova observação a partir da classificação da maioria de seus k vizinhos mais próximos. O número de vizinhos, k, controla a flexibilidade do modelo e ajusta a compensação de viés-variância.

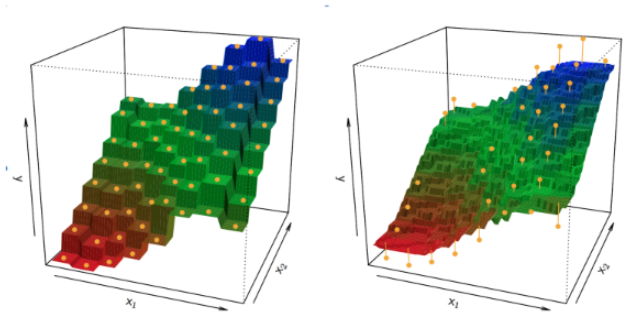

Figura 5. Exemplo de aplicação do KNN em dados de duas dimensões e 64 observações, com k=1 e k=9. [3]

Um problema observado no ajuste do modelo KNN para esse banco de dados é que o desempenho diminuiu à medida que dimensionalidade aumentou. Isso foi observado quando foi ajustado o modelo no banco sem redução de dimensão, pois, há uma redução no número de efetivo de observações ao aumentar a dimensão.

Para o ajuste do modelo foi utilizada a validação cruzada com 5 dobras e foram ajustados os modelos para valores de k variando de 1 até 20. O ajuste foi aplicado, no banco de teste original, no banco após a aplicação do pca com apenas 22 componentes, no banco com 73 componenentes e no banco com PCA e LDA.

D. XGBoost

XGBoost ou Extreme Gradient Boosting, introduzido por Chen et. al, 2015 [4], é uma técnica que combina Gradient Boost (Friedman, 1999) [5] com técnicas de regularização, como o Lasso (Tibshirani, 1996) [6]. O Gradient Boosting ajusta uma série de modelos simples (weak learners) de forma que um modelo tenta diminuir o erro do anterior, minimizando uma função de perda dada, ao final todos os modelos são combinados (ensemble). No XGBoost adiciona-se uma função de regularização a função perda, o que penaliza a complexidade do modelo, evitando-se sobreajuste.

A implementação de Chen et al., 2018, permite utilizar árvores de decisão e modelos lineares nos contextos de regressão e classificação. Neste trabalho utilizou-se ambas as técnicas no contexto de classificação. Os hiperparâmetros, de boosting e de regularização, foram ajustados através de validação cruzada (CV), com 3 dobras. O algoritmo foi aplicado nos banco com e sem redução de dimensão.

E. Floresta Aleatória

O procedimento de floresta aleatória, diferentemente da simples abordagem de árvore de decisão, envolve a implementação de múltiplas sub árvores, que combinadas fazem uma predição mais precisa a partir dos valores das variáveis preditoras.

Para a análise foi gerada diversas árvores por bootstrap de mesmo tamanho a partir do conjunto de dados de treinamento com o objetivo de reduzir a variância do modelo. Porém, diferentemente do processo "Bagging" onde considera-se todas as p variáveis preditoras ao gerar as árvores por bootstrap, em floresta aleatória para cada nó das árvores geradas são utilizadas apenas m preditoras selecionadas aleatoriamente, geralmente o valor de m é a raiz quadrada de p, para problemas de classificação. Esse procedimento faz com que as sub-árvores geradas no modelo produza predições não correlacionadas entre si ou no máximo pouco correlacionadas, assim é obtido uma maior diminuição na variância. Para o caso de classificação, a predição de cada observação pelo modelo de floresta aleatória é determinada de maneira que cada observação pertença a certa classe de acordo com a classe que obtiver maior frequência entre as múltiplas árvores geradas no ajuste.

No projeto foi implementado a floresta aleatória usando o conjunto de dados de treinamento diretamente (com 191 preditoras) e também o conjunto de dados de treino após a implementação de componentes principais seguida de uma análise de discriminante linear (com 5 preditoras resultantes do LDA).

V. Resultados

Os resultados obtidos após a aplicação dos modelos, tal como descrito na metodologia, podem ser visualizados na Tabela 1 e na Figura 6. Foram selecionados os modelos com maior acurácia para cada método, aplicando-se ou não técnicas de redução de dimensão, com exceção de XG Boost. Pode-se observar que muitos modelos tiveram resultados similares, mas é válido ressaltar que a maior acurácia foi obtida no XG Boost com árvores.

Tabela I
Acurácia no banco de dados de teste e treino, para os principais modelos aplicados.

Modelo	Teste	Treino
KNN - 22 Componentes	0.6273	0.9338
KNN - 73 Componentes	0.6482	0.9572
KNN - LDA	0.7466	0.8782
KNN	0.3303	0.4046
XGBoost - Árvore	0.7662	0.9730
XGBoost - Linear	**0.7689**	0.9730
LDA	0.7575	0.8757
Floresta Aleatória	0.7567	0.9400
Floresta Al. - PCA	0.6657	0.8900
Floresta Al. - LDAA	0.7413	0.9800

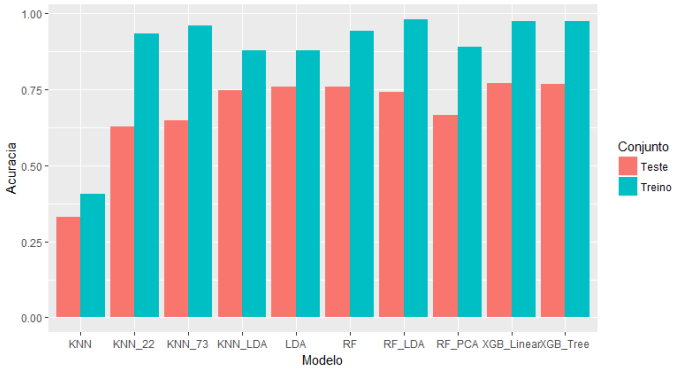

Figura 6. Acurácia no banco de dados de teste e treino, para os principais modelos aplicados.

De forma geral, todos os modelos tiveram uma performance melhor no banco de treino do que no de teste, o que é esperado. Com algumas exceções a maioria deles tem acurácia de cerca de 90% ou superior no banco de treino. Já no banco de testes os valores de acurácia ficam em torno de 75% para a maioria dos modelos.

A LDA foi aplicada como técnica de classificação e redução de dimensão. Como classificador a acurácia foi 88% nos dados de treino e 75,5% nos dados de teste. Como técnica de redução de dimensão foi utilizado em todos os modelos. No entanto é interessante observar que a combinação de técnicas foi melhor, i.e. trouxe maior acurácia no banco de teste, do que sua aplicação isolada como classificador.

O KNN teve o pior resultado quando aplicado no banco original, acertando apenas 40% das classificações no banco de treino e 36% no banco de testes. Este fenômeno, conhecido como maldição da dimensionalidade, é causado pelo grande número de variáveis. Note que quando aplicado a bancos com PCA e LDA sua performance aumenta consideravelmente.

Nas Florestas Aleatórias as técnicas de redução de dimensão não trouxeram efeitos consideráveis. Quando aplicado com PCA, o ajuste piorou a performance. No banco de LDA a acurácia no banco de treino foi maior, mas no banco de testes não.

As performances do XG Boost foram muito próximas,no modelo de árvores e linear, sendo o modelo de árvores um pouco melhor. Para este modelo foram apresentados somente os resultados obtidos nos dados sem redução de dimensão, estas técnicas reduzem a eficiência do algoritmo, assim como nas Florestas Aleatórias. Como este modelo apresentou a melhor performance, analisar-se-á seus resultados com maior profundidade.

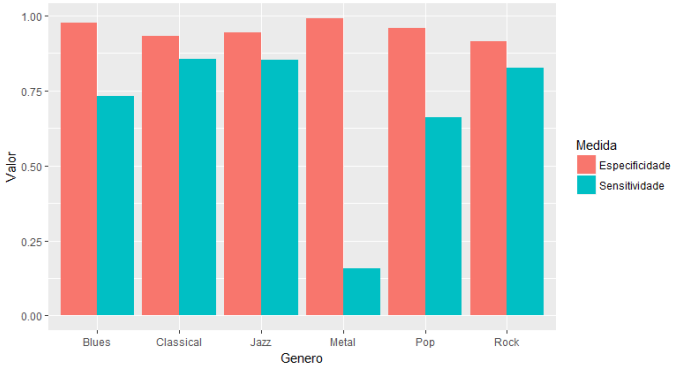

Figura 7. Sensitividade e especificidade, por gênero, do melhor modelo, o XGBoost Linear.

Observando-se o gráfico de Sensitividade e Especificidade, percebe-se que os valores menores de sensitividade estão nos gêneros com menos observações. Este fenômeno pode ser fruto do desbalanceamento dos dados. No entanto, o gênero Metal tem uma Sensitividade muito menor do que os outros gêneros, outro fator contribui para este evento.

De fato, na matriz de confusão pode-se ver que a maioria das observações pertencentes ao gênero metal são classificados como Rock. Intuitivamente são gêneros muito próximos musicalmente, no gráfico de PCA's (Figura 2), na descrição dos dados, são gêneros que se sobrepõem. Outros gêneros que se confundem são Jazz e Música Clássica e em menor escala Pop, Rock e Blues. Considerando-se que as variáveis preditoras são parâmetros que levam em conta, timbre, ritmo, entre outros, é razoável entender a confusão, devido às características musicais destes gêneros.

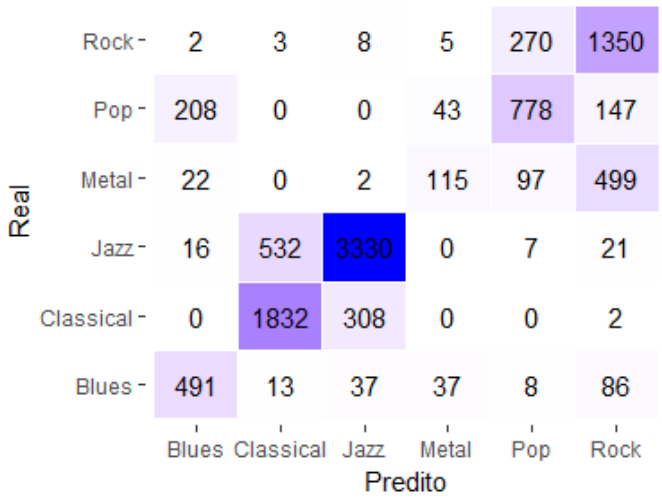

Figura 8. Matriz de Confusão para os dados de teste do modelo XGBoost.

A. *Conclusão*

Neste projeto procurou-se implementar algoritmos que possuíssem uma certa diversidade de representação e estilo de aprendizagem. Apesar de alguns modelos apresentarem uma acurácia similar, o melhor ajuste aos dados de teste foi atingido pelo modelo XBBoost, com 76,89% de acertos com relação ao banco de dados de teste. O resultado obtido com o XGBoost é satisfatório quando comparado com aqueles apresentados na competição TunedIT. Ao considerar o resultado dos 144 participantes que submeteram os projetos, a acurácia obtida com esse modelo traria a 36ª posição na competição.

Referências

[1] Kotesk, B., Czyzewski, A., " Representing musical instrument sounds for their automatic classification," Journal of the Audio Engineering Society. Audio Engineering Society, 49(9):768-785 · September 2001

[2] ISMIS 2011 Contest: Music Information Retrieval. Disponível em: <http://tunedit.org/challenge/music-retrieval/genresfbclid=IwAR0PDNuUSyGs5xIuV31IV6yw5kzDLt63-IeZIhoEpfBRQGp-lTGt1-ZNus > Acesso em: 14/11/2018 às 19h47.

[3] Tibshirani, R., James, G., Witten, D., Hastie, T., "An Introduction to Statistical Learning with Applications in R" Springer, 2013, p. 105.

[4] Chen, T., He, T., "Higgs Boson Discovery with Boosted Trees," JMLR: Workshop and Conference Proceedings, 42:69-80, 2015

[5] Friedman H., J. "Greedy Function Approximation: A Gradient Boosting Machine". ims 1999 reitz lecture. Fevereiro, 1999.

[6] Tibishirani, T. "Regression Shrinkage and Selection via the Lasso". Journal of the Royal Statistical Society. Series B (Methodological) © 1996. 58, N0.1 pp.267-288.

[7] Chen, T., He,T., Benesty, M., Khotilovich, V., Tang,y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie,J., Lin, M., Geng, Y., Li, Y. (2018). "xgboost: Extreme Gradient Boosting. R package version 0.71.2". CRAN. Disponível em: < https://CRAN.R-project.org/package=xgboost> Acessado em 19/11/2018 às 19h22