

Aplicando Métodos de Aprendizado de Máquina para Predição do Risco de Óbito por COVID-19 com Base nos Sintomas Apresentados pelo Paciente

Applying Machine Learning Methods to Predict the Risk of Death by COVID-19 Based on the Symptoms Presented by the Patient

Aplicación de métodos de aprendizaje automático para predecir el riesgo de muerte por COVID-19 según los síntomas presentados por el paciente

Maxwell E. A. Silva¹; Victor G.L. Holanda¹; Eduardo Moares²; Maria H. L. S. Acioli³;
Rodrigo S. da Silva¹; Paulo V.L. Severiano¹; Rafael de Amorim Silva¹

1 Universidade Federal de Alagoas - Instituto de Computação - Centro de Pesquisa em Tecnologias Emergentes, Maceió (AL), Brasil.

2 Instituto de Computação, Universidade Federal de Alagoas – UFAL- AL, Maceió (AL), Brasil.

3 Professora do Ensino Básico Técnico e Tecnológico do Instituto Federal de Alagoas – IFAL-AL, Palmeira dos Índios (AL), Brasil

Autor correspondente: Maxwell E. A. Silva
E-mail: meas@ic.ufal.br

RESUMO

Este artigo investiga o desempenho de três algoritmos de aprendizado de máquina (Random Forest, K-nearest Neighbor e Perceptron Multilayer) para predizer o risco de um paciente de COVID-19 ir a óbito de acordo com seu quadro clínico atual. Três métricas são utilizadas na análise dos algoritmos de aprendizado de máquina: (i) a acurácia; (ii) a precisão; e (iii) a sensibilidade. Um experimento é desenvolvido para testar o desempenho dos algoritmos na predição do risco de óbito, utilizando a ferramenta scikit-learn e a linguagem Python na elaboração desta validação. Os resultados indicam que o perceptron de multicamada obtém um melhor desempenho nas três métricas utilizadas, o que pode sugerir uma técnica adequada para predizer pacientes de COVID-19 com riscos de óbitos. Consequentemente, estes resultados podem ajudar os médicos, profissionais de

saúde e políticos a tomarem decisões mais assertivas nos cuidados e tratamento desta doença.

Descritores: Aprendizado de Máquina; predição do risco de óbito; COVID19

ABSTRACT

This article investigates the performance of three machine learning algorithms (Random Forest, K-nearest Neighbor and Perceptron Multilayer) to predict the risk of a COVID-19 patient dying according to their current clinical condition. Three metrics are used in the analysis of machine learning algorithms: (i) accuracy; (ii) accuracy; and (iii) sensitivity. An experiment is developed to test the performance of the algorithms in predicting the risk of death, using the scikit-learn tool and the Python language in the elaboration of this validation. The results indicate that the multilayer perceptron obtains a better performance in the three metrics used, which may suggest an adequate technique to predict COVID-19 patients with risk of death. Consequently, these results can help doctors, health professionals and politicians to make more assertive decisions in the care and treatment of this disease.

Keywords: Machine Learning; death risk prediction; COVID-19

RESUMEN

Este artículo investiga el rendimiento de tres algoritmos de aprendizaje automático (bosque aleatorio, vecino más cercano K y perceptrón multicapa) para predecir el riesgo de muerte de un paciente con COVID-19 según su estado clínico actual. Se utilizan tres métricas en el análisis de algoritmos de aprendizaje automático: (i) precisión; (ii) precisión; y (iii) sensibilidad. Se desarrolla un experimento para probar el desempeño de los algoritmos en la predicción del riesgo de muerte, utilizando la herramienta scikit-learn y el lenguaje Python en la elaboración de esta validación. Los resultados indican que el perceptrón multicapa obtiene un mejor desempeño en las tres métricas utilizadas, lo que puede sugerir una técnica adecuada para predecir pacientes COVID-19 con riesgo de muerte. En

consecuencia, estos resultados pueden ayudar a médicos, profesionales de la salud y políticos a tomar decisiones más asertivas en el cuidado y tratamiento de esta enfermedad.

Palabra-Clave: Aprendizaje automático; predicción de riesgo de muerte; COVID-19

INTRODUÇÃO

A pandemia causada pelo vírus SARS-CoV-2, também conhecido como COVID-19, tem causado um expressivo impacto na sociedade no âmbito social, econômico, político, histórico e da saúde. O vírus apresenta uma alta facilidade de transmissão, aumentando exponencialmente o número de infectados ao longo do tempo. Com esta elevação no número de pessoas doentes, os sistemas de saúde foram altamente impactados, gerando um colapso sem precedentes na sociedade atual. Consequentemente, foram necessárias políticas de distanciamento entre as pessoas, a fim de reduzir o número de contágios. As medidas adotadas pelos governos de se estabelecer períodos de quarentena afetaram significativamente a saúde econômica dos países. Vários setores do comércio foram fechados, causando uma onda de desempregos nos setores trabalhistas. Do ponto de vista político, os governos tiveram que criar medidas para amenizar os prejuízos causados pela pandemia na sociedade, e intermediar a retomada dos diversos setores que tiveram suas atividades suspensas.

Dado a esta situação pandêmica, inúmeros centros de pesquisa, instituições científicas, governamentais e empresas têm se aliado para combater este vírus, e tentam acelerar o processo de criação de vacinas para produzir anticorpos capazes de neutralizar a ação desse vírus. Como o vírus em questão tem uma alta taxa de transmissão e um número elevado de óbitos entre as pessoas que o contraem, faz-se necessário a utilização de métodos ou tecnologias que sejam capazes de auxiliar na diminuição dos impactos que este possa causar a sociedade.

Uma destas tecnologias é a Inteligência Artificial (IA). A IA pode ser definida como um sistema tecnológico capaz de simular a inteligência humana. Esta tecnologia vem sendo aplicada em diversos setores da sociedade, inclusive como

uma ferramenta subsidiária em diversos aspectos na área médica. Um ramo da IA que tem ganhado um expressivo destaque é o do aprendizado de máquina. Este por sua vez pode ser definido como uma abordagem que é capaz de fazer previsões a partir de um conjunto de dados obtidos como entrada. Na área médica, o aprendizado de máquina tem se tornado um forte aliado na identificação e classificação de padrões, propiciando a prevenção, o diagnóstico e o prognóstico de aspectos relacionados a esta doença.

Diante deste contexto, este artigo traz a aplicação de três algoritmos de aprendizado de máquina para prever o risco de óbito por pacientes acometidos pela COVID-19. Inicialmente, dados abertos de pacientes com caso confirmado de Covid-19 do estado de Alagoas são extraídos para identificar informações que indiquem quais os sintomas estão mais fortemente ligados ao óbito decorrente deste vírus, com a criação de um modelo capaz de estimar a letalidade do vírus a partir de dados clínicos do indivíduo infectado. Desta forma, tais algoritmos determinam quais pacientes devem ter mais atenção no atendimento a partir de seus sintomas notificados. Esta informação tornaria o processo de atendimento a pacientes do Covid-19 mais efetivo, direcionando os recursos aos pacientes mais propensos a complicações. A estrutura do trabalho é definida a seguir. A Seção 'Método' descreve como o experimento foi desenvolvido e descreve as técnicas utilizadas para a predição do risco de óbito dos pacientes de COVID-19. A Seção 'Resultados' apresenta a avaliação de desempenho dos algoritmos implementados. A Seção 'Discussão' interpreta os resultados obtidos no experimento. A Seção 'Conclusão' apresenta as considerações finais deste artigo.

MÉTODO

Este artigo desenvolve um experimento para avaliar o desempenho dos algoritmos de aprendizado de máquina na predição do risco de óbitos por um paciente que esteja acometido com a COVID-19. Para desenvolver esse experimento, três etapas devem ser definidas: (i) realizar o pré-processamento dos dados; (ii) desenvolver os algoritmos a serem utilizados no experimento; e (iii) executar os algoritmos e coletar os resultados do experimento.

Etapa 1: Pré-processamento dos dados

Esta etapa requer uma fonte de dados para que o processamento seja definido. Em relação a este processo de aquisição dos dados, considerou-se uma base de dados que representasse o quadro de sintomas dos pacientes que procuraram alguma unidade de saúde em Alagoas para os primeiros atendimentos. Esta etapa é composta de informações como idade do paciente, sexo, município de residência, data de realização e resultado do exame para detecção do vírus, além de uma lista de sintomas que o paciente possivelmente apresente ao procurar ajuda (e.g. tosse, coriza, febre, diarreia, etc). O conjunto de dados utilizados para fazer este pré-processamento são referentes ao dia 14 de julho de 2020. O número inicial de variáveis apresentadas pela base é 45, em um total de 45788 instâncias. A primeira etapa realizada no pré-processamento dos dados foi a de remover as instâncias com datas inválidas, tais como anotação erradas do ano ou mês do atendimento prestado ao paciente. Além disso, foram removidos os casos em que o paciente está em situação de isolamento domiciliar a menos de 14 dias.

Nem todas as variáveis, em um julgamento inicial, possuíam alguma relevância para o propósito definido no escopo inicial do projeto, que é o de poder prever se o paciente poderá ou não vir a óbito de acordo com o quadro de sintomas apresentados por este. Portanto, algumas destas variáveis foram descartadas, por exemplo, as variáveis **id**, **etnia**, **municipio_residencia**, **data_resultado_exame**, **data_atendimento**, **tipo_coleta**, **data_obito**, **data_confirmacao_obito**, **profissional_saude**. A propriedade **idoso** foi removida por conta de já termos uma variável contendo a idade do paciente. Os atributos **profissional_saude**, **outros** e **outros_fatores** também foram consideradas não relevantes e foram descartadas. Também removeu-se algumas das variáveis que representavam sintomas e não possuíam valores em nenhuma das instâncias da base de dados, são elas: **ausegia**, **anosmia**, **nausea_vomito**, **coriza**, **congestao_nasal**, **calafrio** e **doenca_auto_imune**. As propriedades **fator_nao_informado** e **nao_informado** representam os casos em que não foram

inseridas as informações sobre o paciente no ato de atendimento ao mesmo. Tais casos foram desconsiderados na análise e posteriormente estas variáveis foram descartadas também.

A proposta inicial do projeto levou em conta apenas os casos concluídos, ou seja, ou o paciente já teve alta médica, encerramento do isolamento domiciliar ou o óbito do paciente. Desta forma, foram descartados os casos em que o paciente ainda está em isolamento domiciliar, está internado em algum leito clínico ou UTI. Após as etapas de diminuição de dimensionalidade e numerosidade, foram feitas transformações dos valores categóricos para valores numéricos sobre as variáveis restantes. Dessa forma, a propriedade sexo, que antes é representada pelos valores masculino e feminino, agora ficaram com valores 1 (masculino) e 0 (feminino). Nesta etapa também transformou-se todos os valores NaN (not a number) para valores 0, a maioria de valores NaN estavam nas propriedades relacionadas aos sintomas do paciente. Por sua vez, foi feita a transformação da variável alvo **situacao**, sendo esta propriedade a representação da situação atual do paciente. Neste caso, o valor óbito foi transformado em valor 1, e todas as demais situações foram convertidas no valor 0. Após esta última etapa, foram removidas todas as instâncias em que não haviam marcação de nenhum dos sintomas que o paciente poderia apresentar. Consequentemente, após todas as diminuições de dimensionalidade e numerosidade, a base de dados permaneceu um pouco desbalanceada, ou seja, o número de casos de não óbito era mais que o dobro dos casos de óbito. Para resolver este problema, foram feitas operações para que ambas as classes tivessem o mesmo número de instâncias.

As técnicas de IA adotadas para próxima etapa de desenvolvimento foram: Perceptron Multicamadas (MLP - Multilayer Perceptron), floresta aleatória (*random forest*) e K vizinhos mais próximos (KNN - *k-nearest neighbors algorithm*), comparando e identificando qual delas tem os melhores resultados em nossa aplicação, levando em consideração as métricas de avaliação providas da estatística. Aqui o papel destas técnicas consiste na predição do óbito ou não de um paciente, dado o conjunto de sintomas deste.

Etapas 2: Algoritmos Implementados

Random Forest (RF)

Random Forest (RF) é um algoritmo que pertence a classe dos algoritmos supervisionados de *machine learning*. Ele é um dos algoritmos mais utilizados pela sua simplicidade e diversidade, e também pelos bons resultados apresentados mesmo sem a utilização de hiperparâmetros. Além disso, ele pode ser utilizado tanto para tarefas de classificação ou de regressão.

Em geral, RF funciona através da construção de várias árvores de decisões, ou seja, ele divide o conjunto de dados em vários subconjuntos, e constrói para cada subconjunto uma árvore de decisão. Nessa conjuntura, no lugar de procurar a característica mais importante para construir um nó a partir desta, ele procura a melhor característica entre cada subárvore. Isto resulta em uma melhor diversidade que consequentemente implica em um melhor modelo. Após construir as subárvores, o modelo faz a predição para um determinado dado em cada uma das subárvores e através do sistema de votação, ele faz a inferência da classificação do dado analisado.

A implementação utilizada no projeto foi a definida na biblioteca *scikit-learn*. Como parametrização, foi utilizado como critério (*criterion*) de medida da qualidade da divisão dos dados o atributo *entropy*. Este atributo busca a menor entropia possível entre os dados para obter o maior ganho de informação. Neste sentido, quanto maior o ganho de informação, mais relevante aquele atributo deve ser considerado para estar mais ao topo da subárvore. Outro parâmetro utilizado foi *max_depth*, este atributo define o valor limite da profundidade de uma árvore. Após vários experimentos, o valor mais adequado foi o valor 5. Por fim, foi utilizado o parâmetro *random_state*, este por sua vez controla a aleatoriedade das amostras utilizadas para construir as subárvores.

Além disso, para fazer a validação do modelo, foi utilizada a técnica cross-validation. O número total de folhas utilizadas foi o valor recomendado na maioria das literaturas, ou seja, o valor 10. Por sua vez, as métricas utilizadas para avaliar o modelo foram acurácia, precisão, sensibilidade e f1-score. Neste caso,

para cada uma das métricas, o modelo foi treinado 10 vezes, e para obter o valor da métrica, foi feita uma média dos valores obtidos em cada um dos treinos realizados.

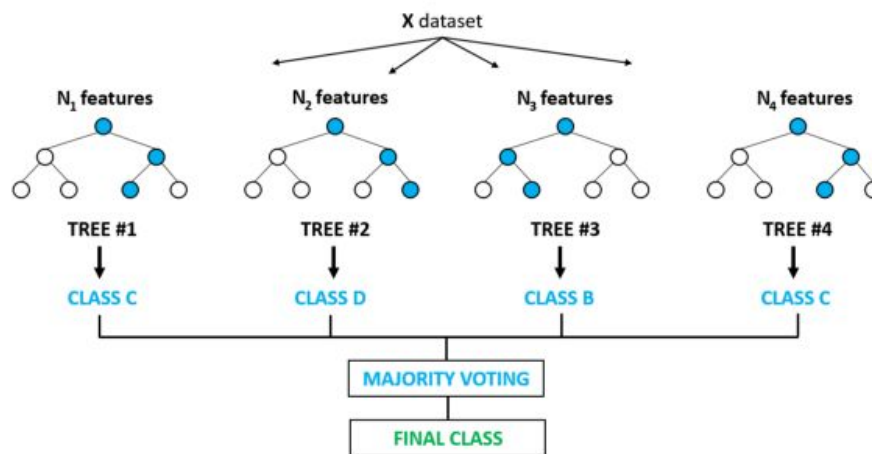


Figura 1. Classificação por Random Forest (Fonte: <https://cutt.ly/La2Orfg>)

K-ésimo vizinho mais próximos (KNN - *K-nearest neighbor*)

O KNN é um dos algoritmos de aprendizagem supervisionada utilizado no campo de mineração de dados e aprendizagem de máquina. Trata-se de um classificador onde o aprendizado toma como base o grau de similaridade de um dado em relação aos demais (vizinhos). O treinamento é formado por vetores de n dimensões. O algoritmo consiste de quatro etapas: (i) Recebe um dado não classificado; (ii) Mede a distância do novo dado em relação a todos os outros dados já classificados, utilizando distância Euclidiana, Manhattan, Minkowski ou ponderada (no nosso caso, o *scikit-learn* adota a distância Euclidiana por *default*); (iii) Identifica as X variáveis de menor distância em relação ao dado analisado, onde $X = K$ definido como o parâmetro do nº de vizinhos mais próximos; (iv) verifica e contabiliza a classe de cada um dos K vizinhos (dados) que estão na menor distância, a classe que for majoritária será o resultado da classificação do novo dado.

A imagem abaixo ilustra o processo de classificação do KNN, onde percebemos que o K adotado influencia significativamente no resultado e desempenho do algoritmo ($K = 3$ implica em classificação B, e $K = 6$ implica em

classificação A). Em relação a definição do número de K, é recomendável sempre utilizar valores ímpares/primos, mas o valor ótimo varia de acordo com a base de dados. Dependendo do problema, pode-se utilizar um algoritmo de otimização (PSO, GA, DE etc), ou definir o número empiricamente.

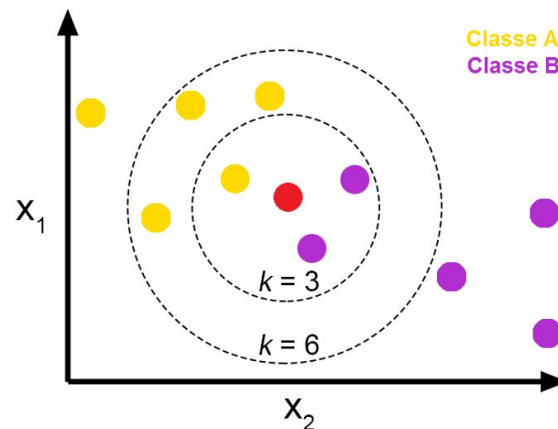


Figura 2. Classificação por KNN (Fonte: <https://cutt.ly/GgtkGsx>)

A scikit-learn é uma biblioteca de aprendizado de máquina de código aberto para a linguagem de programação Python. Dentro do ferramental provido por ela, utilizamos no KNN a função `KNeighborsClassifier`, e as métricas adotadas foram `f1_score`, `precision_score`, `recall_score` e `accuracy_score`. Após treino, teste e validação dos dados preditos na classificação, percebeu-se que o valor adotado por K que obteve melhores resultados foi 7, ou seja, avaliando os 7 vizinhos mais próximos para então classificar o dado analisado.

Em nosso cenário, passamos como parâmetro para o algoritmo a base de dados pré-processada, com 22 atributos, entre idade e sequência de sintomas, onde serão classificadas como não óbito [0] ou óbito [1] pelo KNN. Aplicamos a função `neigh.fit` para treinar nossa base. Em seguida, utilizamos a função de predição `neigh.predict` em nossa base de teste. Posteriormente verificamos se os dados preditos refletem o valor esperado, isto é, se aqueles que foram classificados como óbito, de fato eram óbito na base (o mesmo para o oposto). E avaliamos o algoritmo com as métricas de *accuracy*, *precision*, *recall* e *F1 score*.

Perceptron Multicamadas (MLP - *Multilayer Perceptron*)

Multilayer Perceptron é uma classe de redes neurais artificiais que possuem três ou mais camadas de nós totalmente conectados (cada nó de uma camada é conectado a todos os nós da camada seguinte) que implementam uma combinação linear. A saída de cada nó passa por uma função específica chamada de função de ativação, que mapeia os pesos do nó para uma saída numérica.

O aprendizado de um MLP é do tipo supervisionado e se dá por um processo chamado de *Backpropagation*, onde o valor estimado pela rede é comparado com o valor esperado a partir de uma função de perda que representa o nível de erro da saída. Os pesos dos nós são então ajustados através do algoritmo de gradiente descendente de acordo com o resultado da função de perda e de hiperparâmetros como a taxa de aprendizagem. O objetivo final da rede é encontrar o conjunto de pesos de cada nó que minimize a função de perda. Atualmente, essa atualização dos pesos é feita por algoritmos mais complexos conhecidos como otimizadores.

Neste trabalho foi montada um MLP de 4 camadas utilizando da linguagem de programação Python e da biblioteca PyTorch. Como podemos observar na Tabela 1, as camadas estão dispostas da seguinte maneira:

Camada	Quantidade de Nós	Função de ativação
Oculto 1	50	Relu
Oculto 2	25	Relu
Oculto 3	10	Relu
Saída	1	Sigmóide

Tabela 1. Disposição das Camadas do Multilayer Perceptron

A função de perda utilizada foi a Entropia Cruzada Binária e o otimizador ADAMW. A taxa de aprendizagem teve valor inicial de 0.001 e foi ajustada quando ocorria um *plateau* no valor da perda da rede na etapa de validação. Nesse caso a nova taxa de aprendizagem era definida como $lr = 0.1 * lr$. O processo de treinamento da rede durou 100 épocas, sendo as métricas de teste utilizadas *f1_score*, *precision_score*, *recall_score* e *accuracy_score* da biblioteca *scikit-learn*.

RESULTADOS

Após efetuar o pré-processamento dos dados, foram aplicados os algoritmos definidos na proposta deste trabalho para obter a predição do risco de um paciente vir a óbito ou não de acordo com o sintomas apresentados pelo mesmo. Após obter as predições a partir da aplicação de um modelo de aprendizado de máquina sobre um conjunto de dados, é necessário fazer a validação dos resultados obtidos através de um conjunto de métricas que são capazes de fazer a avaliação das predições geradas pelo modelo.

As métricas adotadas para avaliar o desempenho dos algoritmos foram: (i) *accuracy* ou acurácia, uma das métricas mais simples que verifica a porcentagem de acerto de um classificador. Se de 100 amostras de produtos um classificador conseguir classificar corretamente 80 produtos, teremos uma acurácia de 80%; (ii) *precision* - o coeficiente de precisão é o número de verdadeiros positivos (i.e. o número de itens corretamente rotulados como pertencentes aos positivos), dividido pelo número total de elementos identificados como pertencentes ao conjunto positivo (i.e. a soma de verdadeiros positivos e falsos positivos, que são itens incorretamente rotulados como pertencente ao conjunto); (iii) *recall* - Também chamado de sensibilidade ou revocação, expressa dentre todas as situações de classe Positivo como valor esperado, quantas estão corretas; (iv) F1: média harmônica entre precisão e recall.

Além disso, é necessário fazer a validação dos modelos de aprendizado utilizados, pois através desta validação é capaz de avaliar se a fase de treinamento do mesmo foi bem definida. Este tipo de validação serve para evitar o *overfitting* e *underfitting* dos modelos preditivos ou classificadores, garantindo que estes generalizem o máximo possível as predições realizadas. A técnica utilizada para avaliar os classificadores foi a validação cruzada utilizando o método k-fold, sendo $k = 10$ (*10-fold cross validation*). O método de validação cruzada denominado k-fold consiste em dividir o modelo em k subconjuntos mutuamente exclusivos de mesmo tamanho e, a partir disto, um subconjunto é reservado para teste, enquanto que os k-1 restantes são utilizados para treinamento. Este processo é realizado k vezes,

Algoritmo	Acurácia	Precision	Recall	F1-Score
Multilayer Perceptron	0,84	0,83	0,86	0,84
Random Forest	0.80	0,80	0,82	0,81
K-nearest neighbor	0,79	0,80	0,78	0,83

Tabela 2. Performance dos algoritmos utilizados

alternando o subconjunto de teste. A cada iteração são calculadas as métricas de avaliação para o conjunto de teste da vez. No final dos experimentos de treinamento e teste, a média das métricas é calculada para descrever o desempenho geral do classificador. A Tabela 2 demonstra os resultados obtidos sobre a aplicação dos modelos de aprendizagem utilizados neste trabalho sobre os dados definidos e as métricas de avaliação usadas.

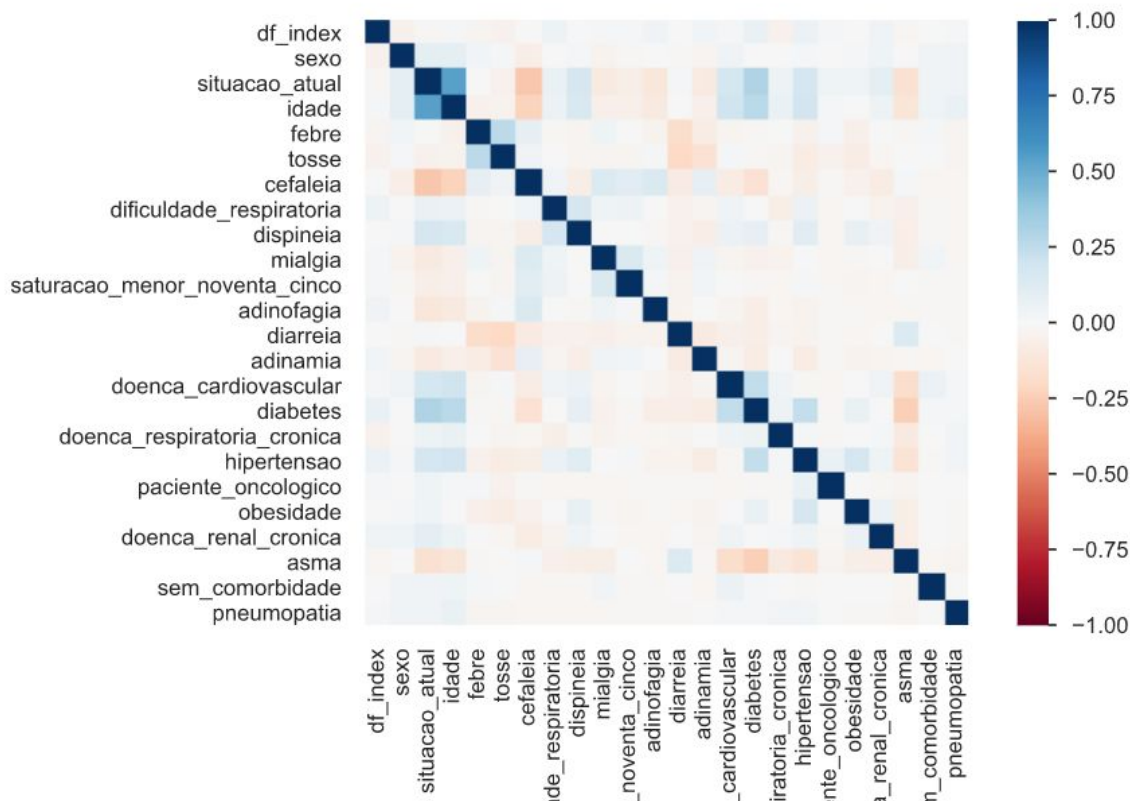
DISCUSSÃO

Esta seção interpreta os principais achados deste artigo. Os resultados indicaram duas questões relevantes: (i) correlação entre a idade e o risco de óbito do paciente; e (ii) a relevância do MLP para predição de óbito por COVID19.

Na questão (i), observou-se que os pacientes que estão situados numa faixa etária mais avançada tendem a ter uma taxa de mortalidade maior quando comparada com pacientes de outras faixas etárias. Esta alta taxa de mortalidade nas pessoas da terceira idade acontece devido ao comprometimento significativo de seus sistemas respiratórios. Portanto, o vírus ataca principalmente este sistema biológico, sendo os sintomas mais nocivos à saúde [4]. Além disso, outro fator que facilita a vulnerabilidade dos idosos é a presença de comorbidade que este grupo apresenta, geralmente com diversas complicações nos organismos.

Na questão (ii), o modelo de aprendizado MLP se mostrou superior ao desempenho dos outros algoritmos. Em primeiro lugar, isto se deve a robustez deste algoritmo, por ser um algoritmo complexo capaz de trazer melhores resultados em relação a outros algoritmos mais simples. Em segundo a lugar, o

algoritmo em questão consegue calcular os melhores pesos para os atributos utilizados, consequentemente os resultados são melhores que outros algoritmos,



embora perde-se um pouco da explicabilidade de como a atribuição da significância das propriedades dos dados são utilizados na predição.

CONCLUSÃO

Este artigo apresentou uma avaliação de desempenho de algoritmos de ML para prever o risco de óbito de um paciente com COVID19. Foi elaborado um experimento incluindo três algoritmos de ML: (i) *Random Forest*; (ii) *K-Nearest Neighbor*; e (iii) *Multilayer Perceptron*. Para avaliar o desempenho destes algoritmos na predição do risco de óbito, escolheu-se três métricas para a análise dos algoritmos de aprendizado de máquina: (i) a acurácia; (ii) a precisão; e (iii) a sensibilidade. Um experimento foi desenvolvido para testar o desempenho dos algoritmos na predição do risco de óbito, utilizando a ferramenta scikit-learn e a linguagem Python na elaboração desta validação.

No momento, conseguimos prever a letalidade do vírus em um paciente dado um conjunto de informações contendo sexo, idade e sintomas apresentados pelo mesmo. Como trabalhos futuros, sugere-se outras abordagens que identifiquem os sintomas que poderiam ter maior peso em prever o risco de óbito. Outro trabalho futuro poderia ser investigar a aplicação de outros algoritmos que possam apresentar um desempenho superior aos aqui apresentados.

REFERÊNCIAS

1. ALAGOAS EM DADOS E INFORMAÇÕES: [CSV] Dados Abertos de Covid-19 em Alagoas. Disponível em: <http://www.dados.al.gov.br/dataset/painel-covid19-alagoas/>. Acesso em 16 de julho de 2020.
2. SCIKIT-LEARN: Machine Learning in Python. Disponível em: <https://scikit-learn.org/stable/index.html>. Acesso em 21 de julho de 2020.
3. Repositório com códigos: Github - projeto-tde. Disponível em: <https://github.com/maxwellacioli/projeto-tde>. Acesso em 22 de julho de 2020.
4. PANDEMIA DA COVID-19 E IDOSOS COMO POPULAÇÃO DE RISCO: ASPECTOS PARA EDUCAÇÃO EM SAÚDE. Disponível em: <http://revistas.ufpr.br/cogitare/article/view/73307>. Acesso em 30 de setembro de 2020