Classificação de músicas de um usuário do *Spotify* utilizando Regressão Logística e *Support Vector Machine*

Felipe Augusto Santos Borges
Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná
Lorayne Veri
Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná
Mariana Kleina
Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná
Marcell Mariano Correa Maceno
Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná
Paola Andrea Rico Belalcazar
Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná

Resumo: O presente artigo tem como objetivo classificar as músicas de um usuário do *Spotify* e classificá-las de acordo com suas características, como o compasso, tempo, energia e detecção de plateia. Estas características são fornecidas pela API *Spotify* e foram coletadas através de bibliotecas do *software* R. Para a classificação serão usadas duas técnicas, Regressão Logística e *Support Vector Machine,* será feita a distinção entre músicas do ritmo sertanejo e outros ritmos. No estudo de caso realizado a técnica com maior acurácia foi *Support Vector Machine*, com resultado levemente superior, porém o classificador carece de estudos mais aprofundados.

Palavras-chave: *Classificação, Regressão Logística, Support Vector Machine, Spotify, Software R.*

Classification of songs from a Spotify user using Logistic Regression and Support Vector Machine

Abstract: This article aims to classify a Spotify user's songs and classify them according to their characteristics, such as tempo, tempo, energy and audience detection. These characteristics are provided by the Spotify API and were collected through R software libraries. Two techniques will be used for classification, Logistic Regression and Support Vector Machine, and a distinction will be made between country rhythm songs and other rhythms. In the case study carried out, the technique with the greatest accuracy was Support Vector Machine, with a slightly better result, but the classifier requires more in-depth studies.

Keywords: *Classification, Logistic Regression, Support Vector Machine, Spotify, R Software.*

1. Introdução

Atualmente, a importância dos dados aumentou significativamente, pois através deles têm-se conseguido obter informações relevantes relativas a diversas áreas. Classificar um paciente como doente ou não-doente a partir dos resultados de seus exames (GUESSE, 2023), identificar o tipo de consumidores dos produtos de uma empresa (LENZ, 2017), classificar um e-mail do tipo SPAM são exemplos de situações recorrentes em que a

aplicação dos algoritmos de classificação de Aprendizado de Máquina (do inglês, *Machine Learning*) é de importante valor.

Assim, com o intuito de aprimorar o conhecimento nos métodos de classificação, este artigo tem o foco na aplicação de classificação, utilizando regressão logística e *Support Vector Machine* (SVM), para a classificação de músicas de um usuário do *Spotify*, através das características das músicas, em sertanejo ou outros estilos. Serão abordados aspectos que vão desde a extração de dados por meio da API do *Spotify* até a manipulação desses dados e a execução da classificação utilizando a linguagem de programação R. Por fim, os resultados serão discutidos.

2. Revisão da Literatura

Nesta seção será apresentado o referencial teórico das ferramentas necessárias para o entendimento e execução da pesquisa.

2.1. Regressão logística

Em um modelo de Regressão Logística, a variável dependente é uma variável binária, enquanto as variáveis independentes podem ser categóricas (mediante dicotomização após a transformação) ou contínuas.
Considere um cenário em que os indivíduos podem ser classificados como bons ou maus clientes (GOUVÊA *et al.,* 2015). A variável dependente binária Y pode assumir os valores 1, quando o i-ésimo indivíduo pertence à categoria dos bons clientes, e 0, quando pertence à categoria dos maus clientes (TSAI, 2010).
Seja $X = \{1, X_1, X_2, \ldots, X_n\}$ um vetor, onde o primeiro elemento é uma constante igual a 1, e os elementos subsequentes representam as n variáveis preditoras do modelo. O modelo de Regressão Logística é um caso particular dos Modelos Lineares Generalizados (NETER et *al.*, 1996), apresentado na Equação 1.

$$\beta' X = \ln\left(\frac{p(X)}{1-p(X)}\right) \quad (1)$$

Onde, $\beta' = (\beta_1, \beta_2, \ldots, \beta_n)$ é o vetor de parâmetros associados às variáveis, e $p(X) = E(Y = 1|X)$ é a probabilidade de um indivíduo ser classificado como bom, com base no vetor X. Segundo Neter et al. (1996), esta probabilidade é expressa pela Equação 2.

$$p(X) = E(Y = 1|X) = E(Y) = \frac{e^{\beta' X}}{1 + e^{\beta' X}} \quad (2)$$

2.2. *Support Vector Machine* para Classificação

Support Vector Machine (SVM), ou no português Máquina de Vetor de Suporte, é uma técnica de aprendizado de máquina amplamente empregada com notável êxito na resolução de problemas de classificação e previsão (JOACHIMS, 2002; PONTIL e VERRI, 1998). Nesta explicação, será abordada a versão da técnica voltada para a classificação de dados, sejam eles linearmente separáveis (com margens rígidas) ou não (com margens flexíveis). A principal missão do SVM consiste em encontrar um hiperplano separador a partir de um conjunto de vetores de treinamento pertencentes a duas categorias, a fim de realizar a classificação precisa dos dados de teste (KLEINA *et al.,* 2001).

2.2.1. SVM com margens rígidas

Seja l o número de pontos de treinamento e $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\} \subseteq (X \times Y)^l$ um conjunto de treinamento, $x_i \subseteq \mathbb{R}^n$ são as entradas e $y_i \in \{-1, +1\}$ são as saídas binárias (duas classes rotuladas em $\{-1, +1\}$), $i = 1, \ldots, l$. A classificação binária é

realizada por meio da função real f: $X \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, atribuindo-se o vetor de entrada $x = (x_1,\ x_2,\ \ldots,\ x_n)$ à classe positiva se $f(x) \geq 0$, e à classe negativa, caso contrário. Tem-se que a função de decisão é definida pela Equação 3.

$$f(x) = \langle w,\ x\rangle + b \tag{3}$$

onde $w \in \mathbb{R}^n$ é o vetor de pesos e $b \in \mathbb{R}$ é o *bias* (BELTRAMI, 2009).

O hiperplano $\langle w,\ x\rangle + b = 0$, cuja dimensão é $n - 1$, atua como uma divisão no espaço de entrada X, criando dois subespaços distintos. A margem negativa é estabelecida por $\langle w,\ x\rangle + b = -1$, enquanto a margem positiva é definida por $\langle w,\ x\rangle + b = +1$. Portanto, um vetor de treinamento é classificado corretamente quando satisfaz o conjunto de inequações 4 (KLEINA *et al.,* 2001).

$$\begin{aligned} \langle w,\ x\rangle + b \geq +1, \text{ se } y_i = +1 \\ \langle w,\ x\rangle + b \leq -1, \text{ se } y_i = -1. \end{aligned} \tag{4}$$

Unificando as duas restrições em uma única expressão, obtem-se o conjunto de inequações 5.

$$y_i(\langle w,\ x_i\rangle + b) \geq 1,\ \forall i = 1,\ \ldots,\ l. \tag{5}$$

A distância entre as duas margens é expressa como $d = \frac{2}{||w||}$. O objetivo é maximizar essa distância, o que é equivalente a minimizar $\left|\left|w\right|\right|$, e para fins de simplificação posterior, isso é equivalente a minimizar $\frac{1}{2}\left|\left|w\right|\right|^2$. Portanto, o problema primal a ser resolvido é representado pela Equação 6 (MULLER *et al.*, 2001)

$$\begin{aligned} &\text{minimizar } \frac{1}{2}\left|\left|w\right|\right|^2 \\ &\text{sujeito a } y_i(\langle w,\ x_i\rangle + b) \geq 1\ \forall i = 1,\ \ldots, l \end{aligned} \tag{6}$$

onde $w \in \mathbb{R}^n$ e $b \in \mathbb{R}$ são as incógnitas do problema. A resolução do problema primal 4 pode ser desafiadora devido às restrições de desigualdade. Portanto, a abordagem é direcionada para a resolução do problema dual, que é formulado a partir da função Lagrangeana, conforme expresso na Equação 7 (MULLER *et al.*, 2001).

$$L(w,\ b,\ \alpha) = \frac{1}{2}\langle w,\ w\rangle - \sum_{i=1}^{l} \alpha_i\ [y_i(\langle w,\ x_i\rangle + b) - 1] \tag{7}$$

onde $\alpha \geq 0$ são os multiplicadores de Lagrange. A solução para o problema de otimização é alcançada pela minimização da função Lagrangeana em relação às variáveis primais e pela maximização em relação aos multiplicadores de Lagrange. Em outras palavras, isso envolve a busca pelo ponto de sela da função. Para minimizar a função Lagrangeana em relação às variáveis primárias, calculam-se as primeiras derivadas dessa função em relação a w e b e, em seguida, iguala-se a zero, resultando em $w = \sum_{i=1}^{l} y_i\,\alpha_i x_i$ e $\sum_{i=1}^{l} y_i\,\alpha_i = 0$. Substituindo esses termos na função Lagrangeana (Equação 7) e considerando um conjunto de treinamento linearmente separável $\{(x_1, y_1),\ (x_2, y_2),\ \ldots,\ (x_l, y_l)\}$, resolver o problema dual significa encontrar os multiplicadores de Lagrange α_i do problema 8 (BELTRAMI, 2009).

$$\begin{aligned} &\text{maximizar } \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i\,y_j\alpha_i\alpha_j\langle x_i,\ x_j\rangle \\ &sujeito\ a \sum_{i\,=\,1}^{l} y_i\,\alpha_i = 0 \\ &\alpha_i \geq 0,\ \forall\ i\ =\ 1,\ \ldots,\ l. \end{aligned} \tag{8}$$

Após encontrar os multiplicadores de Lagrange ótimos α^*, pode-se calcular o vetor de pesos ótimo w^* usando a equação $w^* = \sum_{i=1}^{l} y_i\,\alpha_i^* x_i$. Embora o valor de b não seja explicitamente mencionado no problema dual, o termo de polarização pode ser determinado por meio de $b^* = -\frac{1}{2}\langle w^*,\ (x_r + x_s)\rangle$, onde x_r e x_s representam quaisquer vetores de suporte das

respectivas classes, atendendo às condições $\alpha_r,\ \alpha_s > 0$ e $y_r = -1,\ y_s = 1$ (KLEINA *et al.,* 2001).
As condições de complementaridade de Karush-Kuhn-Tucker (KKT) fornecem informações valiosas sobre a estrutura do problema em questão (CRISTIANINI e SHAWE-TAYLOR, 2006). As soluções ótimas α^*, w^* e b^* devem satisfazer a Equação 9.

$$\alpha_i^*[y_i(\langle w^*,\ x_i\rangle + b^*) - 1] = 0,\ \forall\ i = 1,\ \ldots,\ l. \tag{9}$$

Isso significa que apenas os dados de entrada cuja margem é igual a 1, ou seja, aqueles que satisfazem a expressão $y_i(\langle w,\ x_i\rangle + b\ \ ) = 1$, têm um multiplicador de Lagrange correspondente $\alpha_i \neq 0$. Todos os outros α_i são igualados a zero. Os α_i não nulos, conhecidos como vetores de suporte, são os únicos usados no cálculo de w^*. Conforme afirmado por Beltrami (2009), esses vetores de suporte contêm todas as informações essenciais do conjunto de treinamento necessárias para classificar os dados de teste.

2.2.2. SVM com margens flexíveis

A técnica SVM com margens flexíveis é aplicada a dados que não são linearmente separáveis (ou seja, violam a condição $y_i(\langle w,\ x_i\rangle + b) \geq 1,\ \forall\ i = 1,\ \ldots,\ l$) (KLEINA *et al.,* 2001). Nestes casos, é impossível criar um hiperplano separador sem erros de classificação, mas é viável encontrar um que minimize a probabilidade de erro próximo às amostras de treinamento. Conforme discutido por Beltrami (2009) e Muller *et al.* (2001), variáveis de folga, denotadas por $\xi_i \geq 0$, são introduzidas no problema, associadas a cada vetor de treinamento. Um vetor é considerado corretamente separado quando $\xi_i = 0$. Com a inclusão das variáveis de folga, um vetor de treinamento é corretamente classificado quando a inequação 10 é satisfeita.

$$y_i(\langle w,\ x_i\rangle + b) \geq 1 - \xi_i,\ \forall\ i = 1,\ \ldots,\ l \tag{10}$$

O problema primal a ser solucionado é dado pela Equação 11 (BELTRAMI, 2009).

$$\begin{gathered}\text{minimizar } \frac{1}{2}\left|\left|w\right|\right|^2 + C\sum_{i=1}^{l}\xi_i \\ \text{sujeito a } y_i(\langle w,\ x_i\rangle + b) \geq 1 - \xi_i,\ \forall\ i = 1,\ \ldots,\ l \\ \xi_i \geq 0,\ \forall\ i = 1,\ \ldots,\ l\end{gathered} \tag{11}$$

onde $w \in \mathbb{R}^n$ e $b \in \mathbb{R}$ são as incógnitas do problema e C representa a constante de regularização, que influencia os termos da função de minimização.
O problema dual é expresso na Equação 12.

$$\begin{gathered}\text{maximixar } \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i\, y_j \alpha_i \alpha_j \langle x_i,\ x_j\rangle \\ \text{sujeito a } \sum_{i-1}^{l} y_i\,\alpha_i = 0 \\ 0 \leq \alpha_i \leq C,\ \forall\ i = 1,\ \ldots,\ l\end{gathered} \tag{12}$$

As equações 13 e 14 estabelecem as condições de KKT em relação ao problema dual 12.

$$\alpha_i[y_i(\langle w,\ x_i\rangle + b) - 1 + \xi_i] = 0,\ \forall\ i = 1,\ \ldots,\ l \tag{13}$$

$$\xi_i(\alpha_i - C) = 0,\ \forall\ i = 1,\ \ldots,\ l \tag{14}$$

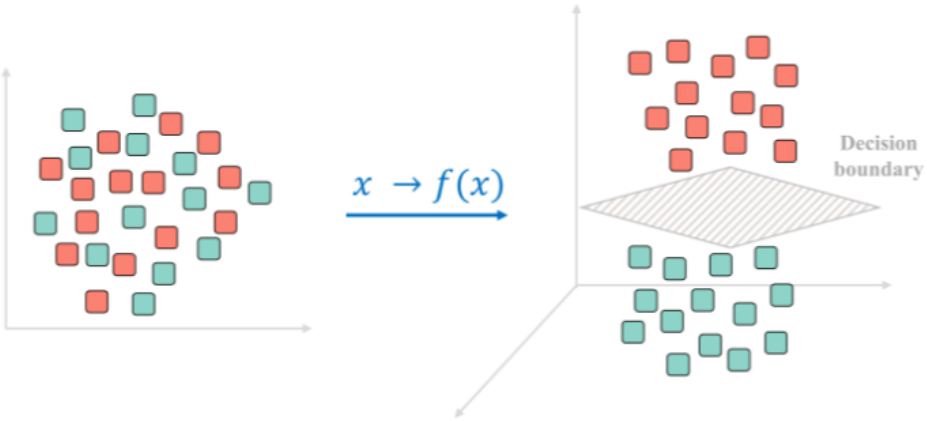
Os pontos de entrada para os quais $\alpha_i \geq 0$ são conhecidos como vetores de suporte (BELTRAMI, 2009). Se $0 \leq \alpha_i \leq C$ (referidos como vetores de suporte não restritos), então, devido a $\xi_i(\alpha_i - C) = 0$, temos $\xi_i = 0$, indicando que esses pontos residem na margem de sua classe. No caso em que $\alpha_i = C$ (conhecidos como vetores de suporte restritos), é possível que $\xi_i > 1$ (indicando erros de classificação) ou $0 < \xi_i \leq 1$ (denotando classificação correta, mas dentro das margens) (KLEINA *et al.,* 2001). Com exceção dos

vetores de suporte, independentemente de serem restritos ou não, os demais pontos (onde $\alpha_i = 0$) não têm influência na determinação do hiperplano separador ótimo. As variáveis w^*, b^* e a superfície de decisão são determinados da mesma maneira que no caso de margens rígidas.

2.2.3. SVM não linear

Quando os dados não podem ser linearmente separados, é necessário expandir a dimensão do espaço, possibilitando assim uma separação linear. Isso é alcançado por meio de um mapeamento não linear dos dados de entrada X para um espaço de alta dimensão conhecido como espaço de características, conforme ilustrado na Figura 1 (KLEINA *et al.,* 2001).

Figura 1 – Fronteira linear no espaço das características gerada pelo mapeamento não linear do espaço de entrada



Fonte: Sheykhmousa *et al.* (2020)

Uma abordagem comum para realizar esse procedimento envolve a transformação dos dados, representando-os como $x = (x_1, x_2, \ldots, x_n) \mapsto \phi(x) = \big(\phi_1(x), \ldots, \phi_N(x)\big)$, onde $F = \{\phi(x) \mid x \in X\}$. Isso é alcançado substituindo as entradas x pela função $\phi(x)$ na função objetivo do problema 10, onde $\phi: \mathbb{R}^n \rightarrow \mathbb{R}^N$, com N sendo muito maior que n (CHAVES, 2006). No entanto, calcular o produto interno $\langle \phi(x_i), \phi(x_j) \rangle$ diretamente no espaço de características pode se tornar computacionalmente inviável devido à sua alta dimensionalidade. Para evitar esse custo computacional, o mapeamento pode ser realizado implicitamente por meio de funções kernel, $K(x, z) = \langle \phi(x), \phi(z) \rangle$, que dependem apenas das variáveis de entrada (CHAVES, 2006). Aqui, ϕ representa o mapeamento de X para o espaço de características (produto interno). As funções kernel comumente utilizadas incluem a linear, polinomial, sigmoidal e de base radial, conforme mencionado por Beltrami (2009) e Muller *et al.* (2001).

Assim, o problema 12 se torna o problema 15.

$$\begin{gathered} \text{maximizar} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i\, y_j \alpha_i \alpha_j K(x_i, x_j) \\ \text{sujeito a} \sum_{i=1}^{l} y_i\, \alpha_i = 0 \\ 0 \leq \alpha_i \leq C, \forall\, i = 1, \ldots, l \end{gathered} \tag{15}$$

onde α_i são os multiplicadores de Lagrange.

Diversos resultados surgirão ao se ajustar valores diferentes para a constante de regularização C e ao empregar distintas funções kernel (com seus respectivos parâmetros), conforme destacado por Beltrami (2009).

2.3. API *Spotify*

O *Spotify*, um serviço de *streaming* de música, foi desenvolvido em 2006 e lançado oficialmente em 7 de outubro de 2008 pela *startup* sueca *Spotify* AB. Atualmente, está disponível em 65 regiões ao redor do mundo, oferecendo acesso a um vasto catálogo de músicas que dispensa a necessidade de aquisição de formatos físicos, como discos e CDs, ou o *download* de arquivos (MATIAS, 2018).
No Brasil, o lançamento do *Spotify* ocorreu em maio de 2014, seis anos após o lançamento global do aplicativo (ARAÚJO; OLIVEIRA, 2014).
O *Spotify* lidera o mercado global de assinaturas de *streaming* de música, com 31% dos assinantes, enquanto a *Apple Music* ocupa o segundo lugar, com aproximadamente 15% do mercado (LISBOA, 2022). Assim, o *Spotify* mantém sua posição como a maior empresa de *streaming* de música do mundo, apesar do crescimento de concorrentes.
"API", que significa "Interface de Programação de Aplicações", é um conjunto de padrões e rotinas de programação que viabiliza o acesso a uma plataforma *web* ou a um aplicativo de *software*. No contexto deste artigo, os dados de análise são obtidos por meio da API gratuita disponibilizada pelo *Spotify*.
Por meio dessa API, os usuários têm a capacidade de desenvolver seus próprios aplicativos que oferecem serviços adicionais relacionados à música, permitindo o acesso a diversos recursos e suporte para autorização do usuário. Para obter informações detalhadas sobre as funcionalidades e recursos disponíveis, é recomendável consultar a documentação da API da *Web* do *Spotify*, conforme sugerido por Lamere (2014). Em resumo, a API do *Spotify* é uma ferramenta que possibilita a criação de novas experiências e serviços relacionados à música com base nos dados e funcionalidades da plataforma.

3. Metodologia

O conjunto de dados selecionado para o desenvolvimento deste projeto consiste no registro das músicas ouvidas por um usuário do *Spotify* durante o período de 22 de agosto de 2021 a 22 de agosto de 2022, bem como uma base de dados construída a partir de informações obtidas por meio da *Web* API do *Spotify*, ambos relacionados à plataforma *Spotify*. Para a manipulação e análise desses dados, o *software* R foi empregado. A seguir, cada uma das etapas do projeto será explicada em detalhes.

3.1. Conjunto de dados utilizados

O *Spotify*, quando solicitado, disponibiliza aos usuários o histórico das músicas ouvidas no período de um ano a partir da data solicitada pelo perfil. Para este estudo, obtive-se acesso ao histórico de um usuário que consentiu com o uso de seus dados, e esse conjunto de dados foi a principal fonte de informação utilizada. O arquivo original continha 10.011 observações, com 4 variáveis - nome do artista da música, nome da música, data e horário em que a música foi ouvida, e duração da reprodução da música. Contudo, para o propósito desta análise, a granularidade original não era necessária. Assim, os dados foram consolidados, focando apenas no nome do artista da música, nome da música e tempo de reprodução, resultando em um conjunto de 2.442 entradas. A Tabela 1 oferece uma descrição das variáveis usadas neste estudo.

Tabela 1 – Variáveis da base de dados do perfil

Nome da variável	Significado	Tipo
artistName	Nome do artista de cada música	*String*
trackName	Nome das músicas	*String*
msPlayed	Quantidade de milésimos de segundos que	Número real maior ou

	uma faixa foi reproduzida	igual a zero

Fonte: Autoria própria com base nos dados de *Spotify* (2023)

Para obter a base de estudo, foi realizada a categorização das músicas em “sertanejo” e “outros” uma vez que o estilo predominante no perfil do usuário é sertanejo.

3.2. Variáveis para a classificação

Com o objetivo de efetuar a categorização das músicas, foram selecionadas as variáveis de classificação com base nas características intrínsecas das músicas. Para isso, utilizou-se a função *getFeatures* do pacote *RSpotify* no *software* R, que extrai informações da base de dados da *Web* API. A extração resultou em um conjunto de 13 variáveis para o conjunto de dados mencionado anteriormente, contendo propriedades que permitem identificar semelhanças entre as músicas. Dentre essas variáveis, foram escolhidas apenas 9, as quais são as mais relevantes para o estudo. Detalhes dessas variáveis estão disponíveis na Tabela 2.

Tabela 2 – Características das músicas

Nome da variável	Significado	Tipo
danceability	Quanto a música é dançável	Números real entre 0 e 1, sendo 0 o valor menos dançável e 1 o mais dançável.
energy	Nível de atividade da música	Números real entre 0 e 1, sendo 0 o valor com menor energia e 1 com maior energia.
tempo	Compasso da música	Número real
valence	Detecta a positividade, sentimento de felicidade da música	Números real entre 0 e 1, sendo 0 o valor com menor positividade e 1 o valor com maior positividade.
acousticness	Indica o quanto a música é acústica	Números real entre 0 e 1, sendo 0 o valor com menor acústica e 1 o com maior acústica.
liveness	Indica a presença de plateia	Números real entre 0 e 1, sendo 0 o valor com menor plateia e 1 o com maior plateia.
loudness	Indica o volume da música em decibéis	Número real.
speechines	Acorde predominante da música	Número real
estilo	Indica se a música é do estilo sertanejo ou não	Variável binária que indica se a música é do estilo sertanejo, 1, ou de outro estilo, 0.

Fonte: Autoria própria com base nos dados de Spotify (2023)

Algumas das músicas na base de dados do usuário não continham as informações obtidas por meio da função *getFeatures* e, consequentemente, essas músicas foram removidas. Como resultado, a base de dados foi reduzida de 2.442 músicas para 2.003.

Após a coleta dos dados, foi realizada a padronização deles, via método de *scale*, que consiste em subtrair os valores de sua média e dividir pelo desvio padrão, e posteriormente foram executados os métodos para classificação.

3.3. Parâmetros da Regressão Logística

A fim de utilizar a Regressão Logística, empregou-se a função *glm* do *software* R, com a família binomial e a função *logit* como função de ligação. Para efetuar as previsões nos dados do conjunto de teste, após o treinamento do modelo, utilizou-se a função *predict()* do mesmo pacote, conforme descrito por Bergmeir (2021).

Inicialmente foram utilizadas todas as 8 variáveis disponíveis - *danceability*, *energy*, *loudness*, *speechiness*, *acousticness*, *liveness*, *valence e* tempo para descrever o estilo, a

variável resposta. Em seguida, as variáveis com maior significância - *loudness*, *acousticness e liveness*, sinalizadas na função *summary* do *software* R; e, por fim, as variáveis com maior correlação com a variável dependente, estilo - *energy*, *loudness*, *acousticness e liveness*.

3.4. Parâmetros do SVM

No método SVM, empregou-se a função *ksvm* do *software* R. Para efetuar as previsões nos dados separamos 70% da base para o conjunto de treinamento (1.402 músicas) e o restante para o conjunto de teste (600 músicas). Para a previsão, utilizou-se a função *predict* do mesmo pacote, conforme mencionado por Meyer *et al.* (2021).
Foram utilizadas todas as 8 variáveis disponíveis. A escolha do parâmetro C foi igual a 1. Além disso, foram testadas as funções kernel do tipo linear, polinomial e gaussiana.

3.5. Avaliação dos Modelos

Os resultados de cada um dos modelos treinados (Regressão Logística e SVM) serão submetidos a análise por meio de uma matriz de confusão (DAVIS e GOADRICH, 2006). Esta abordagem de avaliação permite determinar a quantidade de dados do conjunto de teste que foram classificados corretamente, ou seja, aquelas músicas que eram sertanejo e foram identificados como sertanejo (verdadeiros positivos - VP), bem como os que não eram sertanejo e foram corretamente identificados como outras (verdadeiros negativos - VN). Além disso, a matriz de confusão fornece informações sobre a quantidade de dados classificados de forma incorreta, isto é, casos em que as músicas sertanejas foram erroneamente identificadas como outras (falsos positivos - FP) e casos em que as músicas que não são sertanejo foram incorretamente classificados como sertanejo (falsos negativos - FN).
Com base na matriz de confusão, é possível calcular a acurácia de cada um dos modelos, conforme definido na Equação 16.

$$\text{Acurácia} = \frac{VN + VP}{VN + VP + FN + FP} \times 100\% \quad (16)$$

A Equação 16 representa uma métrica que avalia o desempenho geral do modelo, sendo evidente que um valor de acurácia mais elevado indica um melhor desempenho do modelo.

4. Resultados

Primeiramente, os resultados obtidos pela abordagem da regressão logística são apresentados. Conforme apresentado na metodologia do trabalho, foi aplicado o modelo para 8 variáveis independentes para classificar a variável estilo, cuja matriz de confusão, referente aos dados de teste, é apresentada na Figura 2.

Figura 2 – Matriz de confusão com os resultados do modelo de regressão logística com todas as variáveis para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	240	97
	1	144	120

Fonte: Autoria própria (2023)

Conforme ilustrado na Figura 2, o modelo acertou em 240 casos ao classificar as músicas como não sertanejo e acertou em 120 casos ao classificar as músicas como sertanejo.

Entretanto, cometeu erros em 144 ocasiões, ao afirmar que se tratava de outro estilo quando na verdade era sertanejo, e em 97 situações ao identificar as músicas como sertanejo, mas a classificação real era outros. A acurácia, calculada conforme a Equação 16, atingiu aproximadamente 59,9%.
Com o intuito de buscar uma maior acurácia, foi aplicado o mesmo modelo para as variáveis com maior nível de significância - *loudness*, *acousticness* e *liveness*. Sua matriz de confusão é apresentada na Figura 3.

Figura 3 – Matriz de confusão com os resultados do modelo de regressão logística com as variáveis de maior significância para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	247	106
	1	122	126

Fonte: Autoria própria (2023)

Com esta escolha de variáveis o modelo foi capaz de acertar em aproximadamente 62%.
Finalmente, utilizando as variáveis com maior correlação - *energy*, *loudness*, *acousticness* e *liveness*, obteve-se a matriz de confusão apresentada na Figura 4.

Figura 4 – Matriz de confusão com os resultados do modelo de regressão logística com as variáveis de maior correlação para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	272	68
	1	138	123

Fonte: Autoria própria (2023)

A acurácia atingiu aproximadamente 65,7%.
Ao aplicar o modelo SVM para as 8 variáveis disponíveis e para a variável resposta com $C = 1$ e *kernel* linear os resultados da Figura 5 foram obtidos, por meio da matriz de confusão, também para o conjunto de dados de teste.

Figura 5 – Matriz de confusão com os resultados do modelo SVM com *kernel* linear para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	261	82
	1	132	125

Fonte: Autoria própria (2023)

O modelo SVM com *kernel* linear foi capaz de acertar em aproximadamente 64,3%
Novamente, com o intuito de obter uma maior acurácia, o modelo foi aplicado com um diferente *kernel*, o polinomial, cuja matriz de confusão está na Figura 6.

Figura 6 – Matriz de confusão com os resultados do modelo SVM com *kernel* polinomial para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	277	66
	1	141	116

Fonte: Autoria própria (2023)

Com esta configuração, o modelo obteve uma acurácia de 65,5%.

Finalmente, o modelo SVM foi aplicado com *kernel* gaussiano e os resultados da Figura 7 foram obtidos, por meio da matriz de confusão, também para o conjunto de dados de teste.

Figura 7 – Matriz de confusão com os resultados do modelo SVM com *kernel* gaussiano para os dados de teste

		Valor classificado	
		0	**1**
Valor real	**0**	261	82
	1	132	125

Fonte: Autoria própria (2023)

Assim, o modelo SVM com *kernel* gaussiano foi capaz de acertar em aproximadamente 66,5%

Ao comparar as abordagens utilizadas, observa-se que o modelo treinado com SVM supera o modelo de regressão logística. No entanto, a regressão logística apresentou resultados levemente inferiores, e por ser uma técnica mais simples em comparação com SVM, pode ser empregada para este estudo de caso.

4. Conclusões

No contexto atual, a enorme quantidade de dados que é gerada por meio de plataformas de *streaming* e pesquisas na *internet* torna essencial a capacidade de selecionar músicas relevantes de acordo com o perfil do usuário. Nesse sentido, o objetivo deste artigo consiste em propor um algoritmo de classificação com base nas músicas já ouvidas pelo usuário na plataforma de *streaming* do *Spotify*.

Este estudo apresenta um algoritmo de classificação de músicas entre sertanejo e outros estilos que utiliza o histórico de músicas ouvidas pelo usuário ao longo de um ano no *Spotify*, bem como as características das músicas coletadas. Após a coleta de dados e a definição de parâmetros, os dados foram padronizados e dois métodos de classificação foram aplicados: regressão logística e SVM. Os parâmetros de ambas as técnicas foram escolhidos de forma a maximizar a acurácia no conjunto de teste.

Os resultados obtidos revelaram uma acurácia de aproximadamente 65,7% e 66,5% para o conjunto de teste utilizando, respectivamente, a regressão logística com as variáveis de maior correlação e o SVM com *kernel* gaussiano. Nota-se que os modelos com maior flexibilidade apresentaram uma acurácia superior. Os resultados encontrados não foram satisfatórios, pois esperava-se uma classificação com acurácia superior.

Uma sugestão para trabalhos futuros seria a utilização de outras técnicas de classificação, como Floresta Aleatória, outras variáveis que sejam mais discriminatórias ou um maior conjunto de dados.

É importante ressaltar que, embora esta pesquisa tenha utilizado apenas a amostra de um único usuário do aplicativo, a metodologia desenvolvida pode ser aplicada a qualquer outra amostra de usuários, ampliando seu potencial de uso.

Referências

ARAÚJO, L.; OLIVEIRA, C. **Música em fluxo: experiências de consumo musical em serviços de streaming.** Temática, [s.l.], v. 10, n. 10, p. 122-137, 2014. Mensal. Disponível em <http://www.periodicos.ufpb.br> Acesso em: 22 ago. 2022.

BELTRAMI, M. **Precificação de opções sobre ações por modelos de Support Vector Regression.** Curitiba: UFPR, 2009. 125 p. Dissertação (Mestrado) – Pós-graduação em Métodos Numéricos em Engenharia, Universidade Federal do Paraná.

BERGMEIR, C. **Neural Networks using the Stuttgart Neural Network Simulator (SNNS).** Package 'RSNNS', 2021. Disponível em: < https://cran.r-project.org/web/packages/RSNNS/RSNNS.pdf > Acesso em: 19 setembro 2023.

CHAVES, A. **Extração de regras Fuzzy para máquinas de vetores suporte (SVM) para classificação em múltiplas classes.** Rio de Janeiro, PUC-Rio, 2006. 225 p. Tese (Doutorado) – Pós-graduação em Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

CRISTIANINI, N.; SHAWE-TAYLOR, J. **An Introduction to Support Vector Machines and other kernel-based learning methods.** Reino Unido: Cambridge University Press, 10 ed., 2006.

DAVIS, J.; GOADRICH, M. **The relationship between Precision-Recall and ROC curves.** New York: ACM Press, 2006.

FRANZONI, T. **Motivação de consumo de música por streaming dos usuários do Spotify.** 2017.

GOUVÊA, M. A., GONÇALVES, E. B., MANTOVANI, D. M. N. 2015. **Análise de Risco de Crédito com Aplicação de Regressão Logística e Redes Neurais.** Contabilidade Vista Revista, 24(4), 96–123.

GUESSE, L. **Tipos de problemas de Machine Learning.** 2023. Disponível em: < https://medium.com/@lucasguesse/tipos-de-problemas-de-machine-learning-1517616c66fd >. Acesso em: 20 set. 2023.

JOACHIMS, T. **Learning to classify texts using support vector machines: methods, theory and algorithms.** Kluwer Academic. Publishers, 2002.

KLEINA, M., KACZOROWSKI, B., MARQUES, M., VERI, L. **Classificação de fraudes em pagamentos online por meio de técnicas de machine learning.** In: ENEGEP, 42, 2022, Foz do Iguaçu.

LAMERE, P. spotipy. 2014. Disponível em: <**https://spotipy.readthedocs.io**>. Acesso em: 20 ago. 2022.

LENZ, G. **Uma introdução Didática aos Algoritmos de Classificação de Machine**

Learning. 2017. Disponível em: < https://medium.com/drafter-ai/uma-introdu%C3%A7%C3%A3o-did%C3%A1tica-aos-algoritmos-de-classifica%C3%A7%C3%A3o-de-machine-learning-460be2d73395 >. Acesso em: 20 set. 2023.

LISBOA, A., 2022. Editado por Douglas Ciriaco. **Os apps de música por streaming mais usados no mundo.** Disponível em <https://canaltech.com.br/apps/os-apps-de-musica-por-streaming-mais-usados-no-mundo-207147/>. Acesso em: 22 ago. 2022.

MATIAS, A. **Spotify, 10 anos: como o serviço de streaming mudou a música.** 2018 Disponível em <https://reverb.com.br/artigo/spotify-10-anos-como-o-servico-destreaming-mudou-a-musica>. Acesso em: 20 ago. 2022.

MEYER, D; et al. **Misc Functions of the Department of Statistics, Probability Theory Group.** Package 'e1071', 2021. Acesso em: 19 setembro 2023.

MULLER, K.; et al. **An introduction to kernel-based learning algorithms.** IEEE Transactions on Neural Networks, v. 12, n. 2, p.181-201, março 2001.

NETER, J., KUTNER, M. H., NACHTSHEIN, C. J., & WASSERMAN, W. 1996. **Applied linear statistical models.** first edn. Irwin.

PONTIL, M.; VERRI, A. **Support vector machines for 3-D object recognition.** IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 20, n. 6, p.637-646, 1998.

SHEYKHMOUSA, M., MAHDIANPARI, M., GHANBARI, H., MOHAMMADIMANESH, F., GHAMISI, P., HOMAYOUNI, S. **Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review.** IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, v. 13, p. 6308-6325, 2020.

TSAI, B. 2010. **Comparison of Binary Logit Model and Multinomial Logit Model in Predicting Corporate Failure.** Review of Economics Finance, 1994, 99–111.