Experienciando Aprendizado de Máquina com o problema do Titanic

Experiencing Machine Learning with the Titanic Problem

Thiago Berto Minson*, Rafael Gomes Mantovani†

RESUMO

Com o desenvolvimento crescente das tecnologias e constante aumento na geração de informação, cada vez mais se vê necessário a presença de métodos capazes de organizar, pré-processar e interpretar dados. O amplo crescimento destes setores cria situações em que certos padrões podem ser observados, e podem ser identificados por máquinas por meio de processos de Inteligência Artificial, com o propósito de fornecer os melhores resultados em determinadas atividades. É aqui que se destaca a sub-área do Aprendizado de Máquina, cuja demanda se torna cada vez maior para otimizar o tempo de solução de diversos problemas, dos cotidianos aos mais complexos e isolados. O objetivo deste trabalho é o desenvolvimento de uma solução capaz de prever quem sobrevive ao naufrágio do navio de passageiros do *Titanic*, base de dados pública, por meio de bibliotecas da linguagem *Python* e o algoritmo de Árvore de Decisão. Nos experimentos, foram realizados pré-processamento de dados para a indução do modelo. Os resultados mostraram que a acurácia média dos modelos de treino são próximas do ideal, enquanto os dados de teste demonstram uma diminuição mas ainda uma acurácia de 0,81.

Palavras-chave: Aprendizado de Máquina, Classificação de Dados, Titanic

ABSTRACT

With the growing development of technologies and constant increase in the generation of information, it is increasingly necessary to have methods capable of organizing, pre-processing and interpreting data. The wide growth of these sectors creates situations in which certain patterns can be observed, and identified by machines through Artificial Intelligence processes, with the purpose of providing the best results in certain activities. It is here that the Machine Learning sub-area stands out, whose demand is increasing to optimize the time needed to solve various problems, from everyday to the most complex and isolated. The objective of this work is the development of a solution capable of predicting who survived the sinking of the passenger ship of the Titanic, a public database, through Python language libraries and the Decision Tree algorithm. In the experiments, data pre-processing was performed for the induction of the model. The results showed that the average accuracy of the training models are close to ideal, while the test data demonstrate a decrease but still an accuracy of 0.81.

Keywords: Machine Learning, Data Classification, Titanic

1 INTRODUÇÃO

Com os avanços frequentes das tecnologias e integração dos sistemas de informação, cada vez mais soluções são desenvolvidas diariamente para lidar com os problemas, dos mais simples aos complexos. Dentre essas maneiras de resolução, uma das mais conhecidas é a Inteligência Artificial (IA) (RUSSEL; NOVIG, 2010), que está associada, geralmente, ao desenvolvimento de sistemas inteligentes. Estes sistemas

*Engenharia de Computação, Universidade Tecnológica Federal do Paraná, Campus Apucarana; thiagominson@alunos.utfpr.edu.br , rafaelmantovani@utfpr.edu.br

baseados em conhecimento, construídos principalmente com regras que reproduzem o conhecimento do perito, são utilizados para solucionar determinados problemas em domínios específicos (MENDES, 1997). Estes sistemas permitem atribuir a máquinas funções automáticas baseadas em métodos de análise de dados e incorporação de comandos (LUGER, 2013). Empresas como a Google, a Amazon, a Microsoft e a Apple, têm usado cada vez mais essa forma para entregar funcionalidades como reconhecimento de conteúdo mais acessado, rotina diária, assistente virtual pessoal, reconhecimento espacial para melhoria na resolução de imagens e vídeos, dentre muitos outros.

Na verdade, uma subárea de IA tem crescido nos últimos anos, o Aprendizado de Máquina (AM) (MITCHELL, 1997). Sistemas de AM são preparados internamente para responder principalmente a dois tipos de aprendizado. Algoritmos de AM permitem a identificação de padrões com base em casos e experimentos anteriores, assim como ocorre com a inteligência humana (ERICKSON et al. 2017). Por meio deles podemos considerar que a máquina literalmente analisa informações para devolver uma resposta baseada em uma execução contínua de treino, gerando um "aprendizado".

Ainda sobre algoritmos de AM, podemos classificá-los em: algoritmos supervisionados, que são capazes de analisar dados entregues previamente e já empacotados e rotulados para criar uma função que faça a melhor análise; e algoritmos não-supervisionados, que não possuem os rótulos, fazendo a máquina reconhecer os padrões por conta a partir de um banco (MARSLAND, 2009).

Por meio de uma tarefa de aprendizado supervisionado, mais específico uma classificação, aqui será demonstrado o processo de tentativa de resolver um desafio proposto no site *Kaggle*, chamado de "*Titanic – Machine Learning from Disaster"*, cujo objetivo é tentar determinar os sobreviventes do acidente famoso do transatlântico referido a partir de uma base de dados já rotulados.

O objetivo deste trabalho é implementar o uso das técnicas de AM e Mineração de Dados (MD) (MITCHELL, 1997), para fazer uma análise da base de dados do desafio, e, assim, obter uma predição capaz de demonstrar, nestes parâmetros, quais passageiros sobreviveriam ao desastre baseado apenas nas características previamente informadas dentro dos arquivos. Experimentos foram realizados via linguagem de programação *Python*, e biblioteca scikit-learn.

2 MÉTODO

2.1 Conjunto de Dados

O conjunto de dados usado nos experimentos foi o Titanic, disponível em um desafio proposto no site *Kaggle*, chamado de "*Titanic – Machine Learning from Disaster"*[1], cujo objetivo é tentar determinar os sobreviventes do acidente famoso do transatlântico referido a partir de uma base de dados já rotulados. As amostras foram manejadas no formato de tabelas atributo-valor, e oconjunto de dados já é rotulado. Foram obtidos dois arquivos com dados do site, chamados de *train* e *test*. Ambos contém dados relacionados ao acidente ocorrido no ano de 1912, com o navio de passageiros *RMS Titanic*, operado pela *White Star Line*. Dentre suas informações, eles apresentam uma lista com o nome de todos os passageiros, assim como características específicas de cada indivíduo, sendo elas: *PassengerId* (número relacionado a cada passageiro

1 https://towardsdatascience.com/comprehensive-beginners-guide-to-kaggle-titanic-survival-prediction-competition-solution-21c5be2cec2c

na lista); *Pclass* (qual das três classes o passageiro estava adequado); *Name* (nome do passageiro); *Sex* (sexo do passageiro dentre masculino e feminino); *Age* (idade); *SibSp* (a quantia de conjuges e/ou irmãos); *Parch* (quantia de pais e filhos); *Ticket* (o número do bilhete de passagem); *Fare* (as taxas de cada um); *Cabin*, (cabine relacionada ao indivíduo); e *Embarked* (informa o portão de embarque utilizado, dentro de três opções: *Cherbourg*, *Queenstown* e *Southampton*).

A diferença entre os dois arquivos está na presença do atributo *Survived*, que é a classe, ou seja, o atributo que será previsto pelos métodos, para cada membro da amostra. Este atributo está presente apenas no *train*, que será utilizado para construir o modelo de AM, com o objetivo de prever quais serão os resultados desta classe no outro *arquivo*, o *test*. Para a compreensão destes dados dentro dos arquivos, se faz necessária a análise exploratória.

2.2 Árvore de Decisão

O algoritmo utilizado foi a Árvore de Decisão (MARSLAND, 2009). Em teoria, este modelo cria pontos de decisão, também chamados de "nós". Em cada um desses setores, ele irá optar, baseado em seus parâmetros, por seguir em uma de duas direções possíveis, categorizadas como "Sim" ou "Não", baseado em um questionamento feito em cada uma dessas junções. Ao receber os dados, e com base na análise, ele entrega o melhor resultado. A quantidade de perguntas realizadas ao sistema é controlada pela estrutura do próprio código, dentro das funções.

2.3 Setup Experimental

A medida de desempenho utilizada foi a Acurácia, que representa o número de previsões corretas do modelo em questão. Ela é obtida através da fórmula:

$$acurácia = \frac{Total\ de\ acertos}{Total\ de\ itens} \quad (1)$$

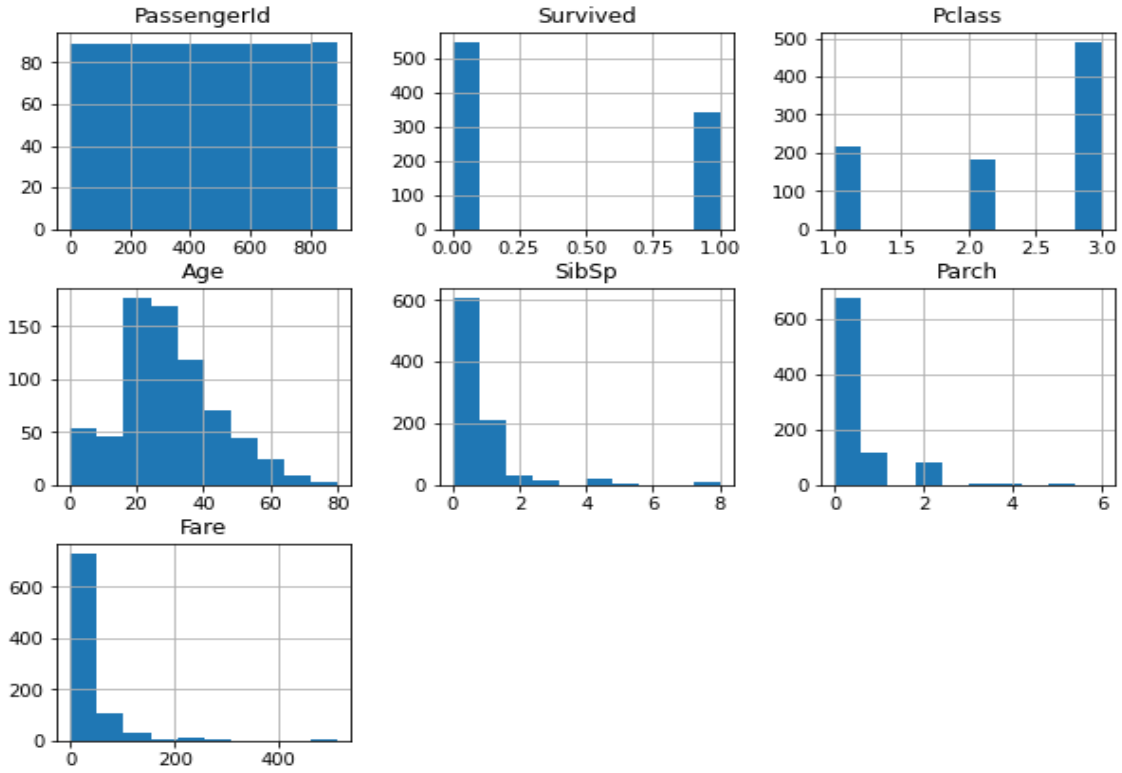
Para programação foi utilizado: o interpretador de Python Spyder 3.8, com as bibliotecas de análise de dados NumPy e Pandas; as de visualização Matplotlib e sua extensão Seaborn; e a biblioteca Scikit Learn para a execução do algoritmo de Árvore de Decisão. O hardware utilizado foi um Intel Core i7 – 4050 com 8GB de Ram e uma placa de vídeo Nvidia GT 750.

3 RESULTADOS

A análise e pré-processamento dos dados foi realizada através da criação e observação de gráficos que comparam seções específicas de informação dentro dos *datasets*. Para isso, foi dividida a função de cada *dataset*, sendo que o *train* seria usado para as instâncias citadas, enquanto o *test* teria seu uso para aplicação do código pronto e verificação da acurácia. O primeiro gráfico gerado na análise exploratória foi um histograma, que demonstra todos os atributos numéricos (que representam números quantitativos), e seus

valores dentro dos intervalos. O gráfico da Figura 1 apresenta uma oscilação entre os dados, em especial no atributo *Age*. Também é possível observar que aproximadamente 560 pessoas não sobreviveram ao acidente dentro destes parâmetros. Estas informações serão utilizadas pelo código para estabelecer padrões no problema.

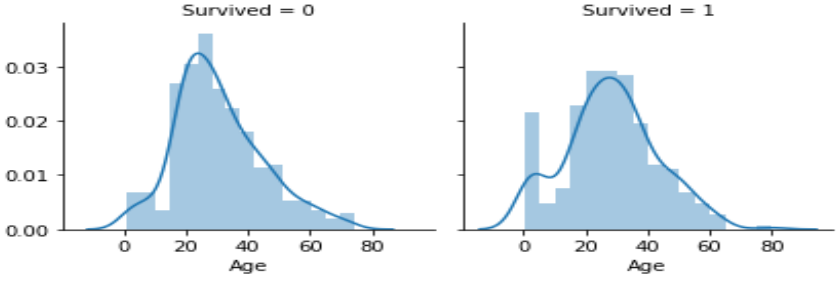
Figura 1 - Relação dos Atributos Numéricos e Seus Valores nos Intervalos no *train*



Fonte: Autoria Própria (2021)

Após a análise do primeiro gráfico, com percepção da oscilação do fator "*Age*", foi feito um gráfico, para verificar a relação da idade com os passageiros da embarcação que sobreviveram ou não. Isso é mostrado na Figura 2. O gráfico nos mostra distribuições iguais entre os passageiros, ou seja, a idade não é um fator tão importante para a previsão final.

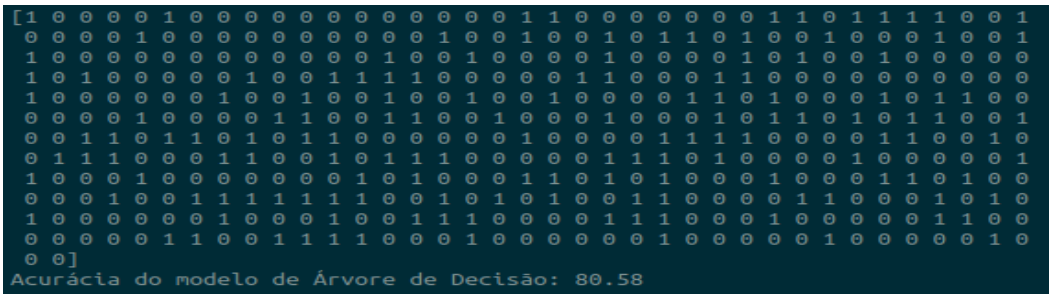
Figura 2 - Relação Entre a Classe e a característica "*Age*" - train



Fonte: Autoria Própria (2021)

Com estes dados e analogias estabelecidas, é possível perceber que os atributos *PassengerID*, *Ticket*, *Name*, *Cabin* e *Embarked* não são relevantes para a situação, pois eles são externos ao que acontece. Isto definiu uma seleção dos atributos que seriam usados diretamente no código. Com a remoção destes atributos o algoritmo de Árvore de Decisão foi estruturado e treinado no conjunto de treinamento (train). As saídas do algoritmo são organizadas por meio de um vetor de números, em ordem de passageiro, usando o conjunto do teste (test) na tentativa de avaliar o modelo induzido. Um exemplo de vetor predito é apresentado na Figura 3.

Figura 3 - Vetor Resultado da Execução e Acurácia Baseada no Aprendizado



Fonte: Autoria Própria (2021)

Após isto, o código foi executado outras 10 vezes. Com base nos resultados, a média aritmética destes foi obtida, de forma a possuir uma tendência central ao qual a acurácia gira. Os resultados são descritos na Tabela 1.

Tabela 1 – Coeficiente de Silhueta obtido pelos Algoritmos de Agrupamento.

Execução	Acc Treino	Acc Teste
1	97.30	80.58
2	96.40	78.68
3	96.18	83.39
4	96.63	80.36
5	96.85	81.59
6	96.63	82.49
7	96.63	81.37
8	96.85	81.82
9	97.30	79.81
10	96.85	81.48
MÉDIA	96.76	81.51

Fonte: Autoria própria (2021).

Desta forma, é possível verificar tanto a acurácia do *dataset train*, usado para a construção do código, que por ser o parâmetro base, sempre estará em torno da perfeição (~100%); quanto o *test*, que se torna a base de dados fundamental para verificar o quão eficiente é o algoritmo, para prever os sobreviventes do naufrágio. Como demonstram os resultados do teste, este modelo possui uma acurácia média de 81,15%, o que o torna um bom algoritmo para previsão dos sobreviventes do naufrágio do *Titanic* dentro deste contexto.

4 CONCLUSÃO

A pesquisa concluiu, que, com base nos dados recebidos e na escolha do algoritmo, o algoritmo de Árvore de Decisão obtém uma performance potencialmente eficiente, para prever um resultado em demanda de dados rotulados, com a presença de uma classe única, de modo a fornecer uma entrega satisfatória de informações relevantes para resolução de problemas análogos. Estes resultados podem ser potencializados pelo uso de hardware mais poderoso em termos de processamento de dados, assim como a exploração mais abrupta dos dados e diversificação dos algoritmos utilizados.

REFERÊNCIAS

ERICKSON, B. J. , KORFIATIS, P. , AKKUS, Z. , et al. **Machine learning for medical imaging**: Radiographics. 2017;37:505-15. Disponível em <https://pubmed.ncbi.nlm.nih.gov/28212054/>, último acesso em Ago 2021.

FERNANDES, G. , MATSUMOTO, F. , COUTINHO, B.; **Modelos de Predição | Decision Tree**. Disponível em<https://medium.com/turing-talks/turing-talks-17-modelos-de-predi%C3%A7%C3%A3o-decision-tree-610aa484cb05>, último acesso em Mai 2021.

LUGER, George F. **Inteligência Artificial** 6° Edição. University of New Mexico, 2013.

MARIANO, D.; **Métricas de avaliação em machine learning**. Disponível em <https://bioinfo.com.br/metricas-de-avaliacao-em-machine-learning-acuracia-sensibilidade-precisao-especificidade-e-f-score/>, último acesso em Ago 2021.

MARSLAND, Stephen. **Machine Learning:** An Algorithmic Perspective. Massey University, Palmerston North, New Zealand, 2009.

MENDES, R. D. **Inteligência Artificial:** Sistemas Especialistas no Gerenciamento da Informação. *In*: Ciência da Informação, v. 26. 1997. Disponível em: <https://doi.org/10.1590/S0100-19651997000100006>, último acesso em Jul 2021.

MITCHELL, Thomas M. **Machine Learning.** McCgraw-Hill, New York, 1997.

RUSSELL, Stuart; NORVIG, Peter. **Inteligência Artificial**. Rio de Janeiro: Elsevier Editora, 2010.