Um Estudo sobre a Predição da Recidiva de Câncer Usando Técnicas de Aprendizado de Máquina

André Eidi Maeda[1,2], Pedro Ferreira Crocco[1,2], Guilherme Cesar Soares Ruppert[1], Mariangela Dametto[1], Rodrigo Bonacin[1]

amaeda@cti.gov.br, pcrocco@cti.gov.br,
guilherme.ruppert@cti.gov.br,mdametto@cti.gov.br
rodrigo.bonacin@cti.gov.br

[1] Centro de Tecnologia da Informação Renato Archer – CTI

[2]Universidade Estadual de Campinas - Unicamp

Abstract. *Machine learning is a technique in which data is provided to a computer system, and from this data the computer creates models capable of analyzing them. Thus, this technique can predict results or even find patterns in the information provided. Prediction and analysis of health data is one of the domains where this technique can be applied. This domain, in general, contains a lot of information with complex relationships, that is, a favorable scenario for the application of computational algorithms based on machine learning. The objective of this article is to present a study that aims to predict, through machine learning techniques and algorithms, the recurrence of cancer by analyzing data obtained in open databases. In this way, it is expected to contribute to the support of health professionals in research, diagnosis and medical decisions.*

Resumo. *O aprendizado de máquina (machine learning) é uma técnica na qual são fornecidos dados para um sistema computacional e a partir destes dados o computador cria modelos capazes de analisá-los. Assim, essa técnica pode prever resultados ou até mesmo encontrar padrões nas informações fornecidas. Entre as áreas que essa técnica pode ser empregada está a predição e análise de dados de saúde, os quais, em geral, apresentam muitas informações com relações complexas, ou seja, um cenário propício para aplicação de algoritmos computacionais baseados em aprendizado de máquina. O objetivo deste artigo é apresentar um estudo que visa predizer, por meio de técnicas e algoritmos de aprendizado de máquina, a recidiva de câncer por meio da análise de dados obtidos em bases abertas. Dessa maneira, espera-se contribuir para o apoio a profissionais de saúde em pesquisas, diagnóstico e decisões médicas.*

Palavras-chaves: Aprendizado de Máquina, Predição, Inteligência Artificial, Informática em Saúde

1. Introdução

O uso de tecnologias em saúde vem ganhando ainda mais relevância no contexto mundial, em especial destaca-se a importância da tecnologia a fim de auxiliar em tratamentos e diagnósticos de doenças. Dentre as inúmeras atividades de profissionais de saúde que podem ser assistidas pelo uso da tecnologia estão o diagnóstico e a predição de recidiva em casos de câncer. Isso é possível graças à aplicação de técnicas de análise nos dados dos pacientes por meio de algoritmos que utilizam o conceito de aprendizado de máquina, bem como a

existência de bases de dados disponíveis, tais como o banco de dados da Fundação Oncocentro do Estado de São Paulo (FOSP)[1]. Este banco de dados possui informações sobre mais de 1 milhão de casos, dos quais aproximadamente 9% registraram recidiva, o que traz evidências sobre a importância do problema em questão.

Neste trabalho, foram inicialmente estudadas e aplicadas algumas técnicas de aprendizado de máquina, tais como SVM (*Support Vector Machines*), Naïve Bayes, entre outros. A escolha dessas técnicas se deu por serem baseadas em algoritmos largamente utilizados e conhecidos [Bishop 2006], sendo assim, um ponto de partida chave para a análise dos dados.

Para a implementação dessas técnicas optou-se pela biblioteca scikit-learn[2]. Sendo essa uma biblioteca *open source* desenvolvida para a linguagem python, cujo foco é apoiar o desenvolvimento de aplicações práticas de aprendizado de máquina.

O restante deste artigo está estruturado da seguinte forma: a seção 2 apresenta os principais conceitos, técnicas e algoritmos utilizados neste artigo; a seção 3 apresenta uma descrição da metodologia adotada no trabalho; em seguida, a seção 4 detalhada os principais resultados obtidos até a presente data; e, por fim, a seção 5 apresenta a conclusão e próximos passos desta pesquisa.

2. Conceitos, Técnicas e Algoritmos

O aprendizado de máquina é um processo no qual, em posse de um conjunto de dados, busca-se tirar conclusões sobre estes. Nesse sentido, espera-se que, com base em informações previamente fornecidas, seja possível fazer predições ou encontrar padrões a fim de obter respostas que manualmente poderiam ser difíceis de visualizar. No caso deste estudo, objetiva-se construir um modelo de aprendizado de máquina que possa predizer casos de recidiva de câncer com base nas características dos pacientes, da doença, dos tratamentos, entre outros parâmetros.

Conforme apresenta a Figura 1, as técnicas de aprendizado de máquina podem ser divididas entre:

- *Aprendizado supervisionado:* Consiste em treinar um modelo, tais como o SVM, com base em informações previamente fornecidas de entrada e saída. Dessa maneira, o modelo pode prever resultados futuros tendo como base parâmetros (*features*) obtidos a partir de resultados anteriores. Os algoritmos supervisionados se dividem em dois grupos: (1) os de classificação, que buscam prever os resultados separando-os em classes, como variáveis discretas; (2) os de regressão, os quais preveem respostas contínuas, tais como variação de temperatura.
- *Aprendizado não supervisionado:* Visa achar padrões ou grupos nos dados fornecidos.

[1] http://www.fosp.saude.sp.gov.br/fosp/diretoria-adjunta-de-informacao-e-epidemiologia/rhc-registro-hospitalar-de-cancer/banco-de-dados-do-rhc/
[2] https://scikit-learn.org/stable/

Figura 1. Na imagem observa-se os tipos de estratégias em aprendizado de máquina. (adaptada de https://opencadd.com.br/o-que-e-machine-learning)

Dos grupos de técnicas supracitadas, neste primeiro momento, o foco foi a aplicação de métodos de aprendizado supervisionado de classificação. Como mencionado, para colocar em prática essas técnicas foi utilizado a biblioteca para Python, scikit-learn. Python é uma linguagem de programação desenvolvida nos anos 90, pelo matemático holandês Guido Van Rossum com o objetivo de otimizar a produção e leitura de códigos.

Dos diversos algoritmos de classificação existentes, este artigo se aprofunda em dois, o Naive Bayes e o SVM [Müller and Guido 2016].

O Naive Bayes é um modelo de classificação em aprendizado supervisionado. Nesse sentido, como o próprio nome indica, ele é baseado no teorema de Bayes, introduz um conceito vindo da probabilidade, que calcula a possibilidade de um evento acontecer, com base na ocorrência de um evento anterior. É dado pela fórmula: $P(A|B) = \frac{P(B|A)\ x\ P(A)}{P(B)}$, onde P(A|B) é a probabilidade do evento A ocorrer dado que B ocorreu, P(B|A) a probabilidade de B ocorrer dado que A ocorreu, P(A) a probabilidade do evento A ocorrer e, finalmente, P(B) é a probabilidade do evento B ocorrer. Vale ressaltar que este algoritmo assume que as *features* observadas são independentes entre si.

O SVM, assim como o Naive Bayes, foi aplicado visando uma abordagem para classificação supervisionada. Conforme apresenta a Figura 2, o princípio geral de funcionamento desta técnica se baseia na criação do melhor hiperplano que separa as classes analisadas. Nessa perspectiva, busca-se criar um hiperplano que contenha a maior margem entre os vetores de suporte (*support vectors*), que são os pontos mais próximos do hiperplano de cada classe. Esse plano, por sua vez, indica a linha de decisão que separa uma classe da outra.

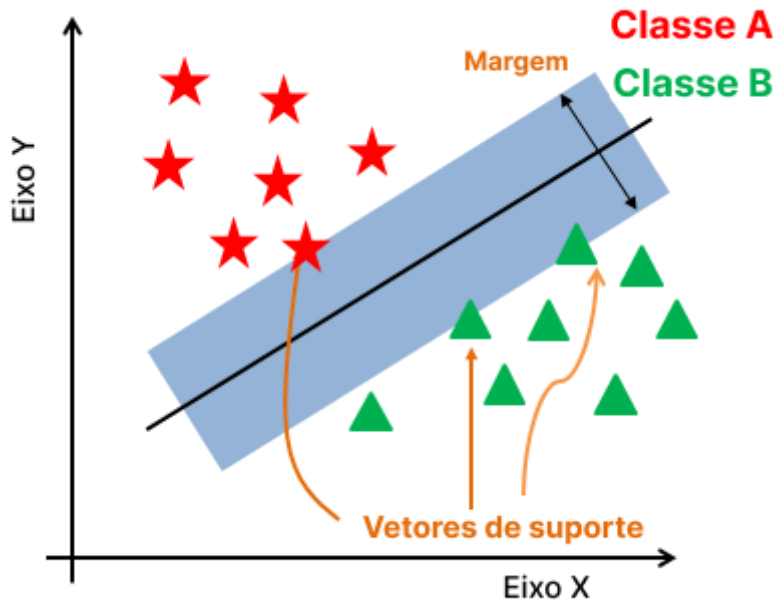

Figura 2. Criação do hiperplano que separa as classes de interesse para o problema buscando a maior margem entre os vetores de suporte. (adaptada de https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python)

A base de dados com a qual este trabalho foi realizado foi fornecida livremente na internet pela FOSP. Essa foi criada em 1967, sob o nome de Centro de Oncologia, por um grupo de professores da Universidade de São Paulo com o objetivo de incentivar os estudos e atividades relacionadas à área da oncologia. Nesse sentido, mudou de nome mais duas vezes, a primeira em 1974 pelo Governo Estadual de São Paulo a fim incentivar o ensino e a pesquisa, estimulando a prevenção e detecção precoce de câncer. Em 1986, após mudanças administrativas, tornou-se a instituição que conhecemos hoje.

Disponibilizado no site da Fundação Oncocentro, o banco de dados utilizado neste projeto é gerado a partir de dados de diversas instituições de saúde no estado de São Paulo e outros estados brasileiros sob coordenação da FOSP. O objetivo é oferecer àqueles interessados informações para suas pesquisas, interesses ou necessidades. Vale destacar que são cadastrados pacientes desde 01/01/2000 e atualmente o banco é atualizado a cada três meses, sendo que há regras específicas para novos cadastros, tais como, somente são considerados casos analíticos aqueles que chegaram à instituição sem tratamento, entre outras regras encontradas no site da fundação[3]. Para a versão do arquivo utilizado para este projeto (obtida no início de 2022) têm-se informações de 1.051.955 pacientes com 99 atributos (*features*) para cada paciente, tais como nome, tipo de câncer, tratamento recebido, entre outros

3. Metodologia aplicada

Para desenvolver este projeto, como mencionado anteriormente, optou-se por técnicas de classificação em aprendizado supervisionado. Isso com o objetivo de fazer predições sobre a recidiva de casos de câncer. Nas subseções a seguir são detalhadas etapas principais desse processo.

[3] http://www.fosp.saude.sp.gov.br/

3.1 Seleção e Pré-processamento

A priori, foram obtidos dados relacionados a pacientes de câncer na base aberta da FOSP. A partir destes dados foram selecionadas as *features* para as quais serão aplicados os algoritmos de aprendizado de máquina supracitados. Nesse sentido, escolheu-se trabalhar de forma a relacionar as *features* que indicam o tipo de tratamento adotado, tais como cirurgia, radioterapia, quimioterapia, entre outros, com a informação da ocorrência ou não de recidiva nos pacientes. A seleção de *features* levou em consideração a qualidade dos dados, disponibilidade e a pertinência aos objetivos propostos (predizer recidiva de câncer). Para tanto, o projeto contou com a participação direta de uma profissional biomédica, com mestrado na área de oncologia, e consultas pontuais a outros especialistas (médicos) por intermédio desta profissional.

Vale destacar que, antes de selecionar, foi necessário desenvolver o software de carga e pré-processamento. Os dados foram originalmente fornecidos como arquivos '.dbf', uma extensão padrão para base de dados, que organiza informações em vários registros com campos armazenados na forma de listas. Para lidar com esses arquivos, foi necessário utilizar a biblioteca "simpledbf"[4] a fim de transformá-la em uma tabela que descrevesse os dados (Pandas dataframe), dessa forma, a manipulação da base torna-se mais simples.

3.2. Treinamento do modelo

Após o processo de seleção e pré-processamento dos dados foi necessário dividir a base para treinar e testar o modelo escolhido. Nesse sentido, foram utilizadas as seguintes estratégias: primeiramente, separou-se a base na proporção 80/20, isto é, 80% dos dados foram dedicados ao treinamento do modelo, enquanto os 20% restantes foram aplicados a teste do mesmo modelo.

Em seguida, a fim de tentar obter um resultado mais confiável aplicou-se uma técnica de validação cruzada denominada "K-Fold". Essa técnica, presente na biblioteca do scikit-learn, consiste em dividir o conjunto de dados em 'K' subconjuntos de mesmo tamanho. Conforme ilustra a Figura 3, posteriormente um desses subconjuntos é selecionado para testes e os restantes para treinamento (observe a figura abaixo). Esse processo é realizado 'K' vezes de forma que ao final dessas 'K' interações calcula-se a acurácia e os erros. Assim, obtém-se resultados mais confiáveis.

[4] https://pypi.org/project/simpledbf/

Figura 3. Ilustração de como o método K-fold separa a base em subconjuntos, nos quais em cada iteração um será utilizado como teste e os outros como treinamento. (adaptada de https://www.section.io/engineering-education/how-to-implement-k-fold-cross-validation/)

Finalmente, notou-se que os dados de recidiva encontram-se desbalanceados, isto é, há muito mais indicativos de não recidiva do que há de recidiva. Esse desbalanceamento na base poderia levar o algoritmo aplicado a retornar resultados viciados ou tendenciosos, visto que retornar como resultado o dado de maior quantidade levaria a bons dados de acurácia, mas que não são de fato corretos e confiáveis. Para isso, aplicou-se uma técnica chamada SMOTE (*Synthetic Minority Oversampling Technique*) [Chawla *et al.* 2002] com auxílio da biblioteca "imblearn"[5]. Conforme ilustra a Figura 4, essa técnica funciona de forma a super amostrar (*oversample*) a classe com menor número de dados, isto é, são criados exemplos sintéticos. Esses exemplos sintéticos são formados pela combinação das características de duas instâncias 'a' e 'b' escolhidas aleatoriamente.

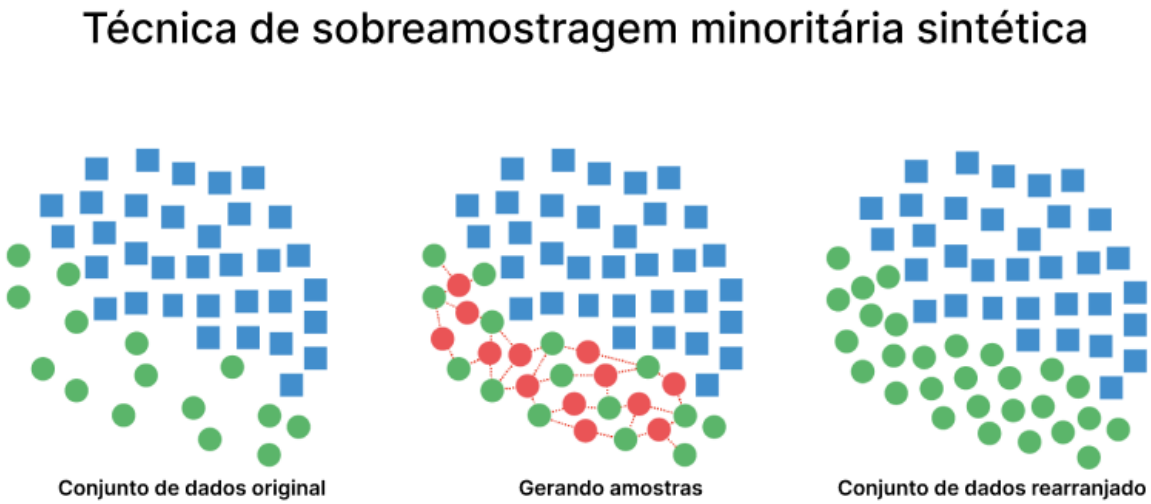

Figura 4. Imagem ilustrativa da criação de dados sintéticos a fim de balancear a base de dados. (adaptada de https://medium.com/analytics-vidhya/bank-data-smote-b5cb01a5e0a2)

[5] https://imbalanced-learn.org/stable/

3.3. Aplicação dos algoritmos

Por fim, com auxílio da biblioteca scikit-learn foram implementados os algoritmos escolhidos (*i.e.*, Naive Bayes e SVM). É importante destacar que em ambos os casos grande parte dos parâmetros foram mantidos com os valores padrão adotados pela biblioteca. Dentre estes, destacam-se, principalmente, no caso do SVM: *C (Parâmetro de regularização) = 1.0*, indica o quanto deseja-se que o algoritmo evite classificar um dado de forma errada. *Kernel = Linear*, informa qual a forma do hiperplano deseja-se ajustar os dados. No caso do Naive Bayes, vale destacar que dos três modelos desse algoritmo presente na biblioteca scikit learn, optou-se pelo uso da *BernoulliNB* pelo fato de ser destinada a problemas cujas *features* possuem valores discretos ou booleanos, tal como é as classes que deseja-se relacionar inicialmente. Assim como no caso do SVM os parâmetros foram mantidos padrão.

4. Testes e Resultados

Terminada a aplicação dos algoritmos coletou-se os resultados, os quais serão demonstrados abaixo seguido de uma interpretação e discussão da validade dos mesmos. Inicialmente, serão exibidos os resultados referentes ao algoritmo Naive Bayes, em seguida aqueles relacionados ao algoritmo SVM (support Vector Machines).

A Tabela 1 apresenta os resultados obtidos com a técnica Naive Bayes, incluindo as medidas de precisão, acurácia e F1-Score (média harmônica entre precisão e revocação/cobertura). A Figura 5 apresenta a matriz de confusão na estratégia de treinamento Naive Bayes para a divisão 80/20. As Figuras 6, 7 e 8 detalham os valores obtidos em cada fold da validação cruzada para a precisão, acurácia e F1-Score, respectivamente. Finalmente, a Figura 9 apresenta a matriz de confusão do algoritmo para a aplicação do ‘SMOTE’ no treinamento do modelo.

Tabela 1.Resultados obtidos para cada estratégia de treinamento do modelo com Naive Bayes

Naive Bayes	***80/20***	***K-fold (Média)***	***Smote***
Acurácia	0.9100436805756901	0.9109144402564749	0.6344758093264445
Precisão	0.9104839630179203	0.9118629773790585	0.9484022372589667
F1 Score	0.9528981359348945	0.9533671866601054	0.7592123513922243

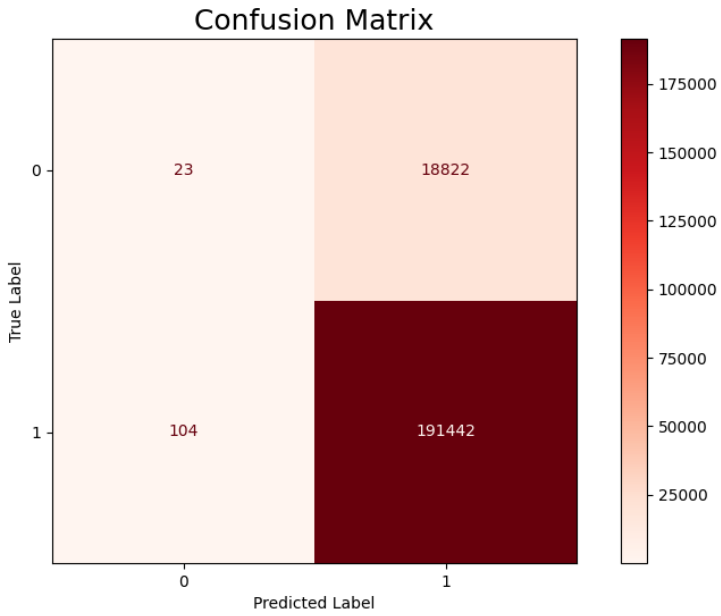

Figura 5. Matriz de confusão para a estratégia de treinamento 80/20 no algoritmo Naive Bayes sem o SMOTE

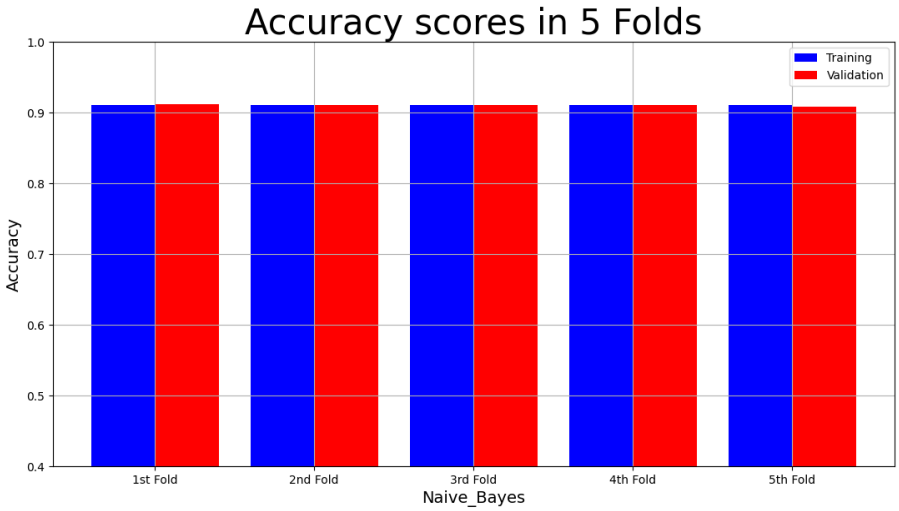

Figura 6. Comparativo entre os valores obtidos para acurácia em cada uma das iterações usadas pela estratégia K-Fold

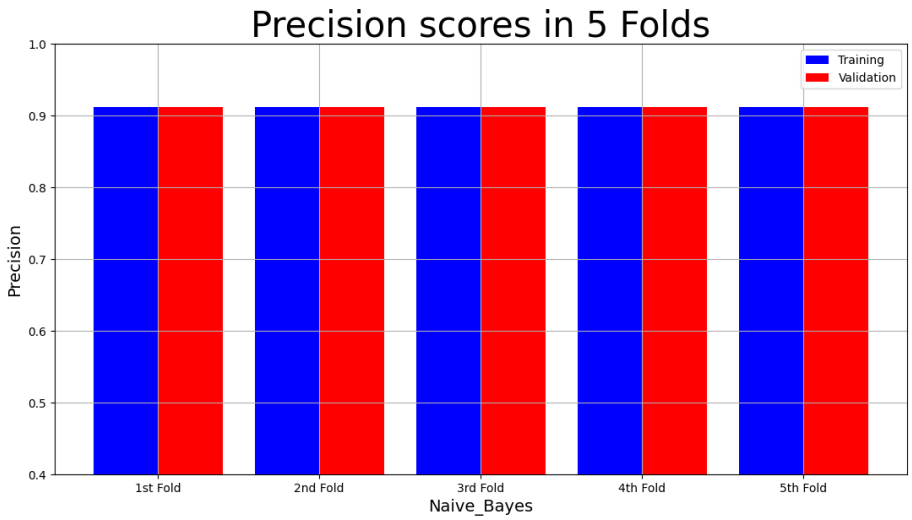

Figura 7. Comparativo entre os valores obtidos para precisão em cada uma das iterações usadas pela estratégia K-Fold

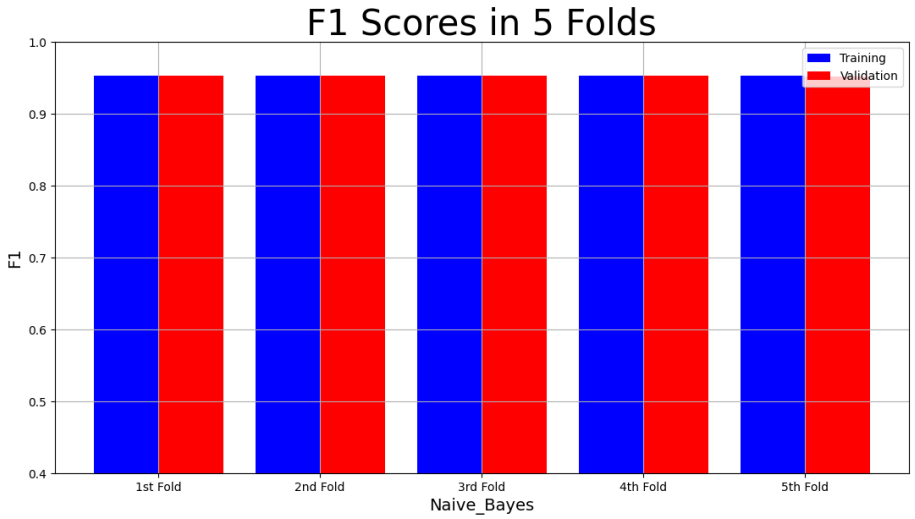

Figura 8. Comparativo entre os valores obtidos para F-1 em cada uma das iterações usadas pela estratégia K-Fold

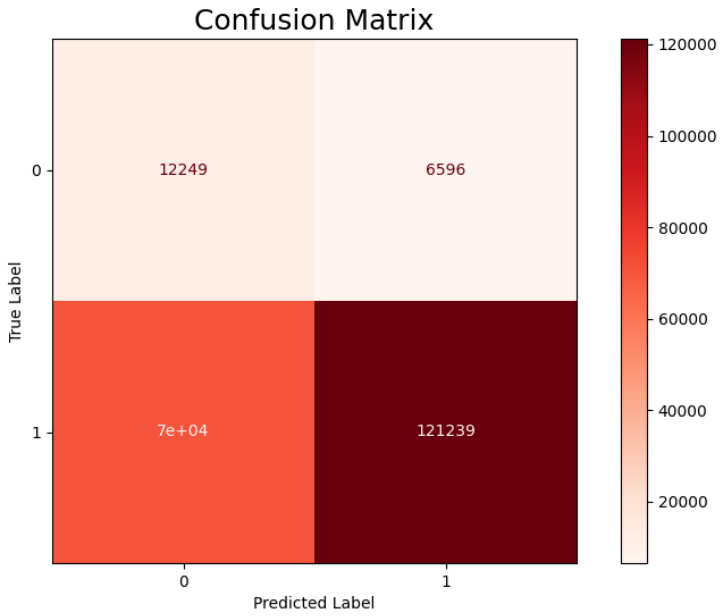

Figura 9. Matriz de confusão para a estratégia de treinamento 'SMOTE' para o algoritmo Naive Bayes

A Tabela 2 apresenta os resultados obtidos com a técnica SVM, incluindo as medidas de precisão, acurácia e F1-Score (média harmônica entre precisão e revocação/cobertura). A Figura 10 apresenta a matriz de confusão na estratégia de treinamento SVM para a divisão 80/20. As Figuras 11, 12 e 13 detalham os valores obtidos em cada fold da validação cruzada para a precisão, acurácia e F1-Score, respectivamente. Por fim, a Figura 14 apresenta a matriz de confusão do algoritmo para a aplicação do 'SMOTE' no treinamento do modelo SVM.

Tabela 2.Resultados obtidos para cada estratégia de treinamento do modelo com SVM

SVM	***80/20***	***K-fold (Média)***	***SMOTE***
Acurácia	0.9104286780328056	0.9920838183934806	0.6654894933718648
Precisão	0.9104286780328056	0.9920838183934807	0.9469494651420141
F1 Score	0.9531145428263634	0.9960261598875692	0.7848399246704332

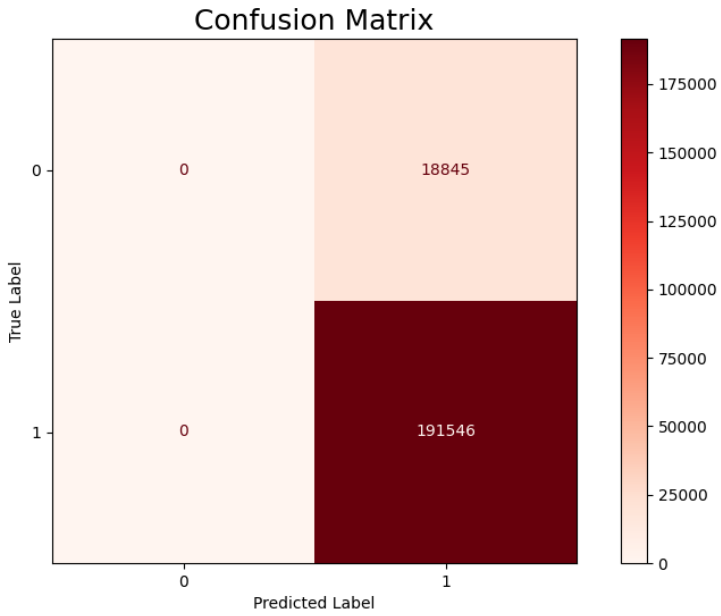

Figura 10. Matriz de confusão para a estratégia de treinamento 80/20 no algoritmo SVM sem o SMOTE

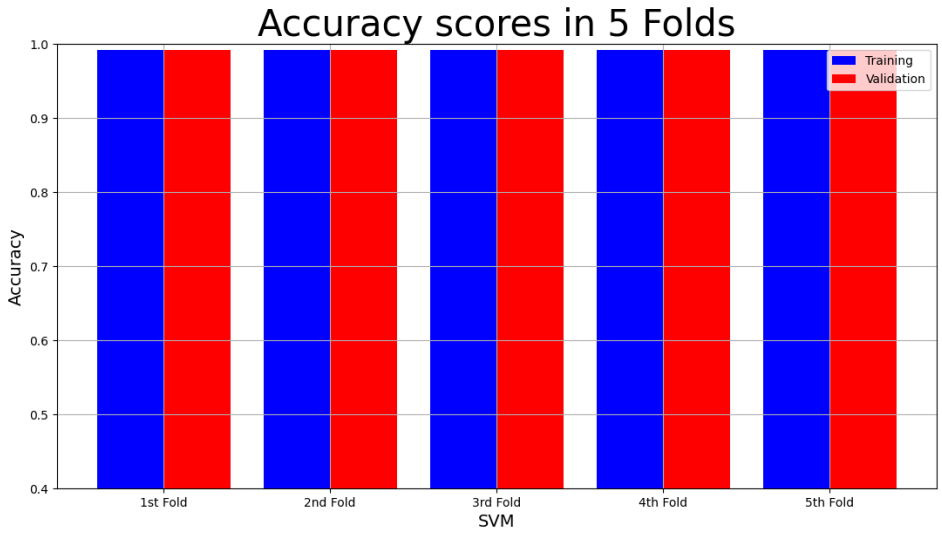

Figura 11. Comparativo entre os valores obtidos para acurácia em cada uma das iterações usadas pela estratégia K-Fold

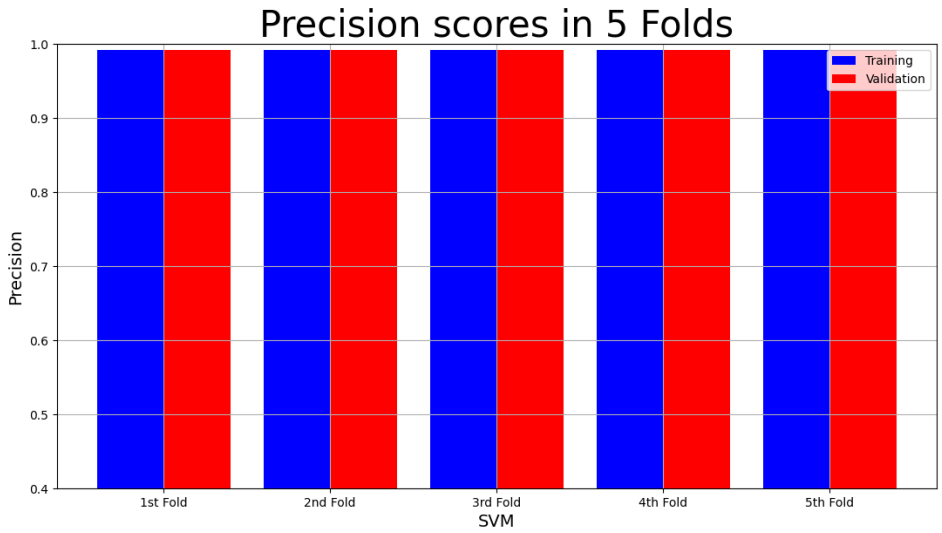

Figura 12. Comparativo entre os valores obtidos para precisão em cada uma das iterações usadas pela estratégia K-Fold

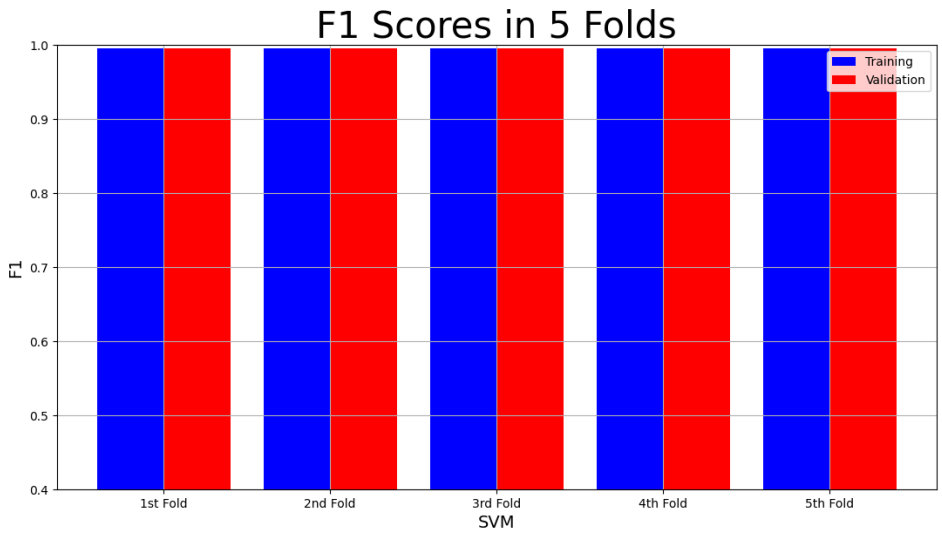

Figura 13. Comparativo entre os valores obtidos para F-1 em cada uma das iterações usadas pela estratégia K-Fold

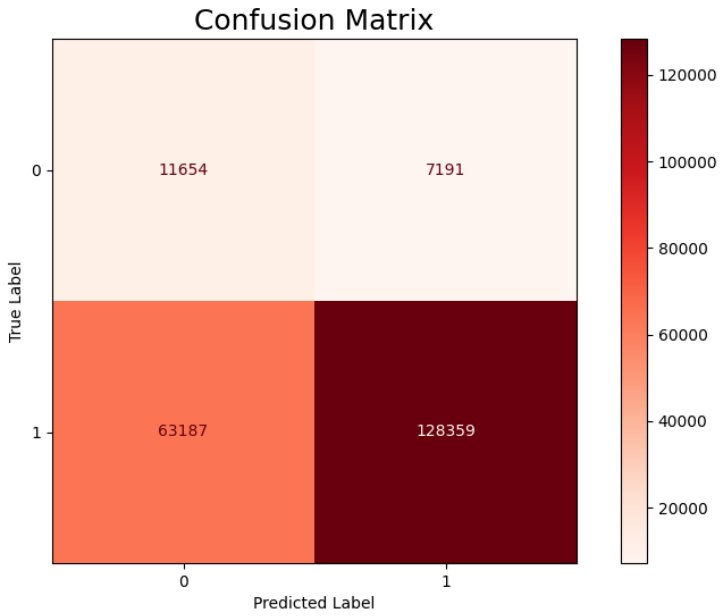

Figura 14. Matriz de confusão para a estratégia de treinamento 'SMOTE' para o algoritmo SVM

4.1. Interpretação e discussão dos resultados

No total foram aplicados 2 algoritmos na base de dados coletada e pré-processada. Com isso, foi possível obter resultados em 6 diferentes modelos, cada um com suas particularidades

Inicialmente, nota-se que em ambos os algoritmos para a técnica de separação de dados no qual 80% eram dedicado ao treinamento e 20% era dedicado aos casos de testes e validação do modelo obteve-se altos valores de acurácia, precisão e F1 Score para o modelo (Tabelas 1 e 2), entretanto por serem valores muito altos suspeitou-se que os modelos estavam enviesados. Isto é devido ao desbalanceamento dos dados de recidiva (havia muitos casos de não recidiva para poucos casos de recidiva) os modelos retornavam como resposta o valor de maior incidência e coincidentemente acertavam a predição muitas vezes. Para confirmar essa hipótese foi plotada a matriz de confusão para cada um dos algoritmos (Figuras 5 e 10). Uma tabela que indica as frequências que o modelo devolve cada um dos valores das classes de interesse. Observe que em ambas as matrizes de confusão o modelo retornou majoritariamente o valor 1, o que indica que nossa hipótese de que o modelo estava enviesado estava, possivelmente, correta

Para lidar com esse problema foram tomadas duas estratégias, usar as técnicas *k-fold* e, em seguida, o SMOTE, ambas mencionadas e explicadas na seção anterior. Para o caso do k-fold nota-se pelos valores da tabela que o problema persistiu visto os resultados tão altos quanto sem a aplicação da técnica. Finalmente, ao aplicar o SMOTE notou-se que os resultados de acurácia diminuíram consideravelmente, entretanto, possivelmente mais confiáveis dados os valores da precisão e, principalmente, o valor do F1-Score. Observa-se, ainda, pelas matrizes de confusão, após a aplicação do SMOTE (Figuras 9 e 14), que as classes de predição estão mais distribuídas. Isso indica que o algoritmo possivelmente aprendeu a desenvolver suas predições de maneira correta, isto é, sem devolver respostas enviesadas pela quantidade de dados de cada classe.

5. Conclusão

Os resultados da classificação indicaram que, apesar de serem necessários estudos complementares, este trabalho já contribui ao apontar uma provável relação entre o tratamento utilizado e a recidiva do câncer. Dessa maneira, torna-se possível estudos mais aprofundados sobre o tema. Uma das principais dificuldades encontradas foi a de custo computacional, isto é, tendo em vista a extensão da base de dados o custo computacional para executar os algoritmos é alto. Sendo assim, estes demoram a serem completamente executados. Com o acesso às máquinas mais rápidas será possível aplicar o conjunto de técnicas, e amplitude dos parâmetros de configuração, e assim obter melhores resultados.

A fim de buscar por resultados melhores, para trabalhos futuros planeja-se aplicar mais técnicas de separação e tratamento dos dados, bem como outras técnicas de classificação, tais como redes neurais artificiais. Além disso, também deseja-se separar a base por tipos de câncer e aplicar os algoritmos, bem como observar o comportamento dos modelos aplicados para dados mais específicos. Pretende-se também aprofundar no estudo da relação do tratamento utilizado (*e.g.*, cirurgia, radioterapia, quimioterapia, entre outros) e a predição da recidiva de câncer.

6. Referências

Bishop, C. M. (2006) Pattern recognition and machine learning. [S.l.]: springer.

Chawla, N. V. et al. (2002) Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research, v. 16, p. 321–357.

Müller, A. C., and Guido, S. (2016). Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.