Imputação de Dados via Programação Genética: Uma Estratégia de Ensemble

Giovana Assis da Matta Machado
Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
giovana.assis@dcc.ufmg.br

Orientador: Gisele Lobo Pappa
Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
glpappa@dcc.ufmg.br

***Abstract*—A presença de dados ausentes é um problema presente em tarefas de mineração de dados e aprendizado de máquina, podendo introduzir vieses significativos e comprometer o desempenho de modelos preditivos. Embora existam diversas técnicas de imputação, desde substituições estatísticas simples até modelagens multivariadas complexas, muitas falham em capturar as especificidades de cada atributo ou em modelar relações não-lineares complexas entre as variáveis. Este trabalho propõe o GP-Imputer, uma abordagem evolutiva baseada em Programação Genética (GP) que opera sob uma estratégia de ensemble. Diferentemente de métodos que geram valores diretamente, o GP-Imputer evolui funções de combinação não-linear ótimas a partir das saídas de imputadores base consolidados (Média, Mediana, KNN, MICE e SVD). O método adota uma representação multi-tree para tratar a dimensionalidade dos dados e utiliza uma abordagem orientada a tarefa final, onde a aptidão dos indivíduos é guiada pelo F1-Score de um classificador (Regressão Logística). A validação experimental, conduzida em cinco conjuntos de dados de referência sob mecanismo de ausência MCAR com taxas de 10% a 30%, demonstrou a superioridade do método proposto. O GP-Imputer obteve o melhor desempenho médio em 14 dos 15 cenários avaliados, apresentando ganhos estatisticamente significativos frente aos métodos do estado da arte, especialmente em contextos de alta dimensionalidade e escassez de informações**

***Index Terms*—Imputação de Dados, Programação Genética, Ensemble de Imputadores, Dados Ausentes, Computação Evolutiva, Classificação.**

I. Introdução

A mineração de dados tem como propósito transformar dados obtidos de diferentes fontes em conhecimento útil. Contudo, esses dados frequentemente apresentam problemas de qualidade, como ruídos, valores extremos, distribuições desbalanceadas e dados faltantes. Entre essas questões, a presença de valores ausentes é especialmente comum e, muitas vezes, inevitável. Os dados nulos surgem por diversos motivos, incluindo falhas no registro das informações, problemas técnicos em sistemas ou sensores, ausência de respostas por parte dos participantes, erros durante a coleta ou ainda restrições relacionadas à privacidade [1].

A falta de dados representa um desafio comum que pode comprometer a eficiência dos processos de aprendizado [2]. Muitos algoritmos de mineração de dados e modelos de inteligência artificial não funcionam adequadamente quando os conjuntos de dados estão incompletos, o que limita a precisão das análises. Por esse motivo, é essencial lidar com os valores ausentes nas primeiras etapas do processo analítico, seja eliminando os dados ou realizando a imputação. A forma escolhida para tratar esse problema pode impactar diretamente o desempenho dos modelos desenvolvidos e a confiabilidade das conclusões obtidas a partir dos dados [3].

A literatura apresenta uma ampla variedade de métodos propostos para lidar com observações ausentes, que variam em complexidade, custo computacional e efetividade. Os métodos mais simples consistem na remoção das instâncias ou dos atributos que contêm valores ausentes, ou na substituição desses valores por estatísticas básicas, como mínimo, máximo, média, moda ou mediana. Embora essas abordagens sejam de fácil implementação, elas podem introduzir vieses nos dados ou levar à perda significativa de informação, comprometendo a qualidade da análise [4].

Por outro lado, métodos mais complexos utilizam técnicas de aprendizado de máquina para realizar a imputação. Entre esses, destacam-se algoritmos como k-Nearest Neighbors (KNN), que estima valores ausentes com base em instâncias similares, e árvores de decisão, que utilizam relações entre atributos para prever os valores faltantes. Existem também técnicas mais avançadas, como florestas aleatórias (Random Forests), redes neurais e modelos baseados em aprendizado profundo, que apresentam maior custo computacional.

Grande parte desses métodos tratam os atributos de forma homogênea, aplicando uma única estratégia de imputação para todo o conjunto de dados. Essa abordagem desconsidera as características de cada atributo, como distribuição, variabilidade e relação com outras variáveis, o que pode resultar em imputações imprecisas e, consequentemente, em modelos preditivos menos eficazes. Para lidar com esse problema, este trabalho propõe o desenvolvimento de um método de imputação utilizando programação genética especializada para cada atributo. Com essa abordagem, busca-se gerar imputações mais precisas e consistentes, reduzindo o viés introduzido pelo processo de preenchimento dos valores ausentes e, consequentemente, aprimorando a qualidade dos dados.

O restante deste documento está organizado em cinco seções. A Seção 2 apresenta a fundamentação teórica e os trabalhos relacionados, discutindo os mecanismos de ausência de dados e o estado da arte em imputação via computação evolutiva. A Seção 3 detalha a metodologia proposta, o GP-

Imputer, descrevendo a representação dos indivíduos, a função de aptidão baseada em classificador e os operadores genéticos especializados. A Seção 4 descreve a configuração experimental, incluindo as métricas de avaliação e ajuste dos métodos de referência, seguido pela análise e discussão dos resultados obtidos. Por fim, a Seção 5 apresenta as conclusões e aponta direções para trabalhos futuros."

II. Trabalhos Relacionados

A compreensão e o tratamento de dados ausentes têm sido amplamente estudados na literatura, uma vez que a presença desses valores pode comprometer a qualidade das análises e a performance de modelos preditivos. Esta seção revisa trabalhos anteriores em dois pontos principais: os diferentes mecanismos que explicam a ausência de dados e abordagens que utilizam programação genética como estratégia para imputação.

A. Mecanismos de Ausência de Dados

De acordo com a taxonomia clássica proposta por Little e Rubin [5], os mecanismos que explicam por que os dados estão ausentes podem ser divididos em três categorias: MCAR, MAR e MNAR. A identificação do mecanismo correto é fundamental, pois ele influencia a escolha do método de imputação apropriado e a validade das inferências estatísticas subsequentes.

Missing Completely At Random (MCAR): Nesse cenário, a ausência de dados ocorre de forma completamente aleatória, ou seja, a probabilidade de um valor estar ausente é independente tanto dos valores observados quanto dos não observados no conjunto de dados. Por exemplo, se um paciente esqueceu de relatar sua idade por acaso, sem qualquer relação com sua condição clínica ou outras variáveis, os dados seriam considerados MCAR. Esse é o cenário ideal para análise estatística, pois os dados ausentes não introduzem viés, apenas reduzem o poder estatístico da amostra [6]. No entanto, esse mecanismo é raro em aplicações práticas.

Missing At Random (MAR): Neste caso, a ausência de um dado depende apenas de informações observadas no conjunto de dados, e não do próprio valor ausente. Por exemplo, se pessoas do sexo masculino tendem a omitir a renda, mas a renda em si não influencia a probabilidade de ausência (condicionalmente ao sexo), os dados são considerados MAR. Muitos métodos de imputação, como o *Multiple Imputation by Chained Equations* (MICE) [7], assumem esse tipo de mecanismo. Embora mais realista que MCAR, o cenário MAR ainda requer cuidados na modelagem, pois pressupõe que as variáveis relevantes associadas à ausência foram corretamente observadas e incluídas no modelo.

Missing Not At Random (MNAR): Trata-se do mecanismo mais complexo, no qual a ausência de dados está relacionada ao próprio valor ausente, mesmo após condicionar as variáveis observadas. Por exemplo, pessoas com renda mais alta podem estar menos dispostos a informar esse dado, e justamente o valor da renda influencia sua probabilidade de estar ausente. Nesse caso, os dados ausentes carregam informações importantes e o padrão de ausência é informativo. A imputação sob MNAR exige modelos específicos que incorporem suposições fortes ou fontes de dados adicionais, sendo comum o uso de modelos baseados em inferência bayesiana [8].

B. Métodos Baseados em Programação Genética

O algoritmo GPMI (Multiple Imputation using Genetic Programming) [9] utiliza a programação genética como técnica de regressão não paramétrica para imputação múltipla. Ao regressar iterativamente atributos ausentes em função dos demais, o método descobre a estrutura do modelo sem suposições prévias sobre a distribuição dos dados. Comparado a técnicas tradicionais e métodos avançados como Random Forests, o GPMI demonstrou superioridade na captura de relações não lineares, resultando em menor erro de predição (NRMSE) e maior acurácia de classificação nos dados imputados.

Visando eficiência computacional em classificação, o método GPI (Genetic Programming-based Imputation) [10] foi desenvolvido para contornar os altos custos de teste observados em abordagens como o MICE. O GPI evolui conjuntos de funções de regressão para cada atributo ainda durante o treinamento e, na fase de aplicação, seleciona dinamicamente a função mais apta e com menor dependência de outros atributos ausentes. Essa abordagem mostrou-se cerca de mil vezes mais rápida que o MICE, mantendo acurácia de classificação comparável ou superior.

Posteriormente, uma abordagem híbrida [11] foi proposta para eliminar a necessidade de imputação durante a fase de teste. O método utiliza imputação múltipla para preparar o conjunto de treinamento e identifica padrões de ausência, empregando programação genética para evoluir classificadores específicos para esses padrões. Ao classificar novas instâncias utilizando apenas os modelos que não dependem dos dados faltantes daquele registro, a técnica obteve acurácia superior e tempos de processamento menores do que as abordagens que realizam imputação em tempo real.

Mais recentemente, o modelo WKNN-GP [12] foi apresentado com foco em regressão simbólica com dados incompletos. Para evitar a reconstrução de modelos a cada nova instância, propôs-se uma arquitetura que combina a seleção de instâncias via Weighted KNN com a capacidade preditiva da programação genética baseada em features. O método permite reutilizar modelos evoluídos para imputar novas instâncias desconhecidas, reduzindo drasticamente o custo computacional e superando técnicas como KNN, CART e Random Forests na precisão da imputação.

Apesar desses avanços, a dependência de uma única técnica subjacente para regredir valores ausentes pode limitar a modelagem da complexidade dos dados. Motivada por essa lacuna, esta proposta sugere uma mudança de paradigma: em vez de evoluir o imputador diretamente, a Programação Genética é empregada para descobrir uma função de combinação não linear entre um conjunto diversificado de imputadores existentes (como KNN, MICE e média). Essa estratégia de ensemble busca superar as limitações de métodos isolados, visando imputações mais robustas para os modelos analíticos subsequentes.

III. Metodologia

Esta seção descreve a proposta do método GP-Imputer, detalhando sua representação, mecanismos de avaliação e operadores genéticos.

A. Visão Geral do Algoritmo

A abordagem proposta, denominada GP-Imputer, fundamenta-se no uso de Programação Genética (GP) para evoluir funções de imputação compostas, operando sob uma estratégia de ensemble. Diferentemente dos métodos tradicionais que aplicam uma técnica única de estimativa, o GP-Imputer busca descobrir uma combinação não linear ótima das saídas geradas por diversos imputadores base, especificamente Média, Mediana, KNN, MICE e SVD.

O ciclo evolutivo, ilustrado na Fig. 1, inicia-se com métodos padrão, segue com a avaliação baseada em maximização de performance (F1-Score), aplica seleção e variações genéticas, preservando sempre os melhores indivíduos através do elitismo.

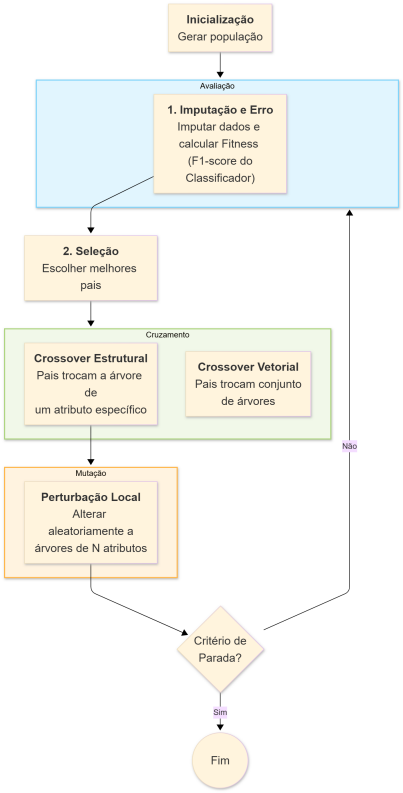

Fig. 1. Fluxograma ilustrando o processo iterativo de imputação de dados utilizando Programação Genética.

B. Representação do Indivíduo

Para lidar com a dimensionalidade dos dados, a representação dos indivíduos na população adota uma estrutura *multi-tree* (múltiplas árvores), onde cada árvore de sintaxe é responsável pela imputação de um atributo específico que contenha valores ausentes (Fig. 2).

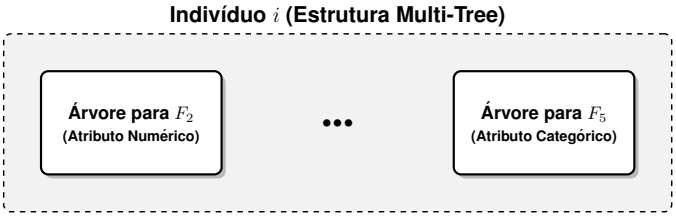

Fig. 2. Representação macro: um único indivíduo contém múltiplas árvores.

Em nível micro, os componentes das árvores possuem papéis definidos: os nós folha não correspondem a valores estáticos, mas sim às saídas dinâmicas dos imputadores base (KNN, Média, MICE). Os nós internos utilizam operadores aritméticos protegidos para evitar instabilidades (Fig. 3).

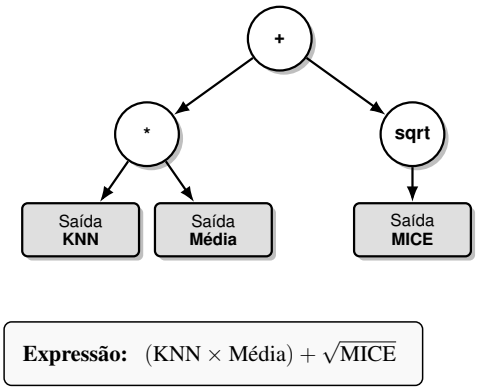

Fig. 3. Estrutura interna de uma árvore de imputação.

C. Seleção (ϵ-Lexicase)

O processo de seleção de progenitores utiliza o método *Epsilon Lexicase Selection*. Diferentemente de métodos baseados em torneio ou roleta que agregam o erro em um único valor escalar (fitness médio), o Lexicase avalia os indivíduos com base em casos de teste individuais (instâncias de dados). O parâmetro ϵ permite relaxar a condição de seleção, permitindo que indivíduos com desempenho suficiente (dentro de um limite ϵ do melhor) passem para a próxima fase, preservando a diversidade populacional e evitando convergência prematura.

D. Avaliação e Função de Aptidão

A avaliação de cada indivíduo na população é realizada em duas etapas principais: a reconstrução dos dados e a validação preditiva. Inicialmente, o indivíduo é utilizado para preencher o conjunto de dados incompleto. Para cada atributo com valores ausentes, a árvore de sintaxe correspondente armazenada no cromossomo do indivíduo é executada, gerando valores imputados baseados na combinação das saídas dos estimadores base.

Uma vez que o conjunto de dados está completo, calcula-se a aptidão (*fitness*) do indivíduo. O GP-Imputer adota uma estratégia orientada à tarefa final. Um classificador auxiliar

(denominado modelo *proxy*) é treinado sobre os dados imputados e avaliado mediante validação cruzada. A métrica escolhida para guiar a evolução é o *F1-Score*.

A escolha por uma métrica baseada em classificação, em detrimento de métricas de erro de reconstrução (como RMSE ou MAE), justifica-se pela aplicabilidade em cenários reais. Em aplicações práticas de imputação, os valores verdadeiros dos dados ausentes são desconhecidos, tornando impossível o cálculo direto do erro de imputação. Portanto, assume-se que a melhor imputação é aquela que maximiza a separabilidade das classes e o desempenho do modelo preditivo subsequente.

Por fim, para evitar o crescimento excessivo e *overfitting*, a função de avaliação incorpora um coeficiente de parcimônia. A aptidão final é penalizada proporcionalmente à complexidade da solução, conforme a Equação 1:

$$Fitness(I) = F1_{score}(D_{imputed}) - (\alpha \times Size(I)) \quad (1)$$

onde $F1_{score}$ é a média harmônica entre precisão e revocação obtida no conjunto $D_{imputed}$, $Size(I)$ representa o número total de nós nas árvores do indivíduo I, e α é o coeficiente de parcimônia que regula o peso da penalidade. Dessa forma, entre duas soluções com desempenho preditivo similar, o algoritmo favorece a que apresenta estrutura mais simples e interpretável.

E. Operadores Genéticos

1) Crossover: O operador de cruzamento recombina informações genéticas de dois progenitores para criar novos indivíduos. Devido à representação *multi-tree* adotada no GP-Imputer, o processo de recombinação atua em dois níveis distintos:

1) **Cruzamento Vetorial :** Considera o indivíduo como um vetor de árvores. Neste processo, árvores completas referentes a diferentes atributos são permutadas entre os pais. Isso permite que um descendente herde, por exemplo, a estratégia de imputação integral para o atributo F_1 do primeiro progenitor e a estratégia para o atributo F_2 do segundo (Fig. 4)
2) **Cruzamento Estrutural:** Opera internamente em uma árvore específica. Dado um mesmo atributo alvo, seleciona-se aleatoriamente um ponto de corte (nó) em cada um dos progenitores, e as subárvores enraizadas nesses pontos são trocadas. Esse mecanismo gera novas combinações matemáticas de funções e imputadores base, refinando a lógica de reconstrução (Fig. 5).

2) Mutação: A mutação introduz diversidade aleatória na população. No GP-Imputer, seleciona-se um nó aleatório em uma árvore de um indivíduo e substitui-se a subárvore enraizada nesse nó por uma nova subárvore gerada aleatoriamente (Fig. 6). Isso permite a exploração de novas áreas do espaço de busca e evita a estagnação em ótimos locais.

IV. Experimentos e Resultados

Esta seção apresenta a validação empírica da abordagem proposta, estruturada para demonstrar a eficácia do GP-Imputer frente a diversos cenários de dados incompletos.

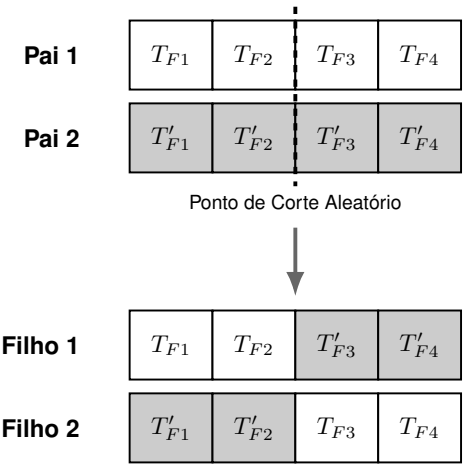

Fig. 4. Crossover Vetorial. Árvores completas referentes a diferentes atributos são trocadas entre os progenitores com base em um ponto de corte.

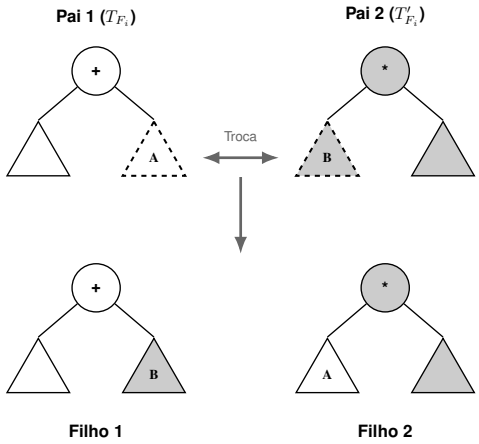

Fig. 5. Crossover Estrutural (Nível da Árvore). Subárvores são trocadas entre os pais, gerando novas combinações lógicas nos filhos.

Inicialmente, descreve-se o protocolo experimental adotado, detalhando os conjuntos de dados selecionados, o método de indução artificial de valores ausentes e a parametrização dos algoritmos evolutivos e dos modelos de base. Na sequência, são reportados os resultados comparativos em relação a técnicas de imputação do estado da arte, avaliando-se o desempenho do método primordialmente pelo seu impacto na performance preditiva (F1-Score) dos classificadores finais, conforme a estratégia orientada à tarefa definida na metodolo-

Fig. 6. Operador de Mutação de Subárvore: uma parte da árvore é removida e substituída por uma nova estrutura gerada aleatoriamente (triângulo tracejado).

gia.

A. Configuração Experimental

A validação empírica do GP-Imputer foi conduzida seguindo um protocolo rigoroso estruturado em três etapas: (i) seleção do classificador; (ii) otimização dos métodos de referência (*baselines*); e (iii) avaliação comparativa em cenários de ausência de dados induzida.

1) Conjuntos de Dados e Indução de Ausência: Foram selecionados cinco conjuntos de dados de referência (*benchmarks*) provenientes do repositório UCI Machine Learning, abrangendo diferentes domínios e dimensionalidades: Sonar, Australian Credit, Ionosphere, Spectf Heart e Statlog Heart.

Como estes dados são originalmente completos, aplicou-se um mecanismo de indução artificial de ausência sob a premissa MCAR (*Missing Completely at Random*), onde a probabilidade de um valor estar ausente é independente de qualquer observação no conjunto de dados. Foram gerados cenários com taxas de ausência de 10%, 20% e 30% para cada dataset, garantindo a avaliação da robustez do método frente a diferentes níveis de severidade na perda de informação.

2) Seleção do Classificador Auxiliar: Dado que o GP-Imputer utiliza uma abordagem orientada a tarefa final, onde um classificador é treinado repetidamente durante a avaliação da aptidão (*fitness*), o custo computacional é um fator crítico. Conduziu-se um experimento preliminar comparando Random Forest (RF), Support Vector Machines (SVM), k-Nearest Neighbors (KNN), Naive Bayes e Árvores de Decisão (DT).

Com base na análise de trade-off apresentada na Fig. 7, optou-se pela utilização da **Regressão Logística** como classificador durante o processo evolutivo. Embora este algoritmo tenha apresentado um tempo de execução superior a alternativas mais simples, como Naive Bayes ou Árvores de Decisão, observou-se que este custo computacional adicional foi marginal quando amortizado pelo tempo total do ciclo evolutivo do GP, não comprometendo a viabilidade da abordagem.

A escolha pela Regressão Logística justifica-se primordialmente por ter obtido o melhor F1-Score médio entre os candidatos e por sua robustez teórica frente ao SVM (*Support Vector Machines*). Em um contexto de imputação evolutiva, onde a distribuição dos atributos é alterada dinamicamente e pode conter ruído significativo especialmente nas gerações iniciais, a sensibilidade do SVM a outliers e a sua dependência de fronteiras rígidas poderiam induzir instabilidades na função de fitness. Em contrapartida, a Regressão Logística otimiza a verossimilhança sobre todo o conjunto de dados, oferecendo uma superfície de avaliação mais suave e estável para guiar a busca genética.

3) Otimização dos Métodos de Referência: Para assegurar uma comparação justa, o GP-Imputer foi confrontado com métodos de imputação simples (Média e Mediana) e técnicas do estado da arte: KNN-Imputer, MICE (*Multivariate Imputation by Chained Equations*), MissForest e SVD (*Singular Value Decomposition*).

Diferentemente de abordagens que utilizam parâmetros padrão, os hiperparâmetros de cada *baseline* foram otimizados

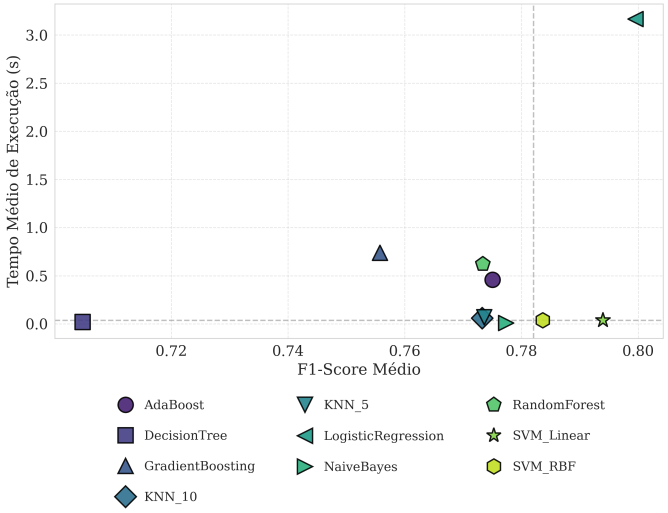

Fig. 7. Análise de trade-off para seleção do classificador. O gráfico relaciona o desempenho preditivo (F1-Score médio) no eixo X com o custo computacional (Tempo de Execução) no eixo Y.

especificamente para cada conjunto de dados utilizando o *framework* Optuna. A função objetivo buscou maximizar o F1-Score através de validação cruzada estratificada ($k = 5$), ajustando variáveis críticas como o número de vizinhos (k) para o KNN e o número de estimadores para o MissForest.

4) Parâmetros do GP-Imputer: A configuração dos parâmetros evolutivos do GP-Imputer foi definida empiricamente com base em testes piloto e nas recomendações da literatura especializada em Programação Genética. A Tabela I sumariza a configuração utilizada nos experimentos finais.

TABLE I
CONFIGURAÇÃO DOS PARÂMETROS DO GP-IMPUTER

Parâmetro	**Valor / Configuração**
Tamanho da População	100
Número de Gerações	100
Método de Inicialização	Full
Profundidade Máx. (Árvore)	7
Taxa de Crossover	0.8
Taxa de Mutação	0.15
Seleção dos Pais	Epsilon Lexicase Selection
Elitismo	5 indivíduos
Coeficiente de Parcimônia	0.0001
Execuções Independentes	10

5) Avaliação e Análise Estatística: Para verificar a estabilidade estocástica do método proposto, cada experimento foi executado 10 vezes com diferentes sementes aleatórias (*random seeds*). O desempenho foi mensurado pelo F1-Score obtido por um classificador final (Regressão Logística) treinado sobre os dados imputados.

A significância estatística dos resultados foi validada utilizando o teste não-paramétrico de Wilcoxon (*Rank-Sum Test*) com nível de significância de $\alpha = 0.05$, comparando o GP-Imputer com os métodos de referência.

B. Resultados

A validação da eficácia do GP-Imputer pode ser visualizada primeiramente através da sua dinâmica de otimização. A Fig. 8 ilustra as curvas de convergência do F1-Score do melhor indivíduo ao longo de 100 gerações para os cinco conjuntos de dados avaliados, sob taxas de ausência de 10%, 20% e 30%. Nestes gráficos, a linha sólida representa a trajetória de aprendizado do GP-Imputer, enquanto as linhas tracejadas horizontais indicam o desempenho estático alcançado pelos métodos de referência (*baselines*) otimizados. A análise visual revela que, na maioria dos cenários, o algoritmo genético é capaz de evoluir estruturas que partem de um desempenho inicial comparável ou inferior aos métodos clássicos e, progressivamente, superam essas barreiras, estabilizando em patamares de aptidão significativamente mais elevados.

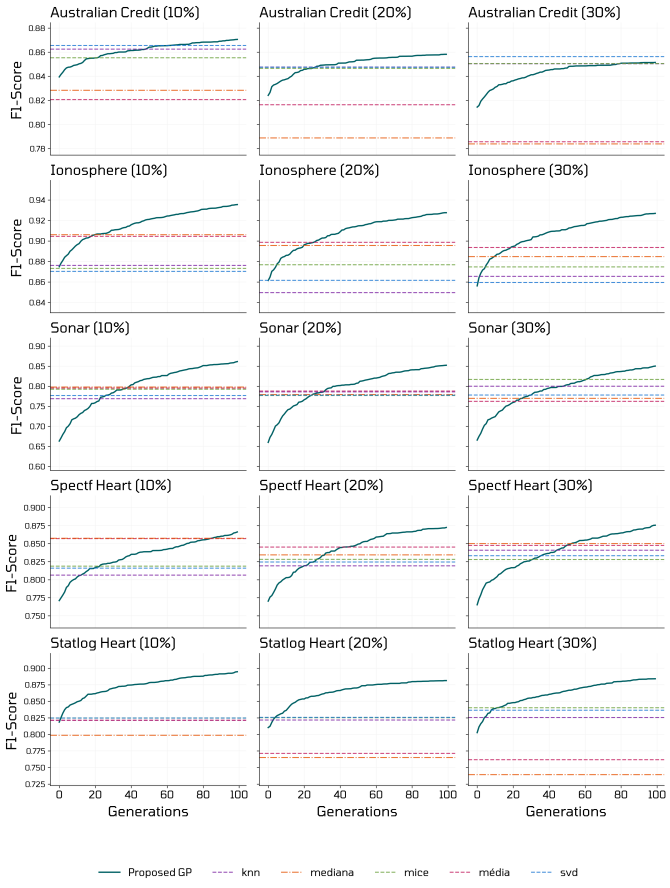

Fig. 8. Curvas de convergência do GP-Imputer (linha sólida) *versus* métodos de referência (linhas tracejadas) ao longo de 100 gerações. O eixo Y representa o F1-Score e o eixo X as gerações. Nota-se a capacidade do método proposto de superar os *baselines* conforme a evolução avança.

Os resultados experimentais consolidados, apresentados na Tabela II, demonstram a eficácia superior do GP-Imputer frente aos métodos tradicionais. A abordagem proposta obteve o melhor F1-Score médio em 14 dos 15 cenários avaliados, apresentando diferença estatisticamente significativa ($p < 0.05$, teste de Wilcoxon) em relação à maioria dos *baselines*. A análise detalhada desses resultados permite identificar comportamentos distintos baseados na natureza dos dados.

1) Desempenho em Alta Dimensionalidade e Dependência Não-Linear: Nos conjuntos de dados com maior número de atributos ou fronteiras de decisão complexas, o ganho de desempenho proporcionado pelo GP-Imputer foi expressivo. O dataset *Sonar*, caracterizado pela maior dimensionalidade (60 atributos), ilustra claramente a limitação dos métodos univariados: enquanto a Média e a Mediana estagnaram em patamares inferiores a 0.80, o GP-Imputer alcançou F1-Scores superiores a 0.85 em todas as taxas de ausência. Similarmente, no *Ionosphere*, o método superou o melhor competidor (Mediana/Média) por uma margem robusta, atingindo 0.9355 no cenário de 10% de dados faltantes. Esses resultados corroboram a hipótese de que a estratégia de *ensemble* evolutivo é particularmente eficaz em explorar redundâncias em espaços de alta dimensionalidade, construindo imputações não-lineares que métodos estáticos não conseguem capturar.

2) Robustez em Baixa Dimensionalidade: A superioridade do método não se restringiu a ambientes complexos, estabelecendo dominância também em *datasets* com espaço de busca reduzido. O caso do *Statlog Heart* (13 atributos) mostra isso: enquanto métodos estatísticos simples como a Mediana sofreram degradação severa com o aumento da taxa de ausência (caindo de 0.79 para 0.73), o GP-Imputer demonstrou notável resiliência, mantendo um F1-Score de 0.8841 mesmo no cenário crítico de 30% de ausência. Isso sugere que, mesmo com menos variáveis disponíveis para permutações genéticas, o algoritmo converge rapidamente para combinações ótimas.

3) Competitividade e Análise de Exceções: A análise do *Spectf Heart* revelou um cenário atípico onde métodos univariados simples, especificamente a Mediana, apresentaram desempenho inicial surpreendentemente alto (0.8575 a 10%), superando técnicas multivariadas complexas como MICE e SVD. Ainda assim, o GP-Imputer foi capaz de evoluir soluções superiores (0.8662), provando sua capacidade de adaptação mesmo quando métodos triviais já oferecem uma linha de base forte. A única exceção à dominância do método proposto ocorreu no *Australian Credit* sob 30% de ausência, onde o SVD obteve um desempenho marginalmente superior (0.8563 contra 0.8515 do GP-Imputer). Contudo, a diferença é sutil e a estabilidade do método proposto (baixo desvio padrão ± 0.0055) permanece competitiva frente à abordagem de fatoração de matrizes.

4) Estabilidade Estocástica: Além das métricas de tendência central, é crucial destacar a baixa dispersão dos resultados obtidos pelo GP-Imputer. O desvio padrão (σ) manteve-se abaixo de 0.02 na vasta maioria dos experimentos, conforme reportado na Tabela II. Essa consistência evidencia a estabilidade estocástica do algoritmo: independentemente da semente aleatória inicial, o processo evolutivo converge consistentemente para soluções de imputação que maximizam a separabilidade das classes, validando a escolha da função de aptidão baseada no classificador.

TABLE II

RESULTADOS EXPERIMENTAIS CONSOLIDADOS (F1-SCORE). OS MELHORES RESULTADOS POR DATASET E TAXA DE MISSING ESTÃO EM **NEGRITO**. O SÍMBOLO † INDICA DIFERENÇA ESTATISTICAMENTE SIGNIFICATIVA ($p < 0.05$) ENTRE O GP-IMPUTER E O BASELINE (WILCOXON SIGNED-RANK TEST).

Dataset	Modelo	10%	20%	30%
Australian Credit	Knn	0.8625†	0.8476†	0.8504
	Média	0.8206†	0.8164†	0.7856†
	Mediana	0.8283†	0.7888†	0.7838†
	Mice	0.8554†	0.8465†	0.8505
	Svd	0.8655†	0.8475†	**0.8563**†
	GP-Imputer	**0.8705 ± 0.0030**	**0.8582 ± 0.0041**	0.8515 ± 0.0055
Ionosphere	Knn	0.8761†	0.8496†	0.8654†
	Média	0.9045†	0.8987†	0.8937†
	Mediana	0.9061†	0.8956†	0.8848†
	Mice	0.8732†	0.8769†	0.8747†
	Svd	0.8703†	0.8615†	0.8595†
	GP-Imputer	**0.9355 ± 0.0082**	**0.9276 ± 0.0077**	**0.9269 ± 0.0072**
Sonar	Knn	0.7687†	0.7858†	0.8000†
	Média	0.7954†	0.7879†	0.7624†
	Mediana	0.7977†	0.7793†	0.7696†
	Mice	0.7924†	0.7766†	0.8170†
	Svd	0.7766†	0.7776†	0.7779†
	GP-Imputer	**0.8613 ± 0.0203**	**0.8522 ± 0.0218**	**0.8501 ± 0.0084**
Spectf Heart	Knn	0.8063†	0.8192†	0.8410†
	Média	0.8569†	0.8453†	0.8476†
	Mediana	0.8575†	0.8344†	0.8501†
	Mice	0.8187†	0.8283†	0.8280†
	Svd	0.8160†	0.8246†	0.8333†
	GP-Imputer	**0.8662 ± 0.0095**	**0.8724 ± 0.0091**	**0.8755 ± 0.0127**
Statlog Heart	Knn	0.8212†	0.8218†	0.8256†
	Média	0.8215†	0.7713†	0.7618†
	Mediana	0.7986†	0.7650†	0.7393†
	Mice	0.8247†	0.8253†	0.8402†
	Svd	0.8248†	0.8256†	0.8364†
	GP-Imputer	**0.8947 ± 0.0061**	**0.8813 ± 0.0095**	**0.8841 ± 0.0089**

V. CONCLUSÃO

Este trabalho abordou o desafio persistente do tratamento de dados ausentes em tarefas de mineração de dados, propondo o GP-Imputer, uma metodologia híbrida baseada em Programação Genética. Ao contrário das abordagens convencionais e dos trabalhos correlatos que utilizam algoritmos evolutivos para gerar regressores diretamente a partir dos dados brutos, o método buscou descobrir combinações não lineares ótimas entre técnicas consolidadas, como KNN, MICE, SVD e estatísticas simples. Essa técnica busca maximizar a qualidade dos dados reconstruídos e o desempenho dos modelos classificadores subsequentes.

Os experimentos realizados em cinco bases de dados de referência, sob mecanismos de ausência MCAR com taxas variando de 10% a 30%, demonstraram a eficácia da abordagem proposta. A análise das curvas de convergência revelou que o GP-Imputer é capaz de superar as limitações individuais dos métodos de base, apresentando uma trajetória de aprendizado consistente ao longo das gerações. Destaca-se a robustez do método em cenários de alta dimensionalidade ou maior escassez de informações (30% de dados faltantes), onde técnicas tradicionais frequentemente apresentaram estagnação de desempenho, enquanto a solução evolutiva manteve ganhos marginais contínuos na métrica F1-Score.

Uma constatação relevante deste estudo foi a capacidade do algoritmo de se adaptar às especificidades de cada distribuição de dados. Em casos onde métodos simples, como a Mediana, mostraram-se surpreendentemente competitivos (ex: Spectf

Heart), o GP-Imputer conseguiu evoluir estruturas complexas suficientes para ultrapassar esse patamar de desempenho, validando a hipótese de que a diversidade de modelos, quando ponderada corretamente, supera a aplicação uniforme de uma única técnica.

Em suma, o GP-Imputer apresenta-se como uma alternativa promissora para o pré-processamento de dados, oferecendo um balanço eficaz entre a automação do aprendizado de máquina e a utilização de heurísticas estatísticas clássicas. Para trabalhos futuros, sugere-se a investigação do método sob mecanismos de ausência mais complexos, como Missing Not At Random (MNAR), e a inclusão de novos imputadores base no conjunto de funções, visando ampliar ainda mais a capacidade de generalização do modelo evoluído.

REFERENCES

[1] H. Kang, "The prevention and handling of the missing data," *Korean Journal of Anesthesiology*, vol. 64, no. 5, pp. 402–406, 2013. [Online]. Available: https://doi.org/10.4097/kjae.2013.64.5.402

[2] J. W. Graham, "Missing data analysis: Making it work in the real world," *Annual Review of Psychology*, vol. 60, pp. 549–576, 2009. [Online]. Available: https://doi.org/10.1146/annurev.psych.58.110405.085530

[3] M. Critical Data, *Secondary Analysis of Electronic Health Records*. Springer Nature, 2016. [Online]. Available: https://doi.org/10.1007/978-3-319-43742-2

[4] A. Donner, "The relative effectiveness of procedures commonly used in multiple regression analysis for dealing with missing values," *The American Statistician*, vol. 36, no. 4, pp. 378–381, 1982. [Online]. Available: https://doi.org/10.1080/00031305.1982.10483055

[5] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*, 2nd ed. Hoboken, NJ: Wiley, 2002.

[6] A. R. T. Donders, G. J. M. G. van der Heijden, T. Stijnen, and K. G. M. Moons, "Review: A gentle introduction to imputation of missing values," *Journal of Clinical Epidemiology*, vol. 59, no. 10, pp. 1087–1091, 2006. [Online]. Available: https://doi.org/10.1016/j.jclinepi.2006.01.014

[7] M. J. Azur, E. A. Stuart, C. Frangakis, and P. J. Leaf, "Multiple imputation by chained equations: what is it and how does it work?" *International Journal of Methods in Psychiatric Research*, vol. 20, no. 1, pp. 40–49, March 2011.

[8] R. J. A. Little and D. B. Rubin, *Analysis of Incomplete Multivariate Data*, 2nd ed. Chapman and Hall/CRC, 2002. [Online]. Available: https://doi.org/10.1201/9781420035388

[9] C. T. Tran, M. Zhang, and P. Andreae, "Multiple imputation for missing data using genetic programming," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation (GECCO '15)*. Madrid, Spain: ACM, 2015, pp. 583–590.

[10] ——, "A genetic programming-based imputation method for classification with missing data," in *Genetic Programming (EuroGP 2016)*, ser. Lecture Notes in Computer Science, M. e. a. Heywood, Ed., vol. 9594. Springer International Publishing, 2016, pp. 149–163.

[11] C. T. Tran, M. Zhang, P. Andreae, and B. Xue, "Multiple imputation and genetic programming for classification with incomplete data," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '17)*. Berlin, Germany: ACM, 2017.

[12] B. Al-Helali, Q. Chen, B. Xue, and M. Zhang, "A new imputation method based on genetic programming and weighted KNN for symbolic regression with incomplete data," 2021, manuscript provided in sources (Filename indicates 2021 final version).