CausalBioCF: Causalidade e otimização bioinspirada para geração de contrafactuais factíveis em tempo real

Gabriel Covello Furlanetto[1], **Alexandro Baldassin**[1], **Aleardo Manacero**[2]

[1]Departamento de Estatística, Matemática Aplicada e Ciência da Computação – Universidade Estadual Paulista (UNESP) – Rio Claro – SP – Brasil

[2]Departamento de Ciência da Computação e Estatística – Universidade Estadual Paulista (UNESP) – São José do Rio Preto – SP – Brasil

{gabriel.furlanetto,alexandro.baldassin,aleardo.manacero}@unesp.br

Abstract. *Machine learning methods have been widely used to support decision-making, but most of the time the decisions cannot be easily explained. Therefore, providing explanations about the results generated by them becomes important. This is particularly relevant in high-risk decision scenarios in order to protect all the participants. In this work we discuss the parallelization of different styles of algorithms to find contrafactual candidates, an important part of the explainability framework.*

Resumo. *Considerando-se que métodos de aprendizado de máquina vem sendo amplamente utilizados para embasar tomadas de decisões, fornecer explicações sobre o resultado gerado por eles torna-se importante. Isso é particularmente valioso em cenários de decisões de alto risco, a fim de proteger todas as partes envolvidas. Neste trabalho é abordada uma parte importante da explicabilidade de aprendizado de máquina, sendo discutida a paralelização de diferentes tipos de algoritmos para encontrar candidatos a contrafactuais.*

1. Introdução

Algoritmos de aprendizado de máquina estão gradativamente sendo utilizados em diversos contextos de tomadas de decisões [Jin 2020]. Suas aplicações encontram-se tanto em cenários de baixo risco para os usuários, como na indicação de conteúdo de mídia [Gomez-Uribe and Hunt 2015], quanto em cenários de elevado risco, como na área médica [Fatima et al. 2017].

Sendo assim, é considerável que as escolhas embasadas pelos dados gerados por esses sistemas sejam confiáveis e permitam a compreensão de como o sistema está fazendo suas previsões, requerendo-se transparência nos algoritmos utilizados. Neste contexto, surge a importância da explicabilidade contrafatual. Esta metodologia trabalha sobre um cenário hipotético, no qual a escolha seria diferente caso certas variáveis de entrada para o modelo tivessem valores diferentes. Assim, a comparação da decisão real (fato) com a hipotética (contrafato) pode ajudar os usuários a entender por que um resultado específico foi alcançado e quais fatores foram mais importantes para esta conclusão [Wachter et al. 2017].

Apesar dos avanços na pesquisa para geração de contrafactuais, existem desafios significativos em relação à eficiência computacional e à qualidade dos resultados.

Há limitações de escalabilidade em muitos dos métodos e isso afeta, principalmente, os cenários de tomada de decisão em tempo real. Neste trabalho, pretende-se encontrar contrafactuais viáveis, de maneira automatizada, por meio do desenvolvimento de uma biblioteca denominada CausalBioCF. A fim de garantir escalabilidade e eficiência computacional do método proposto, serão utilizados algoritmos bioinspirados implementados com estratégias de computação paralela.

2. Motivação e objetivos

Muitos algoritmos de geração contrafactual foram propostos desde o trabalho apresentado por [Wachter et al. 2017]. Dentre seus principais desafios estão [Yang et al. 2021]:

- Ineficiência, uma vez que os algoritmos propostos são dependentes de otimizações e/ou redes neurais profundas, que podem ser computacionalmente custosas;
- Necessidade de aproximação de alta qualidade para uma melhor explicação, o que muitas vezes acaba gerando perturbações não factíveis do dado original;
- Necessidade de levar-se em conta o relacionamento entre variáveis, como a dependência causal entre os atributos, para viabilidade contrafactual;

CausalBioCF propõe um método que contribua para os 3 desafios citados, melhorando a eficiência computacional na obtenção de contrafactuais, por meio de paralelismo em métodos de otimização bioinspirados, escolhidos por sua facilidade de implementação; utilizando análises estatísticas para buscar contrafactuais factíveis por meio de automatização na busca por conhecimento de domínio; e investigando relações entre variáveis por meio de análise de correlação e causalidade.

Estes pilares dão origem ao nome da biblioteca proposta, que utiliza casualidade (**Causal**) e algoritmos bioinspirados paralelizados (**Bio**) na geração de contrafactuais (**CF**). Dentre os diferenciais da proposta estão a obtenção de contrafactuais em um tempo aceitável para apoiar os usuários na análise das tomadas de decisão dos algoritmos de aprendizado de máquina, o que será garantido pela paralelização dos algoritmos bioinspirados, e a automatização da obtenção do conhecimento de domínio, minimizando a interferência humana na geração de contrafactuais. Assim, o projeto busca reduzir lacunas de literatura.

3. Algoritmos e estratégias de paralelização

Uma parte significativa do trabalho é a geração automática de contrafatuais viáveis, utilizando restrições derivadas do conhecimento de domínio. Atualmente, a aplicação dessas restrições e a filtragem dos contrafactuais são realizadas por meio de dados obtidos com especialistas no campo específico, de forma manual, uma tarefa que pretende-se automatizar.

Assim, algoritmos de otimização bioinspirados cujas restrições são obtidas a partir de uma base de dados com o conhecimento de domínio (não discutida neste trabalho), geram contrafactuais que posteriormente podem ser usados para gerar a explicação do modelo (outro ponto não discutido aqui). A geração dos contrafactuais e os algoritmos bioinspirados, alvos deste texto, possuem em sua essência uma implementação sequencial, com alta dependência entre suas iterações. Em particular, estamos usando um Algoritmo Genético (Genetic Algorithm - GA) [Holland 1992] e um Algoritmo de Enxame

de Abelhas (Artificial Bee Colony - ABC) [Karaboga et al. 2005]. Entre eles, o primeiro é o mais utilizado na literatura de geração de contrafactuais que utilizam técnicas de otimização. Embora o segundo não seja frequentemente utilizado, tende a ser mais eficiente e também mais preciso do ponto de vista computacional, motivo que nos levou a considerá-lo [Muthiah and Rajkumar 2014].

Algoritmos genéticos possuem uma estrutura de processamento em gerações, onde uma geração somente pode ser processada após a anterior ter sido gerada. Isso limita o ganho com paralelismo, pois em geral não é possível executar as gerações em paralelo. No entanto, ainda assim há algumas alternativas de paralelização. Dentre elas, [Cantú-Paz et al. 1998] enumera três tipos principais de algoritmos genéticos paralelos, utilizando para isso métodos de paralelização de dados. Neste trabalho, será utilizada a denominada pelo autor como metodologia Global de Única População. Nela, uma ou mais etapas do GA (avaliação, seleção, cruzamento e mutação) podem ser executadas em paralelo, sendo que em qualquer uma delas, será utilizada a população inteira gerada. Para sua implementação é utilizada o paradigma mestre-escravo, em que o mestre realiza o controle de maneira sequencial enquanto os escravos executam as atividades em paralelo.

O algoritmo ABC é uma técnica de otimização que utiliza o comportamento de busca de alimentos das abelhas para guiar a exploração do espaço de soluções em busca da melhor solução possível para um dado problema. Suas etapas compreendem, resumidamente, a exploração de vizinhança, por cada abelha, para encontrar soluções potencialmente melhores; a avaliação de aptidão, determinando o quão boa é cada solução em relação ao problema trabalhado; e a seleção/atualização em que são escolhidas as abelhas mais promissoras para continuar a busca. Utilizaremos como estratégia inicial para paralelização deste algoritmo a proposta descrita em [Narasimhan 2009], que envolve dividir igualmente as abelhas entre os processadores disponíveis, deixando apenas um deles livre para também manter uma cópia completa dos dados e operar como orquestrador. Durante cada ciclo, as abelhas de cada processador buscam melhorar suas soluções locais. Ao final deste ciclo, as soluções são coletadas pelo orquestrador sendo organizadas, substituindo as originais, caso sejam melhores.

Como critério de parada, ambos os algoritmos utilizarão o número de gerações e o critério de estagnação de soluções, ou seja, após N gerações em que o algoritmo não encontrar uma aptidão melhor, o algoritmo será encerrado.

4. Estágio atual da implementação

Neste momento, o trabalho encontra-se na etapa de desenvolvimento e, para isto, estão sendo utilizados como recursos tecnológicos a linguagem Python em sua versão 3.9 e o conjunto de dados *Adult*, inicialmente utilizado como base para testes [1].

Os algoritmos sequenciais (ABC e GA) já estão implementados e para o conjunto de dados trabalhado, estão com duração média aproximada de execução em 30 segundos (ABC) e 2 minutos e 30 segundos (GA). Espera-se que com as soluções paralelas dos algoritmos implementadas, conforme proposto, estes tempos sejam reduzidos, entregando o resultado ao analista mais rapidamente. Com isto, eles poderão ter insumos para realizar as ações necessárias em cada cenário.

[1] https://archive.ics.uci.edu/dataset/2/adult

Como estamos utilizando a linguagem Python, estamos prevendo uma dificuldade inicial para a parelização devido ao *Global Interpreter Lock* (GIL). Essa trava do interpretador limita o número de threads que podem ser usadas ao mesmo tempo e, consequentemente, a melhoria do desempenho. Uma ideia inicial seria utilizarmos a biblioteca de multiprocessamento. Ao contrário do esquema que utiliza multithreading, com o multiprocessamento vários processos são criados e se comunicam através de mensagens que pode ser enviadas e recebidas por meio dos *Pipes* em Python. Uma outra alternativa seria utilizar um interpretador que não tenha a limitação do GIL, como o Jython ou o IronPython. Consideraremos ainda a paralelização usando a própria linguagem C e integrando o binário no framework com o Python. Ademais, vemos a oportunidade de participar do ERAD como um local onde poderemos discutir outras estratégias para contornar o problema.

5. Conclusão

A partir do trabalho proposto, deseja-se obter um maior número de explicações contrafactuais de qualidade, em um período de tempo aceitável, tanto em comparação com as versões sequenciais dos algoritmos GA e ABC, quanto em relação às redes neurais adversárias (Generative Adversarial Networks - GAN), técnica também utilizada na literatura para geração de contrafactuais.

Referências

Cantú-Paz, E. et al. (1998). A survey of parallel genetic algorithms. *Calculateurs paralleles, reseaux et systems repartis*, 10(2):141–171.

Fatima, M., Pasha, M., et al. (2017). Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, 9(01):1.

Gomez-Uribe, C. A. and Hunt, N. (2015). The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)*, 6(4):1–19.

Holland, J. H. (1992). Genetic algorithms. *Scientific american*, 267(1):66–73.

Jin, W. (2020). Research on machine learning and its algorithms and development. In *Journal of Physics: Conference Series*, volume 1544, page 012003. IOP Publishing.

Karaboga, D. et al. (2005). An idea based on honey bee swarm for numerical optimization. Technical report, Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.

Muthiah, A. and Rajkumar, R. (2014). A comparison of artificial bee colony algorithm and genetic algorithm to minimize the makespan for job shop scheduling. *Procedia Engineering*, 97:1745–1754.

Narasimhan, H. (2009). Parallel artificial bee colony (pabc) algorithm. In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pages 306–311. IEEE.

Wachter, S., Mittelstadt, B., and Russell, C. (2017). Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841.

Yang, F., Alva, S. S., Chen, J., and Hu, X. (2021). Model-based counterfactual synthesizer for interpretation. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 1964–1974.