

Classificação de nódulos mamários com máquina de vetores de suporte

Fernanda Costa e Silva* Danton Diego Ferreira**

* Departamento de Automática, Universidade Federal de Lavras, MG,
(e-mail: fernanda.silva10@estudante.ufla.br)

** Departamento de Automática, Universidade Federal de Lavras, MG,
(e-mail: danton@ufla.br)

Abstract: Breast cancer is the most common type among women and its late diagnosis is one of the causes of the high mortality rate associated with this disease. To aid in the interpretation of tests for the correct diagnosis, classification methods have been used to classify nodules as malignant or benign. This work portrays the use of the Support Vector Machine (SVM), with different Kernel functions, to classify the data set of exams results for breast cancer detection, performing the separation of samples into benign and malignant. The implementation was made using the Python language and several open source libraries available for data analysis. A public database available in the *UCI Machine Learning Repository* was used. The results obtained prove the effectiveness of SVM for binary classification problems and demonstrate the importance of choosing its parameters properly during its project.

Resumo: O câncer de mama é o tipo mais comum entre mulheres e seu diagnóstico tardio é uma das causas do alto índice de mortalidade associado a esta doença. Para ajudar na interpretação dos exames para o diagnóstico correto, métodos de classificação têm sido utilizados para classificar nódulos em malignos ou benignos. Este trabalho retrata a utilização da Máquina de Vetores de Suporte (SVM), com diferentes funções Kernel, para classificação do conjunto de dados de resultados de exames para detecção de câncer de mama, realizando a separação das amostras em benignas e malignas, com implementação em Python utilizando diversas bibliotecas de código aberto disponíveis para análise de dados. Foi utilizada uma base de dados pública disponível no *UCI Machine Learning Repository*. Os resultados obtidos comprovam a eficácia da SVM para problemas de classificação binária e demonstram a importância da escolha adequada de seus parâmetros durante seu projeto.

Keywords: Classification, Pattern recognition, Support vector machine, Python

Palavras-chaves: Classificação; Reconhecimento de padrões; Máquina de vetores de suporte; Python

1. INTRODUÇÃO

O câncer de mama é o tipo de câncer mais comum entre mulheres e o segundo mais comum no mundo. Existem vários tipos de câncer de mama e, quando diagnosticado precocemente, apresentam uma alta porcentagem de cura. A doença consiste no crescimento desordenado de células do tecido da mama, formando nódulos que podem ser malignos ou benignos. No Brasil, a doença é comumente diagnosticada em estados mais avançados, o que contribui para que as taxas de mortalidade por câncer de mama continuem elevadas Holsbach et al. (2014). Apesar dos avanços no tratamento da doença, a detecção precoce é muito importante para que o tratamento seja bem sucedido.

A identificação do câncer depende da interpretação médica dos exames, que podem ser de vários tipos: exame clínico da mama, mamografia, análise de tecido da mama, entre outros, podendo ser necessária a realização de mais de um exame para detecção do câncer. A mamografia é um exame que registra imagens da mama e permite identificar

precocemente a presença de nódulos mamários que podem ser tumores malignos. A mamografia é o exame mais utilizado para detecção de câncer de mama, de acordo com o mostrado no trabalho de Borchardt et al. (2013). Os exames de imagem produzem resultados com mais informações, porém a interpretação de seus dados é mais complexa.

Diante disto, métodos de classificação têm sido utilizados para auxiliar profissionais da área da saúde no processamento e interpretação dessas informações para chegar a uma conclusão a respeito dos dados analisados, classificando os nódulos como maligno ou benigno. Alguns dos métodos de classificação mais difundidos na literatura utilizam as redes neurais artificiais, que são modelos computacionais inspirados no sistema nervoso humano, e podem ser usadas em aplicações de aprendizado de máquina e reconhecimento de padrões.

Alguns trabalhos analisam dados referentes ao câncer de mama e o desempenho de diferentes algoritmos de classificação, como no trabalho de Agarap (2017), que é feita

uma comparação entre alguns algoritmos de aprendizado de máquina, analisando a acurácia de cada um na tarefa de classificação de amostras de resultados de exame para detecção de câncer de mama. O banco de dados utilizado foi o *Breast Cancer Wisconsin (Diagnostic) Data Set*, disponibilizado por Wolberg et al. (1992). No trabalho de Borges (2015) também é utilizado o mesmo conjunto de dados para comparar o desempenho dos algoritmos de rede Bayesiana e do algoritmo J48.

Atualmente, a Máquina de Vetores Suporte (SVM do inglês *Support Vectors Machine*), tem uma grande aplicação em problemas de classificação e reconhecimento de padrões, e pode apresentar melhor desempenho, quando comparada a outros algoritmos, desde que seus parâmetros sejam escolhidos corretamente. Esse algoritmo possui aplicações em diversas áreas para análise de dados como biologia, medicina, finanças, comércio, indústria, entre outros. Essa variedade de aplicações se deve à alta capacidade de generalização do modelo.

Dada a importância da detecção precoce do câncer de mama para o sucesso no tratamento e o potencial da utilização da SVM para classificação, este artigo busca analisar a eficácia de um classificador projetado utilizando SVM para classificação de nódulos mamários, sendo utilizada uma base de dados pública disponível no *UCI Machine Learning Repository* (Wolberg et al. (1992)).

Além disso é utilizada a linguagem *Python* para implementação dos códigos, tratamento dos dados e análise de resultados, por ser uma linguagem *open-source*. O *Python* tem grande aplicação em problemas de análise de dados devido às suas bibliotecas de código aberto, que estão em constante atualização por parte da comunidade desenvolvedora.

Este artigo está dividido em cinco seções, conforme descrito a seguir: esta seção introdutória apresenta o tema e proposta do trabalho. Na seção 2 é apresentado o fundamento teórico e alguns conceitos importantes para compreensão do trabalho desenvolvido. Os métodos e materiais são apresentados na seção 3. Os resultados e a discussão são apresentados na seção 4 e a seção 5 apresenta as conclusões deste trabalho.

2. REFERENCIAL TEÓRICO

2.1 SVM para reconhecimento de padrões

A Máquina de Vetores Suporte (SVM do inglês *Support Vectors Machine*) é um algoritmo de aprendizado de máquina supervisionado, ou seja, a informação das classes de cada amostra é conhecida e apresentada ao modelo durante o treinamento. A SVM foi apresentada no trabalho de Boser et al. (1992). Esse algoritmo foi desenvolvido com objetivo de resolver problemas de classificação entre duas classes distintas, como uma alternativa aos métodos convencionais, como as redes neurais artificiais.

Basicamente, este algoritmo de aprendizado supervisionado é formado por quatro conceitos: hiperplanos, vetores de suporte, margem e funções Kernel. De maneira simples, o funcionamento da SVM é baseado na busca de um hiperplano ótimo de separação entre classes em um espaço multidimensional.

O conjunto de características extraídas dos dados é a entrada para o classificador. Os vetores suporte se posicionam no limite de suas classes e servem de base para o cálculo do hiperplano ótimo, que fará a separação entre as classes. Quando os dados são linearmente separáveis, a SVM busca maximizar a distância entre os hiperplanos que separam as classes do problema (Boser et al. (1992)) e minimizar o erro de classificação.

Entretanto, em casos em que os dados não são linearmente separáveis, o uso de um método puramente linear pode levar à classificação errada, pois não é possível estabelecer hiperplanos que separem completamente as amostras de cada classe. Para contornar esse problema, é definido o parâmetro de suavização da margem (*soft-margin*), que permite ao algoritmo realizar a classificação de algumas amostras na classe diferente daquela à que ela pertence, mas ainda assim o resultado da classificação feita pelo algoritmo vai permanecer dentro dos padrões aceitáveis para o usuário.

Porém, existem problemas com dados não são linearmente separáveis, nos quais mesmo com a aplicação da *soft-margin*, o algoritmo ainda não consegue definir o hiperplano ótimo e separar as classes da maneira adequada. Para esta situação, a solução encontrada consiste em mapear os dados não linearmente separáveis para uma dimensão maior, onde estes dados se tornam linearmente separáveis.

Para realizar o mapeamento dos dados para um espaço de dimensões maior, são utilizadas as funções Kernel, que realizam a mudança de dimensão dos dados do espaço de entrada para o espaço de características (em uma dimensão mais alta).

Existem diversas funções Kernel que podem ser utilizadas, mas de acordo com as características dos dados e dos parâmetros escolhidos podem apresentar desempenho diferentes. As funções polinomial, gaussiana, radial e tangente hiperbólica são as mais utilizadas e podem ser encontradas em diversas aplicações em problemas das mais diferentes áreas.

A formulação matemática para a SVM pode ser resumida como se segue. A minimização da distância entre dois hiperplanos leva a um problema dual de otimização quadrática que visa maximizar (1), sujeita às restrições representadas por (2) e (3):

$$\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j x_i \cdot x_j \quad (1)$$

$$\sum_n^{i=1} y_i \alpha_i = 0, \quad (2)$$

$$0 \leq \alpha_i \leq C, i = 1, \dots, n \quad (3)$$

Onde α_i representam os multiplicadores de Lagrange, C é o critério de suavização das margens (*soft-margin*) e x_i e y_i representam o conjunto de dados de treinamento.

O seguinte par (w^*, b^*) (4) e (5) define o hiperplano ótimo de separação entre as classes:

$$w^* = \sum_{i=1}^n \alpha_i^* y_i x_i \quad (4)$$

$$b^* = \frac{1}{2} [\max(i | y_i = -1)(w^* \cdot x_i) + \min(i | y_i = +1)(w^* \cdot x_i)] \quad (5)$$

Após a definição dos parâmetros na etapa de treinamento, os testes são baseados na classificação das amostras de acordo com (6) e (7).

$$g(x) = \text{sgn}(f(x)) \quad (6)$$

$$g(x) = \text{sgn}\left(\sum_{x_i} \alpha_i^* y_i x_i \cdot x + b^*\right) \quad (7)$$

em que o resultado de $g(x)$ pode ser +1 se o valor de $f(x)$ for > 0 , ou -1 se o valor de $f(x)$ for < 0 .

2.2 Funções Kernel

Quando os dados são não linearmente separáveis, nenhum hiperplano pode separar os dados de cada classe completamente, mesmo minimizando a margem entre as classes (utilizando a *soft-margin*). A solução para este tipo de problema é a utilização da SVM não linear, que utiliza uma função Kernel para adicionar uma dimensão aos dados, projetando-os para a alta dimensão, no espaço de características. Quando a função é escolhida corretamente, os dados se tornam linearmente separáveis na alta dimensão, podendo ser separados pelo hiperplano ótimo.

Uma função Kernel é definida como uma função que recebe como dados de entrada dois pontos x e y , no espaço original, e calcula seu produto interno no espaço de características Haykin (1999).

Não existe uma regra para definição da melhor função *Kernel* para um problema. Para realização deste trabalho foram utilizadas as funções Kernel linear, polinomial e a função de base radial (RBF). As funções Kernel utilizadas são descritas a seguir.

Um Kernel linear pode ser usado como o produto entre dois vetores, que é a soma da multiplicação de cada par de valores de entrada. A equação linear é mostrada em (8).

$$K(x, x_i) = \sum (x * x_i) \quad (8)$$

A função polinomial é a forma mais generalizada do kernel linear. Ela pode ser usada para separar dados não linearmente separáveis. Na equação polinomial (9) o parâmetro d representa o grau do polinômio.

$$K(x, x_i) = 1 + \sum (x * x_i)^d \quad (9)$$

A função de base radial (RBF) é uma função Kernel popularmente utilizada para classificação com SVM e é representada em (10). O parâmetro γ varia de 0 a 1. Um valor muito alto de γ pode fazer com que o algoritmo “decore” os dados.

$$K(x, x_i) = \exp(-\gamma * \sum (x - x_i^2)) \quad (10)$$

3. MATERIAIS E MÉTODOS

3.1 Características dos dados

Para realização deste trabalho foram utilizados os dados sobre resultados de exames para detecção do câncer de mama no *dataset* de Wolberg et al. (1992) disponíveis no repositório da UCI (*University of California - Irvine, UCI Machine Learning Repository*). Este banco de dados é de autoria da Universidade de Wisconsin, Madison, sendo as informações coletadas pelo Dr. William H. Wolberg.

Cada instância de dado contém seu identificador, 10 atributos e uma classe. Os valores dos atributos são calculados a partir da imagem digitalizada de uma amostra da massa retirada de uma mama. Tais valores descrevem as características dos núcleos celulares apresentados na imagem. Cada instância tem uma das 2 classes possíveis: benigno ou maligno, sendo, portanto, um problema de classificação binário.

Os atributos para cada instância de dado são: raio (*radius*), textura (*texture*), perímetro (*perimeter*), área (*area*), suavidade (*smoothness*), compactidade (*compactness*), concavidade (*concavity*), pontos côncavos (*concave points*), simetria (*symmetry*), dimensão fractal (*fractal dimension*). Cada atributo tem seus valores de média, erro e pior valor, totalizando 30 valores para cada instância.

Nos dados originais, os valores são apresentados separados por vírgula, contendo 32 campos para cada instância, sendo um campo com código da amostra, 10 atributos e um campo com as classes. O campo 1 é o código da amostra, um valor numérico inteiro; o campo 2 é a classe e possui os valores B para benigno e M para maligno; os atributos nos campos 3 a 32 são representados por valores inteiros.

O conjunto de dados analisado é composto por 569 instâncias. Para realização deste trabalho foram retirados as amostras com valores faltantes. De acordo com a classificação fornecida nos dados originais, a base de dados continha 212 amostras classificadas como malignas e 357 classificadas como benignas.

3.2 Tratamento dos dados

Para tratamento dos dados e implementação das funções utilizadas foi utilizada a linguagem *Python* e suas bibliotecas *open source* para aprendizagem de máquina, como *pandas*, *numpy*, *matplotlib*, *seaborn* e *keras*. Essas bibliotecas são muito utilizadas para o desenvolvimento de projetos na área de inteligência artificial, aprendizado de máquina e manipulação de dados, como forma de facilitar as implementações e agilizar o processo de análise e obtenção de resultados.

Em relação aos dados utilizados, primeiramente, a base de dados foi separada em treino e teste, sendo que o conjunto de treino ficou com 70% dos dados e o conjunto de teste ficou com os 30% restantes. Assim sendo, o conjunto de treino contou com 398 amostras e o conjunto de testes com 171 amostras.

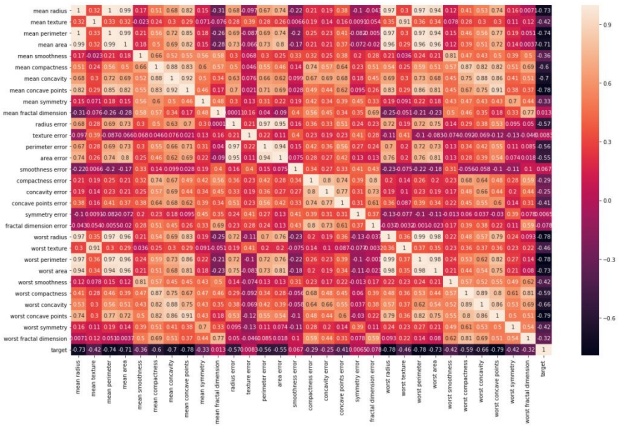


Figura 1. Correlação entre os atributos de uma instância de dado analisada.

3.3 Funções Kernel

A implementação da SVM foi feita utilizando funções das bibliotecas de código livre disponíveis nos pacotes do *Python*. A escolha da função Kernel foi feita de acordo com as opções disponíveis nas bibliotecas utilizadas, sendo utilizadas algumas das funções mais comuns em aplicações com SVM, já que não há critérios bem definidos e consolidados para escolha da melhor função Kernel. Neste trabalho as funções Kernel utilizadas foram: função de base radial (RBF), sigmóide (sigmoid) e linear (linear).

Os valores dos parâmetros de cada função Kernel foram ajustados de acordo com os resultados obtidos, buscando sempre maximizar os índices usados para análise dos resultados. Não foi utilizado nenhum método de otimização para escolha dos parâmetros, sendo necessários novos testes a cada mudança dos valores.

A partir das implementações realizadas com as diferentes funções Kernel escolhidas, depois de escolhidos os parâmetros que gerasse os melhores resultados para cada uma, foram geradas as matrizes de confusão para cada caso. A matriz de confusão mostra o quanto o classificador confunde uma classe com outra.

Para analisar a acurácia e a porcentagem de acertos de cada função, foram utilizados os resultados da classificação dispostos na matriz de confusão de cada função. Além disso foram calculados valores de precisão, recall e do índice F1.

4. RESULTADOS E DISCUSSÃO

Para analisar a correlação dos dados foi desconsiderada a coluna que identifica a amostra. O valor de correlação indica o grau de associação entre duas variáveis. A correlação dos dados é mostrada na Figura 1.

Analisando a Figura 1 é possível perceber que as maiores correlações ocorrem entre o raio médio e o perímetro médio (1,0), bem como entre a área média e o perímetro médio (1,0). Essa alta correlação entre essas variáveis é explicada pois são variáveis relacionadas às medidas do tamanho da amostra, e os valores do perímetro e da área são calculados utilizando o raio da amostra.

A Figura 2 apresenta a relação entre os atributos de área e perímetro médio. Os pontos laranjas representam amostras

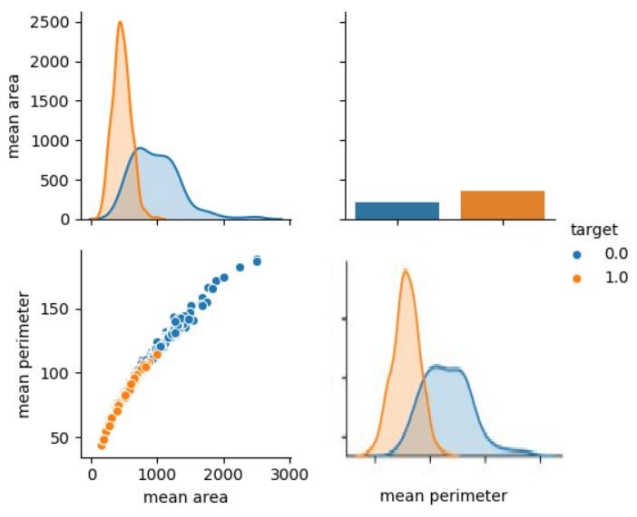


Figura 2. Relação entre a área média e o perímetro médio.

classificadas benignas, ou seja, não apresentam câncer. Já os pontos azuis representam amostras classificadas como malignas.

4.1 Funções Kernel

As funções Kernel utilizadas foram: função de base radial, sigmóide e linear. Os valores dos parâmetros de cada função Kernel foram ajustados ao longo do experimento buscando maximizar o número de acertos do classificador. Os valores finais de cada parâmetro utilizado se encontram na Tabela 1.

Tabela 1. Parâmetros de cada função Kernel.

Função Kernel	Parâmetros (%)
Linear	$C=1$ e $\gamma = 0.1$
Sigmóide	$\gamma = 2^{-7}$
Base radial	$\gamma = 1^{-4}$

A matriz de confusão para a implementação da função Kernel linear está representada na Tabela 2.

Tabela 2. Matriz de confusão para função Kernel linear.

	Valor Previsto	
	Maligno	Benigno
Maligno	103	4
Benigno	4	60

A matriz de confusão para a implementação da função sigmóide linear está representada na Tabela 3.

Tabela 3. Matriz de confusão para função Kernel sigmóide.

	Valor Previsto	
	Maligno	Benigno
Maligno	105	2
Benigno	15	49

Da mesma forma que disposto acima, a matriz de confusão para a implementação da função Kernel de base radial está representada na Tabela 4.

De acordo com os dados acima, foi possível calcular a porcentagem de acertos para cada escolha de função Kernel.

A implementação da SVM tendo como função Kernel a função linear apresentou porcentagem de acertos de 95.3%, sendo que seus parâmetros foram configurados com os valores de $C=1$ e $\gamma = 0.1$.

A implementação da SVM tendo como função Kernel a função sigmóide apresentou porcentagem de acertos de 90.0%, sendo que seu parâmetro foi configurado com valor de $\gamma = 2^{-7}$. A implementação da SVM tendo como função Kernel a função de base radial (RBF) apresentou porcentagem de acertos de 93.6% com o parâmetro $\gamma = 1^{-4}$.

Para análise dos resultados foram utilizados os valores da acurácia, calculada conforme descrito na Equação (Equação 11), onde TP, FP TN, FN representam as instâncias Verdadeiras Positivas, Falso Positivas, Verdadeiras Negativas e Falso Negativas, respectivamente. As porcentagens de acerto (valores de acurácia) foram colocadas na Tabela 5.

$$Acuracia = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

No trabalho de Agarap (2017) a acurácia obtida com o algoritmo SVM foi de 96.09%, porém o autor não especificou qual a função Kernel foi utilizada, nem os valores dos parâmetros adotados. A implementação da SVM nesse trabalho obteve valores de acurácia próximos aos obtidos por Agarap (2017) na implementação com SVM, sendo o mais próximo o da função Kernel linear.

Os resultados podem variar de acordo com a escolha dos parâmetros utilizados na construção da SVM. A implementação com função Kernel sigmóide obteve o pior desempenho dentre os algoritmos analisados. Com os parâmetros escolhidos nesse trabalho para a função Kernel linear, a acurácia foi superior àquela obtida com os métodos SVM treinada com uma Unidade de Processamento Gráfico (GPU, do inglês *Graphics Processing Unit*), e *Nearest Neighbors* utilizados por Agarap (2017), que obteve acurácias de 93.75% e 93.56% respectivamente.

Vale ressaltar que uma SVM treinada com GPU conta com maior poder de processamento do que uma SVM simples, e o modelo proposto neste trabalho com função de base radial alcançou resultados bem próximos daqueles obtidos com os algoritmos e parâmetros do trabalho de Agarap (2017). Os autores de ambos os trabalhos utilizaram a mesma base de dados disponibilizada por Wolberg et al. (1992).

Tabela 4. Matriz de confusão para função Kernel de base radial.

	Valor Previsto	
	Maligno	Benigno
Maligno	102	5
Benigno	6	58

Tabela 5. Porcentagem de acertos para as diferentes funções Kernel.

Função Kernel	Acertos (%)
Linear	95.3
Sigmóide	90.0
Base radial	93.6

Da mesma forma que o valor da acurácia, os valores de precisão (Equação 12), *recall* (Equação 13) e F1 (Equação 14), foram calculados usando os valores da matriz de confusão e são descritos abaixo, onde TP, FP TN, FN representam as instâncias Verdadeiras Positivas, Falso Positivas, Verdadeiras Negativas e Falso Negativas, respectivamente. O valor F1 é obtido usando os valores de precisão e *recall*, gerando um número único que indica a qualidade geral do modelo.

$$Precisao = \frac{TP}{TP + FP} \quad (12)$$

$$Recall = \frac{TP}{TP + FN} \quad (13)$$

$$F1 = \frac{2 * Precisao * Recall}{Precisao + Recall} \quad (14)$$

Os valores de precisão, recall e F1 foram calculados para cada função Kernel e foram colocados na Tabela 6, juntamente com o valor da acurácia (em porcentagem) obtidos com cada função Kernel, para possibilitar melhor visualização dos resultados.

Tabela 6. Resultados para as diferentes funções Kernel.

Função Kernel	Acurácia	Precisão	Recall	F1
Linear	95.3	96,3	96,3	96,3
Sigmóide	90.0	87,5	98,1	92,5
Base radial	93.6	94,4	95,3	94,8

De acordo com os valores da Tabela 6, pode-se notar que a função linear teve os melhores resultados, pois apresentou os maiores valores de acurácia, precisão e o índice F1, que apresenta a qualidade geral do modelo. A função sigmóide apresentou o maior valor de recall. A análise do valor recall é importante em problemas nos quais a ocorrência de falsos negativos são considerados mais prejudiciais que os falsos positivos.

No problema considerado, de detecção do câncer de mama, um falso negativo é mais prejudicial do que um falso positivo, já que uma mulher que recebe um resultado de que está saudável provavelmente não irá buscar outros exames para confirmar o diagnóstico e, no caso desse resultado ser um falso negativo, pode comprometer a chance de cura da paciente, por retardar o início do tratamento. No caso de um falso positivo, um exame complementar que poderia ser usado para conseguir mais informações provavelmente determinaria que a paciente uma vez diagnosticada como doente na verdade se encontra saudável.

5. CONCLUSÃO

Este artigo apresentou os resultados obtidos com a implementação de uma SVM, utilizando a linguagem Python e bibliotecas de código aberto, para classificação de dados referentes a nódulos mamários, que podem ser malignos ou benignos. Também foi apresentada uma comparação do desempenho da SVM de acordo com a função Kernel escolhida.

A SVM se mostrou eficaz para classificação dos dados, sendo importante notar que seu desempenho pode ser

afetado pela escolha dos parâmetros no momento de sua construção. Em comparação com outros métodos encontrados na literatura, a Máquina de Vetores de Suporte apresentou um resultado similar.

Para trabalhos futuros, tem-se a possibilidade de fazer a seleção dos atributos que são analisados pela SVM, podendo ser aplicado algum algoritmo genético. Outra sugestão seria diminuir a quantidade de atributos analisados, fazendo a análise de correlação entre as variáveis e retirando algumas variáveis para comparação de resultados e desempenho do modelo com menos atributos sendo analisados. Também é possível fazer a implementação de outros métodos para comparação dos resultados. Outra possibilidade é a utilização de algum método de otimização para selecionar os melhores parâmetros para a SVM.

AGRADECIMENTOS

Os autores agradecem à CAPES e FAPEMIG por apoiarem este trabalho.

REFERÊNCIAS

- Agarap, A.F. (2017). On breast cancer detection: An application of machine learning algorithms on the wisconsin diagnostic dataset. *CoRR*, abs/1711.07831.
- Borchardt, T.B., Rita, A.C., Lima, C.F., Resmini, R., and Sanchez, A. (2013). Breast thermography from an image processing viewpoint: A survey. *Signal Processing, Elsevier*, 93, 2785–2803.
- Borges, L.R. (2015). Analysis of the wisconsin breast cancer dataset and machine learning for breast cancer detection. *Proceedings of XI Workshop de Visão Computacional*.
- Boser, B., Guyon, I.M., and Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, 144–152.
- Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation. International edition*. Prentice Hall.
- Holsbach, N., Fogliatto, F.S., and Anazanello, M.J. (2014). Método de mineração de dados para identificação de câncer de mama baseado na seleção de variáveis. *Revista Ciência Saúde Coletiva*, 19, 1295–1304.
- Wolberg, W.H., Street, W.N., and Mangasarian, O.L. (1992). Breast cancer wisconsin (diagnostic) data set. [https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+\(Diagnostic\)](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)).