UNIVERSIDADE PRESBITERIANA MACKENZIE
Faculdade de Computação e Informática

Predição de Dançabilidade e Energia em Músicas Populares com Base em Características Sonoras

Enzo Ferroni[1], Luiz Gabriel Profirio Mendes[2]

[1]Universidade Presbiteriana Mackenzie (UPM)
Rua da Consolação, 930 Consolação, São Paulo - SP, 01302-907 - Brasil

{[1]10417100@mackenzista.com.br, [2]10382703@mackenzie.com.br}

https://github.com/omgitsgm/projeto-ciencia-dados

https://colab.research.google.com/drive/1AUWrsbPxsZjI4UUgtzov0A4k0d61mnHe

1. Descrição do Dataset

O dataset utilizado neste trabalho reúne informações sobre músicas populares do Spotify durante o ano de 2023. Cada linha representa uma faixa musical, acompanhada de diversas variáveis numéricas relacionadas às suas propriedades sonoras, como batidas por minuto (BPM), valência, energia, nível de fala (speechiness), intensidade acústica (acousticness), entre outras. Essas variáveis são derivadas da análise computacional do áudio e fornecem um panorama técnico do perfil musical de cada faixa.

Este conjunto de dados oferece uma base interessante para explorar relações internas entre atributos musicais. Em vez de focar em popularidade medida por streams, o trabalho busca prever o grau de dançabilidade de uma música, uma métrica que reflete o quanto uma faixa é ritmicamente adequada para dançar, utilizando como base outras características sonoras. Esse tipo de análise permite investigar se o perfil técnico de uma música pode explicar ou antecipar seu potencial de engajamento corporal.

2. Definição da tarefa

A tarefa definida neste trabalho consiste em prever o nível de dançabilidade de uma música com base em suas características sonoras. A variável dançabilidade (danceability_%) é uma medida contínua, fornecida pelo próprio Spotify, que expressa o quanto uma faixa é considerada apropriada para dançar, levando em conta elementos como ritmo, estabilidade do tempo, força de batida e regularidade. Essa variável será tratada como a "variável alvo" do problema.

As variáveis preditoras selecionadas incluem atributos objetivos e numericamente expressos do áudio, como BPM (batidas por minuto), valência (valence_%), energia (energy_%), intensidade acústica (acousticness_%), presença de elementos instrumentais (instrumentalness_%), nível de fala (speechiness_%), liveness (proximidade de uma performance ao vivo) e o modo musical (mode), que indica se a música está em escala maior ou menor.

A proposta é investigar se existe uma relação consistente entre essas variáveis técnicas e o nível de dançabilidade atribuído a cada música. Para isso, serão utilizados dois modelos de regressão: Regressão Linear, pela sua simplicidade e interpretabilidade, e K-Nearest Neighbors (KNN), que permite capturar padrões locais baseados em similaridade entre observações. O objetivo é comparar a performance dos dois modelos na tarefa de estimar a dançabilidade com base em atributos sonoros previamente conhecidos.

3. Escolha dos Modelos

Para a tarefa de predição de características sonoras das músicas, foram selecionados dois algoritmos de regressão supervisionada com abordagens distintas: Regressão Linear e K-Nearest Neighbors Regressor (KNN). A escolha por esses modelos se baseia em suas naturezas complementares, permitindo analisar tanto relações lineares globais quanto padrões locais baseados em similaridade entre as observações.

A Regressão Linear foi utilizada por ser um modelo simples, eficiente e amplamente adotado em tarefas de regressão com variáveis contínuas. Sua principal vantagem está na capacidade de interpretar diretamente a influência de cada variável preditora sobre a variável alvo. Neste projeto, o modelo foi aplicado para prever os níveis de dançabilidade (danceability_%) e energia (energy_%) com base em atributos técnicos do áudio, como BPM, valência, intensidade acústica e outros. Por ser linear, o modelo fornece uma visão clara sobre a relação entre os atributos e os comportamentos musicais analisados.

Já o modelo K-Nearest Neighbors foi escolhido por sua abordagem não paramétrica, baseada na proximidade entre exemplos. Ele estima os valores alvo considerando a média das músicas mais semelhantes no espaço das variáveis, o que permite capturar padrões locais e relações não lineares. Para esse modelo, foi feita uma análise detalhada da variação do número de vizinhos (k), identificando o valor mais adequado com base no desempenho das métricas de erro. A comparação entre os dois modelos possibilitou avaliar se as variáveis analisadas apresentam relações predominantemente lineares ou se padrões de similaridade local são mais representativos.

4. Pré-processamento básico

Antes da aplicação dos modelos, foram realizadas algumas etapas essenciais de pré-processamento. Inicialmente, os dados foram carregados e foi necessário especificar a codificação correta do arquivo CSV devido a erros de leitura relacionados à acentuação. Em seguida, foram removidas linhas com valores ausentes na coluna “key”, por se tratar de uma variável importante e frequentemente utilizada na análise musical. Também foram excluídas colunas associadas a outras plataformas de streaming (como Apple Music, Deezer e Shazam), uma vez que o foco do projeto está voltado apenas para os dados do Spotify.

Além disso, foi feita a conversão da coluna “streams”, originalmente interpretada como texto, para o formato numérico, garantindo sua compatibilidade com operações matemáticas. A linha correspondente à música “Love Grows (Where My Rosemary Goes)” foi removida do conjunto por apresentar valores inconsistentes. Por fim, foram selecionadas apenas as variáveis numéricas relevantes para a tarefa, incluindo as variáveis preditoras e a “variável

alvo” (dançabilidade), garantindo que o conjunto de dados estivesse limpo, padronizado e adequado para aplicação dos modelos de regressão.

5. Implementação

Os dois modelos foram implementados separadamente utilizando o mesmo conjunto de dados de treino e teste. A seguir estão os trechos de código correspondentes a cada um.

Regressão Linear

```
[11] from sklearn.linear_model import LinearRegression
     from sklearn.model_selection import train_test_split

     # Seleção de variáveis preditoras e variável alvo
     features = ['bpm', 'energy_%', 'valence_%', 'acousticness_%',
                 'instrumentalness_%', 'speechiness_%', 'liveness_%', 'mode']

     df['mode'] = df['mode'].apply(lambda x: 1 if x == 'Major' or x == '1' or x == 1 else 0)

     X = df[features]
     y = df['danceability_%']

     # Divisão em treino e teste
     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

     # Treinamento do modelo
     reg = LinearRegression()
     reg.fit(X_train, y_train)

     # Previsões
     y_pred_lr = reg.predict(X_test)
```

KNN

```
[12] from sklearn.neighbors import KNeighborsRegressor

     # Treinamento do modelo KNN
     knn = KNeighborsRegressor(n_neighbors=5, weights='uniform')
     knn.fit(X_train, y_train)

     # Previsões
     y_pred_knn = knn.predict(X_test)
```

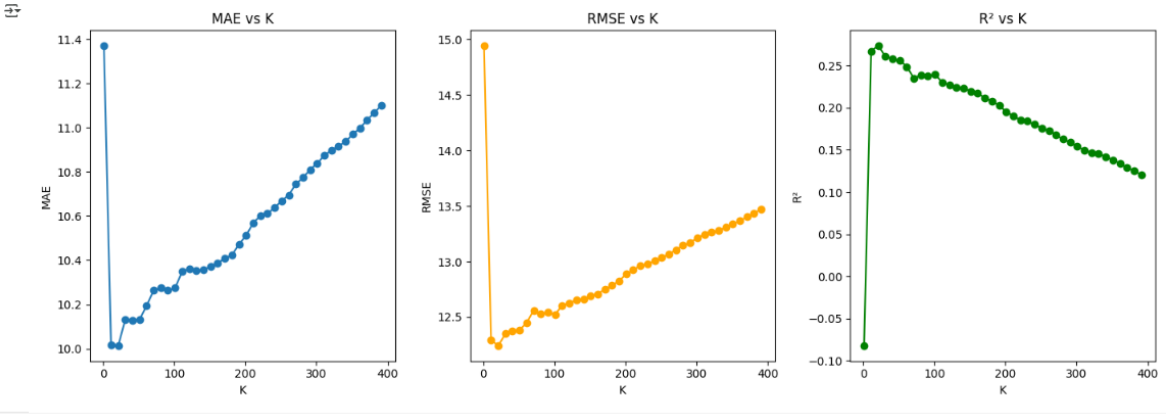
Esses dois modelos serão comparados na próxima seção por meio das métricas MAE, RMSE e R^2.

Além da predição da dançabilidade, os mesmos modelos foram aplicados para prever a variável energy_%, utilizando o mesmo conjunto de atributos preditores e a mesma divisão entre treino e teste. Essa extensão da análise permitiu avaliar se os modelos se comportam de maneira consistente ao lidar com diferentes características musicais. Para o KNN, foi realizada uma varredura de valores de k, testando múltiplas configurações entre 1 e 400. As métricas de desempenho (MAE, RMSE e R^2) foram calculadas para cada valor, e os resultados foram visualizados graficamente. Com base nessa análise, foi possível identificar

o valor de k que apresentou o melhor desempenho para o problema, permitindo uma escolha mais embasada do parâmetro.

```
for k in k_values:
    model = KNeighborsRegressor(n_neighbors=k)
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    mae_scores.append(mean_absolute_error(y_test, y_pred))
    rmse_scores.append(np.sqrt(mean_squared_error(y_test, y_pred)))
    r2_scores.append(r2_score(y_test, y_pred))
```



A partir dos gráficos acima, podemos observar que um valor ideal para K está entre 10 e 30. É possível chegar a essa conclusão ao observar que MAE e RMSE possuem valores bem baixos nesse intervalo do gráfico, além de o valor R² ser alto.

Testando os modelos para prever a energia de uma música

```
# Seleção de variáveis preditoras e variável alvo
features = ['bpm', 'danceability_%', 'valence_%', 'acousticness_%',
            'instrumentalness_%', 'speechiness_%', 'liveness_%', 'mode']

X = df[features]
y = df['energy_%']

# Divisão em treino e teste
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Regressão Linear

```
# Treinamento do modelo
reg = LinearRegression()
reg.fit(X_train, y_train)

# Previsões
y_pred_lr = reg.predict(X_test)
```

KNN

```
# Treinamento do modelo KNN
knn = KNeighborsRegressor(n_neighbors=11, weights='uniform')
knn.fit(X_train, y_train)

# Previsões
y_pred_knn = knn.predict(X_test)
```

6. Métricas

Para avaliar o desempenho dos modelos aplicados, foram utilizadas três métricas apropriadas para problemas de regressão: erro absoluto médio (MAE), raiz do erro quadrático médio (RMSE) e coeficiente de determinação (R^2). O MAE indica, em média, o desvio entre os valores previstos e os valores reais. O RMSE penaliza mais fortemente erros maiores, sendo sensível a valores discrepantes. Já o R^2 representa a proporção da variabilidade da variável alvo explicada pelo modelo, sendo que valores mais próximos de 1 indicam maior qualidade na predição.

Essas métricas foram calculadas tanto para a tarefa de predição da dançabilidade quanto para a tarefa de predição da energia das músicas, utilizando os mesmos modelos e métodos. Isso permitiu avaliar a robustez dos algoritmos aplicados frente a diferentes variáveis alvo, mantendo consistência na comparação entre as abordagens testadas.

Abaixo está o código utilizado para o cálculo das métricas:

```python
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import numpy as np

# Regressão Linear
mae_lr = mean_absolute_error(y_test, y_pred_lr)
rmse_lr = np.sqrt(mean_squared_error(y_test, y_pred_lr))
r2_lr = r2_score(y_test, y_pred_lr)

print("Regressão Linear:")
print(f"MAE: {mae_lr:.2f}")
print(f"RMSE: {rmse_lr:.2f}")
print(f"R²: {r2_lr:.2f}")

# KNN Regressor
mae_knn = mean_absolute_error(y_test, y_pred_knn)
rmse_knn = np.sqrt(mean_squared_error(y_test, y_pred_knn))
r2_knn = r2_score(y_test, y_pred_knn)

print("\nKNN Regressor:")
print(f"MAE: {mae_knn:.2f}")
print(f"RMSE: {rmse_knn:.2f}")
print(f"R²: {r2_knn:.2f}")
```

```
Regressão Linear:
MAE: 10.59
RMSE: 12.72
R²: 0.22

KNN Regressor:
MAE: 10.71
RMSE: 13.21
R²: 0.15
```

7. Comparação dos Resultados

Os modelos de Regressão Linear e KNN foram avaliados com base nas métricas MAE, RMSE e R^2. Na tarefa de prever a dançabilidade, a Regressão Linear apresentou desempenho levemente superior. O modelo alcançou um MAE de 10.59, RMSE de 12.72 e R^2 de 0.22. O

KNN obteve valores um pouco inferiores, com MAE de 10.71, RMSE de 13.21 e R^2 de 0.15. Esses resultados indicam que a Regressão Linear teve melhor desempenho na tarefa, ainda que a capacidade explicativa geral dos modelos tenha sido limitada.

Na tarefa adicional de prever a energia da música (energy_%), os resultados mostraram ganhos consideráveis para ambos os modelos, especialmente para a Regressão Linear. Nesse caso, a Regressão Linear obteve um MAE de 9.60, um RMSE de 11.76 e um R^2 de 0.45, indicando uma melhora significativa na capacidade de explicação da variabilidade da variável alvo. O KNN também teve desempenho superior ao observado na tarefa anterior, com MAE de 10.06, RMSE de 12.63 e R^2 de 0.36, mas ainda ficou atrás da Regressão Linear.

De forma geral, a Regressão Linear apresentou resultados mais estáveis e precisos nas duas tarefas. Já o KNN, mesmo após ajuste do parâmetro k, mostrou ser mais sensível à dispersão dos dados e obteve desempenho inferior. Isso reforça a ideia de que as relações entre os atributos sonoros e as variáveis alvo analisadas são em grande parte lineares, o que favorece o uso de modelos como a Regressão Linear.

8. Conclusão Preliminar

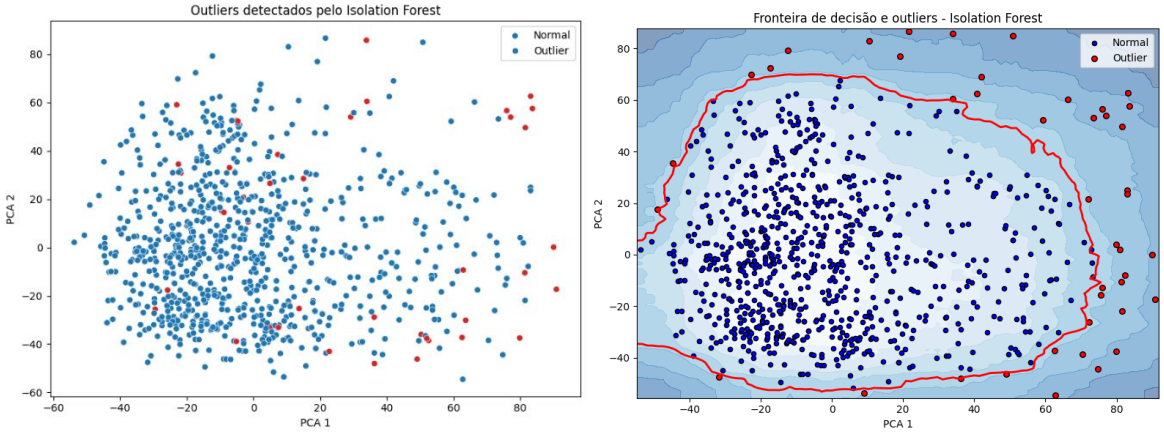
Com base nos resultados obtidos, a Regressão Linear apresentou desempenho superior ao KNN tanto na tarefa de prever a dançabilidade quanto na de prever a energia das músicas. Na predição da dançabilidade, a Regressão Linear alcançou um MAE de 10.59, um RMSE de 12.72 e um R^2 de 0.22, enquanto o KNN obteve um MAE de 10.71, RMSE de 13.21 e R^2 de 0.15. Na predição da variável energy_%, os resultados foram ainda mais favoráveis à Regressão Linear, com MAE de 9.60, RMSE de 11.76 e R^2 de 0.45, contra MAE de 10.06, RMSE de 12.63 e R^2 de 0.36 no modelo KNN.

Esses números reforçam que, para o conjunto de dados utilizado, as relações entre características sonoras como BPM, valência, acústica e modo com variáveis como dançabilidade e energia são melhor modeladas por relações lineares globais. A Regressão Linear demonstrou ser mais eficaz, estável e precisa na captura dessas relações.

Portanto, conclui-se que a Regressão Linear foi o modelo mais adequado entre os dois testados, não apenas por sua simplicidade, mas por seu desempenho superior em ambas as tarefas realizadas.

9. Análise de Outliers com Isolation Forest

Além das análises de regressão, é importante investigar a presença de outliers no conjunto de dados, pois valores atípicos podem influenciar significativamente o desempenho dos modelos. Para isso, utilizaremos o método Isolation Forest, que detecta anomalias de forma eficiente em grandes conjuntos de dados. A seguir, apresentamos a visualização dos outliers detectados por essa abordagem.



10. Conclusão sobre a Análise de Outliers

A análise de outliers realizada com o Isolation Forest permitiu identificar músicas que apresentam características sonoras significativamente diferentes do padrão observado no conjunto de dados. Esses outliers podem ser resultado de erros de registro, particularidades extremas de produção musical ou até mesmo de estilos muito distintos das demais faixas analisadas.

A presença de outliers pode impactar negativamente a performance dos modelos de regressão, como Regressão Linear e KNN, pois eles tendem a distorcer as relações estatísticas e influenciar as métricas de avaliação. Por isso, é fundamental identificar, analisar e, quando apropriado, remover ou tratar esses pontos antes de realizar a modelagem preditiva.

No contexto da predição de dançabilidade e energia em músicas populares, a remoção ou o tratamento adequado dos outliers contribui para uma análise mais robusta e resultados mais confiáveis, permitindo que os modelos aprendam padrões reais do comportamento musical, sem serem influenciados por valores extremos ou inconsistentes. Dessa forma, garantimos que as conclusões obtidas sobre os fatores que influenciam a dançabilidade e a energia das músicas sejam mais precisas e representativas do universo musical analisado.