PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Programa de Pós-Graduação em Informática

Felipe Augusto Lara Soares

Automated Machine Learning Pipeline Enhanced by a Large Language Model for Healthcare Prediction Based on Panel Data

Belo Horizonte

2024

Felipe Augusto Lara Soares

Automated Machine Learning Pipeline Enhanced by a Large Language Model for Healthcare Prediction Based on Panel Data

Thesis presented to the Graduate Program in Informatics at the Pontifícia Universidade Católica de Minas Gerais, as a partial requirement to obtain the title of Doctor in Informatics.

Advisor: Professor Dr. Henrique Cota de Freitas

Co-advisor: Professor Dr. Cristiane Neri Nobre

Belo Horizonte

2024

FICHA CATALOGRÁFICA
Elaborada pela Biblioteca da Pontifícia Universidade Católica de Minas Gerais

Soares, Felipe Augusto Lara
S676a Automated machine learning pipeline enhanced by a Large Language Model for healthcare prediction based on panel data / Felipe Augusto Lara Soares. Belo Horizonte, 2024.
170 f. : il.

Orientador: Henrique Cota de Freitas
Coorientadora: Cristiane Neri Nobre
Tese (Doutorado) – Pontifícia Universidade Católica de Minas Gerais. Programa de Pós-Graduação em Informática

1. Aprendizado do computador - Automação. 2. Coleta de dados. 3. Processamento de dados. 4. Redes neurais (Computação). 5. Mineração de dados (Computação). 6. Algoritmos. 7. Sistemas de recuperação da informação - Saúde. 8. Observatórios de Saúde. I. Freitas, Henrique Cota de. II. Nobre, Cristiane Neri. III. Pontifícia Universidade Católica de Minas Gerais. Programa de Pós-Graduação em Informática. IV. Título.

SIB PUC MINAS

CDU: 681.3.091

Ficha catalográfica elaborada por Fabiana Marques de Souza e Silva - CRB 6/2086

Felipe Augusto Lara Soares

Automated Machine Learning Pipeline Enhanced by a Large Language Model for Healthcare Prediction Based on Panel Data

Thesis presented to the Graduate Program in Informatics at the Pontifícia Universidade Católica de Minas Gerais, as a partial requirement to obtain the title of Doctor in Informatics.

Prof. Dr. Henrique Cota de Freitas (advisor) – PUC Minas

Prof. Dr. Cristiane Neri Nobre (co-advisor) – PUC Minas

Prof. Dr. Luiz Enrique Zárate Galvez – PUC Minas

Prof. Dr. Alexei Manso Corrêa Machado – PUC Minas

Prof. Dr. Rodrigo da Rosa Righi – Unisinos

Prof. Dr. Gisele Lobo Pappa – UFMG

Belo Horizonte, 19 de Dezembro de 2024.

PREFACE

This thesis presents the research I have done during my Ph.D. studies (started in 2020) at the Computer Architecture and Parallel Processing Team (CArT), PUC Minas. My study in this area of research began in my master's degree (2017-2019) and, under the supervision of Professor Dr. Henrique Cota de Freitas, we have been reaping the fruits of our work so far. My Ph.D. is also co-supervised by Professor Dr. Cristiane Neri Nobre from the Applied Computational Intelligence Laboratory (LICAP). Professors Freitas and Nobre contributed to improving my expertise in Machine Learning and AutoML for this doctoral thesis defense. During my Masters and Ph.D. studies, I participated in conferences and workshops, such as SSCAD (formerly called WSCAD 2019), SBAC-PAD 2019, SBBD 2021, WPerformance 2021 and KDMile 2021, which were essential to my learning about scientific research. Over these years, I achieved some accepted publications related directly and indirectly to my Ph.D. thesis, as follows:

Publications on Machine Learning in healthcare directly related to the subject of this thesis

- **Automated and Intelligent System for World Health Organization Data Forecasting.** Felipe A. L. Soares, Henrique C. Freitas. *Expert Systems*, 2024.
- **Analysis and Prediction of Childhood Pneumonia Deaths using Machine Learning Algorithms.** Felipe A. L. Soares, Efrem E. O. Lousada, Tiago B. Silveira, Raquel A. F. Mini, Luis E. Zárate, Henrique C. Freitas. *Anais do IX Symposium on Knowledge Discovery, Mining and Learning (KDMiLe 2021)*, 2021.

Publications on Machine Learning related to the subject of this thesis, but applied to different contexts

- **Fast and Efficient Parallel Execution of SARIMA Prediction Model.** Tiago B. Silveira, Felipe A. L. Soares, Henrique C. Freitas. *Lecture Notes in Business Information Processing, Springer International Publishing*, 2021.
- **Hybrid Approach based on SARIMA and Artificial Neural Networks for Knowledge Discovery Applied to Crime Rates Prediction.** Felipe A. L. Soares, Tiago B. Silveira, Henrique C. Freitas. *International Conference on Enterprise Information Systems*, 2020.

Publications not directly related to this thesis

- **A Systematic Mapping of Autonomous Vehicle Prototypes: Trends and Opportunities.** Ericson Silva, Felipe. Soares, Wenderson Souza, Henrique C. Freitas. *IEEE Transactions on Intelligent Vehicles (T-IV)*, 2024.
- **Automatic Code Parallelization in CUDA Using Reinforcement Learning.** Felipe A. L. Soares, Tiago B. Silveira, Humberto T. M. Neto, Henrique. C. Freitas. *Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance)*, 2021.
- **Heterogeneous Parallel Architecture for Inverted Index Generation.** Tiago B. Silveira, Felipe. A. L. Soares, Wladmir C. Brandão, Henrique C. Freitas. *Simpósio em Sistemas Computacionais de Alto Desempenho (WSCAD)*, 2019.
- **Teaching Parallel Programming to Freshmen in an Undergraduate Computer Science Program.** Leonardo B. A. Vasconcelos, Felipe. A. L. Soares, Pedro H. M. M. Penna, Max V. Machado, Luis F. W. Goes, Carlos A. P. S. Martins, Henrique C. Freitas. *IEEE Frontiers in Education Conference (FIE)*, 2019.
- **Parallel Programming in Computing Undergraduate Courses: a Systematic Mapping of the Literature.** Felipe A. L. Soares, Cristiane N. Nobre, Henrique. C. Freitas. *IEEE Latin America Transactions*, 2019.
- **Uma análise sobre o desenvolvimento participativo de jogos educacionais voltados para a terceira idade.** Ezequiel Duque, Michelle N. Nascimento, Guilherme Fonseca, Felipe A. L. Soares, Hebert Pereira, Lucila Ishitani. *Simpósio Brasileiro de Informática na Educação (Brazilian Symposium on Computers in Education)*, 2018.

ACKNOWLEDGMENTS

To God, for all the companionship in every moment of my life.

To my advisor, Prof. Dr. Henrique Cota de Freitas, and my co-advisor, Prof. Dr. Cristiane Neri Nobre, for their guidance, dedication, and teachings throughout this journey.

To my parents, Altamar and Mirian, and my wife, Cibelle, for their love, dedication, and companionship in every moment.

To my entire family, friends, and all those who have been with me throughout this journey.

Finally, I express my gratitude to the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES, code 001), the Foundation for Research Support of Minas Gerais State (FAPEMIG, codes APQ-03076-18 and APQ-05058-23), National Council for Scientific and Technological Development of Brazil (CNPq, codes 311573/2022-3 and 311697/2022-4), and Pontifical Catholic University of Minas Gerais.

To them, my heartfelt thanks.

ABSTRACT

Technological advances and social transformations have enabled the circulation of a large amount of data in the healthcare sector. Analyzing this data becomes more critical and more challenging as the volume of data increases. Therefore, due to the characteristics of this data, the diversity and heterogeneity of the data and its sources, the manual knowledge discovery process can be costly and not sufficient for all existing possibilities, mainly because this process requires a Machine Learning specialist and a domain expert. Thus, Automated Machine Learning (AutoML) emerged as a prominent solution, automating the multiple steps of the Machine Learning process, from data collection and preprocessing to post-processing. These techniques work mainly on automating the Machine Learning pipeline, minimizing user intervention, increasing productivity and efficiency in this workflow and democratizing Machine Learning for users with limited experience in this area. Although there are several AutoML frameworks, the literature does not yet present a proposal that automates all stages of the Machine Learning process. This issue is even more critical in the healthcare sector, where few studies have proposed the application of AutoML. In this context, this work proposes an Automated Machine Learning (AutoML) in all stages of a pipeline for healthcare data prediction, which uses data analysis techniques, Machine Learning algorithms and a Large Language Model (LLM). Our AutoML pipeline composes a system designed in this thesis named HEAL, which means **He**althcare Data - **A**utomated Machine **L**earning. It is specifically applied to classification problems involving structures organized in spatial dimension and temporal dimension (panel data structure) and encompasses its all stages: 1 - Domain Understanding and Data Collection; 2 - Pre-processing; 3 - Data Mining; 4 - Post-processing. Our AutoML pipeline was developed using ChatGPT-4 as the LLM tool, applied in steps 1 - Domain Understanding and Data Collection and 4 - Post-processing. In the first step, our approach collects all data in an automated way from the World Health Organization using Application Programming Interface (API) and data analysis, collecting all indicators in the panel data structure. After that, this LLM was used to be a user interface and interpret the context to answer questions and, consequently, help the user understand the domain. In the last step, this LLM is used to validate and analyze results to find knowledge discoveries about them. In step 2 - Pre-processing, there are some algorithms such as missing data analysis, feature selection, outlier detection, discretization, and data balancing. In step 3 - Data Mining, we have implemented a set of five algorithms for classification problems: Artificial Neural Network, Decision Tree, Random Forest, Support Vector Machine and K-nearest Neighbors, and used Combined Algorithm Selection and Hyperparameter Optimization (CASH) to perform model selection and hyperparameter

optimization. The efficiency of the system's choices and forecasts was verified for four different areas of health: sexual health; nutritional health; safe sanitation; and mental health. In these contexts and in the best scenario, HEAL achieved precision of up 97.63%, recall of up 90.28% and f1-score of up 93.81%, hitting even the most complex cases. After these experiments, we performed the evaluation and comparison of the prediction results with two different AutoML frameworks found in the literature (H2O and TPOT), showing that HEAL obtained superior results in 3 of the 4 experiments. Finally, three case studies were carried out to expand the studies and present the potential of the system in different contexts: prevalence of anemia in children in the countries of the Americas in 2019; age-standardized suicide rates in men (per 100,000 population) in the countries of Africa in 2019; and influence of tuberculosis and hypertension in prevalence of anemia in children in India in 2021 - using Global Burder of Disease (GBD) database. This final case study was conducted to evaluate HEAL using a different WHO database (GBD), which is also structured as panel data.

Keywords: Automated Machine Learning (AutoML), AutoML Pipeline, Healthcare, Panel Data.

RESUMO

Os avanços tecnológicos e as transformações sociais permitiram a circulação de um grande volume de dados no setor da saúde. Analisar esses dados se torna mais crítico e desafiador à medida que o volume de dados aumenta. Portanto, devido às características desses dados, à diversidade e heterogeneidade dos dados e de suas fontes, o processo manual de descoberta de conhecimento pode ser custoso e insuficiente para todas as possibilidades existentes, principalmente porque esse processo exige um especialista em Aprendizado de Máquina e um especialista no domínio. Assim, o Aprendizado de Máquina Automatizado (AutoML, do inglês *Automated Machine Learning*) surgiu como uma solução de destaque, automatizando as múltiplas etapas do processo de Aprendizado de Máquina, desde a coleta de dados e o pré-processamento até o pós-processamento. Essas técnicas funcionam principalmente na automação do *pipeline* de Aprendizado de Máquina, minimizando a intervenção do usuário, aumentando a produtividade e a eficiência nesse fluxo de trabalho e democratizando o Aprendizado de Máquina para usuários com experiência limitada nessa área. Embora existam diversas ferramentas de AutoML, a literatura ainda não apresenta uma proposta que automatize todas as etapas do processo de Aprendizado de Máquina. Essa questão é ainda mais crítica no setor de saúde, onde poucos estudos propuseram a aplicação do AutoML. Neste contexto, este trabalho propõe um AutoML em todas as etapas de um *pipeline* para predição de dados de saúde, que utiliza técnicas de análise de dados, algoritmos de Aprendizado de Máquina e um *Large Language Model* (LLM). Esse AutoML proposto compõe um sistema projetado nesta tese denominado HEAL, que significa **He**althcare Data - **A**utomated Machine **L**earning. O HEAL é aplicado especificamente a problemas de classificação envolvendo estruturas organizadas em dimensão espacial e dimensão temporal, chamadas de estruturas de dados em painel (do inglês *panel data*) e abrange todas as suas etapas: 1 - Compreensão do Domínio e Coleta de Dados; 2 - Pré-processamento; 3 - Mineração de Dados; 4 - Pós-processamento. O AutoML proposto foi desenvolvido usando ChatGPT-4 como ferramenta LLM, aplicado nas etapas 1 - Entendimento do Domínio e Coleta de Dados e 4 - Pós-processamento. Na primeira etapa, a abordagem proposta coleta todos os dados de forma automatizada da Organização Mundial da Saúde (OMS) usando Interface de Programação de Aplicativos (API, do inglês *Application Programming Interface*) e análise de dados, coletando todos os indicadores na estrutura de dados em painel. Depois disso, o LLM foi usado como uma interface de usuário e como ferramenta para interpretar o contexto com o objetivo de responder perguntas e, consequentemente, ajudar o usuário a entender o domínio do problema. Na última etapa, esse LLM é usado para validar e analisar resultados para encontrar descobertas de conhecimento sobre eles. Na etapa 2 - Pré-processamento, existem

alguns algoritmos, como análise de dados ausentes, seleção de recursos, detecção de outliers, discretização e balanceamento de dados. Na etapa 3 - Mineração de Dados, foram implementamos um conjunto de cinco algoritmos para problemas de classificação: Rede Neural Artificial (ANN, do inglês *Artificial Neural Network*), Árvore de Decisão (DT, do inglês *Decision Tree*), Floresta Aleatória (RF, do inglês *Random Forest*), Máquina de Vetores de Suporte (SVM, do inglês *Support Vector Machine*) e K-vizinhos mais próximos (KNN, do inglês *K-nearest Neighbors*), e foi usado *Combined Algorithm Selection and Hyperparameter Optimization* (CASH) para realizar a seleção de modelos e a otimização de hiperparâmetros. A eficiência das escolhas e previsões do sistema foi verificada para quatro áreas diferentes de saúde: saúde sexual; saúde nutricional; saneamento seguro; e saúde mental. Nesses contextos e no melhor cenário, o HEAL alcançou precisão de até 97,63%, recall de até 90,28% e f1-score de até 93,81%, conseguindo predizer até os casos mais complexos. Após esses experimentos, foram realizadas avalizações e comparações dos resultados de predição com dois AutoML diferentes encontrados na literatura (H2O e TPOT), mostrando que o HEAL obteve resultados superiores em 3 dos 4 experimentos. Finalmente, três estudos de caso foram realizados para expandir os estudos e apresentar o potencial do sistema em diferentes contextos: prevalência de anemia em crianças nos países das Américas em 2019; taxas de suicídio padronizadas por idade em homens (por 100.000 habitantes) nos países da África em 2019; e influência da tuberculose e hipertensão na prevalência de anemia em crianças na Índia em 2021 - usando o banco de dados Global Burder of Disease (GBD). Este estudo de caso final foi conduzido para avaliar o HEAL usando um banco de dados diferente da OMS (GBD), que também é estruturado como dados em painel.

Palavras-chave: Aprendizado de Máquina Automatizado (AutoML), AutoML *Pipeline*, Saúde, Dados em Painel.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACL – Autonomous Continuous Learning

AI – Artificial Intelligence

ANN – Artificial Neural Networks

API – Application Programming Interface

ATM – Auto-Tuned Models

AutoML – Automated Machine Learning

CAAFE – Context-Aware Automated Feature Engineering

CASH – Combined Algorithm Selection and Hyperparameter Optimization

CNN – Convolutional Neural Network

Continuous PNAD – Continuous National Household Sample Survey

DBMS – Database Management System

DC – Data Collection

DCNNs – Deep Convolutional Neural Networks

DHS – Demographic and Health Surveys

DNN – Deep Neural Network

DU – Domain Understanding

GAMA – General Automated Machine Learning Assistant

GBD – Global Burden of Disease

GGP – Grammar-Based Genetic Programming

GHO – Global Health Observatory

GS – Grid Search

HEAL – **He**althcare Data - **A**utomated Machine **L**earning

HO – Hyperparameter Optimization

HTN – Hypertension

IBGE – Brazilian Institute of Geography and Statistics

IHME – Institute for Health Metrics and Evaluation

ILO – International Labour Organization

IQR – Interquartile Range

KDD – Knowledge Discovery in Databases

KNN – K-Nearest Neighbors

LIS – Luxembourg Income Study

LLM – Large Language Model

ML – Machine Learning

MLP – Multilayer Perceptron

MS – Model Selection

NAS – Neural Architecture Search

NLP – Natural Language Processing

OECD – Organisation for Economic Co-operation and Development

POF – Consumer Expenditure Survey

PoP – Post-processing

PrP – Pre-processing

PSO – Particle Swarm Optimization

RF – Random Forest

ROC – Receiver Operating Characteristic

RS – Random Search

SAH – Systemic Arterial Hypertension

SMBO – Sequential Model-based Optimization

SMOTE – Synthetic Minority Oversampling Technique

SVM – Support Vector Machine

TPOT – Tree-based Pipeline Optimization Tool

VS Code – Visual Studio Code

XMC – Extreme Multilabel Classification

WHO – World Health Organization

WVS – World Values Survey

SUMMARY

1 INTRODUCTION

The healthcare sector generates enormous amounts of data that can be used to find hidden patterns and knowledge to collaborate with decision-making in diagnosis and treatment (ALI et al., 2023). These decision-making are essential to preventing diseases, prognostic interventions, new treatments, and therapies, and to model human behavior, bringing improvements to people (SHARMA; WICKRAMASINGHE; GUPTA, 2005).

According to Iwendi et al. (2020), the context of Data Analysis is constantly changing, and the accelerated growth of the data volume to be analyzed creates challenges for processing this information, a complicated task even with all the existing technologies. Thus, as the amount of data grows, the demand for quality software and computing resources also increases (NESHATPOUR; SASAN; HOMAYOUN, 2016).

Machine Learning (ML) aims to create algorithmic solutions that have the ability to train and identify complex patterns and find solutions to new problems using previous data and solutions (ALEEM et al., 2022). Manogaran et al. (2018) state that several algorithms in these areas have been developed to overcome the difficulties presented in recent years. Moreover, according to Salazar et al. (2022), these are used in the healthcare sector, primarily for disease prevention and to assist doctors in diagnosing various medical conditions.

Before using these algorithms, the dataset needs to be pre-processed to prepare it for the algorithms that will be used, improving accuracy and allowing the discovery of patterns for knowledge generation (GOVINDARAJAN et al., 2020).

Given the emphasis that Machine Learning techniques have received in recent years, research is increasingly focused on deepening the topic of automating these techniques to increase productivity and efficiency in the knowledge discovery process (SALEHIN et al., 2024). Consequently, Automated Machine Learning (AutoML) has emerged as a prominent solution, automating the various steps of the ML process, from pre-processing (which includes domain understanding and data collection) to post-processing (such as knowledge interpretation) (BARBUDO; VENTURA; ROMERO, 2023). These techniques work mainly on automating the ML pipeline, minimizing user intervention (ALSHAREF et al., 2022), increasing productivity and efficiency in this workflow and democratizing ML for users with limited experience in this context (SALEHIN et al., 2024).

Although there are several AutoML systems like H2O (LEDELL; POIRIER, 2020), AutoKeras (JIN et al., 2023), EvalML (ALTERYX, 2020), Auto-WEKA (KOTTHOFF

et al., 2017) and Auto-Sklearn (FEURER et al., 2020), it is still an open field of research to develop an AutoML pipeline that automates all steps of the Machine Learning process (ALSHAREF et al., 2022). This issue is even more critical in the healthcare sector, where few studies have proposed the application of AutoML (LUO, 2016b). This is justified because building a high-quality, representative, and diverse health dataset is difficult, and the lack of transparency in black-box AutoML systems, where users do not have access to which model settings, reduces confidence in their application in critical scenarios (WARING; LINDVALL; UMETON, 2020).

According to Zhang et al. (2023), Large Language Model (LLM) techniques have demonstrated the ability to understand natural language and produce coherent and contextually appropriate responses. Developments in recent years have opened up new opportunities for the application of these techniques in different contexts, such as education (KASNECI et al., 2023), healthcare (PAL et al., 2023), software development (Moradi Dakhel et al., 2023), law (NOONAN, 2023) and others (TORNEDE et al., 2023).

These techniques can also be incorporated into the AutoML steps on different occasions, such as when defining the AutoML system configuration or interpreting and validating the results found in each step (TORNEDE et al., 2023). Despite these opportunities, the integration between AutoML and LLM technologies still remains relatively unexplored, limited to some research that focuses on using LLM for individual tasks in AutoML (TSAI et al., 2023).

1.1 Problem

Although the healthcare sector generates enormous amounts of data that can be used to find hidden patterns and knowledge (ALI et al., 2023), this accelerated growth in the volume of data to be analyzed creates great challenges for processing this information and, often, several possibilities for discovering knowledge end up not being found (ALI et al., 2023).

Therefore, given this enormous amount of data, knowledge could be discovered aiming at alternatives to combat diseases that are causes of death around the world. However, due to the characteristics of this data, the diversity and heterogeneity of the data and its sources, the manual knowledge discovery process can be costly and not sufficient for all existing possibilities, mainly because this process requires a Machine Learning specialist and a domain expert (ALSHAREF et al., 2022).

Although the AutoML technique offers an automated pipeline for Machine Learning steps, addressing the mentioned problems, the quest for a method that fully automates the Machine Learning process remains an immature field of research (ALSHAREF

et al., 2022). This issue is particularly critical in the healthcare sector, where relatively few studies have been conducted to propose and apply AutoML (WARING; LINDVALL; UMETON, 2020).

According to Waring, Lindvall and Umeton (2020), in addition to the issues related to fully automating Machine Learning pipeline steps, there are other key challenges to applying Machine Learning in the healthcare space that make it difficult to deploy AutoML solutions. Firstly, it is challenging to assemble a diverse, representative, and high-quality dataset about a healthcare context. Secondly, and a much bigger problem, is the lack of transparency in these black box AutoML systems, where users do not have access to which configurations were searched by the model, leading to a lack of trust in the system and, consequently, they will hesitate to apply these AutoML results in critical applications.

1.2 Objectives

This thesis proposes to investigate the integration of data analysis techniques, Large Language Models, and Machine Learning across multiple steps of a pipeline, aiming to enhance automated data prediction in classification problems applied to healthcare panel data structures. By promoting this integration, it is expected to develop a solution that minimizes human intervention in the process, making the pipeline accessible and applicable to both experts and non-experts. To achieve this goal, this pipeline comprises the following steps: data collection, domain understanding, pre-processing, model selection, hyperparameter optimization, prediction, and post-processing.

According to Biørn (2017), panel data are data that contain observations on different cross sections across time. Extensively utilized for structuring data in the healthcare sector, panel data refers to datasets that contain repeated observations on a selection of variables from a set of observation units, covering simultaneously the temporal and the spatial dimension. There are several dataset sources that use panel data structure like World Health Organization (WHO) (WHO, 2024), Global Burden of Disease (GBD) (GBD, 2024), World Bank (WBOD, 2024), Eurostat (EUROSTAT, 2024), Organisation for Economic Co-operation and Development (OECD, 2024) and Demographic and Health Surveys Program (DHS, 2024)

Health can be measured on three major parameters: Physical, Psychological, and Nutritional (SALIAN; KUMAR, 2022). Regarding these three main parameters, the authors highlight the importance of maintaining hygiene and having adequate sanitation. Therefore, to explore the possibilities of scenarios and the results of our proposed pipeline, four experiments were proposed involving different areas of health that consider these

parameters:

1. *Sexual health*: prediction of the number of people living with HIV.
2. *Nutritional health*: prediction of the prevalence of anemia in children.
3. *Safe sanitation*: prediction of population using at least basic sanitation services.
4. *Mental health*: prediction of age-standardized suicide rates in men.

After these experiments, three case studies were proposed to demonstrate the system's potential in studying a certain indicator. The first two case studies were applied to WHO data (collected by our system), while the third was applied to another data source that also follows the panel data structure. Therefore, the chosen indicators referred to: prevalence of anemia in children in the countries of the Americas in 2019; age-standardized suicide rates in men in the countries of the Africa in 2019; and influence of tuberculosis and hypertension in prevalence of anemia in children in India in 2021 (using Global Burden of Disease data).

To complete this main objective, we consider the following specific objectives:

a) To develop the data collection step, creating a consistent dataset in the panel data structure in the health sector, using mainly Application Programming Interface (API) and data analysis techniques.

b) To develop the pre-processing and data mining steps of the AutoML pipeline for classification problems, using development, data analysis and Machine Learning techniques.

c) To integrate LLM into our pipeline and design the system to enable the implementation of domain understanding and post-processing steps, helping the user understand questions about the dataset and assisting in the interpretation of the results found by our AutoML during the knowledge discovery processes.

d) To evaluate the AutoML proposal in the scenarios of sexual health, nutritional health, safe sanitation and mental health, comparing it with other AutoML frameworks found in the literature, and to propose case studies to demonstrate its potential in the study of certain indicators, showing how our AutoML proposal could be used in real contexts by expert and non-expert users.

1.3 Hypothesis

According to Hu and Huang (2017), AutoML refers to proposing an automated pipeline of Machine Learning (ML) steps so that no (or little) manual effort is required. Thus, AutoML aims to allow non-experts to apply Machine Learning techniques to solve specific tasks without requiring technical knowledge or prior knowledge about the context (FEURER et al., 2018).

Although there are several AutoML systems and works that address this context, according to Alsharef et al. (2022), finding a procedure that automates the entire ML process for predictions is not yet a mature field of research. This is mainly evident in the steps of domain understanding, dataset creation, pre-processing and post-processing, because the current set of AutoML solutions still struggles to effectively handle these steps (TRUONG et al., 2019). This is even worse in healthcare solutions, as despite the growth of research in the area of AutoML, few studies address the application of this technique in this area (LUO, 2016b).

According to Tsai et al. (2023), the integration between AutoML and Large Language Models (LLM) technologies still remains relatively unexplored. There is some research focusing on using LLM for small tasks in AutoML. However, the potential of using these models to help automate the entire AutoML pipeline has not been extensively studied.

With these questions open, our main hypotheses follow:

1. An AutoML pipeline supported by data analytics and Machine Learning techniques and enhanced by an LLM enables the development of a solution that covers all steps of AutoML.
2. The integration of LLM techniques into an AutoML pipeline enables the implementation of steps that have been relatively unexplored, such as domain understanding and post-processing.
3. An AutoML pipeline supported by LLM techniques allows the development of a more transparent solution, which is still a challenge for these solutions and a barrier to the use of these tools by healthcare professionals.

1.4 Contributions

The main contribution of this thesis is an AutoML pipeline capable of collecting, pre-processing, selecting models and hyperparameters, forecasting and analyzing data in the health area. To achieve this, it requires data analysis, Machine Learning algorithms,

a Large Language Model (LLM), and a dataset organized in a panel data structure, thus allowing a specialist or non-specialist user to parameterize it for different health contexts, democratizing the use of predictive models in the healthcare sector, supporting the use of Machine Learning techniques by professionals with different levels of technical knowledge.

Two relevant characteristics support the main contribution as follows:

1. Transparent process that allows users to interact and configure the system when necessary and all results and decisions adopted by the system at each step are presented to the user. This opens up opportunities for users who are not experts in Machine Learning and the problem domain to apply a full or partial Machine Learning solution.

2. Process that includes all AutoML steps dedicated exclusively to knowledge discovery in the healthcare sector, an area still little explored in the context of AutoML. Thus, with a solution for all steps, this process also includes the steps of data collection, domain understanding and post-processing, which are still little explored and analyzed in the literature.

In addition to these main and specific contributions, this thesis provides the following **innovation contribution**: exploration of the potential for applying Large Language Model (LLM) techniques and algorithms to automate certain steps of the Machine Learning pipeline, supporting a change in the way users (including non-experts) interact with AutoML systems. These techniques enhance configuration processes, accessibility, and the interpretability of results.

Finally, this thesis provides publicly available **technical contributions** for user experimentation on our website*. This is a web application to configure and execute our AutoML system, i.e., **He**althcare Data - **A**utomated Machine **L**earning (HEAL), covering all AutoML steps and presenting the results obtained in each of them. The main screens of our solution's interface can be found in the Appendix A. Unlike most solutions in the literature that adopt a black box approach, our transparent solution allows user interaction at each of the AutoML steps, both for previous configurations and for analysis of the automatic choices made and the results found.

1.5 Rationale

This section presents the reasons for: the choice of the research area, the context of application, the problem, the objectives and the contributions in a context where there are beneficiaries.

*Our website can be found in our repository: https://github.com/cart-pucminas/HEAL

AutoML pipeline capable of covering all steps of the Machine Learning process, from data collection to post-processing, enables analysis in different knowledge discovery contexts, including healthcare (SARKER, 2022). Therefore, the thesis developed promotes several advances in the area of AutoML.

Furthermore, the process of knowledge discovery using Machine Learning requires Machine Learning and domain-experienced analysts, which makes the process highly time-consuming, expensive, and rare (ALSHAREF et al., 2022). This restricts the development of solutions based on Machine Learning to those who have the resources to hire the services of these professionals (CARVALHO; MENEZES; BONIDIA, 2024).

Given the popularity of this topic, there is a lack of well-trained professionals in Data Science (CARVALHO; MENEZES; BONIDIA, 2024). Considering all these factors, this has led to an even more growing demand for systems that automate the Machine Learning pipeline and, consequently, collaborate to make these solutions more accessible to non-experts (SCHMITT, 2023).

Therefore, this work proposes an AutoML pipeline capable of covering all these AutoML steps in the healthcare context automatically. In this way, domain-experienced analysts can apply Machine Learning without in-depth knowledge in this area and Machine Learning analysts can explore different contexts in the healthcare, without in-depth knowledge in the healthcare area. The fact that our solution is not a black box method offers more control to the user, makes it easier to understand what was done throughout the process, helps the user identify problems at each step and offers all information about all steps.

The consequence of this is more accessible solutions for applying Machine Learning techniques, in addition to making the process cheaper financially and reducing the time required for execution. Thus, this AutoML system has the potential to offer different possibilities for exploring the knowledge discovery process in different sub-areas of healthcare, which has a positive impact in this area, as it can help identify and prevent diseases, carry out prognostic interventions, help with public policies and modeling human behavior, bringing improvements to people.

Beyond this context, exploring the application of the Large Language Model (LLM) in the context of AutoML results in advancements in the context, as its comprehensive integration of LLM into the AutoML pipeline remains relatively unexplored (TSAI et al., 2023). Therefore, we explore the application of LLM at different steps of the AutoML pipeline.

1.6 Thesis outline

We organized the remaining chapters of the thesis as follows:

- Chapter 2 presents the background for the development of the work, defining important concepts for understanding the work from data analysis, panel data structure and knowledge discovery (methodologies, dataset pre-processing steps, Machine Learning algorithms, evaluation metrics for classification problems), reporting the challenges of discovering knowledge in datasets in the healthcare area, AutoML (concepts and pipeline steps) and Large Language Models (LLM).
- Chapter 3 presents the state of the art in knowledge discovery, from simple Machine Learning models to AutoML applied to the healthcare area. First, we provide an overview of how the knowledge discovery process is applied in the healthcare area. Subsequently, we present research that proposes solutions to automate a specific step in the AutoML pipeline (domain understanding and dataset creation; preprocessing; data mining or post-processing). Next, we present an overview of the main AutoML systems known and currently used on the market. Finally, we review related works that have explored the proposal or application of AutoML systems in healthcare. Additionally, we discuss the gaps in these solutions in addressing the problem outlined in this thesis. Furthermore, we summarize why our work advances in this context.
- Chapter 4 defines the methodology proposed and used in our work, presenting each of the main steps and the connection between them.
- Chapter 5 presents in detail our proposed pipeline and the development of the system according to the proposed methodology. In this chapter, the AutoML pipeline is explained step by step, showing the details of the method applied at each step and how it was developed.
- Chapter 6 presents and discusses the experiments and results. Firstly, four experiments were carried out involving different areas of health: sexual health, nutritional health, safe sanitation and mental health. Subsequently, three case studies were proposed to show the predictive capacity of the system and its contribution: prevalence of anemia in children in the countries of the Americas in 2019; age-standardized suicide rates in men in the countries of the Africa in 2019; and influence of tuberculosis and hypertension in prevalence of anemia in children in India in 2021.
- Chapter 7 presents the conclusions of the work, showing an overview of everything that was discussed, the limitations and challenges of the proposed AutoML pipeline and future work.

2 BACKGROUND

This chapter describes the background researched for the development of the work, covering basic and main concepts in the context of data analysis and knowledge discovery, Machine Learning, Automated Machine Learning (AutoML) and Large Language Models (LLM) used in the work.

Therefore, Section 2.1 (page 19) presents the main methodologies for knowledge discovery in databases, Section 2.3 (page 23) the main techniques and algorithms present in the dataset pre-processing steps, Section 2.4 (page 29) the main concepts of the Machine Learning algorithms used, Section 2.6 (page 36) presents a reflection on challenges of knowledge discovery in database in the healthcare area, Section 2.7 (page 38) presents the main concepts about AutoML and tasks in the AutoML pipeline and, finally, Section 2.8 (page 43) presents the main concepts of LLM.

2.1 Methodologies for Knowledge Discovery in Databases

According to Shamim, Shaikh and Malik (2010), Knowledge Discovery in Databases (KDD) is a process that searches for unknown patterns and hidden information in databases to generate new knowledge for decision making. This process has become increasingly important, mainly due to the amount of data that circulates daily due to advances in different areas of technology,the use of the Internet as a tool for disseminating information and the social transformations that have occurred in recent years.

Analyzing this data can bring several benefits in different areas and sectors, finding knowledge, changing the way we live, work and play, creating new opportunities and providing improvements.

The KDD process has five basic steps to transform data into useful, non-obvious knowledge or information: 1 - data selection and organization; 2 - pre-processing, the data is analyzed and goes through adequacy; 3 - data storage to facilitate the use of data mining techniques; 4 - data mining application; and 5 - interpretation and evaluation of the results, verifying if the generated information has validity for the proposed problem (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

In the context of Knowledge Discovery in Databases, different types of datasets are used depending on the nature of the problem at hand. One such dataset structure is panel data, which offers a combination of time series and spatial data, providing insights

into both temporal and spatial relationships.

2.2 Panel Data

Panel data, sometimes referred as longitudinal data, or combined time-series/cross section data are terms used in statistics for datasets that contains repeated observations on a selection of variables from a set of observation unit. Therefore, these observations cover simultaneously the temporal and the spatial dimension (BIØRN, 2017).

Example of groups that may make up panel data include countries, individuals, demographic groups, time series for production, factor inputs and others. Panel data is similar to times series data, in the point of view that both contains observations collected at a regular frequency, chronologically.

In order to represent individuals and temporal observations, panel data often refer to groups as i and time as t. For example, a panel data observation Y_{it} is observed for all individuals i (1 to N) across all time periods t (1 to T). Table 1 shows an example of a structure used in panel data (Table 1.a), where Group and Time Period represent the spatial and temporal dimensions, respectively, and Value the value referring to the combination of these two dimensions, and in this table, there are 3 groups (1 to 3) and 2 time periods (1 to 2). Table 1.b shows an example with values following the same structure as Table 1.a, in which the spatial dimension is represented by countries (Brazil, Argentina and Peru), the temporal dimension in years (2018 and 2019) and the data related to these dimensions are Systemic Arterial Hypertension (SAH) and Anemia.

Table 1 – Example of Panel Data Representation

(a) Example of a structure for representing panel data with the spatial dimension (Group), the temporal dimension (Time) and the value related to these two dimensions (Value)

Group	Time	Value
1	1	$Y_{1,1}$
1	2	$Y_{1,2}$
2	1	$Y_{2,1}$
2	2	$Y_{2,2}$
3	1	$Y_{3,1}$
3	2	$Y_{3,2}$

(b) Example of panel data with real indicators, spatial dimension (Country), temporal dimension (Year) and the value related to these dimensions being Hypertension (HTN) and Anaemia

Group Country	Time Year	Value1 HTN	Value2 Anaemia
Brazil	2018	47.8%	16.2%
Brazil	2019	47.9%	16.1%
Argentina	2018	53.4%	11.8%
Argentina	2019	54.0%	11.9%
Peru	2018	22.4%	20.4%
Peru	2019	22.8%	20.6%

In addition to the World Health Organization (WHO) (WHO, 2024)[†] and Global

[†]https://www.who.int/en/. Accessed October 20, 2024

Burden of Disease (GBD) (GBD, 2024)[‡] databases used in this work, several other sources provide detailed information on economic, social and health indicators, organized in the panel data structure, thus organized into specific temporal and regional dimensions (WBOD, 2024; EUROSTAT, 2024; OECD, 2024; DHS, 2024; ILO, 2024; LIS, 2024; WVS, 2024).

The World Bank (WBOD, 2024)[§] provides economic and development indicators for countries, offering free access to these data through World Bank Open Data. Another example of a database is Eurostat (EUROSTAT, 2024)[¶], which has data in a panel data structure on the European Union, which includes economic, demographic and social indicators.

The Organisation for Economic Co-operation and Development (OECD) (OECD, 2024)[‖] provides information on public policies covering more than 100 countries worldwide. The Demographic and Health Surveys (DHS) (DHS, 2024)[**] Program has collected, analyzed, and disseminated representative data on population, health, HIV, and nutrition about different countries.

The International Labour Organization (ILO) (ILO, 2024)[††] provides data on labour statistics for many countries around the world, including information such as unemployment rates, working conditions and productivity. The Luxembourg Income Study (LIS) (LIS, 2024)[‡‡] provides datasets on wealth, employment and demographic data for many countries. Finally, the World Values Survey (WVS) (WVS, 2024)[§§] provides data in panel data format on political trust, climate change and social tolerance. Table 2 shows the most important data sources found in the panel data structure.

The Brazilian Institute of Geography and Statistics (IBGE) (IBGE, 2024)[¶¶] is responsible for collecting, analyzing and disseminating geographic and socioeconomic data from Brazil, essential for the development of public policies and academic studies. Among the various data possibilities made available by IBGE, there are structured datasets such as panel data, such as: Continuous National Household Sample Survey (Continuous PNAD) and Consumer Expenditure Survey (POF). The Continuous PNAD provides socioeconomic information on the Brazilian population, including data on work, income, education and housing. The POF provides detailed data on expenses, income and living conditions of Brazilian families, collected periodically.

[‡]https://www.healthdata.org/research-analysis/gbd. Accessed October 20, 2024
[§]https://data.worldbank.org/. Accessed October 20, 2024
[¶]https://ec.europa.eu/eurostat. Accessed October 20, 2024
[‖]https://www.oecd.org/en/data.html. Accessed October 20, 2024
[**]https://dhsprogram.com/. Accessed October 20, 2024
[††]https://www.ilo.org/data-and-statistics. Accessed October 20, 2024
[‡‡]https://www.lisdatacenter.org/. Accessed October 20, 2024
[§§]https://www.worldvaluessurvey.org/WVSContents.jsp. Accessed October 20, 2024
[¶¶]https://www.ibge.gov.br/estatisticas/downloads-estatisticas.html. Accessed October 20, 2024

Finally, the Information Technology Department - DATASUS - (DATASUS, 2024)*** is responsible for the collection, processing and dissemination of public health data in Brazil. Its datasets include epidemiological, hospital, mortality and vaccination coverage data, among others, and many are organized as panel data.

Table 2 – Main area and issues addressed in major data sources using panel data structure

Source	Area	Examples of issues that can be explored
WHO	Health	Food security (child malnutrition, overweight patterns and food security), hygiene and sanitation (disease outbreaks and hygiene campaigns), sexual and reproductive health (reproductive health services)
GBD	Health	Non-communicable diseases (analysis of cardiovascular diseases, diabetes, and obesity), communicable diseases (infectious disease outbreaks and vaccine impact), mental health (depression and anxiety disorders).
The World Bank	Economy and Development	Education (prediction of the impact of educational policies), infrastructure (impact of transportation investments), poverty (modeling of poverty eradication and income inequality analysis).
Eurostat	Socioeconomic Statistics of the European Union	Economy and finance (prediction of economic growth, analysis of inflation and deflation, and unemployment rate prediction), population and social conditions (population aging, gender inequality, and study of poverty and social exclusion), environment (energy consumption and social impact).
OECD	Economy and Public Policies	Labor market (employment rate prediction), innovation and technology (modeling of innovation incentive policies), social welfare (impact of pension reforms).
DHS	Demography and Reproductive Health	Maternal and child health (maternal and child mortality), food security (malnutrition analysis).
ILO	Labor and Employment	Working conditions (study of workplace accidents and occupational safety), gender inequality in the workplace (gender pay disparity).

***https://datasus.saude.gov.br/informacoes-de-saude-tabnet/. Accessed October 20, 2024

LIS	Income and Inequality	Income inequality (analysis of wealth concentration and impact of income redistribution policies), consumption and savings (modeling of consumption patterns).
WVS	Social Values and Political Behavior	Culture and religion (analysis of changes in cultural values and study of the influence of religion on political behavior), political behavior (prediction of electoral patterns and analysis of political polarization).
IBGE	Socioeconomics, Demographics and Industry	Employment trends and social mobility (analyzing labor market dynamics, predicting unemployment trends, and assessing social mobility across regions) and economic inequality (Measuring and predicting patterns of income and consumption inequality across demographic groups and regions).
DATASUS	Health	Epidemic Prediction and Control (forecasting outbreaks based on hospitalization and disease incidence data), chronic disease risk factors (Identifying and modeling risk factors for chronic diseases - e.g., diabetes, hypertension - aiding preventive health policies, and resource allocation in public health (optimizing distribution of health resources to underserved areas).

2.3 Dataset Pre-processing Steps

For all problem categories, pre-processing steps are important to prepare the basis for performing calculations, avoiding future problems or errors. These steps include: removal or imputation of missing data, attribute selection and analysis, outlier analysis and dataset imbalance. According to Mandhare and Idate (2017), most of the time, approximately 10% of the dataset is incorrect, has missing data or outliers.

According to Austin et al. (2021), the missing data analysis phase consists of finding and processing data with no values in the dataset. After finding data without values, processing can be performed in two ways: eliminating data from the dataset or imputing values using a specific metric. This decision can be made by analyzing the percentage of missing data and the importance and influence of this data for the knowledge discovery.

The selection and analysis of attributes consists of a study with the application of techniques to seek information about the attributes and their importance for that context of knowledge discovery (ZEBARI et al., 2020). Thus, searching for important information that is implicit in the dataset can be useful in data analysis for knowledge discovery.

In this context, with the aim of defining the best attributes in a dataset, the Information Gain (KENT, 1983) technique allows you to calculate a relationship of Information Gain in relation to the output of each attribute, allowing you to compare attributes and define which ones can add greater Information Gain to Machine Learning algorithms.

According to Theng and Bhoyar (2024), redundant data can bias the model or increase complexity unnecessarily. Thus, redundancy elimination focuses on removing duplicate information, which is done by identifying repeated records, correlating variables, or applying redundant data elimination algorithms.

The correlation matrix technique quantifies the relationship between two attributes in a dataset, ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation), with 0 indicating no correlation (ASGARI; MIRI; ASGARI, 2022). Using some coefficient, such as Pearson's (BENESTY et al., 2009), this matrix provides insights into how variables behave within a dataset.

Typically, outlier analysis occurs after analysis of missing data. In the area of data analysis, outliers are information that has discrepant characteristics in relation to other data in the dataset, and, therefore, can affect the results obtained in the knowledge discovery process. Furthermore, they can also offer important information about the characteristics of the dataset. It is extremely important to find the outliers and carry out the necessary treatment (MANDHARE; IDATE, 2017).

Visual methods are used to represent data and, consequently, represent outliers, such as histograms and boxplots. These methods are good for visualizing the data and outliers, but do not allow them to be manipulated in the dataset.

Thus, statistical methods are important to standardize the calculations to find and process this data. Among these methods, the Interquartile Range (IQR) can be used to find the most discrepant elements in the dataset (DASH et al., 2023). These are found by subtracting the highest quartile from the base by the lowest quartile, according to the Equation 2.1:

$$IQR = Q3 - Q1 \tag{2.1}$$

According to this statistical method, all x values of the base that respect Equations 2.2 or 2.3:

$$x < Q1 - 1.5 \times IQR \tag{2.2}$$

$$x > Q3 + 1.5 \times IQR \tag{2.3}$$

Another important factor to be addressed mainly in multiclass problems is class balancing, which is important mainly because it is a critical component for forecasting effectiveness. In contexts where the distribution of classes is unbalanced, that is, when there is a significant disparity in the number of instances between different classes, the imbalance can result in poor model performance.

This imbalance can negatively impact the model's ability to generalize patterns present in the minority classes, resulting in favorable predictions for the majority classes. Therefore, strategies to balance the dataset are an essential approach to mitigate such disparities and improve the robustness and reliability of predictions.

Among the techniques used to balance the dataset, two are most used: oversampling and undersampling. Oversampling involves reproducing the instances of the minority class, while undersampling aims to reduce the number of instances of the majority class. Both strategies aim to balance class distributions, allowing the model to learn the patterns present in the minority classes with greater quality.

The choice between these techniques must consider the specific characteristics of the dataset, since oversampling can increase the risk of overfitting, while undersampling can lead to the loss of valuable information. A detailed understanding of these techniques is crucial to improving the effectiveness of predictive models.

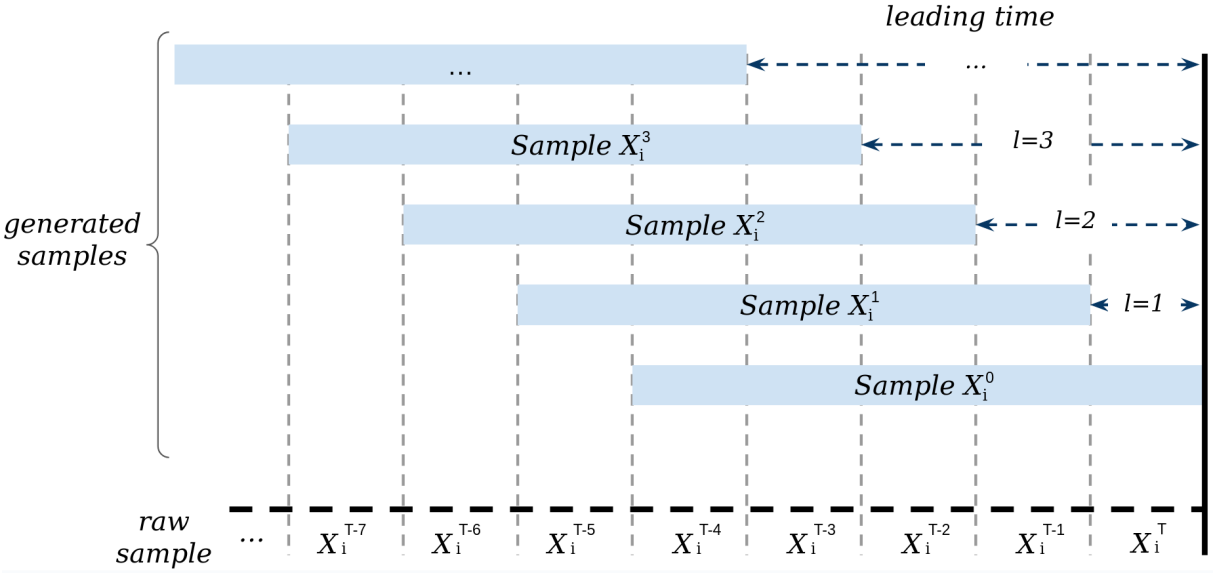
Problems involving temporal dimension, such as time series and panel data, also suffer from the class imbalance problem. Therefore, Zhao et al. (2022) proposed T-SMOTE, an oversampling approach derived from SMOTE (CHAWLA et al., 2002) that generates synthetic examples for the minority classes without compromising the existing temporal dependence.

Unlike traditional SMOTE, which does not take this temporal dimension into account, in T-SMOTE synthetic examples are created in a way that respects the temporal order of the data. To do this, it applies a sliding window over the time series, generating new examples only within consecutive time windows, which preserves the temporal structure and avoids the creation of unreal points that would violate this dependence.

Figure 1 illustrates the sliding window behavior for generating synthetic examples by T-SMOTE. In this figure, it is possible to observe a sliding window of size 5, so that it will always generate synthetic examples repeating 5 time units and sliding between them until the last time unit. This way, new examples are generated only within these time windows.

To ensure that estimated performance during training is a robust reflection of the model's ability to generalize to new data, validating Machine Learning algorithms is a critical step in the model development process. Using appropriate validation techniques

Figure 1 – Example of how T-SMOTE works, using a sliding window and generating samples with different execution times



Source: Based on Zhao et al. (2022).

is critical to mitigating the risk of overfitting, where the model memorizes training data instead of learning general patterns.

Among validation techniques, the holdout method, which divides the dataset into training and testing sets, offers an initial approach. However, to better evaluate the stability of the model in the face of different partitions, the k-fold cross validation technique presents a valuable alternative, subdividing the dataset into k folds and iterating over them for evaluation.

The dataset is divided into two subsets: training data and test data. The first is divided into k subsets, and in each division k-1 of these k parts are used as training, while one of these k parts is used as validation. Finally, after performing the k divisions and defining the best combination of parameters, the model is applied to the test data, which is a unique dataset not used in the training steps.

These techniques not only improve the reliability of performance estimates, but also promote transparency and generalizability of models in complex scenarios. The k-fold cross-validation technique offers a more comprehensive approach by partitioning the data into k mutually exclusive subsets, allowing each subset to act as a test set in one of the k iterations.

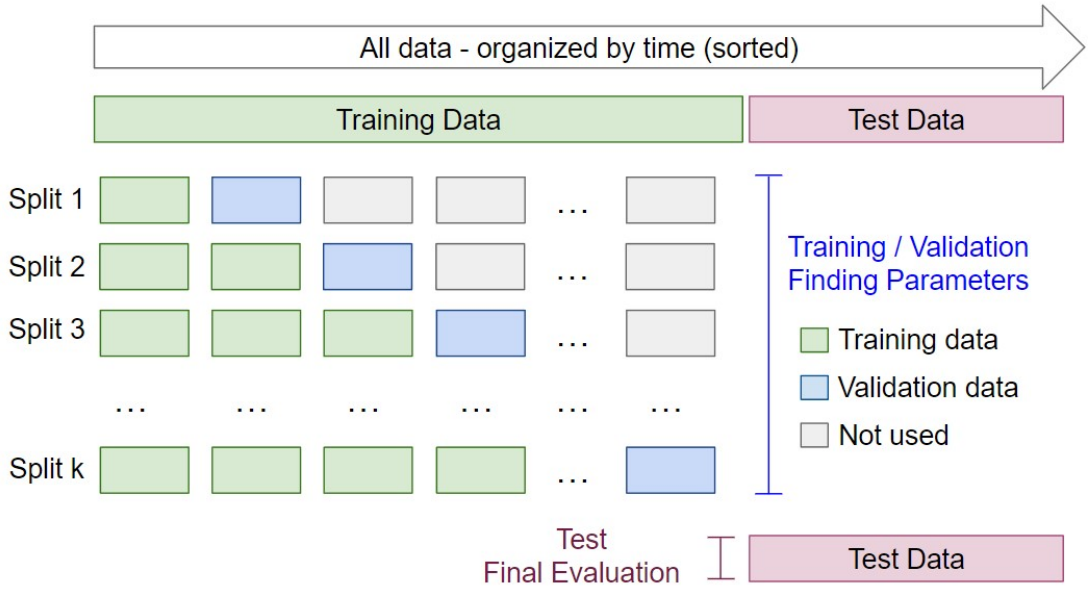
According to Roberts et al. (2017), although the k-fold cross-validation technique is useful in several contexts, when the data has a temporal dimension and is classified as a time series, the application of cross-validation is not trivial. In these contexts, it is

important to ensure that future data is not used for training, being a temporal dependency between observations, and it is important to preserve this relationship during testing.

In these contexts of data with a temporal dimension, there are some alternatives to k-fold cross-validation: time series split cross validation and blocked cross validation. The first starts with a small subset of data for training and frequency prediction for subsequent data points. The same predicted data points are then included as part of the next training dataset, and subsequent data points are predicted by evaluating each of these interactions (MOHAMMED et al., 2021).

Figure 2 illustrates this technique, in which the data must be initially ordered chronologically and later divided into training and testing, repeating the chronological question to separate the test data being the most recent. The training data is then split into each division as training and validation, with the validation set always needing to be the most current, while the training set corresponds to all data chronologically prior to the validation set. Data after validation is not used during this split.

Figure 2 – Time Series Split Cross Validation Representation



Source: Elaborated by the Author.

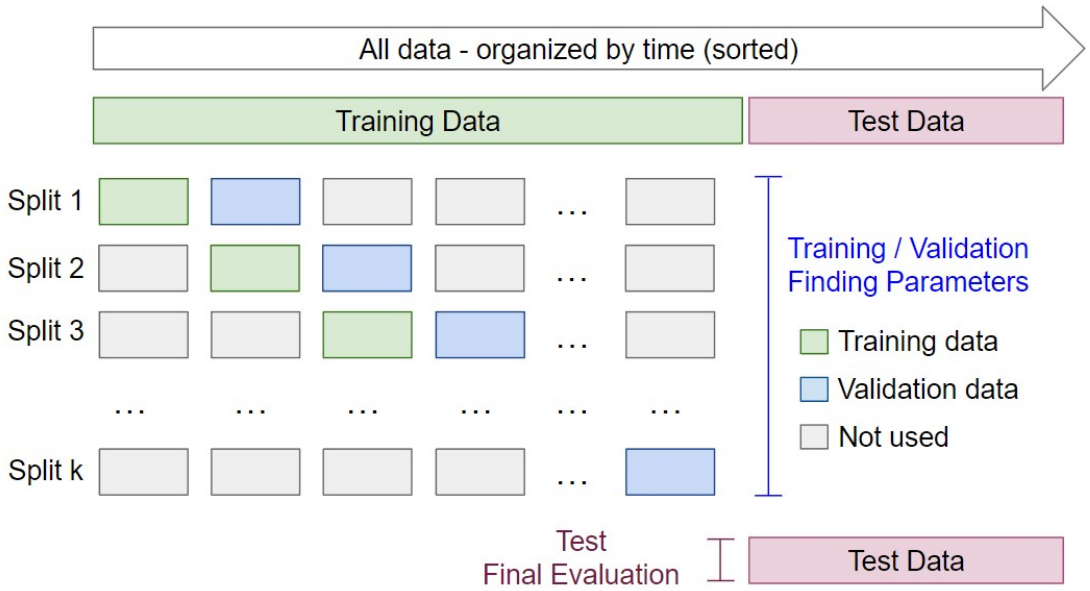
If the time interval were years (for example from 2000 to 2024), an example could be: initially separate 2024 for testing and in the first iteration, the technique starts using data from 2000 to 2010 for training and validates in 2011. After that, it uses data from 2000 to 2011 for training and validation in 2012, and enrolls until the validation data reaches the limit (until 2023 is used as validation). These training and validation steps are used to find the best combination of hyperparameters to perform the final test on data not yet known in the training steps.

The second option, Blocked Cross Validation, is a variation of the first. In this method, the window size is determined in advance, that is, a time interval for training data and another for validation data, and they may or may not be the same size. Regardless of the size, the most important thing is to apply the windows always respecting the chronological order of the data: training always precedes the validation data.

The difference from the previous method is that instead of using just one split, you will now slide both windows forward and repeat the process, creating something similar to traditional cross validation, as the training data will always have the same set size, something that grew in the previous technique (SULANDARI et al., 2024).

Figure 3 illustrates the representation of this technique. Again, all data is ordered chronologically and test data is separated as the most current. In the training data, it is necessary to divide it by looking for the set used for training and validation. The issue of chronological order needs to be respected, always maintaining a training set that precedes the validation set. Any other data before the training set or after the validation set is not used at this step.

Figure 3 – Blocked Cross Validation



Source: Elaborated by the Author.

If the time interval were years (for example from 2000 to 2024), an example could be: initially separate 2024 for testing and in the first iteration, the technique starts using data from 2000 to 2010 for training and validates in 2011. After that, it uses data from 2001 to 2011 for training and validation in 2012, and enrolls until the validation data reaches the limit (until 2023 is used as validation). These training and validation steps are used to find the best combination of hyperparameters to perform the final test on

data not yet known in the training steps.

2.4 Machine Learning Algorithms

According to Shalev-Shwartz and Ben-David (2014), Machine Learning algorithms are capable of learning during the steps and improving based on the experiences gained. Two aspects of a given problem may require the use of these algorithms: the complexity of the problem and the need for adaptability.

The two commonly known categories of Machine Learning algorithms are supervised learning and unsupervised learning (SHALEV-SHWARTZ; BEN-DAVID, 2014). In supervised learning, the user knows what they are looking for, so they specify a target variable and the machine learns from the data, identifying patterns. In unsupervised learning, the user does not know what he is looking for and the machine is responsible for defining this for him, looking for what the data provided have in common.

In the supervised method, the algorithm's tasks are: classification and regression. In the first task, the algorithm predicts which class a data instance is classified into, while in the second, it predicts a numerical value. In unsupervised learning, there are two algorithm tasks: clustering and density estimation. In clustering, the algorithm searches and groups similar data, while in density estimation, the algorithm finds statistical values that describe the data (SHALEV-SHWARTZ; BEN-DAVID, 2014).

2.4.1 Artificial Neural Networks (ANN)

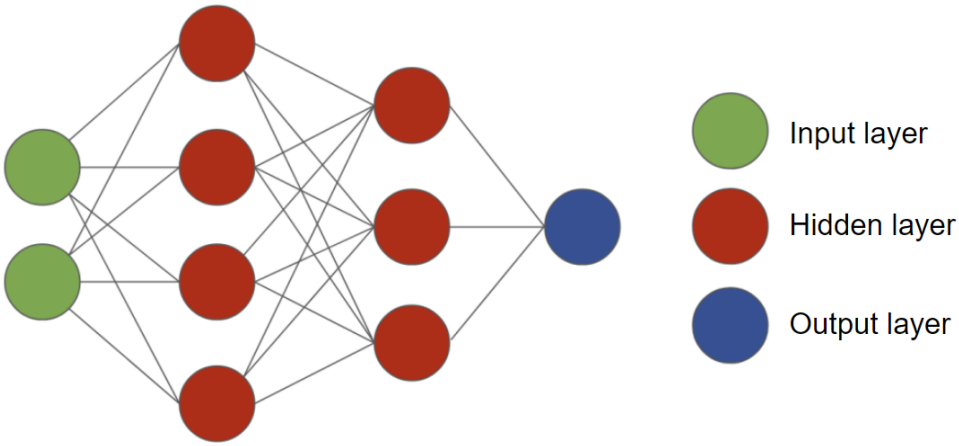
According to Haykin (2009), Artificial Neural Networks (ANN) are an efficient Machine Learning approach, which uses techniques developed based on the behavior of neurons in the human brain to learn from diverse information and be able to recognize hidden patterns, find correlations between raw data and predict future scenarios. Figure 4 illustrates the representation of a simple Artificial Neural Network, having an input layer, an output layer and, between them, two hidden layers.

According to Uhlmann et al. (2021), ANN has a large number of hyperparameters and, generally, the choice of these is made based on user experience. Among the main hyperparameters of this algorithm, we can mention:

a) Number of Layers (Depth): determines its depth and ability to learn complex data representations. Deeper networks can capture more complex relationships, but they are also more sensitive to overfitting and require more data for effective training.

b) Number of Neurons per Layer: directly affects the network's ability to process in-

Figure 4 – Artificial Neural Network Representation



Source: Elaborated by the Author.

formation. An insufficient number of neurons can lead to underfitting, while too many can cause overfitting and increase the computational cost.

c) Activation Function: determines the output of each neuron based on its inputs. The choice of activation function influences the network's ability to capture nonlinearities and affects the speed of training convergence.

d) Learning Rate: controls the size of adjustments to network weights during training. High learning rate can cause oscillations and divergences, while too low rate can result in slow convergence.

e) Number of Epochs: determines how many times the algorithm will run on the entire training set before finishing training. Too many epochs can lead to overfitting, especially if early stopping is not used.

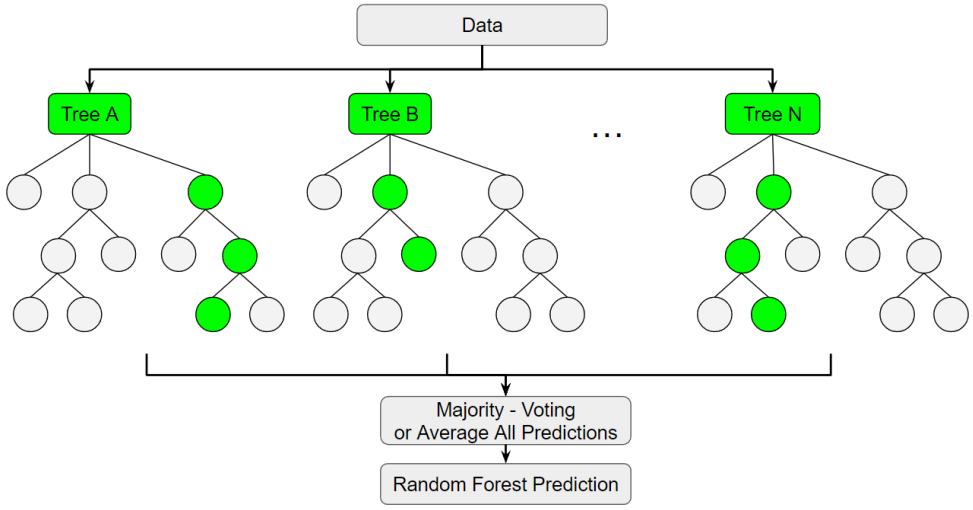
2.4.2 Random Forest (RF)

According to Breiman (2001), Random Forest (RF) is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. This approach is based on estimates and probabilities associated with the outcomes of competing courses of action, with each course of action being weighted by the probability associated with it. After calculating the results, the alternative that represents the highest expected value is chosen.

The Random Forest supervised learning algorithm is an ensemble method that creates a combination of tree predictors to obtain a more accurate prediction. This algorithm can be used in both classification problems and regression problems, presenting itself as a good solution in these contexts. Figure 5 illustrates a representation of the

Random Forest algorithm using n trees in which each tree defines the prediction class and, finally, a majority vote is made to define the final class predicted by the algorithm (in classification problems) or an average calculation is applied (in regression problems).

Figure 5 – Random Forest Representation



Source: Elaborated by the Author.

According to Probst, Wright and Boulesteix (2019), most Random Forest hyperparameters are called tuning parameters, and their values need to be carefully optimized as the optimal values depend on the dataset in question. Among the main hyperparameters of this algorithm, we can mention:

a) Number of Trees: it is essential to balance accuracy and computational efficiency when defining the number of trees in a Random Forest model. A larger number increases the robustness of the model, contributing to reducing the variance of predictions and stabilizing performance. However, this also implies an increase in computation time and memory requirements.

b) Maximum Depth of Trees: deeper trees can capture more complex relationships between features and the target variable, but are also more sensitive to overfitting.

c) Minimum number of samples to split a node: higher values prevent the creation of nodes that represent very specific relationships, potentially reducing overfitting.

d) Minimum number of samples in a leaf node: adjusting this parameter helps to smooth the model, avoiding very specific rules and, consequently, reducing overfitting.

e) Maximum number of features considered for split: choosing an appropriate number of features can significantly influence the diversity among trees and therefore model performance.

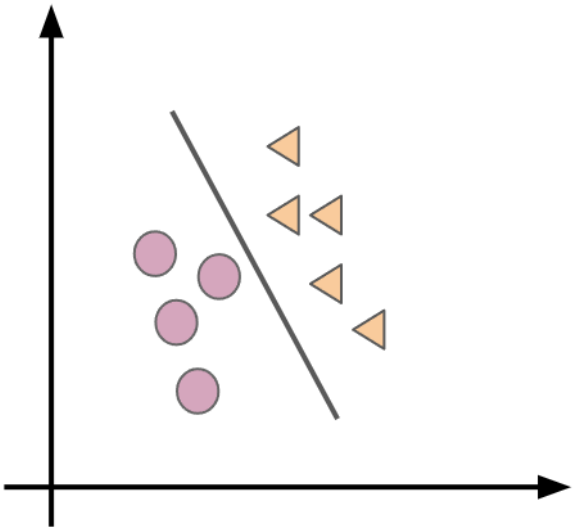
f) Criterion Function: defines the function used to measure the quality of a division. The choice of criterion can affect the model's performance, depending on the nature of the problem.

2.4.3 Support Vector Machine (SVM)

According to Vapnik (1998), Support Vector Machine (SVM) implements the idea of mapping input vectors into a high-dimensional feature space through some non-linear mapping, chosen a priori. In this space, an optimal separating hyperplane is constructed to divide the dimensional space into distinct classes. To achieve this, the algorithm identifies critical points known as Support Vectors, which are instrumental in establishing the optimal hyperplane.

This algorithm can be divided into two categories: linear type SVM and non-linear type SVM. The first scenario occurs when the data needs to be linearly segregated, indicating that the dataset can be divided into classes separated by a straight line. In contrast, non-linear type SVM refers to when data must be segregated in a non-linear way, so straight lines cannot perform class division (BANSAL; GOYAL; CHOUDHARY, 2022). Figure 6 shows a representation of an example of applying linear SVM to a binary classification problem.

Figure 6 – Support Vector Machine Representation



Source: Elaborated by the Author.

Duan, Keerthi and Poo (2003) and Keerthi, Sindhwani and Chapelle (2006) discuss the importance of choosing optimal hyperparameter values for SVM and the importance of this task. Among the main hyperparameters of this algorithm are:

a) Kernel: SVM uses a kernel function to transform the input space into a higher-dimensional feature space where the data can be linearly separable. The choice of kernel directly affects the model's ability to capture non-linear relationships between data, being crucial for the model's performance in complex tasks.

b) Regularization Parameter (C): controls the trade-off between obtaining the widest possible separation margin and minimizing the number of classification errors. A higher C value implies a greater penalty for misclassifications, encouraging the model to better fit the training data, but may lead to overfitting. On the other hand, a lower value favors a wider margin and a more generalizable model, potentially at the cost of some classification errors.

c) Polynomial Kernel Degree (degree): when the polynomial kernel is selected, this hyperparameter defines the degree of the polynomial. A higher degree allows the model to capture more complex relationships between data, but can also increase the risk of overfitting.

2.4.4 K-Nearest Neighbors (KNN)

According to Bansal, Goyal and Choudhary (2022), in the context of supervised Machine Learning algorithms, K-Nearest Neighbors (KNN) also presents as a good technique for classification and regression problems, being predominantly used to classify objects. From a given dataset, it predicts the connection between unknown data and existing data by classifying data points based on the arrangements of their neighbors.

Setting the appropriate value of K for the KNN algorithm is the main process in this algorithm and this process is called parameter tuning. A value of K that is too low can result in noisy outcomes, whereas excessively high values may lead to confusion. Generally, K is set at 5, meaning that, for the majority voting process, the five nearest neighbors of the new data point are taken into consideration. According to Bansal, Goyal and Choudhary (2022), the value of K can also be calculated according to Equation 2.4:

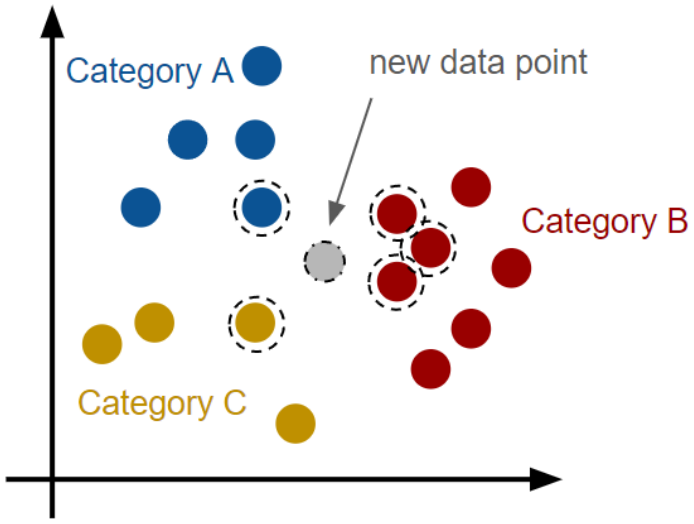
$$K = \sqrt{n} \tag{2.4}$$

where n is the overall count of data points.

Figure 7 illustrates an example of categorization using the KNN algorithm configured with K equal to 5 and for 3 categories, in which the new point will belong to class B, since, of the 5 closest neighbors, 3 belong to class B.

According to Wazirali (2020), hyperparameter tuning helps to get better accuracy in KNN algorithm. Among the main ones, we can mention:

Figure 7 – K-Nearest Neighbor Representation



Source: Elaborated by the Author.

a) Number of Neighbors (K): determines the number of nearest neighbors to be considered for class voting (in classification) or weighted average (in regression). Too low a value for K can cause the model to be sensitive to data noise, resulting in overfitting. A very high value can cause the model to consider a very large portion of the sample space, resulting in underfitting.

b) Weights of Neighbors: can be configured to consider all neighbors equally or to give more weight to closer neighbors. The choice between these options influences how the model balances between the characteristics of the closest neighbors and those that are more distant within the set of K neighbors considered.

c) Distance Metric: different metrics can lead to different sets of nearest neighbors. Selection of the appropriate distance metric should reflect the geometric nature of the data and can have a significant impact on model performance. Common metrics include Euclidean distance, Manhattan distance, and Minkowski distance.

2.5 Evaluation Metrics for Classification Problems

According to (LORENA; CARVALHO; GAMA, 2008), classification problems are described as the application of techniques to predict the class of new data from a domain containing data whose classes are known. When this problem dataset has only two classes, this is named binary classification problems. Whereas problem dataset that requires discrimination of examples into more than two classes is named multiclass classification problems.

Given this contexts, multiclass classification problems present more challenges than binary classification, for instance: the complex task of defining between multiclass categories, the demand for smarter modeling strategies and a more precise definition of evaluation metrics. These challenges are mainly a consequence of the need for models capable of capturing nuances and hidden patterns in the diversity of categories.

Defining accurate metrics for model evaluation is also a challenge in multiclass problems. For example, in binary classification problems, the accuracy metric is a sufficient representation of performance, while in multiclass contexts, accuracy can be misleading, especially in class imbalance scenarios.

Thus, the introduction of more refined metrics such as precision, recall, f1-score, becomes crucial to evaluate the specific performance of each class and the model as a whole.

Metrics such as accuracy and hit rate provide initial insight into the overall effectiveness of the model in correctly classifying instances. However, it is important to consider class distribution and the presence of imbalance, as these metrics can be misleading in scenarios where classes are not balanced.

Yacouby and Axman (2020) explain that additional metrics such as precision, recall and f1-score, offer a more detailed understanding of model performance, providing insights into the ability to discriminate between classes. According to the authors, when dealing with multiclass classification problems, metrics such as the confusion matrix and the area under the Receiver Operating Characteristic (ROC) curve stand out as essential tools for the comprehensive assessment of model performance in different aspects of classification.

Equation 2.5 shows the formula for calculating the precision of a class A in a multiclass problem:

$$Precision = \frac{TP_A}{TP_A + FP_A} \tag{2.5}$$

where, TP_A is the total True Positive and FP_A is the total False Positive. Equation 2.6 shows the calculation of the precision micro-average:

$$Precision_{micro-average} = \frac{TP_A + TP_B + ...TP_N}{TP_A + FP_A + TP_B + FP_B + ...TP_N + FP_N} \tag{2.6}$$

where TP_A, TP_B and TP_N is the total True Positive for each class and FP_A, FP_B and FP_N is the total False Positive for each class. Equation 2.7 shows the formula to calculate the recall of a class A in a multiclass problem:

$$recall = \frac{TP_A}{TP_A + FN_A} \tag{2.7}$$

where, TP_A is the total True Positive and FN_A is the total False Negative. Equation 2.8 shows the calculation of the recall micro-average:

$$recall_{micro-average} = \frac{TP_A + TP_B + ...TP_N}{TP_A + FN_A + TP_B + FN_B + ...TP_N + FN_N} \tag{2.8}$$

where TP_A, TP_B and TP_N is the total True Positive for each class and FN_A, FN_B and FN_N is the total False Negative for each class. Equation 2.9 shows the calculation of the F_w-score:

$$F_w score = \frac{(w+1) \times recall \times precision}{recall + w \times precision} \tag{2.9}$$

where coefficient w determines the balance between precision and recall, with high values favouring recall and low values favouring precision (CARVALHO; MENEZES; BONIDIA, 2024). Finally, when this value is equal to 1, it is equivalent to giving the same importance to recall and precision, the result in the f1-score can be calculated according to Equation 2.10:

$$F1score = 2 \times \frac{precision \times recall}{precision + recall} \tag{2.10}$$

In addition to metric definitions, class balancing strategies and advanced modeling techniques are often needed to deal with the complexities of these problems, ensuring that the model is able to effectively generalize to all categories present in the data.

Finally, it is important to highlight that understanding the challenges and distinctions between multiclass and binary classification problems is essential for developing good models. Addressing the challenges presented requires a more specific approach, integrating advanced modeling strategies with the appropriate selection of metrics and evaluation strategies.

2.6 Challenges of Knowledge Discovery in Dataset in the Healthcare Area

Although the availability of an increasing volume of data in different open sources on the internet brings new opportunities to discover knowledge and make decisions in different contexts (ALSHAREF et al., 2022), there are many challenges to finding relevant information, processing it and applying the Machine Learning process (SALEHIN et al.,

2024).

These same problems exist in the health sector (ALI et al., 2023; WARING; LINDVALL; UMETON, 2020), and possible discoveries of knowledge for preventing diseases, prognostic interventions, new treatments, and therapies, or to model human behavior (KAMBATLA et al., 2014) are lost daily due to the challenges in this Machine Learning process.

Other points that increase these challenges when working with health data are: the enormous diversity of data sources, which contain data with different structures and organizations; the non-standardization of most of these data; the privacy of much health-related information, and there is often sensitive data; and the enormous amount of unstructured data such as manuscripts and natural language (BANAN; FATAH; HAMAAMIN, 2022).

According to Barbudo, Ventura and Romero (2023), another significant challenge accompanying this issue, related to the large volume of data, is the complex task of extracting useful knowledge. This process involves multiple phases and demands extensive expertise in several fields, including pattern mining, statistics, data visualization, and Machine Learning.

Therefore, data scientists are responsible for identifying relevant data sources, automating the process of collection, pre-processing, analysis of acquired data, and a domain expert working together with this scientist is often important. Although health researchers are familiar with this data, the iterative process often requires a lot of time and effort on both sides (WARING; LINDVALL; UMETON, 2020).

In the face of these challenges, AutoML technologies offer an opportunity to make Machine Learning steps and the knowledge discovery process more accessible for users with limited Machine Learning experience (SALEHIN et al., 2024). However, despite the expanding research in the field of AutoML, there has been limited exploration of these tools' application within the healthcare sector (LUO, 2016b).

Among the various challenges of applying AutoML in the healthcare context, one of the main ones is building a diverse, representative and high-quality dataset (WARING; LINDVALL; UMETON, 2020). Another point that contributes to these challenges is that most of these AutoML tools do not perform the data collection and dataset generation process.

According to Waring, Lindvall and Umeton (2020), another big problem is the lack of transparency in these AutoML systems, such as which model configurations were analyzed, resulting in doubts about the automatic results. Therefore, if users lack trust in the system, they may hesitate to apply its results in a critical context such as healthcare.

In the next section, we will discuss important AutoML concepts and AutoML tasks.

2.7 Automated Machine Learning – Concepts and Pipeline Steps

In this context of KDD process, Automated Machine Learning (AutoML) is very important for the next generation of medical mining systems. This can be useful to allow users to automatically find knowledge discovery in large data repositories (CERQUITELLI et al., 2016). Therefore, this important and intelligent data mining process attracts many studies and many researchers are working on it (SHAMIM; SHAIKH; MALIK, 2010).

It is important to emphasize that these automated methodologies are not intended to provide more accurate results, but rather to enable expert or non-expert users to achieve reasonable results with minimal effort (RAJAN; SARAVANAN, 2008). According to Liu, Huang and Zhang (2011) and Campos, Stengard and Milenova (2005), this can save time and costs and help expert and non-expert to solve problems.

According to Barbudo, Ventura and Romero (2023), before generating the model, data pre-processing includes the following phases: dataset creation, data cleaning and pre-processing, and data reduction and projection. Subsequently, the data mining phase unfolds, encompassing both supervised learning contexts (such as classification and regression) and unsupervised learning (including clustering and outlier detection). Finally, post-processing entails the phases of knowledge interpretation and integration, reflecting the knowledge discovered throughout the process and its presentation to the user.

According to Salehin et al. (2024), four main metrics are important to evaluate the performance of an AutoML:

a) Accuracy: models must achieve high prediction accuracy, minimizing errors and maximizing the correctness of results generated by automated processes.

b) Efficiency: models need to be efficient in terms of time and computational resources, optimizing the total time needed to train models and make predictions.

c) Scalability: models need to be scaled up to handle large datasets and complex Machine Learning tasks, being able to work on increasing amounts of data and computational demands, without compromising efficiency.

d) Flexibility: Models should support multiple Machine Learning algorithms, techniques, and models, allowing the user to experiment with different approaches and customize automated processes to their preferences.

2.7.1 Tasks in AutoML Pipeline

Barbudo, Ventura and Romero (2023) defines the following steps for an AutoML model:

a) Pre-processing: domain understanding, dataset creation, data cleaning and pre-processing, and data reduction and projection.

b) Data mining: divided into supervised learning, including classification and regression, and unsupervised learning, encompassing clustering and outlier detection.

c) Post-processing: interpretation of knowledge and integration of knowledge, which act on the knowledge discovered in previous phases.

Waring, Lindvall and Umeton (2020) already defines five steps to consider AutoML: 1 - Data preparation and cleaning; 2 - Feature selection and Engineering; 3 - Model building and training; 4 - Hyperparameter Optimization; 5 - Model validation, selection and deployment. Tsai et al. (2023) also defines five steps, namely: data pre-processing, feature engineering, model selection, and hyperparameter tuning.

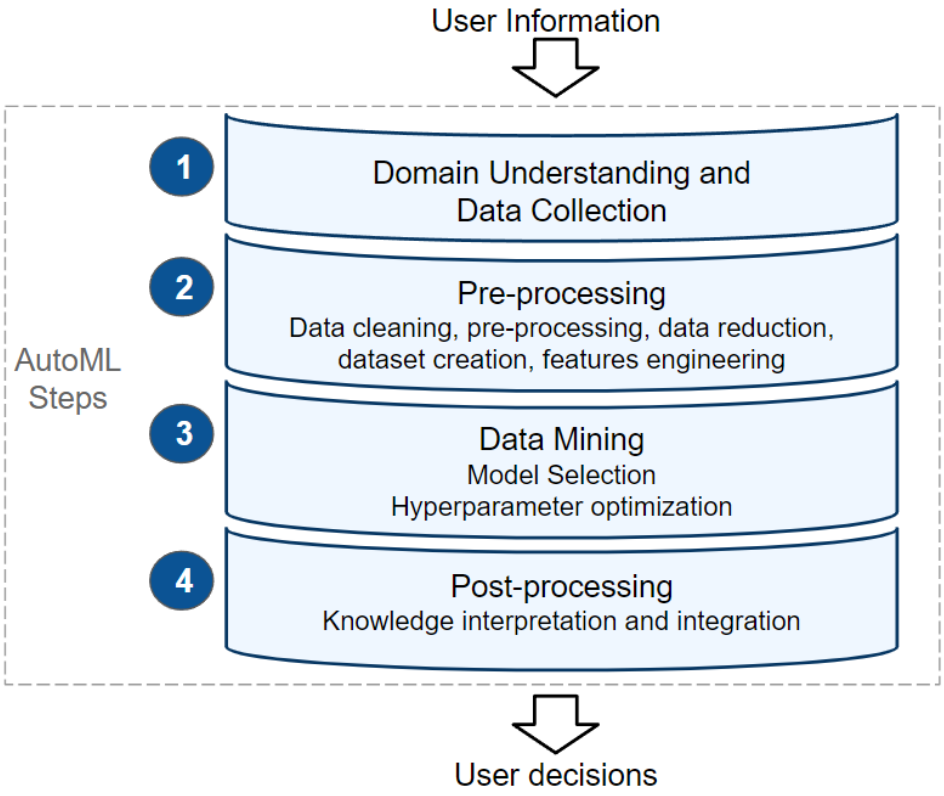
Valle, Mantovani and Cerri (2023) and Salehin et al. (2024) define the ML pipeline with the steps: pre-processing data, feature engineering, ML model search and the interpretability of results. The authors also state the importance of the user in defining the algorithms and possible hyperparameters to be tested.

Finally, He, Zhao and Chu (2021) also divided the AutoML pipeline into four steps: data preparation, feature engineering, model generation and model estimation. It is important to highlight that the authors subdivided the data preparation step into 3 substeps, namely: data collection, data cleaning and data augmentation. There is a consensus that high-quality datasets are of critical importance to the ML process. However, it is a challenging task to perform the first of these substeps (data collection), and there are two methods used to solve this problem: data searching and data synthesis. The first refers to searching for data on the web, while the second refers to the process of creating artificial data to replicate characteristics of real data.

From these steps defined in previous studies, we defined a pipeline of AutoML steps as shown in Figure 8. It should be noted that, while the goal is to automate the Machine Learning steps, it remains crucial for the tool's user to have the ability to input configuration information, such as context definition and certain algorithm parameter settings. It is also worth noting that the final decisions are always made by the user.

Step 1 - Domain Understanding and Data Collection: unlike the proposals presented in previous works, we divided the pre-processing into two steps, defined as 1

Figure 8 – AutoML Steps



Source: Elaborated by the Author.

and 2 in the figure. The first step is related to the applications of Artificial Intelligence (AI) techniques for domain understanding, information extraction and data collection. These steps can be performed from online data sources (such as websites, Application Programming Interface (API), social networks and others) or from digital documents (such as spreadsheets or text documents).

Step 2 - Pre-processing: the pre-processing step is responsible for treating this collected data and preparing it for application in Data Mining. Thus, this step includes data cleaning, outlier detection, dataset creation, data reduction and projection, and other steps. We add feature engineering to this broader context of pre-processing, where this step involves transforming and creating features from data to improve prediction and model performance (ALSHAREF et al., 2022).

Step 3 - Data Mining: finally, the data is ready to be used as input to Machine Learning algorithms. However, it is still important to define, based on the context of the problem (supervised or unsupervised), the most appropriate model or set of learning models for the specific dataset, testing different algorithms and their configurations. Within the context of these configurations, it is important to optimize the hyperparameters of

these Machine Learning models, and this adjustment is done automatically within this AutoML process.

For Machine Learning model selection, He, Zhao and Chu (2021) subdivide spatial search into two approaches: 1. Combined Algorithm Selection and Hyperparameter Optimization (CASH) and 2. Neural Architecture Search (NAS).

CASH is based on the creation of models from several algorithms and their hyperparameters, operating in a search space aiming to identify the ideal configuration for a given task. For example, in the KNN algorithm, the number of neighbors is a hyperparameter that influences the behavior and accuracy of the model. Therefore, by considering combinations of algorithms with their tuned hyperparameters, CASH can explore a diverse model landscape, improving its generalization ability across different distributions and data requirements.

On the other hand, NAS is a specialized method for the automatic design and optimization of neural network architectures (mainly Deep Learning). Unlike CASH, NAS explores a more complex search space, which includes potential neural network layers, connections, and other structural parameters. This search space represents the set of possible architectures that the optimization algorithm can explore, which may include variations in layer types, widths, depths, activation functions, and other architectural elements. NAS enables the discovery of new task-specific architectures that are difficult to design manually, as it systematically navigates possible network configurations to achieve optimal model performance.

When applying these methods to neural networks, the distinction becomes particularly relevant. While CASH optimizes traditional hyperparameters (such as learning rate, number of layers, and number of neurons per layer), it does not delve deeply into the internal structure of the network in the way NAS does. NAS goes beyond basic hyperparameters by optimizing the neural network's architecture itself, adjusting elements like the number and arrangement of layers, types of connections, and specialized configurations that can significantly impact performance.

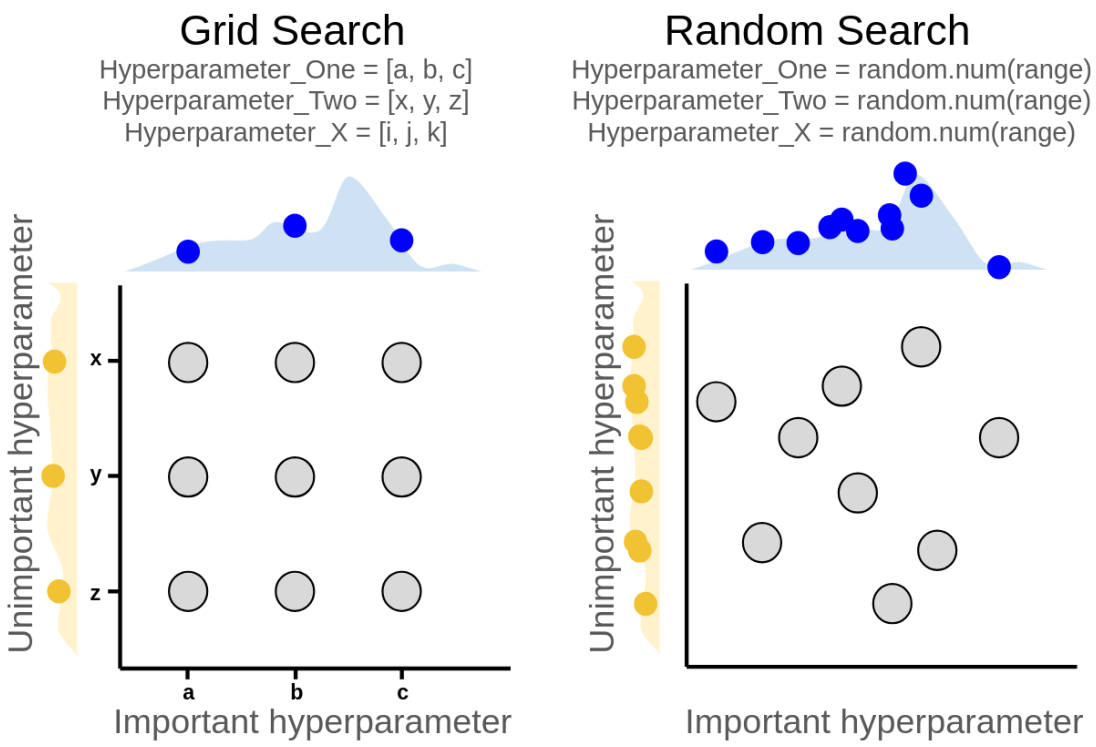
It is important to note that NAS is designed exclusively for neural networks due to its focus on architectural search within layered structures. Consequently, NAS is not applicable to algorithms like Random Forest, K-Nearest Neighbors, Decision Tree, and Support Vector Machines (SVM), as these algorithms do not have the layered, configurable architecture that NAS is designed to optimize. Instead, CASH is the appropriate approach for these algorithms, focusing on selecting and tuning their hyperparameters rather than altering structural configurations.

There are some techniques to optimize hyperparameters, such as Grid Search (GS) and Random Search (RS) (VALLE; MANTOVANI; CERRI, 2023). GS is the simplest

and least effective optimization technique, as it is a brute force algorithm that increases processing time proportionally to the increase in data dimensionality and the number of hyperparameters. From the definition of the search space (grid), that is, the list of possible combinations for the hyperparameters to be optimized, an exhaustive evaluation is carried out through training and evaluation of each combination in the grid. After this, the combination that resulted in the best performance according to a pre-defined metric is selected for the final configuration.

Unlike GS, RS randomly selects combinations of hyperparameters to test. This introduces a component of randomness into the search, which can lead to finding a good hyperparameter configuration faster and with fewer calculations. Figure 9 illustrates a comparison between the execution of Grid Search and Random Search, in which it is possible to observe the configurations of the tested hyperparameters.

Figure 9 – Grid Search and Random Search



Source: Based on Baratchi et al. (2024).

Step 4 - Post-processing: as a final step, there is post-processing, responsible for evaluating the results collected in the previous steps and working on them, seeking to generate visual information for the system user to make decisions.

2.8 Large Language Models (LLM)

Zhao et al. (2023) divides the life cycle of a Large Language Models (LLM) into three main steps: pre-training, fine-tuning, and inference. In the first step, the model is trained on a large corpus of unlabeled text to learn and produce useful sequence representations with a pre-textual task. According to the authors, this is the most expensive task and only a few groups and companies around the world can pre-train a basic model.

Fine-tuning specializes the model for specific tasks or domains by adapting the learned representations to concrete use cases, which is essential for the model's effectiveness in specific applications and is less computationally demanding. Inference is the final phase, where the LLM uses the acquired knowledge to generate text in response to specific linguistic tasks, with transfer learning being a key component in unlocking the potential of LLM in diverse applications.

According to Achiam et al. (2023), the various advances in the LLM area have allowed the development of powerful tools, such as ChatGPT (OpenAI, 2024), LLaMA (Meta, 2024) and Gemini (Google, 2024). These tools are distinguished primarily by their capabilities to reason, understand, interact, and perform new tasks based on instructions (ZHANG et al., 2023).

Among these tools, ChatGPT-4 demonstrates a deep understanding of natural language and the ability to produce coherent and contextually appropriate responses, opening up new application possibilities for challenging tasks involving data from different domains, such as image and text processing (ZHANG et al., 2023).

According to OpenAI (2024), GPT-4o (o for omni) provides much more natural human-computer interaction by accepting any combination of text, audio, image, and video as input and generating any combination of text, audio, and image output. It matches the performance of GPT-4 Turbo (previous version) on English text, with significant improvement on non-English text, while being much faster and 50% cheaper in API.

According to Tsai et al. (2023), although Large Language Models (LLM) have been used successfully in several different application contexts, their comprehensive integration into the AutoML pipeline remains relatively unexplored. The application of LLM in this context mainly focuses on individual tasks such as data pre-processing and feature engineering. This wastes the opportunity to automate the entire AutoML pipeline, from the data collection and preparation step to the post-processing step.

According to Tornede et al. (2023), among the applications of LLM in the context of AutoML, we can mention:

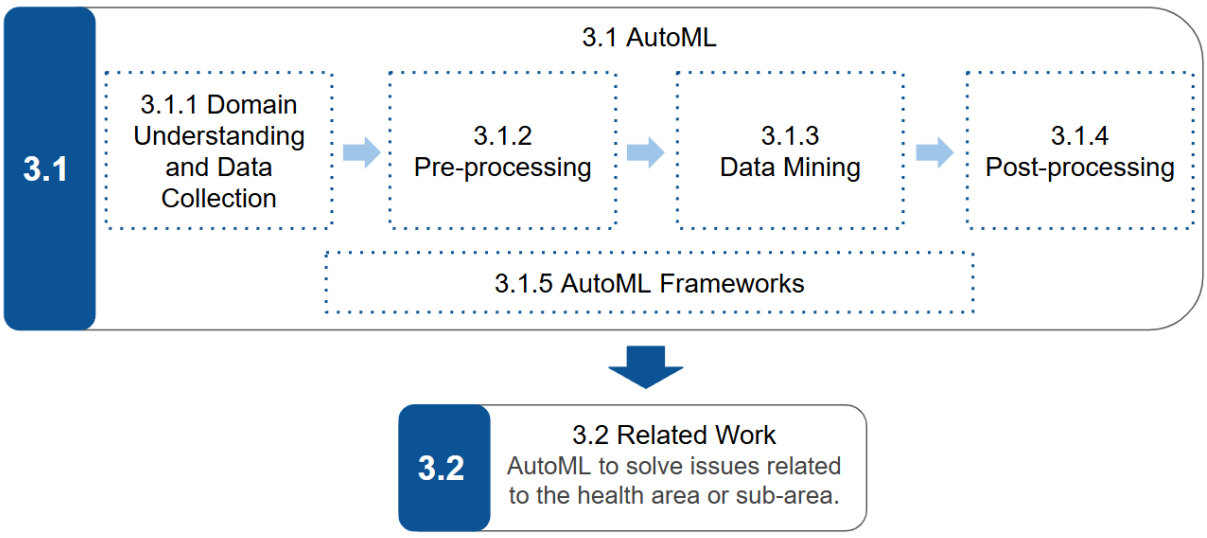
a) Faced with the challenge of interacting with AutoML systems for non-expert users, the use of Natural Language Processing (NLP) from the LLM context offers the possibility of changing the way humans interact with AutoML systems, both from the point of view of their configuration, as well as in the interpretation of the results.

b) AutoML systems need to be configured in the most appropriate way to reach their full potential, which often requires an expert. The capabilities of LLMs provide the opportunity to suggest an initial configuration of an AutoML system for a specific problem.

3 STATE OF THE ART OF MACHINE LEARNING SYSTEMS APPLIED TO HEALTHCARE

This chapter reports the state of the art of Machine Learning systems applied to healthcare, presenting concepts related to knowledge discovery and AutoML that are still under construction and represent the frontier of knowledge on this subject. Therefore, Section 3.1 presents works focusing mainly on works that presented solutions for the steps of the AutoML process: domain understanding and data collection (Subsubsection 3.1.1, page 46), pre-processing (Subsubsection 3.1.2, page 47), data mining (Subsubsection 3.1.3, page 47) and post-processing (Subsubsection 3.1.4, page 50). Subsequently, the main existing AutoML framework* are presented in Subsubsection 3.1.5 (page 50). Furthermore, we present work related to our in Section 3.2 (page 53), which are works that proposed the AutoML system to solve issues related to the area or subarea of health. Figure 10 illustrates the organization of this chapter related to the state of the art.

Figure 10 – State of the Art - Steps



Source: Elaborated by the Author.

*The terms framework and system are used interchangeably in the AutoML literature

Thus, the following section presents works that applied the knowledge discovery process in the health area without using AutoML techniques.

3.1 AutoML

Barbudo, Ventura and Romero (2023) carried out a systematic review in the area of AutoML, examining how primary studies address the steps of AutoML, as well as presenting gaps that are not yet well explored. This study was carried out on 447 articles and Table 3 shows for each step what approximate percentage of articles included it.

Table 3 – Percentage of articles that performed the AutoML steps following Systematic Literature Review presented in (BARBUDO; VENTURA; ROMERO, 2023)

Step	Percentage
Domain understanding and Data Collection	1.96
Pre-processing	6.72
Data Mining	93.00
Post-processing	1.00

Source: Adapted from (BARBUDO; VENTURA; ROMERO, 2023).

Among the works analyzed, there are some that developed specific solutions for one of the AutoML steps. The next subsections analyze these works and others found in the literature that propose solutions of this nature.

3.1.1 Domain Understanding and Data Collection

Few works present solutions for step 1 of the AutoML process, known as domain understanding and dataset creation. This inherently human activity has been automated by explicitly selecting or generating the best visualization techniques according to the dataset's properties (BARBUDO; VENTURA; ROMERO, 2023).

These visualization recommendation systems reduce the barrier to exploration of basic visualizations by automatically generating results for analysts to search and select, rather than manually specifying (HU et al., 2019).

Ehsan, Sharaf and Chrysanthis (2016) proposed MuVE for multi-objective vision recommendation for visual data exploration, with the aim of supporting effective data exploration by visualizing useful insights into the analyzed data. Demiralp et al. (2017) introduced Foresight, a system that helps users quickly discover visual insights from large, high-dimensional datasets. The system also presents visualizations of the top k instances in the data, based on an appropriate ranking metric.

Within the context of information extraction for dataset creation, Mahule and Vyas (2016) presented an automatic way to extract cloud features from linked open data using a linked traversal approach in a local environment with previously identified and known RDF datasets. This way, the user has the flexibility to determine the depth of neighboring properties to be traversed to extract information.

Dibia and Demiralp (2019) introduced Data2Vis, a neural translation model to automatically generate visualizations from given datasets. Hu et al. (2019) presented a new approach in Machine Learning for visualization recommendation that learns visualization design options from a large corpus of datasets and associated visualizations.

3.1.2 Pre-processing

According to Bilalli et al. (2018), Machine Learning techniques may work differently according to the characteristics of the dataset, for example, they may work better on datasets with continuous rather than categorical attributes, or vice versa, or they may be required to process data missing and outliers. Therefore, to improve results, the dataset needs to be pre-processed, and there are a large number of alternatives for this. Given this, the authors proposed PRESISTANT, which leverages meta-learning ideas to assist the user by recommending pre-processing operators to improve classification performance.

Kandanaarachchi et al. (2020) performed an instance space analysis of combinations of normalization and outlier detection methods, offering insights into which combination of methods can achieve the best performance for a given dataset.

Finally, Hollmann, Müller and Hutter (2023) developed Context-Aware Automated Feature Engineering (CAAFE), a feature engineering method for tabular datasets that uses an LLM to iteratively generate additional semantically meaningful features for tabular datasets based on the dataset's description. The method generates both Python code for creating new features and explanations for the utility of the generated features. This methodology improved performance on 11 of the 14 datasets tested, using OpenAI's GPT-4 and GPT-3.5 as LLMs.

3.1.3 Data Mining

In recent years, as the demand for data analysis has constantly increased, the application of Machine Learning has spread to different areas and people with different profiles and no experience in the area of Machine Learning have started to analyze their data (BILALLI; ABELLÓ; ALUJA-BANET, 2017). This resulted in a major challenge for non-experts: selecting the best algorithm for application in the study scenario. This challenge becomes even more difficult due to the need for the high level of knowledge

required to use these algorithms and Machine Learning techniques (COHEN-SHAPIRA et al., 2019) .

Given this scenario, Reif et al. (2014) developed an open-source meta-learning system capable of predicting the accuracy of target classifiers by empirically evaluating five categories of classifiers such as SVM, Neural Networks, Random Forest, Decision Trees, and Logistic Regression.

Lacoste et al. (2014) developed an extension of the Sequential Model-based Optimization (SMBO) methods, which performs model and hyperparameter selection. The method is based on agnostic Bayesian learning and has been experimented on classification and regression datasets.

Miranda et al. (2014) used meta-learning with Particle Swarm Optimization (PSO) algorithm to propose a solution to the parameter selection problem of the Support Vector Machine (SVM) algorithm.

Bilalli, Abelló and Aluja-Banet (2017) used a meta-learning approach, exploring an exploratory factor analysis method to study the predictive power of different meta-courses collected in OpenML.

Ali, Lee and Chung (2017) presented a multi-criteria decision methodology that empirically evaluates and ranks classifiers and allows end users or experts to choose the classifier with the best performance for the context.

Suganuma, Shirakawa and Nagao (2017) and Sun et al. (2020) used genetic programming techniques to construct a technique to automatically construct CNN architectures for an image classification task.

Abdulrahman et al. (2018) introduced A3R to combine algorithm rankings observed in previous datasets to identify the best algorithms for a new dataset. Sá, Freitas and Pappa (2018) also developed a method to automate algorithm selection and hyperparameter optimization in multi-label classification problems. To do this, they used the MEKA tool and Grammar-Based Genetic Programming (GGP).

Martín et al. (2018) developed EvoDeep, an algorithm dedicated to evolving the parameters and architecture of a Deep Neural Network (DNN) to maximize its classification accuracy.

Cohen-Shapira et al. (2019) presented AutoGRD, a classification approach capable of accurately recommending high-performance algorithms for never-before-seen datasets.

Maher and Sakr (2019) developed SmartML, a meta-learning-based framework to automate the model selection and hyperparameter optimization steps, simulating the role of the Machine Learning expert. The technique stores the meta-features of all processed

datasets, the classifiers and parameters used. Thus, for a new set of data, SmartML extracts its characteristics and searches its knowledge base for the best algorithm to start the optimization process.

Some works sought to solve the same problem using a new algorithm based on Particle Swarm Optimization (PSO), which is capable of automatically searching for significant CNN architectures for image classification tasks (Fernandes Junior; YEN, 2019; FIELDING; LAWRENCE; ZHANG, 2019; LI et al., 2019).

Ma et al. (2020) proposed an Autonomous Continuous Learning (ACL) algorithm to automatically generate a Deep Convolutional Neural Networks (DCNNs) architecture in image classification and other vision applications. The algorithm was evaluated on six image classification tasks such as MNIST, CIFAR100 and others.

Pauletto et al. (2020) developed a Neural Architecture Search (NAS) solution for the Extreme Multilabel Classification (XMC) task, proposing an approach that automatically finds architectures with competitive results in relation to the state of the art.

Silva et al. (2021) proposed a technique to automatically recommend the structure of a set of classifiers, using the classifiers: K-Nearest Neighbors (KNN), Decision Tree, Random Forest, Naive Bayes and Logistic Regression.

Nguyen et al. (2021) experimented with random search techniques and Tree Parzen estimators to solve the Combined Algorithm Selection and Hyperparameter Optimization (CASH) problem for imbalanced classification, proving that the Tree Parzen estimators approach (which is based on Bayesian) outperforms other approaches.

According to Dias et al. (2021), within the context of Neural Networks, the selection of the architecture of a Convolutional Neural Network (CNN) is generally carried out by trial and error, which can consume a lot of time and resources. Therefore, the authors developed the ImageDataset2Vec method, which uses a pre-trained deep neural network, to extract meta-features to describe an image classification dataset. Thus, these meta-features are used by Meta-learning to select the optimal CNN architecture for the context.

Finally, Custode et al. (2024) used two open-source Large Language Models tools (Llama2-70b and Mixtral) for a hyperparameter optimization solution, analyzing optimization logs and recommending new hyperparameters in real-time. The results found by authors showed that LLMs can be used as an effective method for automated hyperparameter optimization.

3.1.4 Post-processing

In the final step of post-processing, which encompasses the interpretation and integration of knowledge, the actions are inherently human in nature, akin to those in step 1. Thus, few studies address strategies and tools in this context, and these few aim to collaborate in a relatively superficial way with the interpretation and integration of knowledge.

Lloyd et al. (2014) presented a system that explores an open space of statistical models to find a good explanation of a set of data and produce a report with figures and texts in natural language. Finally, Tiddi, d'Aquin and Motta (2014) developed Dedalo, a framework that dynamically traverses Linked Data to find commonalities that form complete explanations for items in a cluster, enabling understanding for people without such specialized knowledge.

3.1.5 AutoML Frameworks

According to Alsharef et al. (2022), in recent years, several AutoML frameworks have been proposed that combine the automation of different steps of the ML process, from data generation and preparation to post-processing steps. Among the main examples, we can mention: TuPAQ (SPARKS et al., 2015), TPOT (OLSON et al., 2016a), Auto-WEKA (KOTTHOFF et al., 2017), ATM (SWEARINGEN et al., 2017), Automatic Frankensteining (WISTUBA; SCHILLING; SCHMIDT-THIEME, 2017), MLS-Plan (MOHR et al., 2018), Autostacker (CHEN et al., 2018), Google Cloud AutoML (BISONG; BISONG, 2019), H2O (LEDELL; POIRIER, 2020), EvalML (ALTERYX, 2020), Auto-Sklearn (FEURER et al., 2020), Oracle AutoML (YAKOVLEV et al., 2020), AlphaD3M (DRORI et al., 2021), General Automated Machine Learning Assistant (GAMA) (GIJSBERS; VANSCHOREN, 2021), AutoKeras (JIN et al., 2023), AutoML-GPT (ZHANG et al., 2023) and AutoM3L (LUO et al., 2024)

Firstly, TuPAQ (SPARKS et al., 2015) is built on the MLBase (KRASKA et al., 2013) architecture and also seeks to solve the CASH problem, using bandit search for its optimization based on the data properties and the computational budget defined by the user. Bandit search is a technique similar to Random Search, which balances the exploration of new options from previously found results, aiming to maximize efficiency by focusing on promising solutions while exploring new possibilities.

Tree-based Pipeline Optimization Tool (TPOT) (OLSON et al., 2016b) was originally developed to automate biomedical data science but has now been implemented to handle different ML tasks. This is an open-source genetic programming-based AutoML system that is intended to handle pre-processing, model selection, and hyperparameter

optimization tasks. TPOT uses supervised classification algorithms such as Random Forest, KNN, Logistic Regression and others), feature pre-processing operators (i.e. Scalers, PCA, Polynomial, etc.), and feature selection operators (i.e. Variance thresholders, RFE, etc.) (WARING; LINDVALL; UMETON, 2020).

Auto-Weka is a system based on the WEKA data mining platform and was one of the first AutoML systems to consider the problem of simultaneously selecting an ML algorithm and optimizing hyperparameters, a problem known as Combined Algorithm Selection and Hyperparameter Optimization (CASH). Auto-Weka supports classification and regression problems (WARING; LINDVALL; UMETON, 2020).

Intending to be a distributed, collaborative, and scalable system for AutoML, Auto-Tuned Models (ATM) (SWEARINGEN et al., 2017) allows users to load a dataset, select the Machine Learning method, and choose a variety of hyperparameters to search. Thus, this tool presents the automation of the resource engineering and hyperparameter optimization steps, using a hybrid Bayesian/Bandit optimization system for this purpose. In the same period as ATM, the Automatic Frankensteining technique (WISTUBA; SCHILLING; SCHMIDT-THIEME, 2017) also emerged to solve the CASH problem using ensemble learning. This alternative builds and selects optimized models and groups them to further increase prediction performance.

MLS-Plan (MOHR et al., 2018) solves two phases of the AutoML problem: algorithm selection and algorithm configuration, using hierarchical task networks. Autostacker (CHEN et al., 2018) also works with these two steps and does not perform data preprocessing or feature engineering. This tool uses an evolutionary algorithmic approach to perform hyperparameter optimization on hierarchical stacked Machine Learning models.

According to Bisong and Bisong (2019), Google Cloud AutoML is a tool that automates crucial aspects of the Machine Learning process, making it easier to implement ML models for users without in-depth knowledge of programming or data science. It covers data pre-processing, feature engineering, model selection and optimization in an automated way, providing an efficient solution for developing and implementing deep learning models.

H2O is an open source program written in Java that offers pre-processing techniques and algorithms like random forest, extremely randomized trees, generalized linear models, XGBoost, H2O gradient boosting machine, and deep neural networks (ALSHAREF et al., 2022). Considered in the literature to be one of the most mature AutoML frameworks, it stands out mainly for its speed and performance. According to Gijsbers et al. (2019), H2O achieves better performances in classification and regression tasks. This framework defines algorithms and optimizes hyperparameters using a random grid search and making adjustments through cross-validation (ALSHAREF et al., 2022; SCHMITT,

2023).

EvalML is an open source AutoML library written in python that supports different types of supervised ML problems: regression, binary classification, multiclass classification, time series regression, multiclass and time series classification. This library automates much of the ML process such as: feature selection, model selection, hyperparameter optimization (using Bayesian Optimization), cross-validation and others (ALSHAREF et al., 2022).

Based on the popular Python ML library scikit-learn, Auto-sklearn is also an AutoML solution that seeks to solve the CASH problem (WARING; LINDVALL; UMETON, 2020). This tool performs well and its PosSH Auto-sklearn version uses the BOHB algorithm, creating a huge boost in system speed (FEURER et al., 2018). However, it is not yet well implemented to handle large clinical datasets (WARING; LINDVALL; UMETON, 2020).

Aiming to not only provide accurate models but also significantly reduce execution time, Oracle AutoML (YAKOVLEV et al., 2020) eliminates the need for continuous iterations across multiple pipeline configurations. Using a feed-forward approach, each pipeline step makes decisions based on meta-learned proxy models, which are able to predict the performance of candidate pipeline configurations before building the final model. This methodology allows only the best candidate pipeline to be built and tuned.

AlphaD3M (DRORI et al., 2021) uses a reinforcement learning approach to optimize the AutoML pipeline problem. The authors use a sequence modeling technique that employs deep neural networks and Monte Carlo tree searches to solve the optimization problem.

In contrast to current, often black-box systems, General Automated Machine Learning Assistant (GAMA) (GIJSBERS; VANSCHOREN, 2021) allows users to connect different AutoML and post-processing techniques, record and visualize the research process and supports easy benchmarking. It currently features three AutoML search algorithms, two model post-processing steps, and is designed to allow you to add more components.

AutoKeras uses the Keras API and mainly focuses on solving problems in Deep Learning, rather than simple models. To achieve this, this AutoML framework uses Neural Architecture Search (NAS), to improve the modeling task (JIN; SONG; HU, 2019).

AutoML-GPT (ZHANG et al., 2023) is an AutoML Framework that uses ChatGPT-4 as a core component to automate some steps of the AutoML pipeline. AutoML-GPT dynamically receives user requests and automatically conducts experiments, from data processing to model architecture and hyperparameter tuning. Thus, AutoML-GPT can

be used for numerous complex AI tasks across diverse tasks and datasets.

Finally, AutoM3L (LUO et al., 2024) uses LLMs (HuggingFace and Timm) to understand data modalities and select appropriate models based on user requirements, providing automation and interactivity. The solution proposed by the authors is used to automate the Machine Learning pipeline for multimodal scenarios.

Table 4 shows a comparison between the analyzed AutoML frameworks and our solution, comparing which steps each one proposes to solve, namely: Domain Understanding (DU), Data Collection (DC), Pre-processing (PrP), Model Selection (MS), Hyperparameter Optimization (HO) and Post-processing (PoP). The symbol ✔ is used when that framework automates that step, while ⁛ is used when there is some automation, but it is limited. The LLM column shows whether the AutoML framework used any LLM tools (symbol ✔) in its pipeline. When comparing the AutoML framework with our proposed solution (HEAL), we observed that none of the AutoML frameworks performs complete automation of all AutoML steps, only our solution goes through all the steps from domain understanding to post-processing.

3.2 Related Work

Despite the growing research in the area of AutoML and the need to apply these techniques in the healthcare area, few studies have been developed in this context (WARING; LINDVALL; UMETON, 2020). These works are considered related works to this thesis.

Luo (2016a) proposed PredicT-ML, an automated version of Weka for big clinical data that aims to automate the construction of Machine Learning models with large sets of clinical data. This solution focuses on overcoming common barriers in predictive modeling in healthcare, such as algorithm selection and optimization and temporal aggregation of clinical attributes. The goal is to make clinical big data more accessible for healthcare research and application, improving the efficiency and accuracy of predictive modeling.

The pipeline proposed in PredicT-ML is composed of pre-processing, model selection and hyperparameter optimization. Therefore, the proposed AutoML does not perform the domain understanding and data collection steps, and it is the user's responsibility to submit the medical dataset. As pre-processing, it performs missing data analysis, with imputation and deletion, automatic feature selection, data balancing, normalization, and feature engineering. When operating in the context of clinical big data, it automates the selection of algorithms and feature selection techniques based on the task, using advanced methods such as Auto-WEKA, being faster than other methods, as it uses advanced hyperparameter optimization using progressive sampling and techniques

Table 4 – Overview of AutoML frameworks, comparing each step of the AutoML process: domain understanding (DU), data collection (DC), pre-processing (PrP), model selection (MS), hyperparameter optimization (HO), post-processing (PoP) and use of LLM tool (LLM)

AutoML	DU	DC	PrP	MS	HO	PoP	LLM
TuPAQ (2015)				✔	✔	◌	
TPOT (2016)			✔	✔	✔		
Auto-WEKA (2017)			✔	✔	✔		
ATM (2017)					✔		
Frankensteining (2017)				✔	✔	◌	
MLS-Plan (2018)			✔	✔	✔	◌	
Autostacker (2018)				✔	✔	◌	
Google AutoML (2019)		◌	✔	✔	✔		
H2O (2020)		◌	✔	✔	✔	◌	
EvalML (2020)		◌	✔	✔	✔	◌	
Auto-Sklearn (2020)			✔	✔	✔	◌	
Oracle AutoML (2020)		◌	✔	✔	✔		
AlphaD3M (2021)		◌	✔	✔	✔	◌	
Gama (2021)			✔	✔	✔	✔	
AutoKeras (2023)			✔	✔	✔	◌	
AutoML-GPT (2023)			✔	✔	✔	◌	✔
AutoM3L (2024)			✔	✔	✔		✔
HEAL - Our work (2024)	✔	✔	✔	✔	✔	✔	✔

to reduce the search space. Finally, it only presents the metrics of the model results, but does not perform post-processing analysis of the results.

Faes et al. (2019) used Google's Cloud AutoML API and a public dataset to develop a tool that aims to allow healthcare professionals without coding experience to develop medical image diagnostic classifiers using AutoML. This research demonstrated the feasibility of developing deep learning models with performance comparable to advanced algorithms, promoting the democratization of deep learning design in medicine.

The AutoML used by the authors has an approach that uses deep learning architecture and has pre-processing, model selection and hyperparameter optimization as pipeline steps. Therefore, it does not include the domain understanding and data collection steps, and the authors used publicly available open source datasets. In the pre-processing step, Google's AutoML Cloud Vision API relies on some techniques such as: automatic deletion of duplicates and division of data into training, validation and testing. Finally, the pipeline does not have a post-processing step, providing only metrics such as precision, sensitivity, and others.

Karhade et al. (2021) implemented an AutoML tool to classify early childhood caries status in children using oral health data. The tool automates several crucial steps of the Machine Learning pipeline, including data pre-processing, feature engineering, model selection, and hyperparameter optimization, with the goal of facilitating early cavity identification and diagnosis. An Automated Machine Learning (AutoML) deployment on Google Cloud was used to identify the best performing model, using an appropriate network architecture and optimizing its parameters to maximize discriminative performance.

Kausar et al. (2022) proposed an AutoML pipeline for automatic fall detection in older adults using sensors, applying Machine Learning algorithms and AutoML to classify activities as falls or normal daily activities. The proposed AutoML pipeline has only the pre-processing steps (data cleaning and feature extraction), model selection, and hyperparameter optimization. For model selection and hyperparameter optimization, the Classifier Learner application in MATLAB was used to automatically select the best-fitting model.

Zhang et al. (2022) presented a tool based on the BrewAI platform, which seeks to solve challenges in predicting the risk of disease outbreaks, a crucial problem in public health. This enables the automatic creation of Machine Learning (ML) models without the need for prior ML knowledge on the part of users, making advanced ML techniques accessible to researchers and healthcare professionals.

This is done by automating critical steps in the AutoML pipeline such as preprocessing, model selection and optimization, with the aim of improving epidemic prevention and control. The authors created an algorithm for automatic data collection using API to access sources such as WHO and World Bank, but it was presented as something external to the AutoML used. In the hyperparameter optimization step, BrewAI uses tools like Optuna and HyperOpt, and in the last post-processing step, AutoML only provides metrics like confusion matrix and f1-score but does not generate insights into the results. Finally, as limitations presented, the authors stated that many AutoML tools, including BrewAI, do not have the ability to deal with data imbalance and the database contained a lot of missing data.

Khan et al. (2024) used AutoML to develop an optimal regression model, evaluating the effectiveness of the hybrid strategy for detecting abnormal data based on heuristic and stochastic approaches, aiming at patient rehabilitation. This study provides reliable information to healthcare professionals, improving decision-making and patient records management. The proposed AutoML pipeline performs only the pre-processing steps (missing value analysis, normalization, and duplicate removal), model selection, and hyperparameter optimization. At the end of the execution, the framework presents metrics such as MAPE, but does not perform a post-processing analysis.

Finally, Mohammadi and Nguyen (2024) used OpenAI ChatGPT as an LLM tool integrated with Vertex AI, an AutoML platform, to develop two classification models. The first model is a binary classifier to detect the presence of diabetic retinopathy, achieving a precision and recall rate of 84.48%, while the second determined the severity of diabetic retinopathy, achieving an area under the precision-recall curve of 0.81, with a precision rate of 81.81% and recall of 72.83%.

The AutoML pipeline proposed by the authors has the steps of domain understanding (partial), pre-processing (normalization and application of the CLAHE technique to improve image quality), model selection and hyperparameter optimization. For the domain understanding step, ChatGPT provided scripts and instructions in R and Python to improve image quality and also generated scripts to calculate metrics such as sensitivity and specificity. There is no data collection step, as the dataset used was extracted from the Messidor-2 dataset. In the model selection and hyperparameter optimization step, Vertex AI was used to train the classification models and automatically select the most appropriate ones for different tasks. The pipeline does not have a post-processing step, it only presents metrics such as precision and recall.

Table 5 shows a comparison between related works that propose the development and/or application of the AutoML framework applied to the healthcare area and our solution, comparing the techniques used and the objectives of applying AutoML. It is observed that some works did not develop their own technique, opting to apply an existing technique such as Google AutoML (FAES et al., 2019; KARHADE et al., 2021), MATLAB Classifier Learner (KAUSAR et al., 2022), BrewAI (ZHANG et al., 2022) and Vertex AI (MOHAMMADI; NGUYEN, 2024). The other works (LUO, 2016a; KHAN et al., 2024) developed their own AutoML pipeline to discover knowledge in healthcare.

Table 6 shows a comparison between related works that propose the development and/or application of the AutoML framework applied to the healthcare area and our solution, comparing which steps each proposes to solve: domain understanding (DU), data collection (DC), pre-processing (PrP), model selection (MS), hyperparameter optimization (HO) and post-processing (PoP). The symbol ✔ is used when that framework

Table 5 – Overview of related work, the techniques used and the objectives of each work

Works	Tool	Objective
(LUO, 2016a)	PredicT-ML (Own tool)	Develop AutoML to automate building Machine Learning models with large clinical datasets
(FAES et al., 2019)	Google AutoML	Medical imaging diagnosis
(KARHADE et al., 2021)	Google AutoML	Early childhood caries situation in children
(KAUSAR et al., 2022)	MATLAB Classifier Learner (Own tool)	Develop an AutoML pipeline to detect falls in the elderly
(ZHANG et al., 2022)	BrewAI	Predict risk of disease outbreaks
(KHAN et al., 2024)	Own tool	Develop an AutoML to assist in efficient patient rehabilitation
(MOHAMMADI; NGUYEN, 2024)	Vertex AI and ChatGPT	Classifier to detect the presence and severity of diabetic retinopathy
Our work	**HEAL (Own tool)**	Development of an AutoML system applied to panel data structure in the healthcare area

automates that step, while ⁘ is used when there is some automation, but it is limited. The LLM column shows whether the AutoML solution used any LLM tools (symbol ✔) in its pipeline. When comparing the work related to our proposed solution, we observed that no work performed the complete automation of all AutoML steps, only our solution goes through all the steps from domain understanding to post-processing.

Based on the analysis of related work and despite the importance of each step of the AutoML pipeline described in the Background (page 19) and State of the Art (page 45), we can observe that none of the related works propose the development of an AutoML pipeline applied to the health area that covers all steps: 1 - Domain Understanding and Data Collection; 2 - Pre-processing; 3 - Data Mining; 4 - Post-processing. Therefore, our solution presents itself as a unique AutoML pipeline in these comparative questions.

Table 6 – Overview of related work comparing each step of the AutoML process: domain understanding (DU), data collection (DC), pre-processing (PrP), model selection (MS), hyperparameter optimization (HO), post-processing (PoP) and use of LLM tool (LLM)

AutoML	**DU**	**DC**	**PrP**	**MS**	**HO**	**PoP**	**LLM**
(LUO, 2016a) PredicT-ML			✔	✔	✔		
(FAES et al., 2019) Google AutoML			✔	✔	✔		
(KARHADE et al., 2021) Google AutoML			✔	✔	✔		
(KAUSAR et al., 2022) MATLAB Classifier Learner			✔	✔	✔		
(ZHANG et al., 2022) BrewAI		✔	✔	✔	✔		
(KHAN et al., 2024) Unnamed			✔	✔	✔		
(MOHAMMADI; NGUYEN, 2024) Vertex AI and ChatGPT	◌		✔	✔	✔		✔
HEAL (our work)	✔	✔	✔	✔	✔	✔	✔

4 METHODOLOGY

This chapter presents the methodology adopted for the development of this thesis. Therefore, it is divided into materials and methods, mainly presenting the steps followed for its execution.

4.1 Materials

To develop HEAL, we built the database using the PostgreSQL Database Management System, a high-performance and highly scalable open source project. For the data presented in a graphical web interface, we used the Firebase Real Time Database, a cloud-hosted database that stores and synchronizes data in real time for free.

The programming language choice was Python due to its large community of experts and a growing number of libraries for Data Analysis and Machine Learning solutions. As a Large Language Model (LLM) tool, we used OpenAI ChatGPT (4o and 4o-mini). Visual Studio Code (VS Code) was used as a code editor to build, debug, and version control code.

To develop the framework, we used the Python libraries: pandas and numpy (for data analysis and manipulation), sklearn (for Decision Tree, Support Vector Machine, K-Nearest Neighbors, Random Forest, metrics, KMeans, Min Max Scaler and Time Series Split), tensorflow keras (for Artificial Neural Network), math and statistics (to help with the application of some mathematical equations), infogain (for Information Gain), psycopg2 (for database connection), imblearn (for data balancing). To perform web scraping, we use Beautiful Soup, a library that makes it easy to extract information from web pages.

We use React for web interface development, an open-source JavaScript front-end library. To build this interface, in addition to the Realtime Database, we used some tools provided by Firebase: Storage (to store files sent by the user to the system) and Authentication (to facilitate user authentication when accessing our system).

We used TensorFlow Artificial Neural Network Feedforward (Multilayer Perceptron) trained using the backpropagation algorithm, Decision Tree (using the Classification and Regression Trees algorithm - CART), Random Forest (ensemble of multiple CART decision trees), Support Vector Machine (using Sequential Minimal Optimization - SMO) and K-Nearest Neighbors algorithms for predicting future values. The choice of these algorithms was defined with the objective of using some algorithms referring to different

categories of Machine Learning algorithms, such as: Ensemble (RF), Neural Networks (ANN) and Instance Based (KNN).

The test environment used for all experiments was a *Intel® Core™ i5* 9400f, 2.90 *GHz*, six-core processor, six threads without Simultaneous Multithreading support, 48 GB *RAM* and the video card *Geforce GTX 1660* with 6 GB, Linux Ubuntu 22.04.1 *bits* operational system.

The next subsection presents in detail the structure of the datasets supported by HEAL: panel data structure. Furthermore, it presents how this structure is found in the main data source used in this work, the data repository of the Global Health Observatory (GHO) of the World Health Organization (WHO).

4.1.1 WHO dataset and panel data: data structures supported by HEAL

The Global Health Observatory (GHO) data repository of the World Health Organization (WHO)* is the main data source used in this work. We used the Application Programming Interface (API) GHO Open Data Protocol to extract all indicators (attributes) from WHO, and all these data are found in the panel data structure (Section 2.2, page 20).

It is important to emphasize that the WHO database served as the starting point for our research. However, any dataset that respects the same standard as WHO data (panel data) can be imported into HEAL, and all other steps can be carried out normally.

The GHO repository contains WHO data for health statistics from 194 member states divided into six regions, providing access to more than 2,000 indicators for each country. The data include different indicators, such as the mortality rate related to different diseases, data related to alcohol and tobacco consumption, sexually transmitted diseases, psychiatric problems, hygiene, and environmental health, among others.

Most of these datasets represent the World Health Organization's best estimates of specific indicators for cross-country comparisons over time (WHO, 2024). Those responsible for the repository update it frequently, keeping the data updated and available as close to official national estimates as possible.

Following the structure of the data panel, the indicators in this repository are divided mainly into countries (194 WHO member countries) and years (1932 to 2030 as projection), with countries being considered the spatial dimension and years the temporal dimension. For each indicator, the database has the following column structure:

a) *CODE* - indicator identifier.

*https://www.who.int/en/. Accessed March 15, 2024

b) *DISPLAY* - indicator name.

c) *YEAR* - indicator year (temporal dimension).

d) *REGION* - indicator region.

e) *COUNTRY* - indicator country (spatial dimension).

f) *DISPLAY VALUE* - indicator value.

g) *DIMENSION* - each indicator has 0 or more dimensions.

In addition to the temporal and spatial dimensions, WHO also considers other dimensions of the indicators, such as sex (male, female, and both), type of resident area (urban or rural), age (ranging from newborn to elderly), and others.

These other dimensions can offer a better refinement of the indicator, as they allow defining attributes based on specific characteristics. For instance, if attribute X has the gender dimension, information can be extracted from this indicator for males, females, and both genders. This attribute refinement can be helpful for the knowledge discovery process, in addition to increasing the possibilities and detail of predictions.

In the same structure as the WHO data, HEAL also supports any other dataset in the panel data structure, as long as the spatial and temporal dimensions are present in that dataset and are identified by the user of the system. Table 7 illustrates an example of the dataset structure supported by HEAL with 3 spatial dimensions (1 to 3), 2 temporal dimensions (1 to 2) and X different features.

Table 7 – Example of dataset structure supported by HEAL with 3 spatial dimensions, 2 temporal dimensions and *X* features

Spatial Dimension	**Temporal Dimension**	**Feature F1**	**Feature F2**	**...**	**Feature FX**
Region 1	Time 1	$F1_{11}$	$F2_{11}$	...	FX_{11}
Region 1	Time 2	$F1_{12}$	$F2_{12}$	...	FX_{12}
Region 2	Time 1	$F1_{21}$	$F2_{21}$	...	FX_{21}
Region 2	Time 2	$F1_{22}$	$F2_{22}$	...	FX_{22}
Region 3	Time 1	$F1_{31}$	$F2_{31}$	...	FX_{31}
Region 3	Time 2	$F1_{32}$	$F2_{32}$	...	FX_{32}

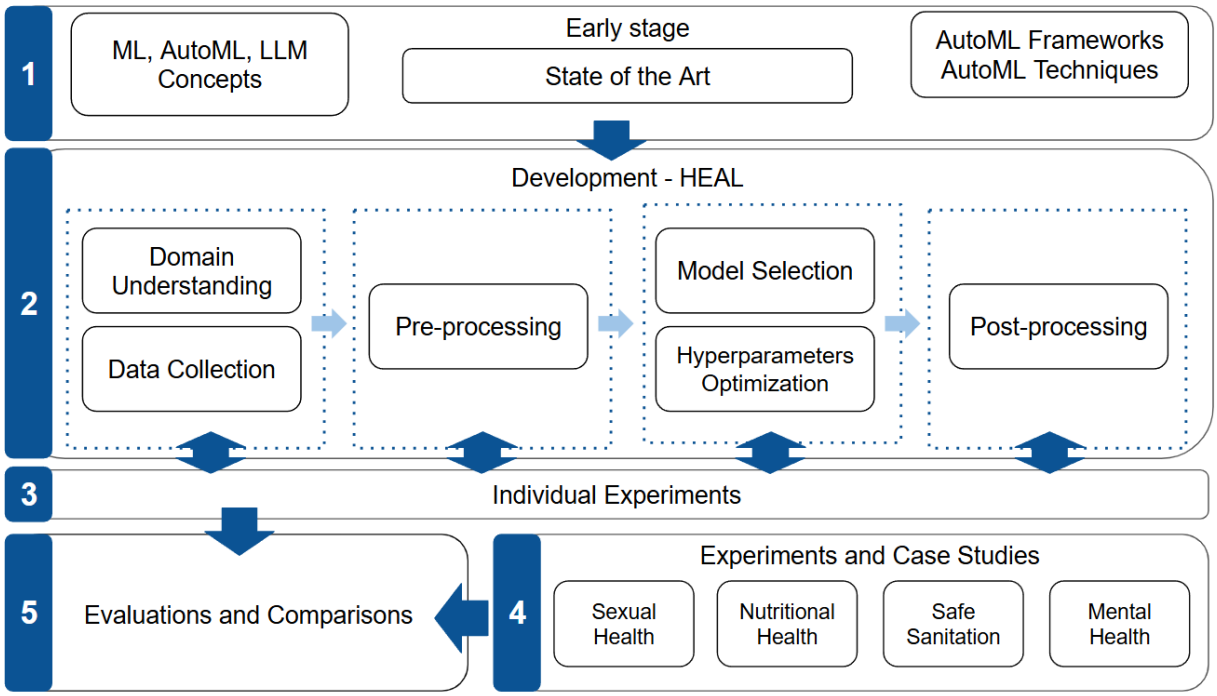
WHO datasets are already in this format, with the spatial dimension referring to countries and the temporal dimension referring to years. We have already carried out the processing for extraction and import of these.

Finally, it is important to emphasize that the spatial dimension can be any existing regional context, such as a country, continent, city or social group. The temporal dimension can also include different periods of time, such as years, months, days, semesters, hours, and others. Therefore, although HEAL works mainly with WHO data, it supports datasets that respect the conditions of a panel data structure, regardless of the spatial and temporal dimension.

4.2 Method

To develop this thesis, we followed the steps of the method illustrated in Figure 11. This is divided into the following steps: initial state, development of the HEAL system, individual experiments, case studies and evaluations.

Figure 11 – Steps of Method



Source: Elaborated by the Author.

Step 1 - Early stage: initially, the first step of the proposed method is responsible for the initial study necessary to prepare the following steps. This study encompasses a state-of-the-art survey of researched papers in IEEE, ACM, Science Direct and Google Scholar, an examination of Machine Learning and AutoML concepts, Large Language Models (LLM), and a review of the most commonly used AutoML frameworks and techniques.

All of this initial study has already been presented in the chapters relating to

Background (Chapter 2, page 19) and State of the Art of AutoML (Chapter 3, page 45). The study definitions in these chapters will be used to guide the steps developed in HEAL, for instance: the tasks in the AutoML pipeline and the opportunity to apply LLM to them.

Therefore, this comprehensive study of key concepts within the AutoML and LLM context, along with an analysis of major contributions in this field, is crucial for identifying future directions and gaps in the literature. This enables the proposition of solutions that contribute significantly to the area.

Step 2 - Development - HEAL: the second step is responsible for controlling tasks related to the development of HEAL. This includes the four related phases of the AutoML pipeline: 1 - Domain Understanding and Data Collection; 2 - Pre-processing; 3 - Model selection and Hyperparameters Optimization; and 4 - Post-processing.

By developing these phases, we were able to cover the entire AutoML pipeline, thus addressing a gap identified in the literature. Notably, our review did not reveal any works that cover all these steps comprehensively, nor did we find a mature solution specifically in the healthcare context.

Therefore, the primary goal is to introduce an AutoML system solution that spans all phases of the AutoML Pipeline. This will empower users lacking experience in data analysis and/or healthcare to automate the Machine Learning process within the healthcare context, as long as the dataset is in the panel data structure.

Step 3 - Individual Experiments: for each step developed in the previous stage, it is necessary to carry out individual tests and experiments seeking to validate the results of the proposal presented. Therefore, this step 3 is responsible for individually experimenting and evaluating each AutoML step developed in the previous step. After these individual experiments, the phases defined in step 2 will be joined together to form the complete version of HEAL.

Step 4 - Experiments and Case Studies: as the step 4 of our method, experiments and case studies were proposed to validate the HEAL pipeline developed, applying it to predictions in real healthcare contexts using data collected from the World Health Organization (WHO) and other databases. The objective of this step is to confirm in applied experiments the efficiency of the proposed system and the contributions it can generate in the health area.

These experiments and case studies were carried out in different sub-areas of healthcare, covering: 1 - Sexual Health; 2 - Nutritional Health; 3 - Safe Sanitation; and 4 - Mental Health.

Step 5 - Evaluations and Comparisons: finally, after the previous steps that

prove that the earlier steps work in the pipeline, in this last step, we evaluate the results found in the earlier steps, and, as a previous task, we compare our system with others found in the literature.

5 AUTOMATED MACHINE LEARNING PIPELINE FOR HEAL

This chapter presents the proposed pipeline for HEAL in Section 5.1, and each of its steps. After that, we present in more detail the design of the HEAL AutoML for each of the proposed steps: Domain Understanding and Data Collection (Subsection 5.1.1, page 67), Pre-processing (Subsection 5.1.2, page 73), Data Mining (Subsection 5.1.3, page 77) and Post-processing (Subsection 5.1.4, page 79).

5.1 HEAL Pipeline

The developed HEAL system is characterized by four main steps, as presented in the Background (Chapter 2, page 19) and Methodology (Chapter 4, page 59), as follows: 1 - Domain Understanding and Data Collection ; 2 - Pre-processing; 3 - Data Mining and 4 - Post-processing. The solutions developed for each of these steps are described in the following sections. Figure 12 illustrates our proposed pipeline, highlighting the application of key techniques including ChatGPT-4o (steps represented in green and with the GPT icon), API processing, web scraping and data analysis techniques at each step of AutoML.

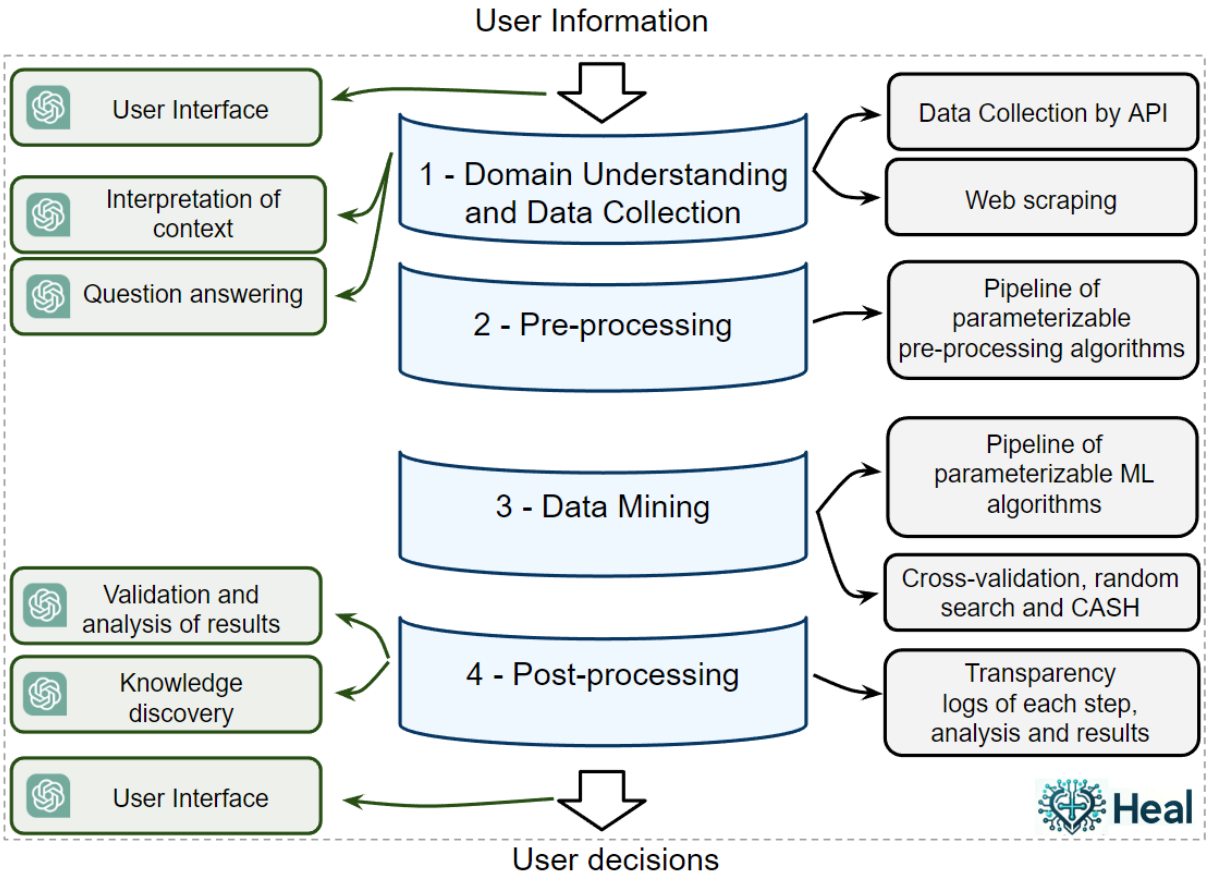
LLMs can be used to simplify and facilitate interaction between users and AutoML systems, making them more accessible tools for expert and non-expert users. Thus, this user interface improvement application was applied both in the initial step of information entered by the user, and in the final step of presenting the interpretations of knowledge generated by AutoML to the user.

Step 1: in the first step of the AutoML (Domain understanding and Data Collection) process, data analysis techniques were applied to collect data from the WHO Data API. In addition, we applied web scraping techniques on the WHO website to enrich our data related to WHO attributes.

To perform domain understanding, we use LLM techniques to interpret the context (from the most relevant indicators and information present in the dataset). Based on this interpretation, we can provide functionality to help the user understand the domain, which can answer questions the user asks related to the context.

Step 2: in the second step (pre-processing), we developed a configurable pipeline with some of the main pre-processing steps in the context of Machine Learning, for example: missing data analysis, outliers, discretization, balancing and feature selection. Thus,

Figure 12 – Steps of our pipeline (HEAL)



Source: Elaborated by the Author.

the user can configure these pre-processing steps and the pipeline executes according to this configuration. It is important to note that it is possible to add new steps in future versions of HEAL.

Step 3: in the third step of AutoML (Data Mining), we also developed a parameterizable pipeline with the Artificial Neural Networks (ANN), Random Forest (RF), Decision Tree (DT), Support Vector Machine (SVM) and K-nearest Neighbors (KNN) algorithms. All these algorithms were implemented for the context of multiclass classification problems.

For hyperparameter selection, we implement the Combined Algorithms Selection and Hyperparameter Optimization (CASH) along with cross-validation and random search. This way, the user can choose which hyperparameters will be searched and our pipeline will execute this, searching for the best combination from the defined universe. Finally, it is possible to add new algorithms in future versions of HEAL.

Step 4: LLMs techniques were also used as a tool to help validate the results found in step 4 of the AutoML process (post-processing). Thus, these algorithms received

the results, interpreted them and checked whether they are within the context defined by the user. Finally, LLMs also helped make the AutoML process more transparent and understandable for users by explaining configuration steps and decisions.

The following subsections describe in detail each of the steps performed in HEAL.

5.1.1 Step 1 - Domain Understanding and Data Collection

The first step of the HEAL pipeline is characterized by domain understanding and data collection. Therefore, as the first step of developing the system proposed in this work, we defined a solution to collect health data from WHO in the panel data structure and, consequently, build our database that will be used in the following steps. In addition to the database created by this data collection step, the user can import a dataset into our system, as long as it follows the panel data standard.

Therefore, the first step consisted of creating a database with various indicators relating to health area. Thus, as the first version of HEAL, we designed a relational database using the PostgreSQL Database Management System (DBMS).

To explore health data, the objective of the data collection step is to select and extract data from the World Health Organization (WHO). The Global Health Observatory (GHO) data repository of the World Health Organization (WHO)* is the source for the data used in this work. We used the Application Programming Interface (API) GHO Open Data Protocol.

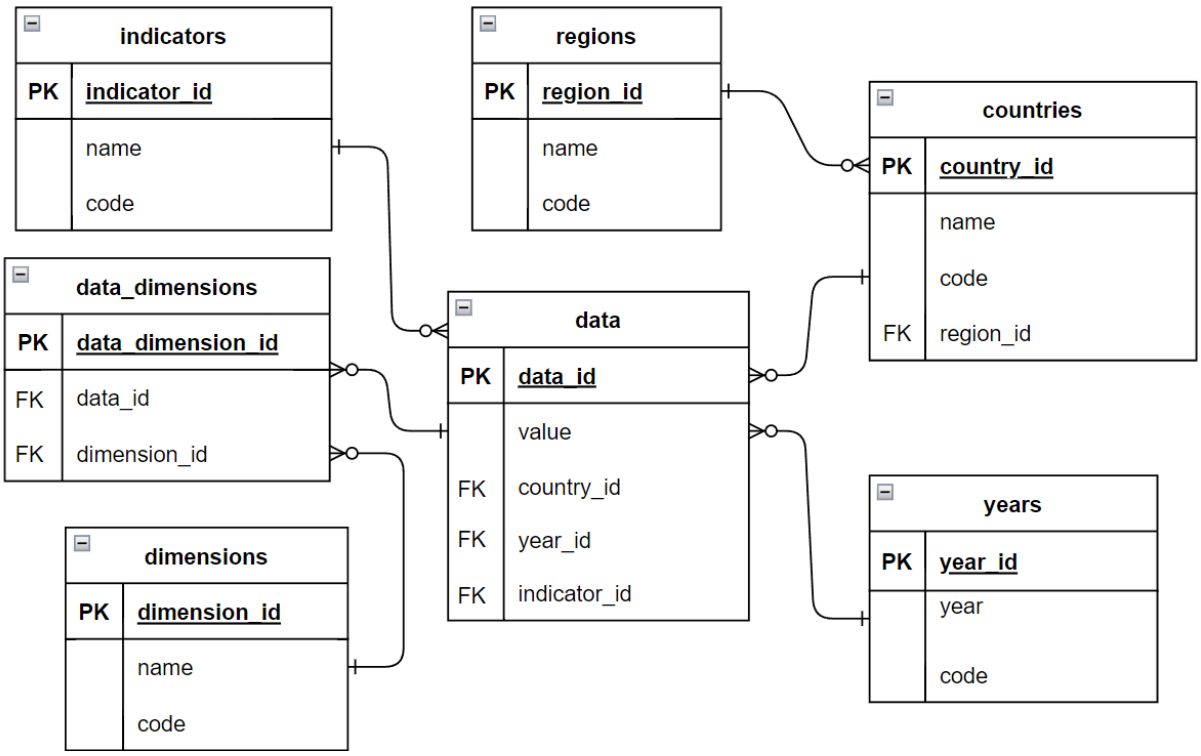
It is important to emphasize that the WHO database serves as the starting point for our research. However, HEAL step 1 is open for the user to submit new datasets in the panel data structure. This way, the user can upload a .csv file and the system will extract and insert the data from the file into the HEAL database, consequently increasing the opportunities for knowledge discovery.

To collect data, we created a HEAL step capable of searching automated (through an API) and processing automated the selected information in the WHO data repository. Then, the system inserted it in an organized way into the database, which has the structure of the illustrated relational model in Figure 13. It is possible to observe that the panel data structure is represented by the spatial dimension (regions or countries) and temporal dimension (years). Other dimensions can be expanded, for example: sex and age groups.

The purpose of the tables and relationships created is to organize the data imported from WHO into an ideal format for forecasting annual data (temporal dimension). Another feature is to scale the amount of data stored as new data is inserted without

*https://www.who.int/en/. Accessed March 15, 2024

Figure 13 – Relational Model SQL



Source: Elaborated by the Author.

losing the relationships necessary to perform the queries.

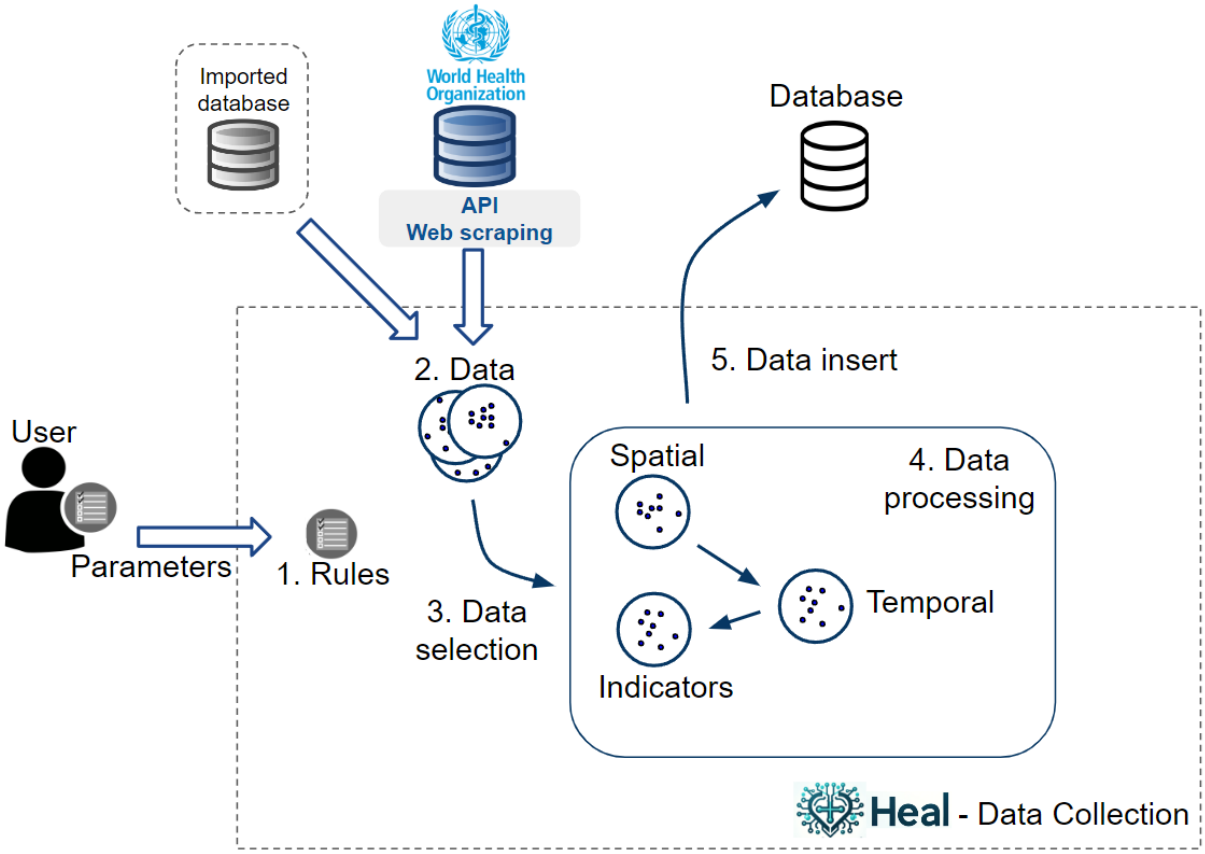
We export the data directly from the WHO website using the API routes provided by the GHO. Then, this data is automated processed to extract the indicators and inserted into the database. Figure 14 illustrates the steps to build the database.

HEAL is enabled to receive parameters as user-defined input about which attributes and dimensions the user would like to import into the system database (step 1 - rules in Figure 14). There is a standard configuration where the user can request the import of all information present in the WHO database. Despite being a step in an AutoML, it is important to provide the user with possibilities to control some minimal tasks, if necessary in specific cases.

After this initial configuration, HEAL can automatically extract all data present in the WHO following user parameters (step 2 - data in Figure 14). After this extraction, the data is processed and information regarding regions, countries, years, indicators and dimensions is organized (step 4 - data processing in Figure 14) and saved in the HEAL database (step 5 - data insert in Figure 14) to be used in the following steps of the pipeline.

In the experiments we carried out, we used the configurations shown in Figure 15, in which we defined the extraction and processing of all WHO indicators and only the

Figure 14 – Steps of Data Collection



Source: Elaborated by the Author.

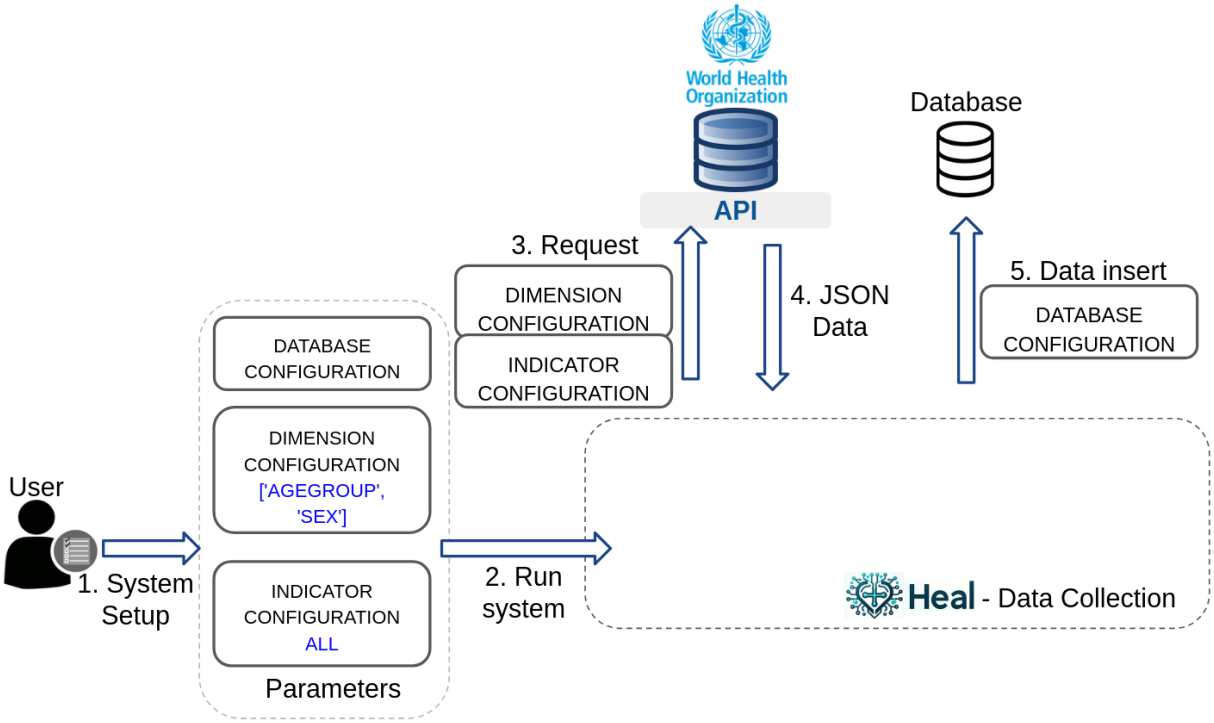
AGEGROUP and *SEX* dimensions (step 1 - system setup and 2 - run system in Figure 15). It's possible to observe that HEAL makes requests to the WHO API based on user-defined parameters and receives JSON containing the data (step 3 - request in Figure 15). From there, it processes this received information to convert it into the required format (represented by the HEAL data collection step in Figure 15, these being the same steps presented in the flow of Figure 14) and saves the information in the database (step 5 - data insert in Figure 15).

The execution of the database construction step required a total of 18,765 seconds (about 5.21 hours) in the test environment used. Thus, we got 2,915 indicators in 90 different years (from 1932 to 2021) and 194 countries for the bases collected and extracted. This processing time, as well as the total indicators, may change according to the environment used and the initial settings made by the user.

In addition, we applied web scraping techniques on the WHO website to enrich our data related to WHO attributes by searching the WHO website[†] for more detailed

[†]https://www.who.int/data/gho/data/indicators/indicators-index. Accessed October 20, 2024

Figure 15 – Configuration in the Data Collection Step



Source: Elaborated by the Author.

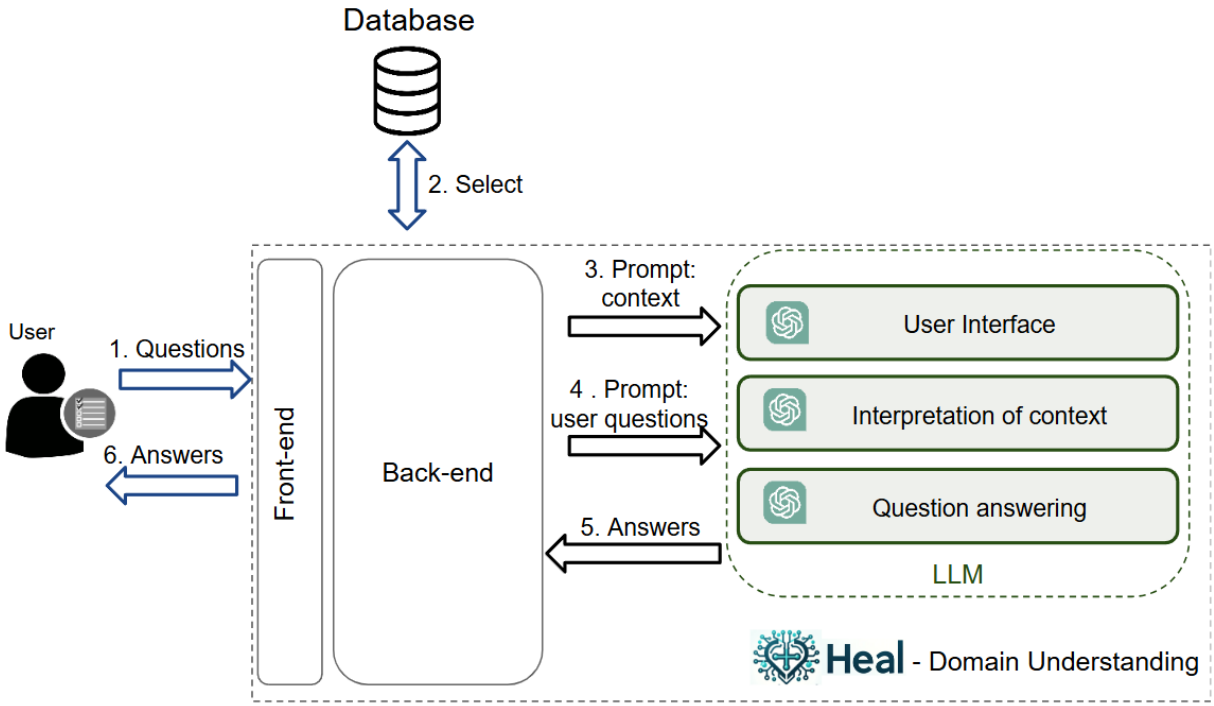
information on each indicator provided, such as: indicator name, short name (code), data type, disaggregation, method of estimation, expected frequency of data, dissemination, definition and rationale.

For web scraping, we used the Beautiful Soup library[‡], searching for the information contained in these fields. This information enriches the HEAL database, so that the LLM application is carried out with more information and, consequently, produces more accurate responses.

After inserting the data extracted from the WHO or receiving the .csv file from the user, we develop a step related to understanding the domain. There are two main objectives in this step, the first being to provide the user with the possibility of interacting with the system in a natural way, thus, the system can interpret the context of the database and the user can learn more about the database and also the possibilities that exist for potential knowledge discoveries. The second objective is to validate the dataset created from the database data and user interactions, ensuring its alignment with the contextual requirements. Figure 16 illustrates the flow responsible for carrying out this step.

[‡]https://pypi.org/project/beautifulsoup4/. Accessed November 05, 2024

Figure 16 – Steps of Domain Understanding



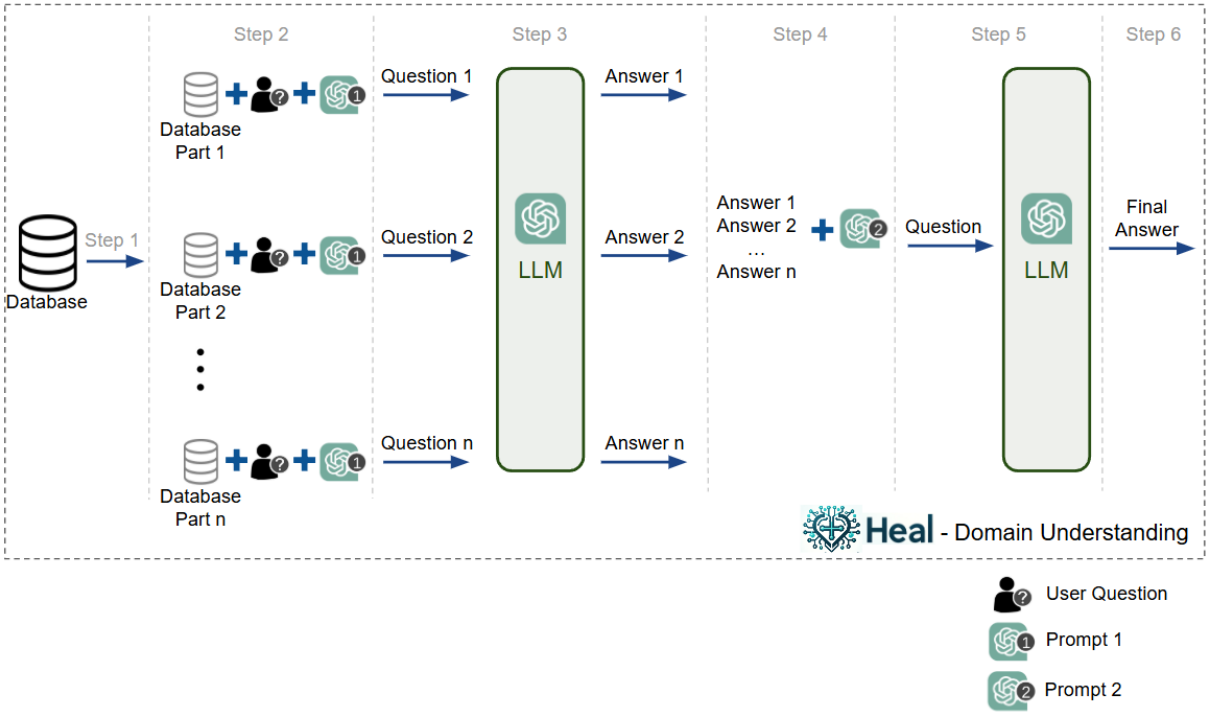
Source: Elaborated by the Author.

The user can ask open-ended questions naturally to the system through a developed front-end (step 1 - questions in Figure 16). When asking questions, HEAL automatically selects information from the database (step 2 - select in Figure 16) and generates prompts corresponding to the questions asked to send them to the ChatGPT-4o API (step 3 - prompt context and 4 - prompt user questions in Figure 16). This LLM will generate responses to the questions (step 5 - answers in Figure 16) and return them for processing and display on the system's front-end (step 6 - answers in Figure 16). The LLM can also validate the dataset extracted from the WHO by checking its integrity and compliance with the user-defined needs and parameters. Furthermore, the questions and observations made by the user can serve as context for the LLM to understand the domain in which the system is operating.

Figure 17 illustrates in detail the steps required to use the LLM ChatGPT-4o API. Therefore, this figure explains steps 3, 4, and 5 of Figure 16.

Firstly, due to the limit of tokens that LLM can process for each request, it is necessary to divide the database into parts, respecting this limit and according to the size of this database (step 1). In step 2, HEAL adds to each of these database parts the user question and prompt 1. The first prompt was created to explain to the LLM its features and provide context about the data sent to them. So, some information was put into this

Figure 17 – Steps to use LLM for Domain Understanding



Source: Elaborated by the Author.

prompt to build the LLM profile:

a) You are a helpful assistant to an AutoML system.

b) In this system, you are helping in the initial stage of domain understanding.

c) You will simulate the behavior of a domain expert in the healthcare area.

d) You will be given some information about the database that the user can explore.

e) Based on this information, you will answer the questions the user asks about that database.

f) Always present the name of the indicators as they are in the original English and always present the indicator code when presenting it to the user.

g) To do this, read the indicators sent previously and work with this data.

h) If you don't know, answer: *I don't know the answer*

i) Always write the answer in English.

Then, this union of the database parts, user question and prompt 1 is sent to the LLM and, for each question sent, the LLM generates a single answer according to each context (step 3). It is then important to merge these responses to create a single answer to the user's question. Therefore, HEAL merges all the answers using a second prompt (step 4) and sends this final question to LLM (step 5), which processes this question and creates a final answer for the user (step 6). Some information is important to build the LLM profile in prompt 2:

a) You are receiving multiple responses that were generated by you when processing different parts of a complete database.

b) Now the goal is to put all these responses together to generate a single answer. Therefore, your answer will be the combination of the previous answers.

c) Interpret the context and see if it will be necessary to add, concatenate or join the responses sent previously.

d) Do not use *I don't know the answer* responses.

e) Always write the answer in English.

f) Format the response into a structure to be presented in html.

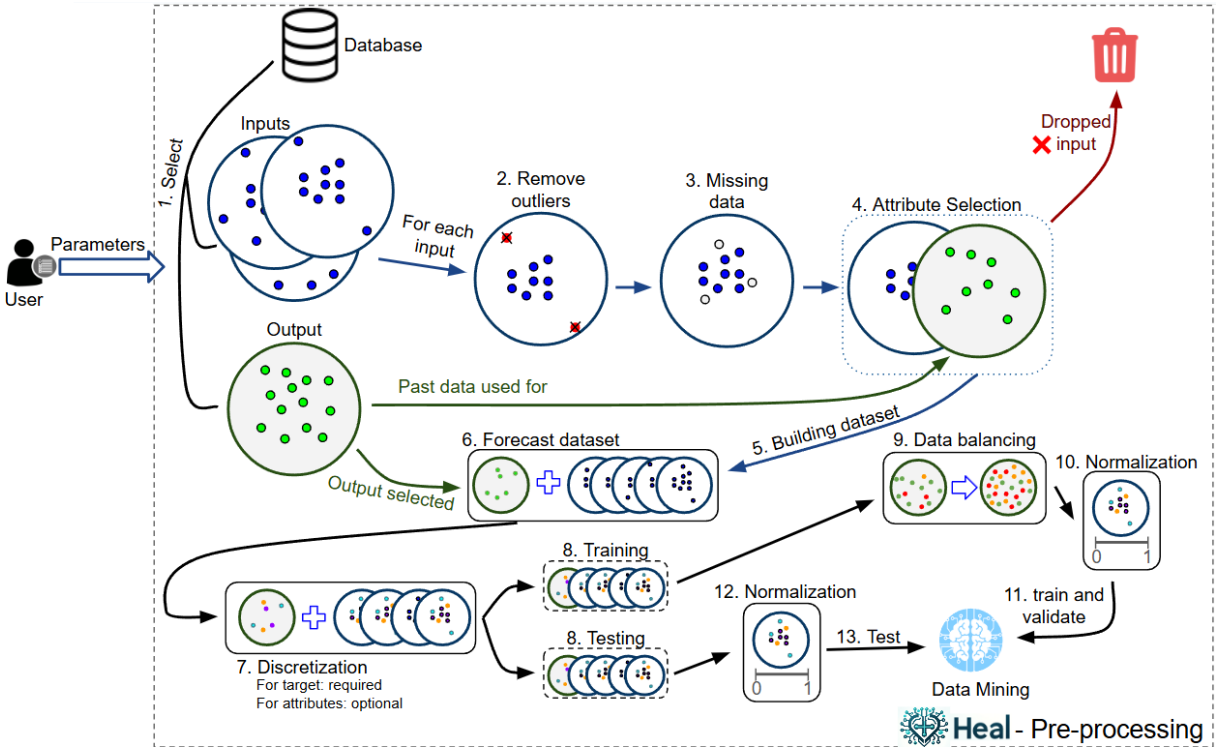
Finally, HEAL allows the user to define a context for knowledge discovery, indicating which attribute is relevant to it along with other settings that the user considers relevant. After this, the system will move on to the next step, responsible for automatically pre-processing the WHO attributes stored in the HEAL database, together with the understanding of the domain acquired by the LLM and the parameterizations made by the user.

5.1.2 Step 2 - Pre-processing

After defining the context based on the parameterization performed by the user, the algorithm applies the steps to define the attributes for building the dataset used for prediction. Therefore, the algorithm analyzes the data for each attribute, checking for missing data, selecting attributes, and removing outliers.

Figure 18 illustrates the steps defined for pre-processing. HEAL is configurable, allowing the user to choose customized options, such as parameters for outliers, missing data analysis, attribute selection parameters for the algorithms, and others. If the user does not define these parameters, HEAL is executed with a default parameterization.

Figure 18 – Steps of Pre-processing



Source: Elaborated by the Author.

After user parameterization, indicators are selected from the database (step 1 - select in Figure 18). Firstly, the outlier identification and exclusion algorithm is applied using the Interquartile Range (IQR) method and the user can choose these parameters in the system (step 2 - remove outliers in Figure 18). For our experiments, we set these values to 0.1 ($Q1$) and 0.9 ($Q3$), respectively.

Attributes with relevant missing data are treated, with two possibilities for this treatment: deletion or imputation (step 3 - missing data in Figure 18). If the definition is by deleting the attribute, HEAL analyzes the attribute dataset and, based on a defined parameter, removes it or not. For example, in the experiments carried out, we parameterized the system to remove attributes with more than 50% missing data. Another option is to perform data imputation, and HEAL is implemented with three algorithms: mean, median and kmeans, and the user can configure by choosing an option to use.

In the next step, the steps for feature selection and feature engineering methods are applied (step 4 - attribute selection in Figure 18). Two different algorithms were implemented for this step: Information Gain and Correlation Matrix. Both can be configured by the user and used to define which attributes will be used in the knowledge discovery process. In the experiments performed to test the system, we set the minimum Information Gain to 2.

In addition, the correlation matrix (Pearson) is also used to compare the previous data of each attribute with the attribute configured as the forecast output. Values close to -1 or 1 indicate a high correlation between the attributes, presenting a linear relationship between them, indicating that the attribute should not be used as input in the prediction algorithm for that proposed output. The cutoff value is also defined by a parameter in the system, and in the experiments we used -0.7 and 0.7.

It is also possible for the user to define which attributes will be analyzed, therefore, the user can define the resources that will be used for the knowledge discovery process. HEAL also addresses redundancies by removing duplicate instances, which may otherwise introduce bias or unnecessarily increase the complexity of the model.

We use empirical testing to define the parameters used in the experiments to determine the efficiency of our system. For each of the tests performed, these steps were followed to define the indicators used for each forecast made. Finally, the system sends the attributes of each dataset resulting (step 6 - forecast dataset in Figure 18) from this pre-processing to the discretization process (step 7 - discretization in Figure 18).

In each forecast that the system performs, the defined output is categorized, dividing the output values by the population of each country, transforming it into a percentage, and later dividing the results into Y different categories (class), which the user can also parameterize.

For discretization, calculations are performed using the lowest value (*Minimum*) and the highest value (*Maximum*) of the characterized data. This *Range* is calculated according to Equation 5.1:

$$Range = (Maximum - Minimum)/Y \tag{5.1}$$

where Y is the number of different classes. Thus, class 1 contains the data between Minimum and Minimum + (*Range**1); class 2 contains the data between Minimum + (*Range**1) and Minimum + (*Range**2); class 3 contains the data between Minimum + (*Range**2) and Minimum + (*Range**3), following up until the final category. This last class contains the data between Minimum + (*Range**(Y-1)) and Maximum. Only the lower bounds are always included, with only the last upper bound also included. Therefore, the problem is converted to a multi-class problem.

Table 8 exemplifies the discretization performed for Y categories. This process characterizes the dataset to divide the data into categories representing all the data. The objective is to send the attributes of each dataset resulting from this pre-processing to the Machine Learning algorithms.

Table 8 – Generalization to create output discretization (using Y categories)

Value Range (button)	Value Range (up)	Class
[Minimum	Minimum + (Range*1))	1
[Minimum + (Range*1)	Minimum + (Range*2))	2
[Minimum + (Range*2)	Minimum + (Range*3))	3
...	...	
[Minimum + (Range*(Y-2))	Minimum + (Range*(Y-1)))	Y-1
[Minimum + (Range*(Y-1))	Maximum]	Y

HEAL makes forecasts of defined categories from the percentage. Thus, in each forecast that the system performs, the defined output can be categorized as described. In the experiments, we used a total of five categories. However, this value can be configured by the system user.

In addition to output discretization, it is possible to apply the discretization process to features according to user parameterization. It is important to highlight that it is also possible to choose the customization of the discretization. Therefore, the method presented in the table presented was not used, and a discretization established by the user was used. Thus, the discretization may be more consistent with real contexts than performing a systematic discretization as presented.

Thus, the dataset is divided into training and testing (step 8 - training and testing in Figure 18). HEAL applies the oversampling or undersampling technique to the training dataset with the aim of artificially increasing the representation of minority classes and, consequently, balancing the defined classes more (step 9 - data balancing in Figure 18. For oversampling, the pipeline uses the Synthetic Minority Oversampling Technique (SMOTE), which creates synthetic examples not only by replicating existing ones, but also by interpolating between similar examples of the minority classes. Oversampling variations that exploit temporal and/or spatial dimensions can also be selected by the user. For undersampling, we use Random Under Sampler, which reduces the number of samples from the majority classes to balance with the number of samples from the minority classes.

Since the data processed by HEAL has a temporal dimension (panel data structure), it is important to offer balancing options that utilize this temporal dimension. Thus, in addition to the traditional methods mentioned, HEAL also offers balancing options based on the spatial and/or temporal dimension (details in Section 2.3), which are:

a) Respecting the temporal dimension (T-SMOTE (ZHAO et al., 2022)): a sliding window is applied over the time series, generating new examples only within consecutive

time windows, which preserves the temporal structure and avoids the creation of unrealistic points that would violate this dependence.

b) Respecting the spatial and temporal dimension: adaptation of the T-SMOTE technique for panel data, also respecting the spatial dimension of the data. Thus, the data is divided among the different spatial features and subsequently T-SMOTE is applied to each spatial dataset.

   For example, if the spatial dimension is related to countries and the temporal region is years, oversampling will be applied to each country separately in a pre-configured time window, such as every 5 years. In this way, the idea of spatial and temporal dimensions is used when applying data balancing.

After balancing, HEAL applies normalization to the training data (step 10 - normalization in Figure 18) and performs the training step on the selected Machine Learning algorithms (step 11 - train and validate in Figure 18). After training and defining the best combination of hyperparameters, the test data is also normalized (step 12 - normalization in Figure 18) and sent to the final testing stage (step 13 - test in Figure 18). Afterward, Machine Learning algorithms are used to predict the defined outputs from the selected indicators and the discretization performed. Therefore, HEAL is intended for classification problems. Despite this, HEAL is fully adjustable and scalable to handle regression issues in the future, with very few changes required.

These steps related to the data mining process are described in the subsequent subsection.

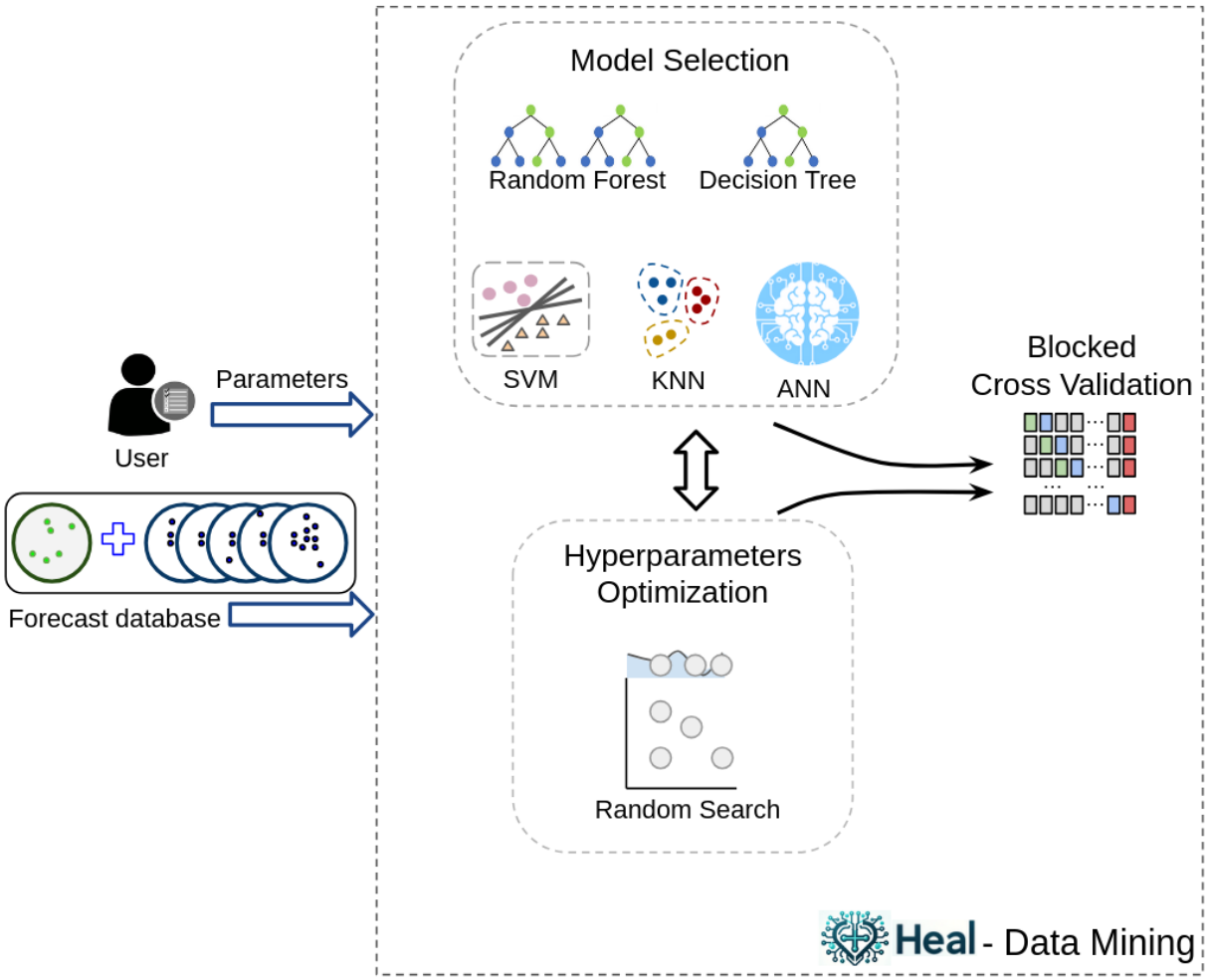
5.1.3 Step 3 - Data Mining

After the data pre-processing steps described in the previous section, this data mining step uses the data generated by the pre-processing and also some possible user-defined parameters, to perform model selection and hyperparameter optimization. Figure 19 illustrates the process carried out in this step.

To this end, Machine Learning algorithms were implemented in HEAL: Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Artificial Neural Network (ANN). As for hyperparameter optimization, we implemented the Random Search method. To validate the results, Blocked Cross Validation was used, respecting the temporal dimension of the data.

Initially, the Combined Algorithm Selection and Hyperparameter Optimization (CASH) method was used, in which several models are created from multiple algorithms and their hyperparameters, seeking to find the one that maximizes the model's perfor-

Figure 19 – Steps of Data Mining



Source: Elaborated by the Author.

mance. To test combinations, we use Random Search, which randomly selects combinations of hyperparameters to test and is able to find good models in a small fraction of the compute time (BERGSTRA; BENGIO, 2012).

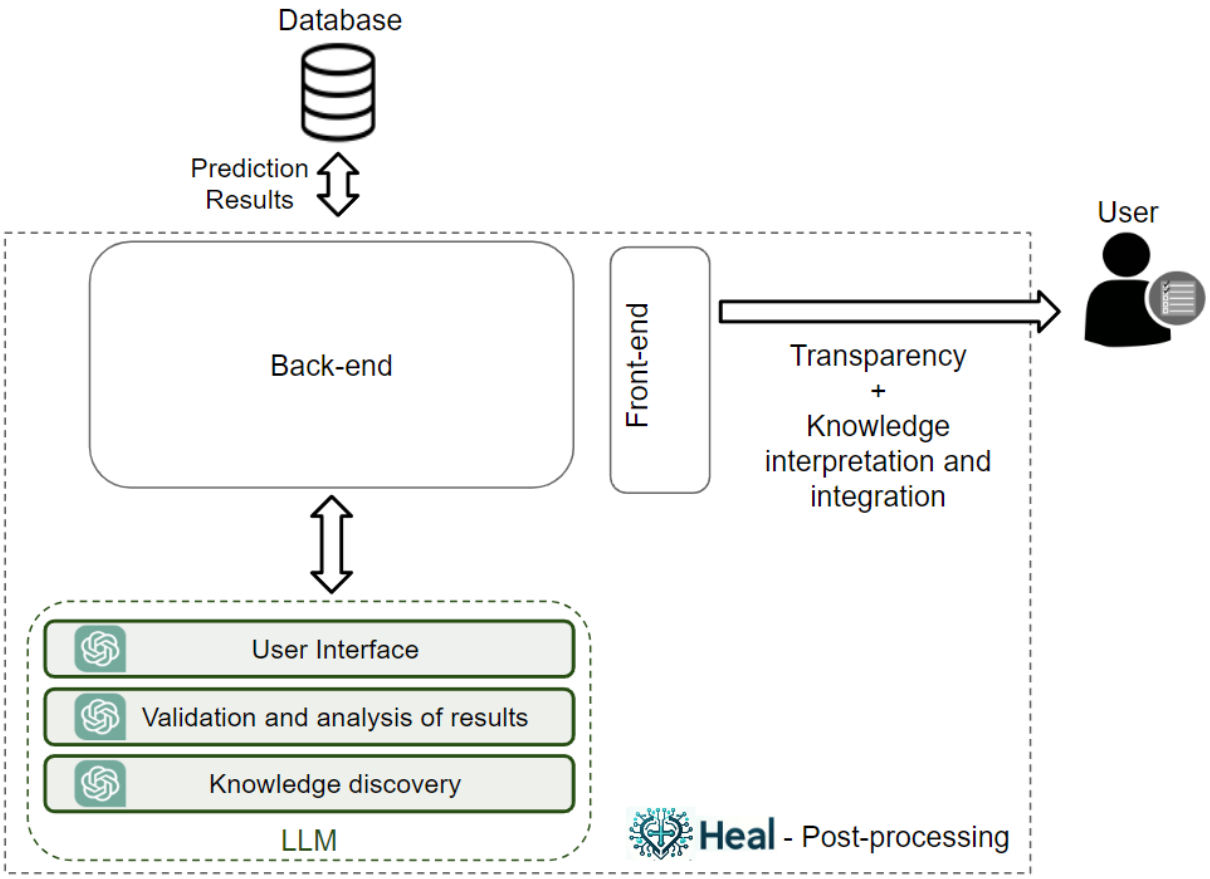
Finally, it is important to highlight that, in order not to execute a black box method, if the user wishes to have more control over HEAL, it is possible for the user to initially define hyperparameter ranges for each algorithm. Furthermore, all results found in each step are saved in a history so that the user can access and understand the process.

The defined combinations are tested using Blocked Cross Validation, which partitions the dataset into test and training/validation and iterates over this second set (always respecting the temporal issue) searching for the best combination of hyperparameters. For each experiment, the system calculates the performance criteria: hits, accuracy, precision, recall, f1-score and the confusion matrix.

5.1.4 Step 4 - Post-processing

Finally, the last step of HEAL aims to provide information about the knowledge discovery process carried out in the previous steps of the HEAL pipeline. Figure 20 illustrates this step.

Figure 20 – Steps of Post-processing



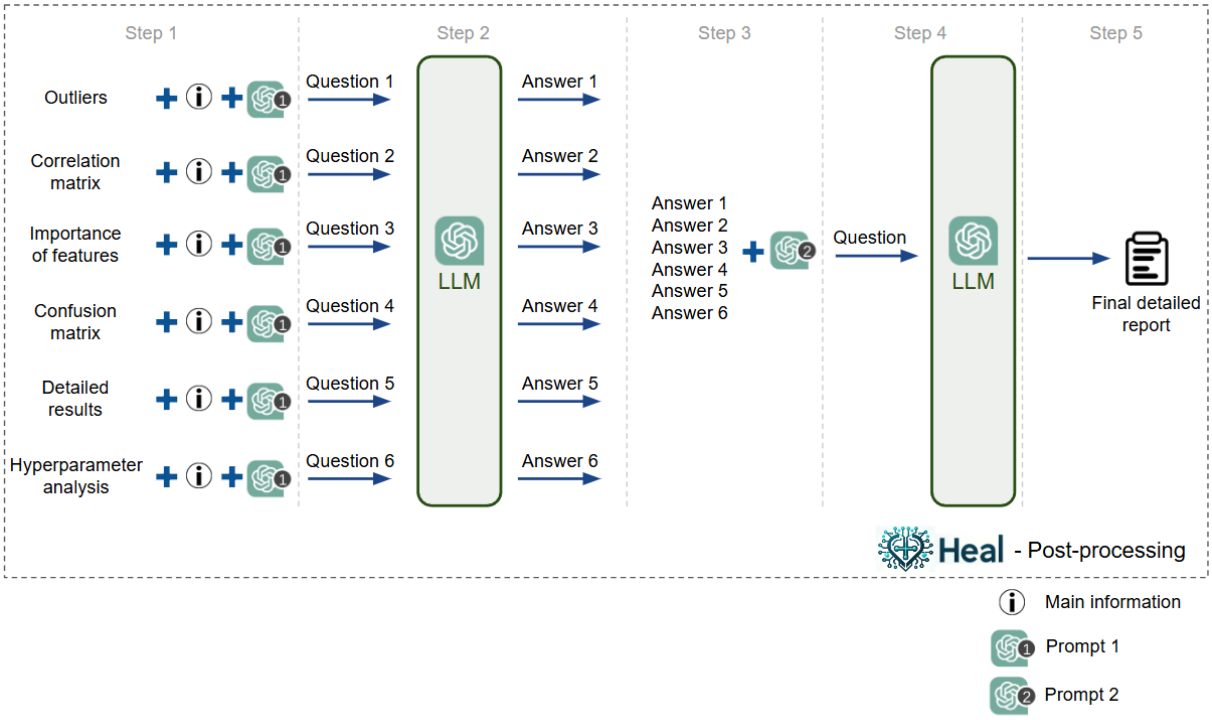
Source: Elaborated by the Author.

For the post-processing step, we developed a dynamic front-end capable of generating a screen with the results and important information for the user.

To complement the final results and assist in the knowledge discovery process, the system sends all found results to ChatGPT-4o following the pattern of a defined prompt. After that, the LLM process provides information and creates a report about the entire executed process. In addition, the LLM processes seeking to discover non-obvious knowledge by analyzing the main attributes defined by each algorithm, the results found and the parameters used.

Figure 21 illustrates in detail the steps required to use LLM in this post-processing.

Figure 21 – Steps to use LLM for Post-processing



Source: Elaborated by the Author.

Firstly, in step 1, HEAL creates different questions to send to the LLM ChatGPT-4o API. These contexts are created using the information generated during the previous steps of the autoML process: outliers, correlation matrix, importance of features, confusion matrix, detailed results and hyperparameter analysis. These contexts are joined with main information (details about the forecasted indicator, such as name, time period, and region) and prompt 1. So, some information was put into this prompt to build the LLM profile:

a) You are a helpful assistant to an AutoML system.

b) In this system, you help with the final post-processing step and then look for useful knowledge discoveries.

c) You will simulate the behavior of a domain expert in the healthcare area.

d) You will receive some information that is the result of a Machine Learning process.

e) Search primarily for non-obvious knowledge discoveries, interpreting the submitted results to assist the user in searching for knowledge discoveries.

f) Always write the answer in English.

In the second stage, each of these questions is sent to the LLM, which provides an answer for each question. After that, to present a detailed final report to the user, it is important to merge all these responses. Therefore, HEAL merges the responses with a second prompt (step 3) to send this final question to LLM (step 4). Some information is important to build the LLM profile in prompt 2:

a) You are receiving multiple responses that were generated from the results found in the AutoML process steps.

b) Now the goal is to put all these responses together to generate a single answer. Therefore, your answer will be the combination of the previous answers.

c) Interpret the context and see if it will be necessary to add, concatenate or join the responses sent previously.

d Write the results found in report format.

e) Always write the answer in English.

f) Write the response in an HTML text format to be used within a website that is already HTML compliant.

Finally, the LLM provides a final response in the form of a detailed final report (step 5). This final report is presented in our front-end to help the user make decisions and discover knowledge.

Finally, this interface presents the user with all the information and data relating to each of the steps taken (each of the responses generated separately), presenting a complete record of each step, in addition to presenting the final detailed report. Therefore, the proposal is that HEAL is not a black box method, being different from most works found in the literature. HEAL presents all the information from all the steps, giving the user more control and making it easier to understand what was done throughout the process. This can also help the user identify errors and issues during each step, which can help correct them in future runs.

6 EXPERIMENTS AND RESULTS

After the HEAL database creation process, we studied the database to define possible health areas for experiments. Four of these possible health areas and indicators for predictions were defined as follows:

- *Experiment 1* - Sexual health category: prediction of the number of people living with HIV.

  Sexual health is related to a respectful approach to sexuality and safe sexual relations, free from discrimination and violence. It is essential for the general health and well-being of individuals, couples, and families, in addition to being fundamental for countries' social and economic development (WHO, 2024).

- *Experiment 2* - Nutritional health category: prediction of the prevalence of anemia in children.

  Malnutrition is found worldwide and threatens human health (mainly in low-income and middle-income countries). Poor nutritional health generates developmental, economic, and social impacts for individuals, families, and countries (WHO, 2024).

- *Experiment 3* - Safe sanitation category: prediction of population using at least basic sanitation services.

  Safe sanitation systems are essential to protect public health, and the absence of these can result in serious illnesses, impacting the health of people and countries (WHO, 2024).

- *Experiment 4* - Mental health category: prediction of age-standardized suicide rates in men.

  Mental health is related to a state of mental well-being that allows people to live with the stresses of life, learn and work well, and contribute to the community. Therefore, poor mental health affects the individual, the community and countries (WHO, 2024).

The four experiments presented in this section aim to prove that the system created allows the automated correlation of various information from different areas of health to make predictions with a high level of assertiveness. Thus, the objective is to present possibilities of forecasts from the developed system and the proposed methodology.

After running the experiments, we performed the evaluation and comparison of the prediction results with two different AutoML frameworks found in the literature (H2O and TPOT). These are well-established AutoML frameworks, with several published works using them for knowledge discovery.

Table 9 illustrates the total output data of each experiment and, according to the parameters chosen for the performed experiments, the total data referring to the training set and the comprehensive test set data, following the Blocked Cross Validation method. This table also presents the total runtime of each experiment in the used test environment. For the first experiment, we set the total training data to 20%, with the remainder (80%) for training. For the other experiments, we use 30% and 70% for testing and training, respectively. This value can be configured in HEAL according to the characteristics of the experiment and user customization.

Table 9 – Total output data, divided into training and testing

Exp.	Training/Validation Data	Testing Data	Total Data	Runtime (sec)
1	2,175	932	3,107	6,429.22
2	2,883	723	3,615	8,782.89
3	3,396	804	4,200	9,949.28
4	2,712	678	3,390	8,446.61

The four results of the experiments carried out are presented below. The complete description of each step of the HEAL processing (such as missing data analysis, outliers analysis, attribute definition, application of Machine Learning algorithms, and others) and also the description of all the results found are detailed in our repository of GitHub, presented in the introduction. To run the system in a new experiment different from those performed in this thesis, parameterize the system, run it, and follow the generated results.

6.1 Experiment 1 - Sexual health category: prediction of the number of people living with HIV

This experiment aims to predict the number of people living with HIV. Table 10 illustrates the results obtained for this experiment, comparing the accuracy, precision, recall and f1-score metrics of the results found by the Artificial Neural Network (ANN), Random Forest (RF), Decision Tree (DT), K-nearest Neighbors (KNN) and Support Vector Machine (SVM) algorithms.

For model selection and hyperparameter optimization, several combinations were tested using Blocked Cross Validation and random search. For each algorithm, some

Table 10 – Experiment Results 1

	Metrics (%)	ANN	DT	RF	KNN	SVM
Total 932 instances	Accuracy	70.39	83.58	86.37	87.88	82.40
	Precision	78.06	52.54	61.29	57.39	61.04
	Recall	22.87	54.13	54.39	62.75	61.27
	F1-score	35.37	53.32	57.63	59.95	61.16
Class 1 666 instances	Precision	75.85	90.63	90.01	97.66	94.57
	Recall	96.70	95.80	98.80	93.84	91.44
	F1-score	85.02	93.14	94.20	95.71	92.98
Class 2 172 instances	Precision	100.00	70.90	89.38	79.52	68.42
	Recall	0.00	55.23	58.72	76.74	52.91
	F1-score	0.00	62.09	70.88	78.11	59.67
Class 3 68 instances	Precision	14.46	65.45	57.81	57.14	42.22
	Recall	17.65	52.94	54.41	76.47	83.82
	F1-score	15.89	58.54	56.06	65.41	56.16
Class 4 11 instances	Precision	100.00	0.00	0.00	0.00	25.00
	Recall	0.00	0.00	0.00	0.00	18.18
	F1-score	0.00	0.00	0.00	0.00	21.05
Class 5 15 instances	Precision	100.00	35.71	69.23	52.63	75.00
	Recall	0.00	66.67	60.00	66.67	60.00
	F1-score	0.00	46.51	64.29	58.82	66.67

default settings were defined, so these were used, although prior user configuration is possible. Table 11 illustrates the best hyperparameter combination found for each of the algorithms used, as well as the average test score (Score (%)), which was calculated across all folds to evaluate the overall performance of the model, and the standard deviation of the test score across all folds was used to assess the consistency of the model performance (Standard Deviation (%)).

In addition to presenting accuracy, precision, recall and f1-score, and showing in more detail the performance of the prediction system, we generated confusion matrices for all tests in each of the experiments carried out. It is essential to highlight the performance of the forecasting system to hit the atypical and consecutively complex cases or, even when the system made a mistake in the forecast, it identified results close to the expected results. Confusion matrices of all tests of all experiments can be seen in our project's GitHub repository, presented in the introduction.

In this experiment 1, the ANN, DT, KNN, RF and SVM algorithms achieved f1-score values higher than 85.00% in the majority class (class 1) and good values in the other classes (class 2, 3 and 5), with an excess of class 4 (minority class), in which no algorithm was able to predict. This difficulty in predicting class 4 is related to the fact that it is the minority class in the dataset and, consequently, the most difficult to predict. Therefore, it is important to evaluate the balancing performed in the pre-processing step,

Table 11 – Best combination of hyperparameters for each algorithm - Experiment 1

Alg.	Hyperparameters	Score (%)	Std (%)
ANN	neurons = 51 and dropout_rate = 0.2 and activation = relu and optimizer = sgd and num_layers = 3	85.61	2.32
DT	splitter = best and min_samples_split = 2 and min_samples_leaf = 1 and max_features = None and max_depth = 10 and criterion = gini	85.00	1.48
RF	n_estimators = 100 and min_samples_split = 2 and max_features = log2 and max_depth = 10 and criterion = gini and bootstrap = True	88.77	4.06
KNN	weights = distance and p = 2 and n_neighbors = 3 and leaf_size = 30 and algorithm = auto	93.51	2.15
SVM	kernel = poly and gamma = scale and degree = 3 and C = 10	90.91	3.14

seeking a balance that improves the representativeness of this class in the dataset and that contributes to a better prediction of this class. For this purpose, different balancing configurations can be tested in HEAL in future experiments.

Finally, it is important to highlight that the Artificial Neural Network (ANN) implemented in HEAL is a Feedforward Multilayer Perceptron (MLP), trained using the backpropagation algorithm. Despite this, the ANN did not perform well on this dataset, failing to correctly predict any of the outcomes beyond the majority class (class 1). This poor performance can be justified by two main possible reasons: the relatively small size of the dataset, which may not provide enough information for effective training of the ANN, and the fact that the MLP architecture used is not ideal for panel data structure contexts. Alternative ANN architectures or additional pre-processing techniques might be necessary to achieve better performance in this context.

6.2 Experiment 2 - Nutritional health category: prediction of the prevalence of anemia in children.

This experiment aims to predict the prevalence of anemia in children aged 6-59 months. Table 12 illustrates the results obtained for this experiment, which are excellent indicators of system performance to make predictions in this context. This experiment presented the best prediction result, obtaining up to 93.81% f1-score using the RF algorithm. It is essential to highlight the performance of the forecasting system to hit the atypical and consecutively tricky cases. The other results, such as the correlation matrix and hyperparameter analysis, can be found in our GitHub repository.

Analyzing the table, it is possible to observe that all algorithms performed well

Table 12 – Experiment Results 2

	Metrics (%)	ANN	DT	RF	KNN	SVM
Total 723 instances	Accuracy	56.71	97.23	99.17	98.34	97.10
	Precision	33.58	83.21	97.63	88.27	83.16
	Recall	58.88	76.17	90.28	96.84	91.45
	F1-score	42.77	79.54	93.81	92.36	87.10
Class 1 669 instances	Precision	100.00	98.38	99.26	99.70	99.24
	Recall	56.65	99.70	100.00	98.51	97.76
	F1-score	72.33	99.03	99.63	99.10	98.49
Class 2 9 instances	Precision	2.54	64.29	88.89	50.00	47.37
	Recall	55.56	100.00	88.89	100.00	100.00
	F1-score	4.85	78.26	88.89	66.67	64.29
Class 3 20 instances	Precision	9.76	90.91	100.00	100.00	78.26
	Recall	40.00	50.00	95.00	100.00	90.00
	F1-score	15.69	64.52	97.44	100.00	83.72
Class 4 11 instances	Precision	21.21	62.50	100.00	91.67	90.91
	Recall	63.64	45.45	81.82	100.00	90.91
	F1-score	31.82	52.63	90.00	95.65	90.91
Class 5 14 instances	Precision	34.38	100.00	100.00	100.00	100.00
	Recall	78.57	85.71	85.71	85.71	78.57
	F1-score	47.83	92.31	92.31	92.31	88.00

in this experiment (with f1-score above 79.00%), with the exception of ANN (f1-score of 42.77%). Repeating the poor performance of experiment 1, the ANN also did not perform well in this experiment. This reinforces the idea that the ANN implemented in HEAL is not the best option for the processed data structure.

Finally, it is important to analyze that, unlike experiment 1, in this experiment 2 the models were able to predict the minority classes with good results, especially the RF which reached more than 88.00% in class 2 and more than 90.00% in classes 3, 4 and 5.

6.3 Experiment 3 - Safe sanitation category: prediction of population using at least basic sanitation services

This experiment aims to predict the number of population using at least basic sanitation services. Table 13 illustrates the results obtained for this experiment. To test different HEAL configurations, in this experiment, we changed the total number of classes in the target discretization to 4 (instead of 5, as in the other experiments). The results found in this experiment were also satisfactory, with f1-Score of up to 87.46% (RF) and good performance even in minority and more difficult to predict classes. However, once again, the ANN algorithm did not perform well in predicting this dataset. Once again, we believe that the poor performance of the ANN is related to the same problems mentioned

in experiments 1 and 2.

Table 13 – Experiment Results 3

	Metrics (%)	ANN	DT	RF	KNN	SVM
Total 804 instances	Accuracy	42.79	95.52	98.01	96.52	93.91
	Precision	32.53	82.81	91.65	73.61	72.07
	Recall	63.25	65.57	85.00	88.35	79.73
	F1-score	27.83	71.88	87.46	79.78	75.12
Class 1 741 instances	Precision	98.68	97.34	98.54	99.45	98.33
	Recall	40.49	98.65	100.00	97.57	95.28
	F1-score	57.42	97.99	99.26	98.50	96.78
Class 2 45 instances	Precision	9.33	70.00	100.00	73.08	55.22
	Recall	71.11	62.22	71.11	84.44	82.22
	F1-score	16.49	65.88	83.12	78.35	66.07
Class 3 10 instances	Precision	5.00	62.50	66.67	53.85	44.44
	Recall	40.00	50.00	80.00	70.00	40.00
	F1-score	8.89	55.56	72.73	60.87	42.11
Class 4 8 instances	Precision	15.69	100.00	100.00	66.67	88.89
	Recall	100.00	50.00	87.50	100.00	100.00
	F1-score	27.12	66.67	93.33	80.00	94.12

6.4 Experiment 4 - Mental health category: prediction of age-standardized suicide rates in men

Experiment 4 tested the use of dimensions to refine the choice of attributes for prediction. Changing the use of dimensions, this experiment predicts the age-standardized suicide rates, selecting only data referring to men. Therefore, we used the gender dimension, choosing only data about men for this prediction. Table 14 illustrates the results obtained for this experiment.

As demonstrated in previous experiments, again the DT, RF, KNN and SVM algorithms performed well in predictions on this dataset, with a slight worsening in the DT results, since this algorithm had difficulty predicting minority classes. Therefore, we believe that this may be due to the hyperparameter optimization of this algorithm for this scenario, and that HEAL may not have found the best combination for this algorithm. This possibility is the most recommended, as all other algorithms (with the exception of ANN) performed well in all classes, including minority ones.

As mentioned, once again the ANN did not perform well. We believe, once again, that the reasons for this are the same as in the previous experiments. A more detailed discussion of this will be given in the following sections.

Table 14 – Experiment Results 4

	Metrics (%)	ANN	DT	RF	KNN	SVM
Total 678 instances	Accuracy	63.27	89.38	96.61	96.46	90.12
	Precision	30.18	58.81	85.66	82.39	74.72
	Recall	40.88	47.73	81.42	87.78	80.42
	F1-score	34.73	52.69	83.49	85.00	77.47
Class 1 607 instances	Precision	98.99	94.90	98.19	99.16	99.10
	Recall	64.42	95.06	98.52	97.36	90.77
	F1-score	78.04	94.98	98.36	98.25	94.75
Class 2 38 instances	Precision	15.08	34.69	86.49	78.72	41.57
	Recall	78.95	44.74	84.21	97.37	97.37
	F1-score	25.32	39.08	85.33	87.06	58.27
Class 3 8 instances	Precision	0.00	50.00	100.00	100.00	100.00
	Recall	0.00	12.50	75.00	100.00	75.00
	F1-score	0.00	20.00	85.71	100.00	85.71
Class 4 11 instances	Precision	5.26	44.44	63.64	57.14	52.94
	Recall	18.18	36.36	63.64	72.73	81.82
	F1-score	8.16	40.00	63.64	64.00	64.29
Class 5 14 instances	Precision	31.58	70.00	80.00	76.92	80.00
	Recall	42.86	50.00	85.71	71.43	57.14
	F1-score	36.36	58.33	82.76	74.07	66.67

6.5 Evaluations and Comparisons

After performing the experiments, we evaluated the results found and compared them with two different AutoML found in the literature: H2O and Tree-based Pipeline Optimization Tool (TPOT). We chose these two AutoML to represent the others found in the literature, as there are several options. Both are established AutoML, with considerable time in operation and several published works using them. More information about them can be found in the Subsection 3.1.5, page 50.

Furthermore, it is important to highlight that both have pre-processing, model selection and hyperparameter optimization steps. However, since they do not have a data collection step, we use HEAL data collection to generate .csv files representing the experiment datasets to use as input for these two AutoML. The HEAL data collection step is implemented together with some pre-processing steps (outlier analysis, missing data, feature selection, and discretization), so these steps were also applied to the generated datasets before running H2O and TPOT.

Given this, it is important to highlight that, even comparing HEAL with other AutoML solutions, we used our system to generate the dataset (data collection) and for some pre-processing steps, as this would be necessary to generate the dataset for use with these AutoML solutions.

Thus, we ran H2O and TPOT once for each of the experiments performed (experiment 1 to 4). Table 15 shows and compares the results found in these two AutoML and also in HEAL for experiment 1, comparing the results of the best model selected by HEAL (SVM for this experiment) with the best models selected by AutoML H2O and TPOT.

Table 15 – Results of Experiment 1 - Comparison of HEAL, H2O and TPOT

	Metrics (%)	**HEAL**	**H2O**	**TPOT**
Total	Precision	61.04	58.13	58.86
932 instances	Recall	61.27	54.00	47.72
	F1-score	61.16	55.57	51.02
Class 1	Precision	94.57	91.16	88.86
666 instances	Recall	91.44	97.60	97.00
	F1-score	92.98	94.27	92.75
Class 2	Precision	68.42	85.37	81.03
172 instances	Recall	52.91	61.05	54.65
	F1-score	59.67	71.19	65.28
Class 3	Precision	42.22	60.27	57.75
68 instances	Recall	83.82	64.71	60.29
	F1-score	56.16	62.41	58.99
Class 4	Precision	25.00	0.00	0.00
11 instances	Recall	18.18	0.00	0.00
	F1-score	21.05	0.00	0.00
Class 5	Precision	75.00	53.85	66.67
15 instances	Recall	60.00	46.67	26.67
	F1-score	66.67	50.00	38.10

For these experiments, H2O defines Stacked Ensembles as the best method, while TPOT defines Gradient Boosting Classifier. It is possible to observe in the table that the model defined by HEAL obtained a better final f1-score and even in the minority classes (4 and 5) and consequently more difficult to predict. In the other classes (majority), all models performed similarly. It is also worth noting that H2O and TPOT do not find a model capable of reaching any value in class 4 (minority), with f1-score equal to 0, unlike HEAL, in which SVM managed to get some cases right (f1-score equal to 21.05%), even though this is a difficult class to predict. The model selected by each AutoML and the main hyperparameter settings were:

HEAL - Support Vector Machine (C: 10, degree: 3, gamma: scale, kernel: poly).

H2O - Stacked Ensembles (training frame: none, response column: none, base models: [ ], metalearner algorithm: auto, metalearner params: none, seed: -1).

TPOT - Gradient Boosting Classifier (learning rate: 0.5, max features: 0.15, min samples leaf: 19, min samples split: 10, random state: 42, subsample: 0.9).

Table 16 shows the comparisons of the best model defined by HEAL (Random Forest) with the models found by H2O (Stacked Ensembles) and TPOT (Decision Tree Classifier). Once again, HEAL obtained a model that was able to predict the dataset better (with a total f1-score equal to 93.81%). In this experiment, the model defined by HEAL was superior in all classes (1 to 5) of the target dataset, resulting in a much higher total f1-score than the H2O and TPOT models.

Table 16 – Results of Experiment 2 - Comparison of HEAL, H2O and TPOT

	Metrics (%)	**HEAL**	**H2O**	**TPOT**
Total	Precision	97.63	85.22	87.10
723 instances	Recall	90.28	75.44	72.39
	F1-score	93.81	78.83	74.15
Class 1	Precision	99.26	98.42	98.62
669 instances	Recall	100.00	99.70	100.00
	F1-score	99.63	99.06	99.31
Class 2	Precision	88.89	100.00	82.35
9 instances	Recall	88.89	62.50	87.50
	F1-score	88.89	76.92	84.85
Class 3	Precision	100.00	70.00	100.00
20 instances	Recall	95.00	46.67	40.00
	F1-score	97.44	56.00	57.14
Class 4	Precision	100.00	68.18	100.00
11 instances	Recall	81.82	83.33	44.44
	F1-score	90.00	75.00	61.54
Class 5	Precision	100.00	89.47	54.55
14 instances	Recall	85.71	85.00	90.00
	F1-score	92.31	87.18	67.92

The model selected by each AutoML and the main hyperparameter settings for this experiment's dataset were:

HEAL: Random Forest Classifier (bootstrap: true, criterion: gini, max depth: 10, max features: 2, min samples split: 2, n estimators: 150).

H2O: Stacked Ensemble (training frame: none, response column: none, base models: [ ], metalearner algorithm: auto, seed: -1).

TPOT: Decision Tree Classifier (criterion: entropy, max depth: 6, min samples leaf: 15, min samples split: 16, random state: 42).

Table 17 shows the results found in experiment 3 by the best models defined by AutoML HEAL, H2O and TPOT, which are Random Forest Classifier, Stacked Ensemble and Random Forest Classifier, respectively. Of the 4 experiments performed, this was the

only one in which HEAL was unable to obtain better results than the other two AutoML, especially in class 3 (minority), in which HEAL had worse results than H2O and TPOT, despite also having found good results (more than 87.00%).

Table 17 – Results of Experiment 3 - Comparison of HEAL, H2O and TPOT

	Metrics (%)	**HEAL**	**H2O**	**TPOT**
Total	Precision	91.65	92.98	93.59
804 instances	Recall	85.00	91.88	93.06
	F1-score	87.46	91.97	93.18
Class 1	Precision	98.54	98.92	98.92
741 instances	Recall	100.00	99.73	99.46
	F1-score	99.26	99.33	99.19
Class 2	Precision	100.00	98.00	89.09
45 instances	Recall	71.11	77.78	77.78
	F1-score	83.12	86.73	83.05
Class 3	Precision	66.67	75.00	86.36
10 instances	Recall	80.00	90.00	95.00
	F1-score	72.73	81.82	90.48
Class 4	Precision	100.00	100.00	100.00
8 instances	Recall	87.50	100.00	100.00
	F1-score	93.33	100.00	100.00

The model selected by each AutoML and the main hyperparameter settings for the experiment 3 dataset were:

HEAL: Random Forest Classifier (bootstrap: true, criterion: gini, max depth: 20, max features: 6, min samples split: 2, n estimators: 20).

H2O: Stacked Ensemble (training frame: none, response column: none, base models: [ ], metalearner algorithm: auto, seed: -1).

TPOT: Random Forest Classifier (criterion: entropy, max features: 0.75, min samples leaf: 2, min samples split: 7, random state: 42, n estimators: 100, bootstrap: true).

Finally, the same process was performed in experiment 4 and the results are presented in Table 18. Once again, HEAL performed better with a total f1-score equal to 85.00% higher than H2O (65.69%) and TPOT (64.24%). In this experiment, the models selected by AutoML were K-nearest Neighbors, Stacked Ensemble, and Random Forest Classifier for HEAL, H2O, and TPOT, respectively.

The model selected by each AutoML and the main hyperparameter settings for the experiment 4 dataset were:

Table 18 – Results of Experiment 4 - Comparison of HEAL, H2O and TPOT

	Metrics (%)	HEAL	H2O	TPOT
Total	Precision	82.39	67.84	65.72
678 instances	Recall	87.78	65.17	63.96
	F1-score	85.00	65.69	64.24
Class 1	Precision	99.16	97.90	96.96
607 instances	Recall	97.36	97.36	98.35
	F1-score	98.25	97.63	97.65
Class 2	Precision	78.72	63.51	77.19
38 instances	Recall	97.37	77.05	72.13
	F1-score	87.06	69.63	74.58
Class 3	Precision	100.00	83.33	85.71
8 instances	Recall	100.00	83.33	100.00
	F1-score	100.00	83.33	92.31
Class 4	Precision	57.14	55.56	25.00
11 instances	Recall	72.73	31.25	12.50
	F1-score	64.00	40.00	16.67
Class 5	Precision	76.92	38.89	43.75
14 instances	Recall	71.43	36.84	36.84
	F1-score	74.07	37.84	40.00

HEAL - K-Nearest Neighbors (algorithm: auto, leaf size: 30, n neighbors: 7, p: 2, weights: distance).

H2O - Stacked Ensemble (training frame: none, response column: none, base models: [ ], metalearner algorithm: auto, seed: -1).

TPOT - Random Forest Classifier (bootstrap: False, criterion: 'entropy', max features: 0.5, min samples leaf: 2, min samples split: 10, random state: 42, n estimators: 100).

Again, it is important to emphasize that the objective here is to compare the results found in step 3 - Data Mining of the AutoML pipeline. In addition to HEAL having found superior results in 3 of the 4 experiments performed (75% of experiments), it is capable of executing all steps of the AutoML pipeline, unlike all other AutoML found in the literature. Finally, it is important to highlight that the datasets used in AutoML H2O and TPOT were generated by the HEAL system itself, since these two do not have the necessary steps to generate the WHO datasets.

In addition to the different models suggested by the compared AutoML (H2O and TPOT), which may have already influenced the worst result, the pre-processing steps performed by HEAL may have been the difference between our solution and the others. We believe that the data balancing performed with respect to temporal issues (applied

mainly to panel data structures) may also have had a positive impact on improving HEAL results, in addition to the models chosen in the Data Mining stage.

All H2O and TPOT codes, the datasets used and the results found for each experiment can be found in our GitHub repository*.

6.6 Case Studies

After the four experiments carried out in different areas of health, we proposed three case studies presenting possibilities of using the system for public policies in the healthcare area:

- *Case Study 1* - Prevalence of anemia in children in the countries of the Americas in 2019.
- *Case Study 2* - Age-standardized suicide rates in men (per 100,000 population) in the countries of the Africa in 2019.
- *Case Study 3* - Influence of tuberculosis and hypertension in prevalence of anemia in children in India in 2021 (using GBD data source).

The contexts of case studies 1 and 2 are justified because they are attributes with data referring to most of the countries and regions (continents) used in each case study. Although this chosen year (2019) is relatively distant from the current year of this work (2024), WHO data may take time to register as they have been updated over the years. Besides, as the intention is to validate the results of the system, we need to perform experiments on known data to evaluate the results. Thus, for these attributes, the most current year in the WHO is 2019. Therefore, the system will be applied to predict 2019 data, but it would work and could be used in different regions (continents) and years.

To this end, step 1 – Domain Understanding and Data Collection – was parameterized using the $AGEGROUP$ and SEX dimensions and using all attributes. The data uploaded was from June 26, 2023. Different parameterizations can increase or decrease the detail of the data in the database and, consecutively, influence the quality of the analyses, the accuracy of the system's predictions, and the knowledge discovery.

These case studies (1 and 2) were applied for 2019 to validate the system through forecast metrics on existing data. However, although not presented here, it can be used for unknown years of future scenarios, such as the prevalence of anemia in children for a future year and, consequently, unknown at the moment.

*https://github.com/cart-pucminas/HEAL

From the results found by the forecasts, both the main attributes directly related to the forecast attribute and the high accuracy of the forecasts, it is possible to define directions, ideas, and policies that aim to combat the high rates of a given disease. For example, based on the main directly related indicators, sectors or public bodies of a given country can propose investments in these indicators to combat or control the high index of this attribute. Furthermore, based on these indicators for any region and year, it is possible to make predictions and determine indicators for any region and year.

Finally, these case studies (1 and 2) aim to show the system's potential for three specific indicators out of the more than 2,000 possible indicators made available by the WHO.

Case Study 3 is an extension of Case Study 1. After carrying out this first case study, we realized that the study could be applied to a specific country, changing the spatial dimension from countries to states. Therefore, in view of this and seeking to explore the capacity of the execution pipeline for any dataset in the panel data structure, we carried out case study 3 testing the system in a dataset other than the WHO, choosing the Global Burden of Disease (GBD).

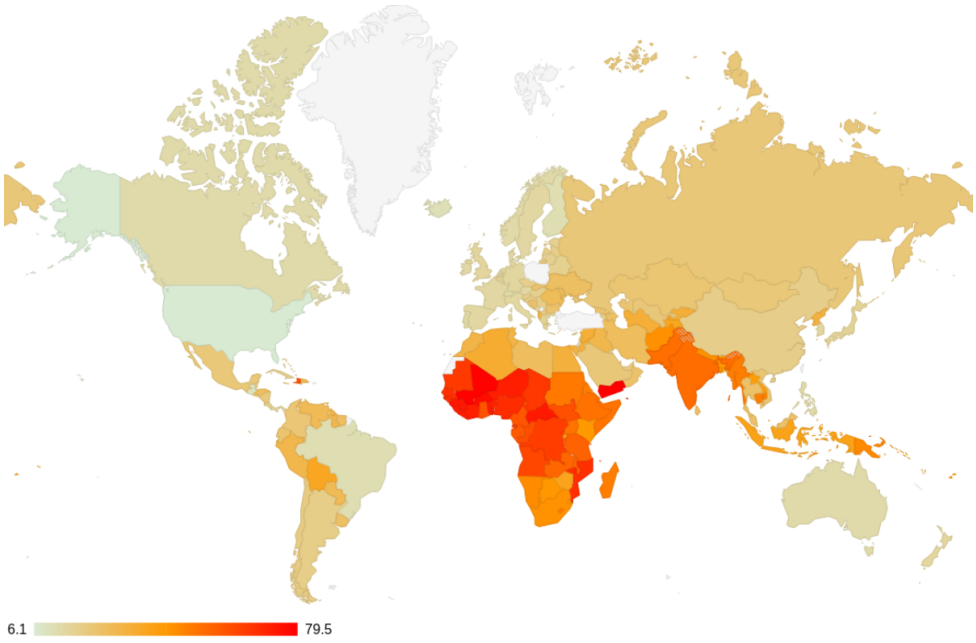
6.6.1 Case Study 1 - Prevalence of anemia in children in the countries of the Americas in 2019

Figure 22 illustrates the world map, in which the color of each country reflects the percentage of prevalence of anemia in children (%) in the year 2019. It is observed that the closer to the red color, the greater the prevalence of anemia in children (maximum value equal to 79.5%). Likewise, colors close to green indicate lower prevalence (minimum value equal to 6.1%), and, finally, countries in white color do not have data registered in WHO.

This study aims to run the prediction system configured to predict the attribute prevalence of anemia in children (%) in the countries of the region (continent) of the Americas in 2019. Based on WHO data, for the year 2019, there are 34 countries in the Americas region. With the generation of system execution outputs, it will also be possible to define the main attributes directly related to the prediction attribute, valuable information for decision-making that seeks to improve the related indexes. Figure 23 illustrates the parameterization of the HEAL carried out for the case study presented. From this figure it is possible to see that the user can have control over some system parameters.

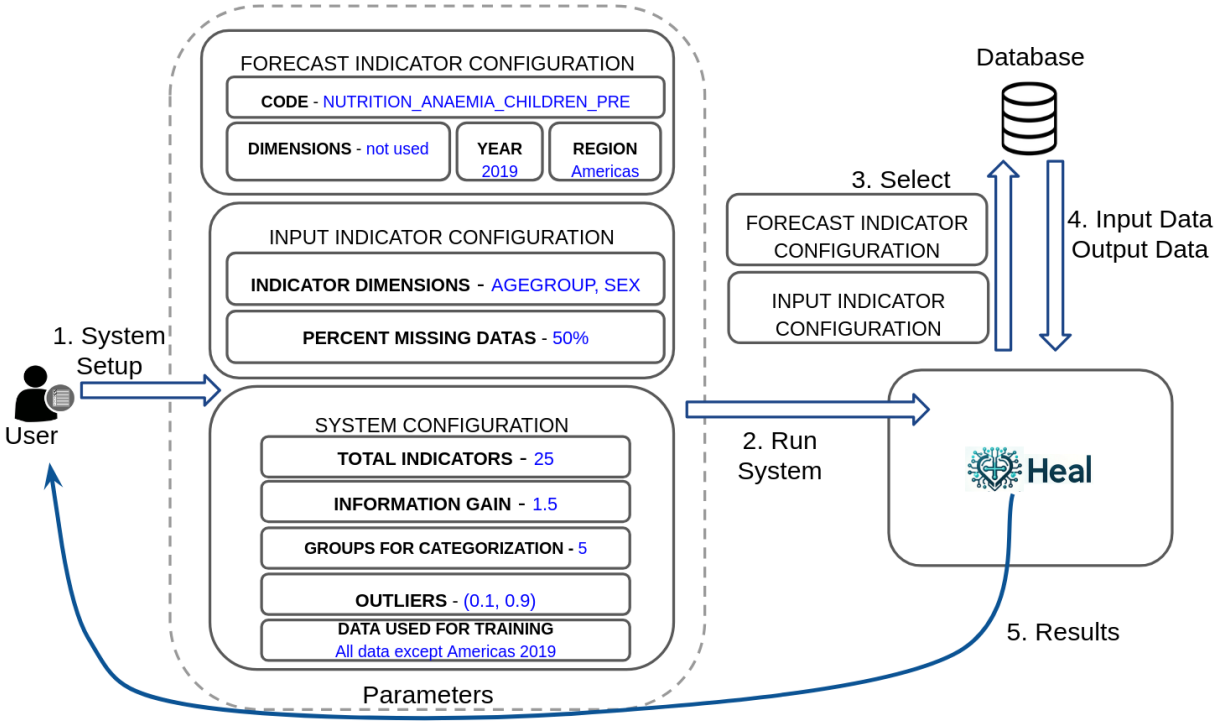
After running the system, lasting 10,069.14 seconds (about 2.79 hours), it is possible to analyze and configure the main attributes chosen by the system that are directly related to the prediction attribute and, consecutively, were used as training data for pre-

Figure 22 – World map depicting the prevalence of anemia in children (%) in the year 2019



Source: Elaborated by the Author.

Figure 23 – HEAL parameterization performed for the case study 1



Source: Elaborated by the Author.

dictions (this information can be found in our GitHub repository). This total of attributes

was found due to the parameterization performed in the system configuration, using 20 for total indicators. Given this information, countries seeking policies to improve data on the prevalence of anemia in children may seek to invest in these attributes.

For this study, the five algorithms proposed in HEAL were executed for the predictions. In all experiments, data related to the Americas region in 2019 was used for testing (34, representing 0.94%), while all other data (3,614, representing 99.06%) were used for model training. The attribute prevalence of anemia in children (%) was categorized into five groups. For each algorithm, accuracy, precision, recall and f1-score were measured and the results can be found in Table 19.

Table 19 – Case Study 1 results

	Metrics (%)	**ANN**	**DT**	**RF**	**KNN**	**SVM**
Total 34 instances	Accuracy	44.12	97.06	97.06	100.00	94.12
	Precision	51.67	100.00	93.75	100.00	90.00
	Recall	80.92	91.67	99.14	100.00	98.28
	F1-score	63.07	95.00	96.37	100.00	93.96
Class 1 29 instances	Precision	100.00	100.00	100.00	100.00	100.00
	Recall	37.93	100.00	96.55	100.00	93.10
	F1-score	55.00	100.00	98.25	100.00	96.43
Class 2 1 instances	Precision	8.33	100.00	100.00	100.00	100.00
	Recall	100.00	100.00	100.00	100.00	100.00
	F1-score	15.38	100.00	100.00	100.00	100.00
Class 3 3 instances	Precision	50.00	100.00	75.00	100.00	60.00
	Recall	66.67	66.67	100.00	100.00	100.00
	F1-score	57.14	80.00	85.71	100.00	75.00
Class 5 1 instances	Precision	100.00	100.00	100.00	100.00	100.00
	Recall	100.00	100.00	100.00	100.00	100.00
	F1-score	100.00	100.00	100.00	100.00	100.00

Figure 24 illustrates the maps with forecast results in the context of 5 categories (f1-score of 100.00% for KNN), showing the countries most critical of the prevalence of anemia in children (larger circles in red) and also countries with lower rates (smaller circles in green), all from the Americas region. In the first context, there were only categories 1, 2, 3 and 5 for the Americas region, although there are categories from 1 to 5 worldwide.

In the post-processing step, the LLM integrated with HEAL provided some important insights into the entire AutoML process, each log presented at each step, and the final results.

Based on these insights, we describe below some observations and knowledge findings presented by LLM. It is important to highlight that, as this is an LLM tool, the information needs to be validated by an expert in the field and human analysis is essential. Finally, it is important to highlight that below only some information provided by

Figure 24 – Case study 1 - Example of visualization of the results found shown on a map to aid decision making

Source: Elaborated by the Author.

LLM will be presented, the full report can be found in Appendix B.

Initial analysis - Initially, LLM generated a report compiling all the results found, which was summarized as follows: the DT and RF algorithms were defined as the most robust, with high precision and recall, with RF being the best with an f1-score of 96.37%. According to LLM, although KNN presents a perfect f1-score (100.00%), validation is necessary to avoid overfitting. SVM had inconsistent results in some classes, and ANN had the worst performance. It is recommended to prioritize DT and RF, in addition to tuning hyperparameters and validating KNN on new data.

Outlier Analysis - LLM also performed an analysis of the outliers identified in the database, generating some interesting insights. Extreme cases such as Sao Tome and Principe and Niue stand out, reflecting regional disparities linked to socioeconomic factors such as poverty, access to health and nutritional education. Relevant correlations were found between high prevalence of anemia, infant mortality and maternal health. Con-

texts such as food security and chronic diseases also have an influence, even in developed countries such as the Cook Islands. Targeted interventions in nutrition, education and health are recommended to mitigate these disparities.

Correlation matrix analysis - By analyzing the correlation matrix, LLM was able to uncover some interesting insights. Interestingly, while many attributes related to mortality have high correlations, the prevalence of insufficient physical activity in adults shows only a weak correlation (about 0.08) with childhood anaemia prevalence. This indicates that factors directly linked to health outcomes, such as nutrition and healthcare access, may be much more significant than lifestyle factors like physical activity when it comes to childhood anaemia.

Detailed analysis of the results - By analyzing all the results found in detail, LLM was able to identify some insights, such as: the algorithms showed similar tendencies in incorrectly classifying specific countries, especially in cases where the actual values were significantly lower than the predicted values. For example, Peru and Colombia were predicted inaccurately across multiple algorithms, with predictions biased toward a higher prevalence category, and Argentina, Belize, and Jamaica also showed inconsistencies, with predictions often exceeding actual anemia prevalence levels.

When examining these errors, one can observe a correlation with socioeconomic factors prevalent in the Americas region during this period. Many countries that are misclassified are those facing political instability, economic challenges or natural disasters, which affect health systems and data availability. For instance, the crisis in Venezuela may have affected the availability and accuracy of health-related data, leading to erroneous predictions.

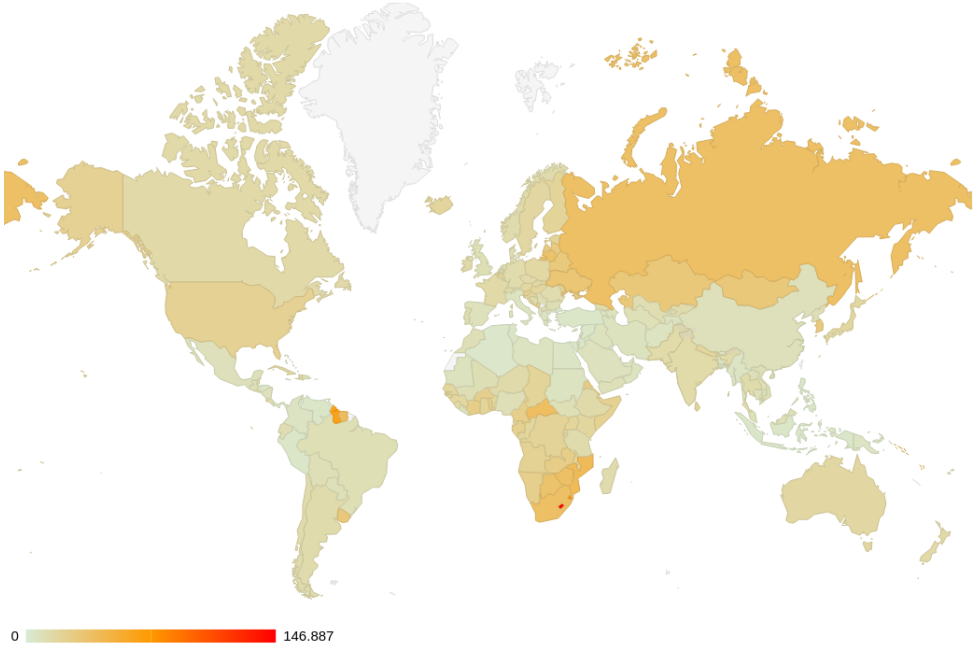
Interestingly, the predictive models performed exceptionally well in estimating anemia prevalence in high-risk countries such as Dominica and Grenada, which have more robust health data. This can be attributed to: more robust health reporting systems in these countries; and community-wide health initiatives and policies aimed at combating malnutrition and anemia.

Further analysis revealed patterns in countries that achieved high prediction accuracy: countries such as Trinidad and Tobago and the United States demonstrated effective public health campaigns, minimizing anemia prevalence and improving prediction accuracy; and countries with lower socioeconomic status but with recent investments in health, such as Guatemala, reflected surprising accuracy in forecasts, suggesting positive effects of recent health initiatives.

6.6.2 Case Study 2 - Age-standardized suicide rates in men (per 100,000 population) in the countries of Africa in 2019

For our second case study, we studied the attribute of age-standardized suicide rates in men (per 100,000 population). Figure 25 illustrates the world map in which the color of each country reflects this attribute in 2019. In this figure, the closer the region is to red, the greater the age-standardized suicide rates in men (maximum value equal to 146.887). Likewise, colors close to green indicate lower rates (minimum value equal to 0), and, finally, countries in white color do not have data registered in WHO.

Figure 25 – World map depicting the age-standardized suicide rates in men (per 100,000 population) in the year 2019



Source: Elaborated by the Author.

This study aims to run the prediction system configured to predict the attribute age-standardized suicide rates in men (per 100,000 population) in the countries of the continent of Africa in 2019. Based on WHO data, for the year 2019, there are 45 countries in the Africa region. The system needed 9,904.33 seconds (about 2.75 hours) to run in this case study. With the generation of system execution outputs (present in the log generated by the system), it is also possible to define the main attributes directly related to the prediction attribute, valuable information for decision-making that seeks to improve the related indices.

For this study, the five algorithms proposed in HEAL were executed for the predictions. In all experiments, data related to the Africa region in 2019 was used for testing (45, representing 1.32%), while all other data (3,344, representing 98.67%) were used for

model training. The attribute age-standardized suicide rates in men (per 100,000 population) were categorized into five groups. For each algorithm, accuracy, precision, recall and f1-score were measured and the results can be found in Table 20.

Table 20 – Case Study 2 results

	Metrics (%)	**ANN**	**DT**	**RF**	**KNN**	**SVM**
Total 45 instances	Accuracy	75.56	84.44	91.11	97.78	91.11
	Precision	78.85	61.82	83.33	91.67	83.33
	Recall	68.75	60.00	97.50	99.38	97.50
	F1-score	73.45	60.90	89.86	95.37	89.86
Class 1 40 instances	Precision	100.00	97.30	100.00	100.00	100.00
	Recall	75.00	90.00	90.00	97.50	90.00
	F1-score	85.71	93.51	94.74	98.73	94.74
Class 2 2 instances	Precision	15.38	0.00	33.33	66.67	33.33
	Recall	100.00	0.00	100.00	100.00	100.00
	F1-score	26.67	0.00	50.00	80.00	50.00
Class 4 1 instances	Precision	100.00	100.00	100.00	100.00	100.00
	Recall	0.00	100.00	100.00	100.00	100.00
	F1-score	0.00	100.00	100.00	100.00	100.00
Class 5 2 instances	Precision	100.00	50.00	100.00	100.00	100.00
	Recall	100.00	50.00	100.00	100.00	100.00
	F1-score	100.00	50.00	100.00	100.00	100.00

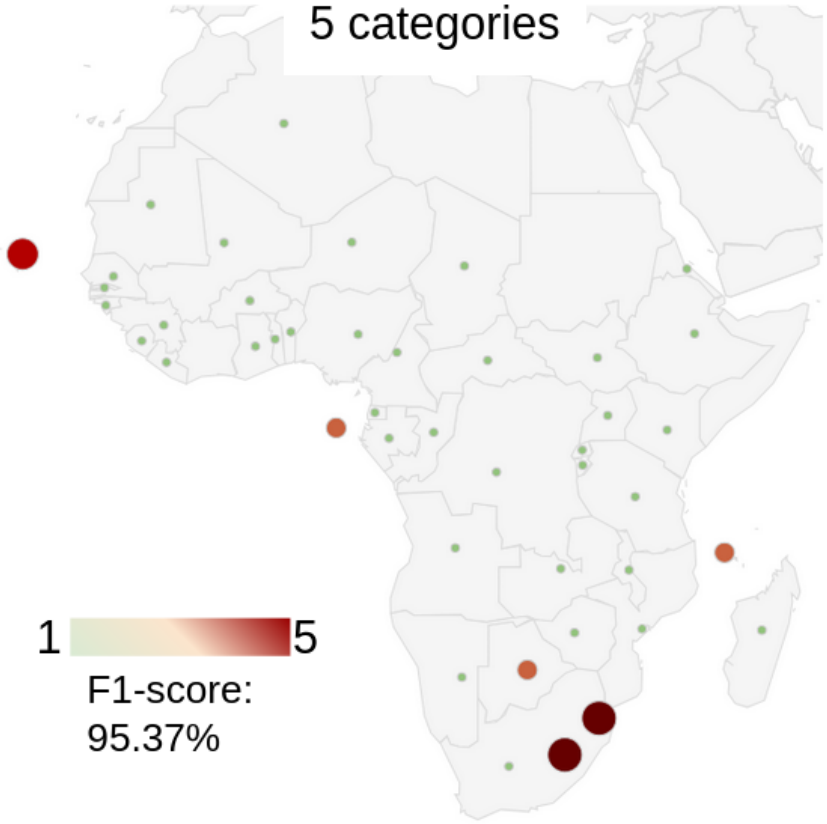
Figure 26 illustrates the maps with predicted outcomes in the context of 5 categories (95.37% accuracy by KNN), showing the most critical countries regarding age-standardized suicide rates in men (larger red circles) and also countries with lower rates (smaller green circles). This is an example of a visualization that can be used to help a HEAL user make decisions.

Finally, in the post-processing step, LLM presented important points of analysis on the results found by AutoML. The main information will be presented below and the detailed report can be found in Appendix B. It is important to highlight that, as this is an LLM tool, the information needs to be validated by an expert in the field and human analysis is essential.

Initial analysis - Initially, LLM presented a compiled report of the results found by HEAL: KNN stood out as the most accurate (97.78% accuracy), followed by RF and SVM (> 91%). Classes 4 and 5 were well detected, with perfect f1-scores, while class 2 showed significant problems, especially in DT (zero recall), suggesting imbalances or flaws in the data. It is recommended to reevaluate the characteristics related to Class 2, apply balancing techniques and explore ensemble methods to improve the results.

Correlation matrix analysis - LLM also processed the correlation matrix to gain insights including: the attribute prevalence of obesity among adults displays a significant

Figure 26 – Case study 2 - Example of visualization of the results found shown on a map to aid decision making



Source: Elaborated by the Author.

negative correlation (-0.845) with age-standardized suicide rates. This could imply that areas with higher obesity rates may experience lower suicide rates, possibly reflecting socioeconomic support or health resources available to populations with higher obesity.

Another observation raised by LLM is that the number of maternal deaths and maternal mortality ratio were both found to be negatively correlated with suicide rates (-0.740 and -0.765, respectively). This observation may suggest that higher maternal mortality rates might correlate with increased community distress and wider societal issues, potentially leading to higher suicide rates.

As a third observation from the correlation matrix, LLM noted that: a strong positive correlation (0.998) was observed between adolescent mortality rate and number of deaths attributed to non-communicable diseases. While these may not directly correlate with suicide rates, they indicate underlying issues with health standards and the well-being of younger populations, which could influence suicide rates indirectly.

Importance of features - LLM also sought important insights by analyzing the importance of features for forecasting. Thus, she observed that: the prevalence of obesity

and overweight emerged as relevant features in most algorithms, however, their impact varied, indicating that they may influence suicide rates differently based on underlying health conditions related to mental health.

Although several features related to violence and homicide rates were present, they showed minimal importance in the KNN and SVM models, possibly suggesting that indirect indicators are more predictive of suicide rates than direct measures of violence. Notably, maternal mortality and adolescent mortality rates also appeared to be highly significant, suggesting the long-term effects of maternal health on youth mental health outcomes.

Detailed analysis of the results - Finally, the LLM evaluated all the results in detail and, from there, considered some issues, such as: Lesotho, for example, was predicted as class 2 by Decision Tree and class 5 by Random Forest, indicating a deep inconsistency between the algorithms. This suggests that the representation of Lesotho characteristics may not be robust; potentially an indication of missing socioeconomic or health data, which may significantly impact mental health and consequently suicide rates.

Regarding the context of the region in 2019, it is essential to consider factors such as political stability, access to healthcare, economic conditions and social programs aimed at mental health. The Southern African region, particularly affected by high rates of poverty and unemployment, coupled with limited access to mental health resources, may contribute to inconsistent predictions. Countries like Lesotho, with unique socioeconomic challenges, may require customized models that take these nuances into account.

The fact that certain countries, such as Sao Tome and Principe, were classified incorrectly raises questions about the availability and quality of the data. The quality of features in rural and urban regions in Africa can significantly affect the results, thus emphasizing the need for localized data collection, as mass data sets can obscure critical details.

6.6.3 Case Study 3 - Influence of Tuberculosis and Hypertension in Prevalence of Anemia in Children in India in 2021

The Global Burden of Disease (GBD) (GBD, 2024) data repository led by the Institute for Health Metrics and Evaluation (IHME)* is the source of the data used in this case study. This repository creates a unique platform to compare the magnitude of diseases, injuries, and risk factors across age groups, sexes, countries, regions, and time.

Therefore, the GBD approach provides an opportunity to compare their counstries' health progress to that of other countries, and to understand the leading causes of health

*https://www.healthdata.org/research-analysis/gbd. Acessed July 24, 2024

loss that could potentially be avoided. This repository also uses the panel data structure for much of its data, so it is possible to use it in our system as long as it is imported by the user.

We use the data collected related to the country India. Thus, these data are mainly divided into spatial dimension (30 states in India) and temporal dimension (years - 1990 to 2021). Other divisions used are sex (male and female) and age (ranging from newborn to elderly). The dataset has the following column structure:

a) *LOCATION* - indicator location (state of India).

b) *YEAR* - indicator year.

c) *AGE* - dimension - age.

d) *SEX* - dimension - sex.

e) *CAUSE OF DEATH OR INJURY* - indicator name.

f) *MEASURE* - how data is displayed.

g) *VALUE* - indicator value.

In order to select and organize the data used in the knowledge discovery process, the first step is responsible for extracting data from the Global Burden of Disease (GBD) (data collection). This extraction was done using .csv files downloaded from GBD India Compare Data Visualization for all attributes related to: tuberculosis, hypertension and anemia using *SEX* and *AGE* dimensions.

Table 21 shows the attributes that were collected and used as input for the experiments performed, the id and description of each one, according to (GBD, 2024). All attributes were collected according to prevalence (%) and for each defined attribute the dimensions *SEX* (Male and Female) and *AGE* (less than 5 years, between 5 and 14, between 15 and 49, between 50 and 69 and greater than 70 years) were used.

Thus, according to the dimensions used, each attribute was subdivided into two dimensions for *SEX* and five dimensions for *AGE*, with the exception of attribute B.2.4 Hypertensive heart disease, which had no values for *AGE* - less than 5 years and *AGE* between 5 and 14 years. Given these steps, the final dataset has a total of 76 input attributes.

Regarding the outputs defined for the study carried out, we used the indicator present in the GBD: A.7.4 Dietary iron deficiency applied to children under 5 years old and without the division into male and female (percent of total prevalent cases). Therefore, this indicator used as output is presented in Table 22.

Table 21 – Attributes used as Input

ID	Attribute name	Description
A.2.1	Tuberculosis	Tuberculosis is an infectious disease caused by Mycobacterium tuberculosis, including pulmonary and extrapulmonary TB, which are bacteriologically confirmed or clinically diagnosed.
A.2.1.1	Latent Tuberculosis Infection	Latent tuberculosis infection is defined as an infection with Mycobacterium tuberculosis without any symptoms or signs of active tuberculosis disease.
A.2.1.2	Drug-susceptible tuberculosis	Drug-susceptible tuberculosis is distinguished as being responsive to isoniazid and rifampicin treatment.
A.2.1.3	Multidrug-resistant tuberculosis without extensive drug resistance	Multidrug-resistant tuberculosis without extensive drug resistance is a form of tuberculosis that does not respond to the two most effective first-line antituberculosis drugs, but is not resistant to any fluoroquinolone and any second-line injectable drugs.
A.2.1.4	Extensively drug-resistant tuberculosis	Extensively drug-resistant tuberculosis is a form of tuberculosis that is not responsive to isoniazid and rifampicin, plus any fluoroquinolone and any second-line injectable drugs.
A.2.2	Lower respiratory infections	This cause incorporates death and disability resulting from lower respiratory infections, including clinician-diagnosed and self-reported cases of pneumonia, bronchiolitis, respiratory syncytial virus, and influenza-like illness.
A.2.3	Upper respiratory infections	Upper respiratory infections include cough, acute nasopharyngitis, sinusitis, pharyngitis, tonsillitis, laryngitis, tracheitis, epiglottitis, rhinitis, rhinosinusitis, rhinopharyngitis, and supraglottitis.
B.2.4	Hypertensive heart disease	Hypertensive heart disease is structural heart disease defined by its underlying cause (systemic hypertension), resulting in left ventricular hypertrophy, diastolic dysfunction, and clinical heart failure with either preserved or reduced systolic function.

Table 22 – Attribute used as Output

ID	Attribute name	Description
A.7.4	Percent of total prevalent cases of dietary iron deficiency in children (less than 5 years old)	Sickle cell disorders, thalassaemias, glucose-6-phosphate dehydrogenase deficiency, sickle cell trait, thalassaemia trait, hemizygous G6PD deficiency, haemoglobinopathies and acquired haemolytic anaemias, red cell aplasia, and aplastic anaemias.

Table 23 – Discretization used for Anemia (WORLD HEALTH ORGANIZATION, 2011)

Category of public health significance	Prevalence of anaemia (%)
Normal	4.9 or lower
Mild	5.0 – 19.9
Moderate	20.0 – 39.9
Severe	40 or higher

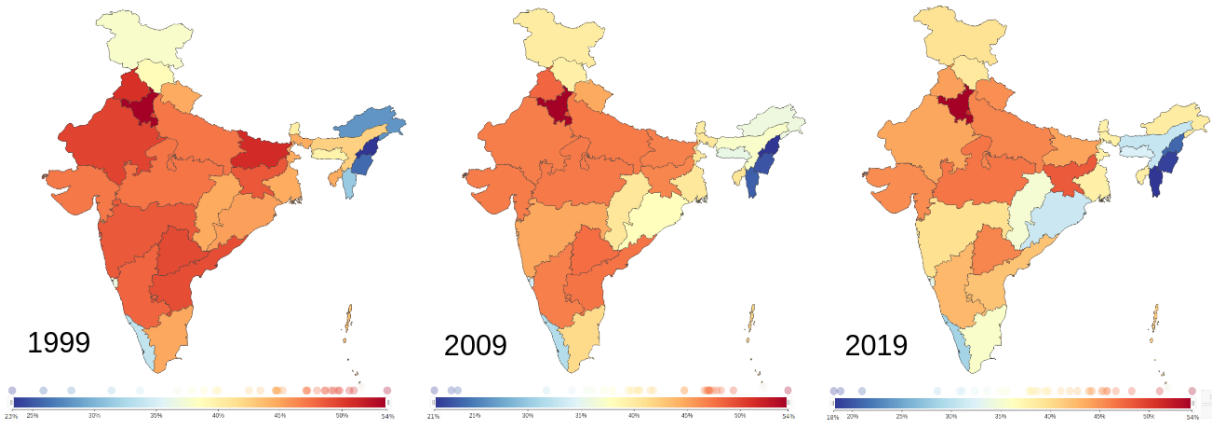
Our study contemplates multi-class problems. Therefore, it's necessary to categorize the output data, dividing the results into different categories (class). Table 23 exemplifies the categorization performed, dividing in 4 possibles categories: normal, mild, moderate and severe. HEAL allows for the configuration of discretization, setting it as needed according to the context.

With the aim of studying the correlation between hypertension and tuberculosis in anemia in children, we carried out different experiments applied to prevalent cases of iron deficiency in children's diets. Given the impact that Anemia, its variations and causes have in India, Figure 27 shows the prevalence of cases of dietary iron deficiency in children (less than 5 years) over the years, ranging from 1999, 2009 and 2019.

It is possible to note that despite this disease having had a higher incidence in 1999, the disease is still very prevalent in this country today and, despite 10 years having passed since 2009, the values in 2019 are still very close, and in some states they are even worse. This reinforces the importance of seeking solutions to this problem, given that despite the public policies implemented in the country, the values have not seen a significant reduction over the last 10 years and, to make matters worse, in some states these values have increased, such as Uttarakhand (northwestern), Arunachal Pradesh (northeast) and Jharkhand (eastern).

Applying the discretization presented in Table 23 to the data relating to the states

Figure 27 – Dietary Iron Deficiency in Children in 1999, 2009 and 2019



Source: Elaborated by the Author.

of India in the year 2021, we can find the map presented in Figure 28. It can be seen in this map that all states in India are classified as Moderate (9 states) or Severe (21 states), with no state being in Normal or Mild cases.

It is important to highlight that in an ideal context it would be interesting to predict data more current than 2021. However, given the database used (GBD), the year 2021 is the most current. As it is important to know what the results were in predictions, we used this year. Good results in this experiment indicate that good results will likely be found in future years.

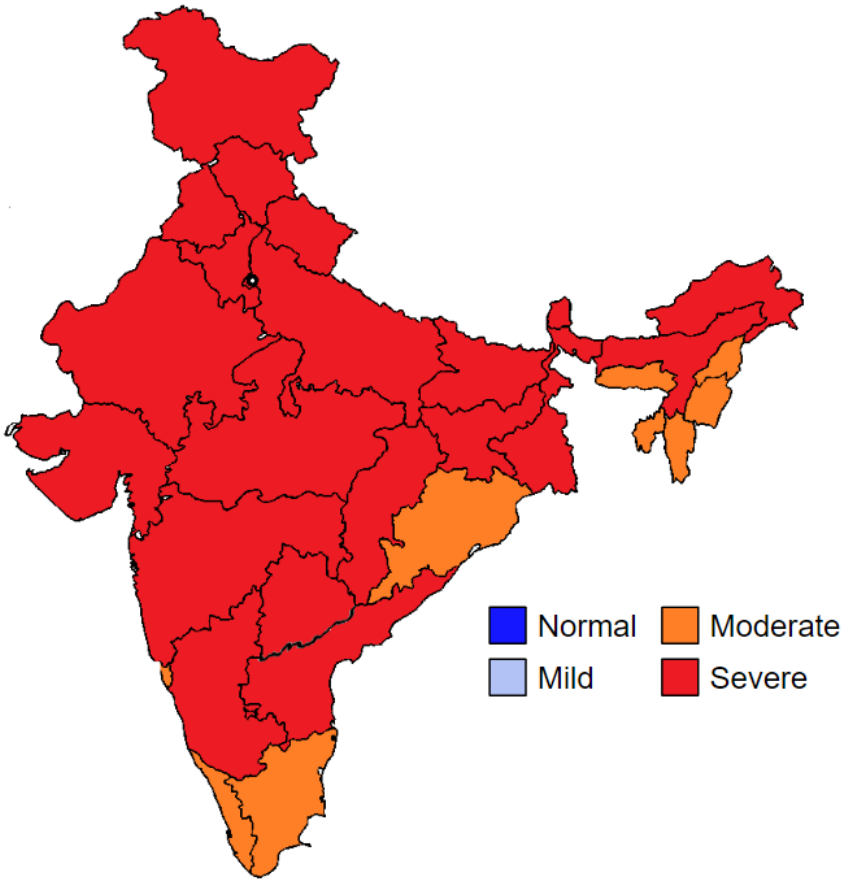
Therefore, data relating to the year 2021 was removed for testing, while the rest of the dataset was used for training and validating the model. After this division, Machine Learning algorithms were used to make predictions. Table 24 shows the results found by each of the algorithms for the total number of instances (30) and for each separate class, with 9 instances for Moderate and 21 for Severe.

Again, the results were satisfactory for all algorithms, with Random Forest together with KNN presenting the best results, with f1-score of 92.65%. For all other algorithms, f1-score always remained above 62.00%. The results of this experiment show that it was possible to select relevant attributes for correlation with dietary iron deficiency, resulting in good predictions of future scenarios. Thus, these can be used to predict future data on prevalent cases of dietary iron deficiency in children.

The application of HEAL, for example, in the generation of graphs to visualize future scenarios and, consequently, assist in decision-making in public policies by the government. Examples of such visualization plots are shown in Figure 29, for the results of predicting prevalent cases of dietary iron deficiency in children in 2021.

These high accuracy and f1-score were found by both the Random Forest (RF) and

Figure 28 – Dietary Iron Deficiency in Children in 2021



Source: Elaborated by the Author.

KNN algorithms. Based on these graphs and metrics, public policies aimed at improving hypertension and tuberculosis rates could be proposed, aiming to consequently reduce the prevalence of anemia in children. A forecast of future scenarios could be generated from the defined methodology.

In the last experiment performed, we interpreted the importance relationship of the features generated by the Random Forest algorithm in one of the HEAL steps. The results of this application can be found in Table 25 for the 20 indicators that present the highest results, in which the importance is presented as a percentage in the Importance column and the features are ordered by importance position.

The feature importance table highlights features with a high degree of importance related to the health of adult men and women. This importance given to health conditions in adults of both genders may indicate an influence of the health patterns of parents or caregivers on the nutritional status of children.

In the last step of HEAL (post-processing), the results were sent together with the script created for the LLM (Chat GPT-4o) technology used, seeking analysis of the

Table 24 – Classification Results - Dietary Iron Deficiency - Case Study 3

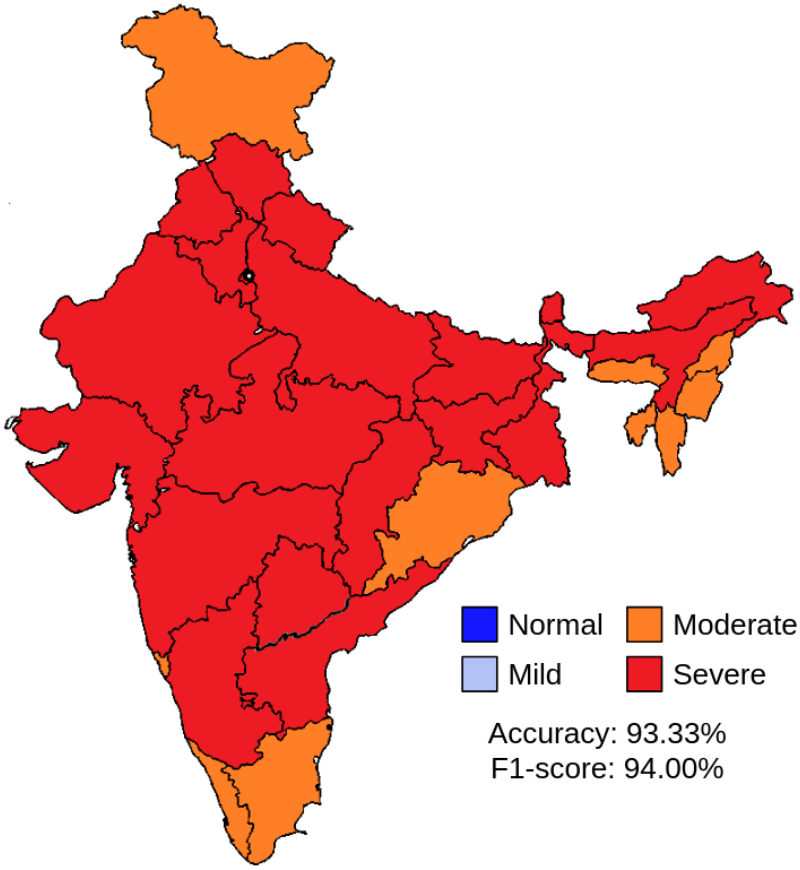
	Metrics (%)	ANN	DT	RF	KNN	SVM
	Accuracy	46.67	73.33	93.33	93.33	76.67
Total	Precision	66.67	69.44	90.00	90.00	72.01
30 instances	Recall	63.64	73.86	95.45	95.45	76.14
	F1-score	65.12	71.59	92.65	92.65	74.02
Moderate	Precision	33.33	50.00	80.00	80.00	54.55
9 instances	Recall	100.00	75.00	100.00	100.00	75.00
	F1-score	50.00	60.00	88.89	88.89	63.16
Severe	Precision	100.00	88.89	100.00	100.00	89.47
21 instances	Recall	27.27	72.73	90.91	90.91	77.27
	F1-score	42.86	80.00	95.24	95.24	82.93

Table 25 – Case Study 3 - Feature Importance in Random Forest

Feature name	Dimension	Importance (%)
Drug-susceptible tuberculosis	Male and 15-49 years	5.69
Upper respiratory infections	Male and 0-5 years	4.55
Lower respiratory infections	Female and 70+ years	4.26
Hypertensive heart disease	Female and 70+ years	3.90
Latent tuberculosis infection	Female and 15-49 years	3.75
Drug-susceptible tuberculosis	Male and 50-69 years	3.70
Lower respiratory infections	Male and 70+ years	3.65
Drug-susceptible tuberculosis	Female and 70+ years	3.63
Latent tuberculosis infection	Male and 50-69 years	3.61
Tuberculosis	Male and 50-69 years	3.48
Tuberculosis	Female and 15-49 years	3.35
Latent tuberculosis infection	Female and 70+ years	3.34
Latent tuberculosis infection	Female and 50-69 years	3.33
Tuberculosis	Male and 15-49 years	3.30
Tuberculosis	Female and 50-69 years	3.13
Latent tuberculosis infection	Female and 5-14 years	3.10
Tuberculosis	Female and 5-14 years	3.06
Latent tuberculosis infection	Male and 70+ years	3.06
Tuberculosis	Male and 70+ years	2.99
Latent tuberculosis infection	Female and 0-5 years	2.88

results and non-obvious knowledge discoveries. Finally, LLM presented important points of analysis on the results found by AutoML. The main information will be presented below

Figure 29 – Example of Information Visualization - Prevalent of Dietary Iron Deficiency in Children in 2021



Source: Elaborated by the Author.

and the detailed report can be found in Appendix B.

Initial analysis - Initially, LLM generates a report compiling all the results found, the summary of which is: RF and KNN performed best, both with 93.33% accuracy and robust precision and recall metrics, especially in the Severe Class (perfect 100.00% accuracy). In contrast, the Moderate Class showed higher variability, with ANN achieving only 50.00% f1-score, suggesting resource limitations or class imbalance. DT and SVM showed moderate performances (71.59% and 74.02% f1-score, respectively), but also faced difficulties with the Moderate Class. It is recommended to prioritize RF and KNN for clinical applications and investigate improvements for the Moderate Class through feature engineering and balancing techniques, aiming at more effective interventions to combat iron deficiency.

Correlation Analysis of features - When analyzing the correlation matrix, LLM generated some interesting insights. Several tuberculosis-related variables, such as "Tuberculosis | Male | <5 years" and "Extensively drug-resistant tuberculosis | Female | <5 years" exhibit strong positive correlations with the target variable. This suggests

a critical link between iron deficiency and tuberculosis prevalence among children under five. Such insights underline the importance of addressing nutritional support as part of the therapeutic approach for tuberculosis in young patients.

Importance of features - By analyzing the characteristics, LLM was able to identify an indirect relationship between anemia in children and the factors hypertension and tuberculosis, since parents or caregivers who have illnesses end up not being able to or having difficulties in acquiring resources and feeding their children. Consequently, this contributes to the development of anemia in children.

It is possible to observe that the majority of attributes classified with a high degree of importance are related to the young, adult or elderly population (Dimension equal to 15-49 years, 50-69 years or 70+ years). This may suggest that public health policies aimed at improving child nutrition should also address health conditions prevalent in young, adult and elderly populations, indicating the need for a more holistic approach.

Detailed analysis of the results - Considering that the only state where all the algorithms made mistakes in their predictions was Jammu & Kashmir and Ladakh, it is possible to reflect on some possible explanations for this. This prediction error in this state, where all algorithms predicted Moderate class while the actual value was Severe, could have a few different explanations. However, among these, it is worth highlighting that this state has specific characteristics that differentiate it from other states in India, mainly in terms of the geopolitical, social and economic context.

Jammu & Kashmir and Ladakh present conflict, political uncertainty, and barriers to access to resources that may have negatively impacted public health in ways that available data have not adequately captured (RAGHAVAN, 2012). These factors may have led to greater severity of nutritional deficiencies, which was not predicted by the models. Therefore, the algorithms predicted the values for tuberculosis and hypertension, but these other issues also increased the final values for iron deficiency.

In addition to these problems, the state has extreme climatic and geographic conditions, such as high altitudes and harsh winters, which can affect the availability of iron-rich foods and access to health care, also contributing to increased iron deficiency (RAINA, 2002). As all these conditions were not adequately represented in the data used, since the objective of the study was to use tuberculosis and hypertension data, the Machine Learning models used may have underestimated the severity of iron deficiency in this region, indicating as Moderate the real value that is Severe.

It is also important to highlight that the results found suggest that interventions to combat anemia in children should be integrated into programs to control diseases such as tuberculosis and hypertension.

Finally, it is worth noting that a study to explore other comorbidities or social factors that were not considered in this study may be useful, to verify whether they also play an important role in anemia in children. The results highlighted in this case study help guide actions and policies that aim to improve child nutrition, especially in contexts where multiple chronic health conditions are prevalent.

7 CONCLUSIONS

The healthcare sector generates a massive amount of data daily that can be explored to find hidden patterns and knowledge to assist in decision-making in diagnosing, treating, and preventing diseases, prognostic interventions, new treatments, and therapies. From this scenario, our work aimed to develop a AutoML pipeline (named HEAL, which means **He**althcare Data - **A**utomated Machine **L**earning) capable of collecting, pre-processing, select model, optimize hyperparameters, forecasting and analyzing data in the health area in the panel data structure. The development of this system explored Data Analysis, Machine Learning and LLM techniques.

Our adopted methodology is divided into five main steps: early stage, responsible for the state of the art and study of important concepts in the context of Machine Learning, AutoML, Large Language Models (LLM), AutoML frameworks and techniques; HEAL development; individual experiments; experiments and case studies; and evaluations and comparison.

To develop HEAL, we use data analysis and Machine Learning techniques, such as: data collection via API, web scraping, pipeline of parameterizable pre-processing algorithms and pipeline of parameterizable Machine Learning algorithms and Combined Algorithm Selection and Hyperparameter Optimization (CASH). HEAL is also enhanced by applying LLM techniques to user interface, context interpretation, answering questions about context, validation, analysis, and interpretation of results to support knowledge discovery. Therefore, the development of an AutoML pipeline supported by data analysis and Machine Learning techniques and enhanced by an LLM allowed the development of a solution that covers all steps of the AutoML process: 1 - Domain Understanding and Data Collection; 2 - Pre-processing; 3 - Data Mining; 4 - Post-processing (**hypothesis 1, page 15**).

In some steps we use ChatGPT-4o as an LLM tool, assisting in the automation process, such as: user interface, interpretation of context, question answering, validation and analysis of result and knowledge discovery. Therefore, integrating LLM into our AutoML pipeline has enabled us to implement steps that were relatively unexplored: domain understanding and post-processing (**hypothesis 2, page 15**).

Currently, our AutoML system (named HEAL) has five classification algorithms: Artificial Neural Networks (ANN), Decision Tree (DT), Random Forest (RF), K-nearest Neighbors (KNN) and Support Vector Machine (SVM). In the initial experiments carried

out, we applied HEAL in four different areas of health: sexual health, nutritional health, safe sanitation and mental health. In these contexts and in the best scenario, HEAL achieved precision of up 97.63% for RF, recall of up 90.28% and f1-score of up 93.81%. After these experiments, we performed the evaluation and comparison of the prediction results with two different AutoML frameworks found in the literature (H2O and TPOT), showing that HEAL obtained superior results in 3 of the 4 experiments.

Finally, we carried out three case studies to expand the studies and present the potential of the system in different contexts: prevalence of anemia in children in the countries of the Americas in 2019; age-standardized suicide rates in men (per 100,000 population) in the countries of Africa in 2019; and influence of tuberculosis and hypertension in prevalence of anemia in children in India in 2021 (using GBD data source).

With the aim of developing a transparent solution, mainly for healthcare professionals, HEAL provides logs of all the steps carried out during processing, as well as all the results found in each of them. Finally, in its last step, HEAL offers a report on the entire process carried out, seeking to raise important points for the discovery of knowledge and assisting the user with guidance on the study carried out. This report is generated by the integration with ChatGPT 4o, which proves that an AutoML pipeline supported by LLM techniques allowed the development of a more transparent solution, which was still a challenge for these solutions and a barrier to the use of these tools by healthcare professionals (**hypothesis 3, page 15**).

The experiments and results validated our proposal in terms of the accuracy of predictions. Therefore, HEAL can be important for future studies in the health area, helping to define the behavior of countries in the fight against diseases. Besides, it can predict what the future scenario will be for different diseases (including in different social groups) and make it possible to accelerate actions that reduce the predicted numbers.

The following section presents and discusses the limitations, generalizations, and challenges of our solution, and proposes explanations on how these can be addressed in future versions.

7.1 Limitations, Generalization, and Challenges

Although the experiments, results, and case study have validated the current version of HEAL, there are some limitations, generalizations, and challenges to discuss. It is important to highlight the importance of the AutoML pipeline (HEAL) for future studies in the health area, helping to define the behavior of countries in combating diseases. HEAL can also be applied in other areas, as long as the data provided is in the panel data structure.

Initially, as it is a system that makes automated decisions regarding the knowledge discovery process, scenarios with unsatisfactory results may present a limitation because manual processing is not applied to the dataset. For instance, experiment 1 (page 83) presented an f1-score in the best model for this context of 61.16%. Thus, in a knowledge discovery scenario in which a domain expert participates in the process, manual work can be performed to improve the dataset and prediction results. In the proposed AutoML pipeline, the main objective is to automate the knowledge discovery steps, making it possible to achieve a greater diversity of prediction contexts without the need for an expert in the domain of each context. For this reason, in some cases, forecast results (accuracy and collected metrics) are not the priority, although they are important.

It is important to highlight that we proposed using five Machine Learning algorithms for predictions, which could be seen as a limitation since there are scenarios in which all of them can present poor results. For all experiments and case studies performed, the Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) algorithms presented satisfactory results. However, although the Artificial Neural Network (ANN) obtains satisfactory results in some scenarios, they are relatively far from the results of other algorithms, finding some scenarios with poor results, such as experiment 3, with an average accuracy of 42.79% and an average f1-score of 27.83%.

Therefore, studying other algorithms for future system versions would be interesting, including deep learning algorithms. Furthermore, HEAL can evolve to solve regression problems in future scenarios, which is a current limitation as it only solves classification problems. This change is simple to make in future scenarios, with small adjustments to the pipeline structure and, consequently, to the system. As for execution time, in some scenarios, the system needed almost 9,949.28 seconds (about 2.76 hours) to execute, which is also a critical point and needs to be studied in future scenarios.

As explained in the background (see Section 2.7), AutoML can be evaluated for its accuracy, efficiency, scalability, and flexibility. Accuracy was evaluated in 4 experiments and 2 case studies, with satisfactory results in all of them. Regarding the flexibility metric, HEAL is implemented with five Machine Learning algorithms and several pre-processing steps, in addition to covering the entire AutoML process. In our system, the user can customize according to their needs and replicate the AutoML process several times and in different scenarios, being a solution that offers flexibility in AutoML to different contexts.

Regarding the generalization and applicability of HEAL, we designed it for new experiments and case studies on different attributes of the WHO database, including the possibility of expanding attributes through the inclusion of new data sources. Therefore, HEAL presents good generalization, flexibility and applicability in the context of the

WHO, as this database has more than 2,000 indicators. It represents many opportunities for predictions and knowledge discovery in health. This number can increase with dimensions, as explained in Subsection 5.1.1 (page 67).

Furthermore, our pipeline was also successful when run from the GBD database, as this data source is also in the panel data structure. Even so, it is important to discuss the generalization and applicability of HEAL in other dataset organized by region (space dimension) and period (time dimension). Therefore, it is important to conduct experiments to explore these other datasets. Given this, another possible advance is the possibility of the system processing data that is not in this structure.

Aiming at solutions to these limitations, generalizations and challenges, the following section presents possibilities for future work.

7.2 Future Work

Considering the satisfactory results found by most algorithms, but which were not repeated in the Artificial Neural Network (ANN) algorithm, a more complete study of this algorithm is suggested. In this way, more experiments could be carried out applying this algorithm and seeking to understand its behavior, since it presented good results in some contexts and unsatisfactory results in others. It is also worth noting that it may be important to look for an ANN solution (different from the feed-forward used in this work) that can identify temporal characteristics in the data, which can significantly improve performance.

As another possibility for future work, we suggest exploring the context of regression problems in HEAL. Our solution is already open to receive new Machine Learning models (mainly because those implemented in our tool have versions for regression problems) and new pre-processing algorithms, in addition to important metrics for regression problems (such as RMSE, MAPE and KGE) already being implemented in our solution. Therefore, inserting this type of problem would not be a difficult task, as long as special attention was paid to the experiments and results.

It is also important to highlight that HEAL was developed ready to receive continuous updates and improvements in the future. Thus, new modules with new algorithms and techniques can be easily incorporated into our solution.

Thus, as future work, we suggest exploring LLM techniques (such as ChatGPT 4o used in this thesis), for other steps of the AutoML pipeline, such as data mining. At this step, we believe that LLM could be used to reduce the number of experiments to find the best model, as it could analyze the dataset and, based on this analysis and its prior knowledge, indicate the best algorithms and their possible hyperparameter combinations.

In the current version, HEAL uses Random Search as a hyperparameter optimization method. In this context, as a suggestion for future work, we believe it is important to explore other methods, such as Bayesian Search. Working differently from Random Search, this method uses information from previous evaluations to model the search space and identify more promising configurations efficiently. Thus, exploring this method can contribute to the evolution of the proposed AutoML pipeline.

More experiments on HEAL in contexts with larger datasets would be highly recommended. This would make it possible to measure important metrics for AutoML solutions that we are unable to measure yet, such as efficiency and scalability. Thus, it would be possible to measure issues of time and computational resources, in contexts with larger datasets, as well as HEAL's ability to work in larger contexts without losing efficiency.

Regarding the scalability context and analyzing processing time issues (with experiments taking almost 3 hours to run), we suggest exploring high-performance computing techniques (such as parallel and distributed computing) in future work, aiming to reduce HEAL processing time and obtain better performance and scalability.

Although we performed comparisons of AutoML H2O and TPOT with HEAL in our experiments, another suggestion for future work is to perform comparisons of HEAL with other AutoML frameworks. Therefore, it is suggested to run other AutoML in the same contexts presented, or in new ones, to have a greater number of comparisons of AutoML tools with HEAL.

Finally, even though the graphical interfaces were developed with good usability in mind for both Machine Learning experts and non-experts, it is important to study their quality and the ease of user interaction with our solution, seeking mainly the transparency of the solution and accessibility to all users.

REFERENCES

ABDULRAHMAN, S. M. et al. Speeding up algorithm selection using average ranking and active testing by introducing runtime. Machine learning, Springer, v. 107, p. 79–108, 2018.

ACHIAM, J. et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.

ALEEM, S. et al. Machine learning algorithms for depression: Diagnosis, insights, and research directions. Electronics, v. 11, n. 7, 2022. ISSN 2079-9292.

ALI, O. et al. A systematic literature review of artificial intelligence in the healthcare sector: Benefits, challenges, methodologies, and functionalities. Journal of Innovation Knowledge, v. 8, n. 1, p. 100333, 2023. ISSN 2444-569X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2444569X2300029X>.

ALI, R.; LEE, S.; CHUNG, T. C. Accurate multi-criteria decision making methodology for recommending machine learning algorithm. Expert Systems with Applications, v. 71, p. 257–278, 2017. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417416306698>.

ALSHAREF, A. et al. Review of ml and automl solutions to forecast time-series data. Archives of Computational Methods in Engineering, Springer, v. 29, n. 7, p. 5297–5311, 2022.

ALTERYX. EvalML 0.83.0 documentation. 2020. Available from: https://evalml.alteryx.com/en/stable/. Accessed: 2024-04-10.

ASGARI, P.; MIRI, M. M.; ASGARI, F. The comparison of selected machine learning techniques and correlation matrix in icu mortality risk prediction. Informatics in Medicine Unlocked, v. 31, p. 100995, 2022. ISSN 2352-9148. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2352914822001381>.

AUSTIN, P. C. et al. Missing data in clinical research: A tutorial on multiple imputation. Canadian Journal of Cardiology, v. 37, n. 9, p. 1322–1331, 2021. ISSN 0828-282X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0828282X20311119>.

BANAN, J. A.; FATAH, C. A.; HAMAAMIN, M. Y. A review of the role and challenges of big data in healthcare informatics and analytics. Computational Intelligence and Neuroscience: CIN, Hindawi Limited, v. 2022, 2022.

BANSAL, M.; GOYAL, A.; CHOUDHARY, A. A comparative analysis of k-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning. Decision Analytics Journal, v. 3, p. 100071, 2022. ISSN 2772-6622. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2772662222000261>.

BARATCHI, M. et al. Automated machine learning: past, present and future. Artificial Intelligence Review, Springer, v. 57, n. 5, p. 1–88, 2024.

BARBUDO, R.; VENTURA, S.; ROMERO, J. R. Eight years of automl: categorisation, review and trends. Knowledge and Information Systems, Springer, v. 65, n. 12, p. 5097–5149, 2023.

BENESTY, J. et al. Pearson correlation coefficient. In: _____. Noise Reduction in Speech Processing. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 1–4. ISBN 978-3-642-00296-0. Disponível em: <https://doi.org/10.1007/978-3-642-00296-0_5>.

BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. Journal of machine learning research, v. 13, n. 2, 2012.

BILALLI, B.; ABELLÓ, A.; ALUJA-BANET, T. On the predictive power of meta-features in openml. International Journal of Applied Mathematics and Computer Science, v. 27, n. 4, p. 697–712, 2017.

BILALLI, B. et al. Presistant: Data pre-processing assistant. In: MENDLING, J.; MOURATIDIS, H. (Ed.). Information Systems in the Big Data Era. Cham: Springer International Publishing, 2018. p. 57–65. ISBN 978-3-319-92901-9.

BIØRN, E. Econometrics of panel data: Methods and applications. [S.l.]: Oxford University Press, 2017.

BISONG, E.; BISONG, E. Google automl: cloud vision. Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners, Springer, p. 581–598, 2019.

BREIMAN, L. Random forests. Machine Learning, v. 45, n. 1, p. 5–32, out. 2001.

CAMPOS, M.; STENGARD, P.; MILENOVA, B. Data-centric automated data mining. In: Fourth International Conference on Machine Learning and Applications (ICMLA'05). [S.l.: s.n.], 2005. p. 8 pp.–.

CARVALHO, A.; MENEZES, A. G.; BONIDIA, R. P. Ciência de Dados - Fundamentos e Aplicações. [S.l.]: LTC, 2024. ISBN 8521638752.

CERQUITELLI, T. et al. Data mining for better healthcare: A path towards automated data analysis? In: 2016 IEEE 32nd International Conference on Data Engineering Workshops (ICDEW). [S.l.: s.n.], 2016. p. 60–63.

CHAWLA, N. V. et al. Smote: synthetic minority over-sampling technique. J. Artif. Int. Res., AI Access Foundation, El Segundo, CA, USA, v. 16, n. 1, p. 321–357, jun. 2002. ISSN 1076-9757.

CHEN, B. et al. Autostacker: a compositional evolutionary learning system. In: Proceedings of the Genetic and Evolutionary Computation Conference. New York, NY, USA: Association for Computing Machinery, 2018. (GECCO '18), p. 402–409. ISBN 9781450356183. Disponível em: <https://doi.org/10.1145/3205455.3205586>.

COHEN-SHAPIRA, N. et al. Autogrd: Model recommendation through graphical dataset representation. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. New York, NY, USA: Association for Computing Machinery, 2019. (CIKM '19), p. 821–830. ISBN 9781450369763. Disponível em: <https://doi.org/10.1145/3357384.3357896>.

CUSTODE, L. L. et al. An investigation on the use of large language models for hyperparameter tuning in evolutionary algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion. New York, NY, USA: Association for Computing Machinery, 2024. (GECCO '24 Companion), p. 1838–1845. ISBN 9798400704956. Disponível em: <https://doi.org/10.1145/3638530.3664163>.

DASH, C. S. K. et al. An outliers detection and elimination framework in classification task of data mining. Decision Analytics Journal, v. 6, p. 100164, 2023. ISSN 2772-6622. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2772662223000048>.

DATASUS. Department of Information Technology - DATASUS - Health Information. 2024. https://datasus.saude.gov.br/informacoes-de-saude-tabnet/. Accessed: 2024-11-01.

DEMIRALP, Ç. et al. Foresight: Recommending visual insights. arXiv preprint arXiv:1707.03877, 2017.

DHS. Demographic and Health Surveys Program. 2024. Available from: https://dhsprogram.com/. Accessed: 2024-08-12.

DIAS, L. V. et al. Imagedataset2vec: An image dataset embedding for algorithm selection. Expert Systems with Applications, v. 180, p. 115053, 2021. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417421004942>.

DIBIA, V.; DEMIRALP, Data2vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks. IEEE Computer Graphics and Applications, v. 39, n. 5, p. 33–46, 2019.

DRORI, I. et al. Alphad3m: Machine learning pipeline synthesis. CoRR, abs/2111.02508, 2021. Disponível em: <https://arxiv.org/abs/2111.02508>.

DUAN, K.; KEERTHI, S.; POO, A. N. Evaluation of simple performance measures for tuning svm hyperparameters. Neurocomputing, v. 51, p. 41–59, 2003. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S092523120200601X>.

EHSAN, H.; SHARAF, M. A.; CHRYSANTHIS, P. K. Muve: Efficient multi-objective view recommendation for visual data exploration. In: 2016 IEEE 32nd International Conference on Data Engineering (ICDE). [S.l.: s.n.], 2016. p. 731–742.

EUROSTAT. The home of high quality statistics and data on Europe. 2024. Available from: https://ec.europa.eu/eurostat. Accessed: 2024-08-12.

FAES, L. et al. Automated deep learning design for medical image classification by health-care professionals with no coding experience: a feasibility study. The Lancet Digital Health, Elsevier, v. 1, n. 5, p. e232–e242, 2019.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. The kdd process for extracting useful knowledge from volumes of data. Commun. ACM, ACM, New York, NY, USA, v. 39, n. 11, p. 27–34, nov. 1996. ISSN 0001-0782.

Fernandes Junior, F. E.; YEN, G. G. Particle swarm optimization of deep neural networks architectures for image classification. Swarm and Evolutionary Computation, v. 49, p. 62–74, 2019. ISSN 2210-6502. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2210650218309246>.

FEURER, M. et al. Practical automated machine learning for the automl challenge 2018. In: International Workshop on Automatic Machine Learning at ICML. [s.n.], 2018. p. 1189–1232. Disponível em: <https://api.semanticscholar.org/CorpusID:51858430>.

FEURER, M. et al. Auto-sklearn 2.0: Hands-free automl via meta-learning. arXiv:2007.04074 [cs.LG], 2020.

FIELDING, B.; LAWRENCE, T.; ZHANG, L. Evolving and ensembling deep cnn architectures for image classification. In: 2019 International Joint Conference on Neural Networks (IJCNN). [S.l.: s.n.], 2019. p. 1–8.

GBD. Global Burden of Disease Study 2021 (GBD 2021). 2024. Available from: https://www.healthdata.org/research-analysis/gbd. Accessed: 2024-09-25.

GIJSBERS, P. et al. An open source automl benchmark. arXiv preprint arXiv:1907.00909, 2019.

GIJSBERS, P.; VANSCHOREN, J. Gama: A general automated machine learning assistant. In: DONG, Y. et al. (Ed.). Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track. Cham: Springer International Publishing, 2021. p. 560–564. ISBN 978-3-030-67670-4.

Google. Gemini. 2024. Available from: https://gemini.google.com/. Accessed: 2024-10-20.

GOVINDARAJAN, P. et al. Classification of stroke disease using machine learning algorithms. Neural Computing and Applications, Springer, v. 32, p. 817–828, 2020.

HAYKIN, S. Neural Networks and Learning Machines. [S.l.]: Prentice Hall, 2009. (Neural networks and learning machines, v. 10). ISBN 9780131471399.

HE, X.; ZHAO, K.; CHU, X. Automl: A survey of the state-of-the-art. Knowledge-Based Systems, v. 212, p. 106622, 2021. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705120307516>.

HOLLMANN, N.; MüLLER, S.; HUTTER, F. Large language models for automated data science: Introducing caafe for context-aware automated feature engineering.

In: OH, A. et al. (Ed.). Advances in Neural Information Processing Systems. Curran Associates, Inc., 2023. v. 36, p. 44753–44775. Disponível em: <https://proceedings.neurips.cc/paper$_f iles/paper/2023/file/8c2df4c35cdbee764ebb9e9d0acd5197-Paper-Conference.pdf$>.

HU, K. et al. Vizml: A machine learning approach to visualization recommendation. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. New York, NY, USA: Association for Computing Machinery, 2019. (CHI '19), p. 1–12. ISBN 9781450359702. Disponível em: <https://doi.org/10.1145/3290605.3300358>.

HU, Y.-J.; HUANG, S.-W. Challenges of automated machine learning on causal impact analytics for policy evaluation. In: 2017 2nd International Conference on Telecommunication and Networks (TEL-NET). [S.l.: s.n.], 2017. p. 1–6.

IBGE. The Brazilian Institute of Geography and Statistics - IBGE Statistics and Downloads Portal. 2024. `https://www.ibge.gov.br/estatisticas/downloads-estatisticas.html`. Accessed: 2024-11-01.

ILO. International Labour Organization. 2024. Available from: https://www.ilo.org/data-and-statistics. Accessed: 2024-08-12.

IWENDI, C. et al. Covid-19 patient health prediction using boosted random forest algorithm. Frontiers in Public Health, v. 8, 2020. ISSN 2296-2565.

JIN, H. et al. Autokeras: An automl library for deep learning. Journal of Machine Learning Research, v. 24, n. 6, p. 1–6, 2023. Disponível em: <http://jmlr.org/papers/v24/20-1355.html>.

JIN, H.; SONG, Q.; HU, X. Auto-keras: An efficient neural architecture search system. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, NY, USA: Association for Computing Machinery, 2019. (KDD '19), p. 1946–1956. ISBN 9781450362016. Disponível em: <https://doi.org/10.1145/3292500.3330648>.

KAMBATLA, K. et al. Trends in big data analytics. Journal of Parallel and Distributed Computing, Elsevier, v. 74, n. 7, p. 2561–2573, 2014.

KANDANAARACHCHI, S. et al. On normalization and algorithm selection for unsupervised outlier detection. Data Mining and Knowledge Discovery, Springer, v. 34, n. 2, p. 309–354, 2020.

KARHADE, D. S. et al. An automated machine learning classifier for early childhood caries. Pediatric dentistry, American Academy of Pediatric Dentistry, v. 43, n. 3, p. 191–197, 2021.

KASNECI, E. et al. Chatgpt for good? on opportunities and challenges of large language models for education. Learning and Individual Differences, v. 103, p. 102274, 2023. ISSN 1041-6080. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1041608023000195>.

KAUSAR, F. et al. Automated machine learning based elderly fall detection classification. Procedia Computer Science, v. 203, p. 16–23, 2022. ISSN 1877-0509. Disponível em: <https://www.sciencedirect.com/science/article/pii/S187705092200610X>. 17th International Conference on Future Networks and Communications / 19th International Conference on Mobile Systems and Pervasive Computing / 12th International Conference on Sustainable Energy Information Technology (FNC/MobiSPC/SEIT 2022), August 9-11, 2022, Niagara Falls, Ontario, Canada.

KEERTHI, S.; SINDHWANI, V.; CHAPELLE, O. An efficient method for gradient-based adaptation of hyperparameters in svm models. In: SCHöLKOPF, B.; PLATT, J.; HOFFMAN, T. (Ed.). Advances in Neural Information Processing Systems. MIT Press, 2006. v. 19. Disponível em: <https://proceedings.neurips.cc/paper$_files/paper/2006/file/cc431fd7ec4437de061c2577a4603995-Paper.pdf$>.

KENT, J. T. Information gain and a general measure of correlation. Biometrika, v. 70, n. 1, p. 163–173, 04 1983. ISSN 0006-3444. Disponível em: <https://doi.org/10.1093/biomet/70.1.163>.

KHAN, M. A. et al. Enhanced abnormal data detection hybrid strategy based on heuristic and stochastic approaches for efficient patients rehabilitation. Future Generation Computer Systems, v. 154, p. 101–122, 2024. ISSN 0167-739X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167739X23004533>.

KOTTHOFF, L. et al. Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka. Journal of Machine Learning Research, v. 18, n. 25, p. 1–5, 2017. Disponível em: <http://jmlr.org/papers/v18/16-261.html>.

KRASKA, T. et al. Mlbase: A distributed machine-learning system. In: Cidr. [S.l.: s.n.], 2013. v. 1, p. 2–1.

LACOSTE, A. et al. Sequential model-based ensemble optimization. CoRR, abs/1402.0796, 2014. Disponível em: <http://arxiv.org/abs/1402.0796>.

LEDELL, E.; POIRIER, S. H2O AutoML: Scalable automatic machine learning. 7th ICML Workshop on Automated Machine Learning (AutoML), July 2020. Disponível em: <https://www.automl.org/wp-content/uploads/2020/07/AutoML$_2020_paper_61.pdf$>.

LI, Y. et al. Evolving deep convolutional neural networks by quantum behaved particle swarm optimization with binary encoding for image classification. Neurocomputing, v. 362, p. 156–165, 2019. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231219309713>.

LIS. Luxembourg Income Study. 2024. Available from: https://www.lisdatacenter.org/. Accessed: 2024-08-12.

LIU, X.; HUANG, J.; ZHANG, C. System framework design of automated data mining technology based on intelligent agents. In: 2011 International Conference on Image Analysis and Signal Processing. [S.l.: s.n.], 2011. p. 653–655.

LLOYD, J. et al. Automatic construction and natural-language description of nonparametric regression models. In: Proceedings of the AAAI Conference on Artificial Intelligence. [S.l.: s.n.], 2014. v. 28, n. 1.

LORENA, A. C.; CARVALHO, A. C. D.; GAMA, J. M. A review on the combination of binary classifiers in multiclass problems. Artificial Intelligence Review, Springer, v. 30, p. 19–37, 2008.

LUO, D. et al. Autom3l: An automated multimodal machine learning framework with large language models. arXiv preprint arXiv:2408.00665, 2024.

LUO, G. Predict-ml: a tool for automating machine learning model building with big clinical data. Health Information Science and Systems, Springer, v. 4, p. 1–16, 2016.

LUO, G. A review of automatic selection methods for machine learning algorithms and hyper-parameter values. Network Modeling Analysis in Health Informatics and Bioinformatics, Springer, v. 5, p. 1–16, 2016.

MA, B. et al. Autonomous deep learning: A genetic dcnn designer for image classification. Neurocomputing, v. 379, p. 152–161, 2020. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231219313797>.

MAHER, M. M. M. Z. A.; SAKR, S. SmartML: A Meta Learning-Based Framework for Automated Selection and Hyperparameter Tuning for Machine Learning Algorithms. In: EDBT: 22nd International Conference on Extending Database Technology. Lisbon, Portugal: [s.n.], 2019. Disponível em: <https://hal.science/hal-02087414>.

MAHULE, R.; VYAS, O. P. Leveraging linked open data information extraction for data mining applications. Turkish Journal of Electrical Engineering and Computer Sciences, v. 24, n. 6, p. 4874–4884, 2016.

MANDHARE, H. C.; IDATE, S. R. A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques. In: 2017 International Conference on Intelligent Computing and Control Systems (ICICCS). [S.l.: s.n.], 2017. p. 931–935.

MANOGARAN, G. et al. Machine learning based big data processing framework for cancer diagnosis using hidden markov model and gm clustering. Wireless personal communications, Springer, v. 102, n. 3, p. 2099–2116, 2018.

MARTíN, A. et al. Evodeep: A new evolutionary approach for automatic deep neural networks parametrisation. Journal of Parallel and Distributed Computing, v. 117, p. 180–191, 2018. ISSN 0743-7315. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0743731517302605>.

Meta. Introducing Llama 3.2. 2024. Available from: https://www.llama.com/. Accessed: 2024-10-20.

MIRANDA, P. B. et al. A hybrid meta-learning architecture for multi-objective optimization of svm parameters. Neurocomputing, v. 143, p. 27–43, 2014. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231214007693>.

MOHAMMADI, S. S.; NGUYEN, Q. D. A user-friendly approach for the diagnosis of diabetic retinopathy using chatgpt and automated machine learning. Ophthalmology Science, v. 4, n. 4, p. 100495, 2024. ISSN 2666-9145. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666914524000319>.

MOHAMMED, A. et al. Time-series cross-validation parallel programming using mpi. In: 2021 International Conference on Data Analytics for Business and Industry (ICDABI). [S.l.: s.n.], 2021. p. 553–556.

MOHR, F. et al. (wip) towards the automated composition of machine learning services. In: 2018 IEEE International Conference on Services Computing (SCC). [S.l.: s.n.], 2018. p. 241–244.

Moradi Dakhel, A. et al. Github copilot ai pair programmer: Asset or liability? Journal of Systems and Software, v. 203, p. 111734, 2023. ISSN 0164-1212. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0164121223001292>.

NESHATPOUR, K.; SASAN, A.; HOMAYOUN, H. Big data analytics on heterogeneous accelerator architectures. Hardware/Software Codesign and System Synthesis (CODES+ ISSS), 2016 International Conference on, p. 1–3, 2016.

NGUYEN, D. A. et al. Improved automated cash optimization with tree parzen estimators for class imbalance problems. In: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA). [S.l.: s.n.], 2021. p. 1–9.

NOONAN, N. Creative mutation: A prescriptive approach to the use of chatgpt and large language models in lawyering. Available at SSRN 4406907, 2023.

OECD. Organisation for Economic Co-operation and Development. 2024. Available from: https://www.oecd.org/en/data.html. Accessed: 2024-08-12.

OLSON, R. S. et al. Evaluation of a Tree-based Pipeline Optimization Tool for Automating Data Science. 2016.

OLSON, R. S. et al. Automating biomedical data science through tree-based pipeline optimization. In: SQUILLERO, G.; BURELLI, P. (Ed.). Applications of Evolutionary Computation. Cham: Springer International Publishing, 2016. p. 123–137. ISBN 978-3-319-31204-0.

OpenAI. Hello GPT-4o. 2024. Available from: https://openai.com/index/hello-gpt-4o/. Accessed: 2024-10-20.

PAL, S. et al. Chatgpt or llm in next-generation drug discovery and development: pharmaceutical and biotechnology companies can make use of the artificial intelligence-based device for a faster way of drug discovery and development. International Journal of Surgery, LWW, v. 109, n. 12, p. 4382–4384, 2023.

PAULETTO, L. et al. Neural architecture search for extreme multi-label text classification. In: YANG, H. et al. (Ed.). Neural Information Processing. Cham: Springer International Publishing, 2020. p. 282–293.

PROBST, P.; WRIGHT, M. N.; BOULESTEIX, A.-L. Hyperparameters and tuning strategies for random forest. Wiley Interdisciplinary Reviews: data mining and knowledge discovery, Wiley Online Library, v. 9, n. 3, p. e1301, 2019.

RAGHAVAN, V. Conflicts in Jammu and Kashmir: Impact on Polity, Society and Economy. [S.l.]: Vij Books India Pvt Ltd, 2012.

RAINA, A. Geography of jammu & kashmir state. Radha Krishan Anand & Co., Pacca Danga, Jammu, v. 9, 2002.

RAJAN, J.; SARAVANAN, V. A framework of an automated data mining system using autonomous intelligent agents. In: 2008 International Conference on Computer Science and Information Technology. [S.l.: s.n.], 2008. p. 700–704.

REIF, M. et al. Automatic classifier selection for non-experts. Pattern Analysis and Applications, Springer, v. 17, p. 83–96, 2014.

ROBERTS, D. R. et al. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography, v. 40, n. 8, p. 913–929, 2017. Disponível em: <https://nsojournals.onlinelibrary.wiley.com/doi/abs/10.1111/ecog.02881>.

SÁ, A. G. C. de; FREITAS, A. A.; PAPPA, G. L. Automated selection and configuration of multi-label classification algorithms with grammar-based genetic programming. In: AUGER, A. et al. (Ed.). Parallel Problem Solving from Nature – PPSN XV. Cham: Springer International Publishing, 2018. p. 308–320. ISBN 978-3-319-99259-4.

SALAZAR, L. H. A. et al. No-show in medical appointments with machine learning techniques: A systematic literature review. Information, v. 13, n. 11, 2022. ISSN 2078-2489.

SALEHIN, I. et al. Automl: A systematic review on automated machine learning with neural architecture search. Journal of Information and Intelligence, v. 2, n. 1, p. 52–81, 2024. ISSN 2949-7159. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2949715923000604>.

SALIAN, P.; KUMAR, A. Focus of health and fitness in life. International Journal of Science Engineering Development Research, v. 7, n. 9, p. 440–442, 2022. Disponível em: <:http://www.ijsdr.org/papers/IJSDR2209073.pdf>.

SARKER, I. H. Ai-based modeling: Techniques, applications and research issues towards automation, intelligent and smart systems. SN Computer Science, Springer, v. 3, n. 2, p. 158, 2022.

SCHMITT, M. Automated machine learning: Ai-driven decision making in business analytics. Intelligent Systems with Applications, v. 18, p. 200188, 2023. ISSN 2667-3053. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2667305323000133>.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. Understanding machine learning: From theory to algorithms. [S.l.]: Cambridge university press, 2014.

SHAMIM, A.; SHAIKH, M. U.; MALIK, S. U. R. Intelligent data mining in autonomous heterogeneous distributed bio databases. In: 2010 Second International Conference on Computer Engineering and Applications. [S.l.: s.n.], 2010. v. 1, p. 6–10.

SHARMA, S. K.; WICKRAMASINGHE, N.; GUPTA, J. N. Knowledge management in healthcare. TeAM YYePG, p. 1, 2005.

SILVA, R. A. D. et al. Automatic recommendation method for classifier ensemble structure using meta-learning. IEEE Access, v. 9, p. 106254–106268, 2021.

SPARKS, E. R. et al. Automating model search for large scale machine learning. In: Proceedings of the Sixth ACM Symposium on Cloud Computing. New York, NY, USA: Association for Computing Machinery, 2015. (SoCC '15), p. 368–380. ISBN 9781450336512. Disponível em: <https://doi.org/10.1145/2806777.2806945>.

SUGANUMA, M.; SHIRAKAWA, S.; NAGAO, T. A genetic programming approach to designing convolutional neural network architectures. In: Proceedings of the Genetic and Evolutionary Computation Conference. New York, NY, USA: Association for Computing Machinery, 2017. (GECCO '17), p. 497–504. ISBN 9781450349208. Disponível em: <https://doi.org/10.1145/3071178.3071229>.

SULANDARI, W. et al. Implementing time series cross validation to evaluate the forecasting model performance. KnE Life Sciences, v. 8, n. 1, p. 229–238, Mar. 2024. Disponível em: <https://knepublishing.com/index.php/KnE-Life/article/view/15584>.

SUN, Y. et al. Automatically designing cnn architectures using the genetic algorithm for image classification. IEEE Transactions on Cybernetics, v. 50, n. 9, p. 3840–3854, 2020.

SWEARINGEN, T. et al. Atm: A distributed, collaborative, scalable system for automated machine learning. In: 2017 IEEE International Conference on Big Data (Big Data). [S.l.: s.n.], 2017. p. 151–162.

THENG, D.; BHOYAR, K. K. Feature selection techniques for machine learning: a survey of more than two decades of research. Knowledge and Information Systems, Springer, v. 66, n. 3, p. 1575–1637, 2024.

TIDDI, I.; D'AQUIN, M.; MOTTA, E. Dedalo: Looking for clusters explanations in a labyrinth of linked data. In: SPRINGER. The Semantic Web: Trends and Challenges: 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014. Proceedings 11. [S.l.], 2014. p. 333–348.

TORNEDE, A. et al. Automl in the age of large language models: Current challenges, future opportunities and risks. arXiv preprint arXiv:2306.08107, 2023.

TRUONG, A. et al. Towards automated machine learning: Evaluation and comparison of automl approaches and tools. In: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI). [S.l.: s.n.], 2019. p. 1471–1479.

TSAI, Y.-D. et al. Automl-gpt: Large language model for automl. arXiv preprint arXiv:2309.01125, 2023.

UHLMANN, E. et al. Hyperparameter optimization of artificial neural networks to improve the positional accuracy of industrial robots. Journal of Machine Engineering, Editorial Institution of Wrocław Board of Scientific Technical Societies ..., v. 21, n. 2, p. 47–59, 2021.

VALLE, A. M. D.; MANTOVANI, R. G.; CERRI, R. A systematic literature review on automl for multi-target learning tasks. Artificial Intelligence Review, Springer, v. 56, n. Suppl 2, p. 2013–2052, 2023.

VAPNIK, V. N. Statistical Learning Theory. [S.l.]: Wiley-Interscience, 1998.

WARING, J.; LINDVALL, C.; UMETON, R. Automated machine learning: Review of the state-of-the-art and opportunities for healthcare. Artificial Intelligence in Medicine, v. 104, p. 101822, 2020. ISSN 0933-3657. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0933365719310437>.

WAZIRALI, R. An improved intrusion detection system based on knn hyperparameter tuning and cross-validation. Arabian Journal for Science and Engineering, Springer, v. 45, n. 12, p. 10859–10873, 2020.

WBOD. World Bank Open Data. 2024. Available from: https://data.worldbank.org/. Accessed: 2024-09-25.

WHO. World Health Organization. 2024. Available from: https://www.who.int/en/. Accessed: 2024-04-28.

WISTUBA, M.; SCHILLING, N.; SCHMIDT-THIEME, L. Automatic frankensteining: Creating complex ensembles autonomously. In: SIAM. Proceedings of the 2017 SIAM International Conference on Data Mining (SDM). 2017. p. 741–749. Disponível em: <https://epubs.siam.org/doi/abs/10.1137/1.9781611974973.83>.

WORLD HEALTH ORGANIZATION. Haemoglobin concentrations for the diagnosis of anaemia and assessment of severity. 2011. Available from: http://apps.who.int/iris/bitstream/10665/85839/3/WHO_NMH_NHD_MNM_11.1_eng.pdf. Disponível em: <http://apps.who.int/iris/bitstream/10665/85839/3/WHO_NMH_NHD_MNM_11.1_eng.pdf>. Accessed: 2024-08-12.

WVS. World Values Survey. 2024. Available from: https://www.worldvaluessurvey.org/WVSContents.jsp. Accessed: 2024-08-12.

YACOUBY, R.; AXMAN, D. Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In: EGER, S. et al. (Ed.). Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems. Online: Association for Computational Linguistics, 2020. p. 79–91. Disponível em: <https://aclanthology.org/2020.eval4nlp-1.9>.

YAKOVLEV, A. et al. Oracle automl: a fast and predictive automl pipeline. Proc. VLDB Endow., VLDB Endowment, v. 13, n. 12, p. 3166–3180, aug 2020. ISSN 2150-8097. Disponível em: <https://doi.org/10.14778/3415478.3415542>.

ZEBARI, R. et al. A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. Journal of Applied Science and Technology Trends, v. 1, n. 1, p. 56 – 70, May 2020. Disponível em: <https://www.jastt.org/index.php/jasttpath/article/view/24>.

ZHANG, S. et al. Automl-gpt: Automatic machine learning with gpt. arXiv preprint arXiv:2305.02499, 2023.

ZHANG, T. et al. Use of automated machine learning for an outbreak risk prediction tool. Informatics in Medicine Unlocked, v. 34, p. 101121, 2022. ISSN 2352-9148. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2352914822002581>.

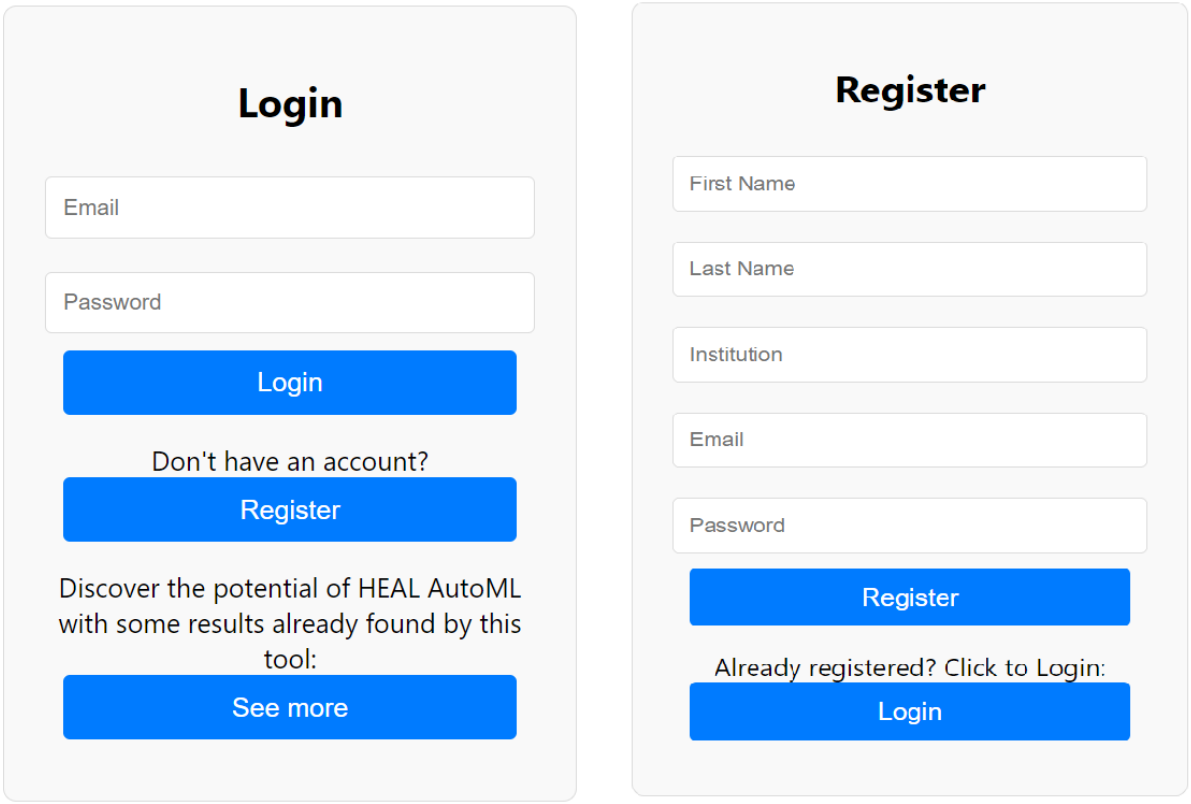
ZHAO, P. et al. T-smote: Temporal-oriented synthetic minority oversampling technique for imbalanced time series classification. In: IJCAI. [S.l.: s.n.], 2022. p. 2406–2412.

ZHAO, W. X. et al. A survey of large language models. arXiv preprint arXiv:2303.18223, 2023.

APPENDIX A – HEAL INTERFACE

This appendix aims to present the screens (Graphical User Interface) developed for the HEAL solution (full title). Therefore, when the user accesses our system, he has access to two main screens: login and registration, as shown in Figure 30.

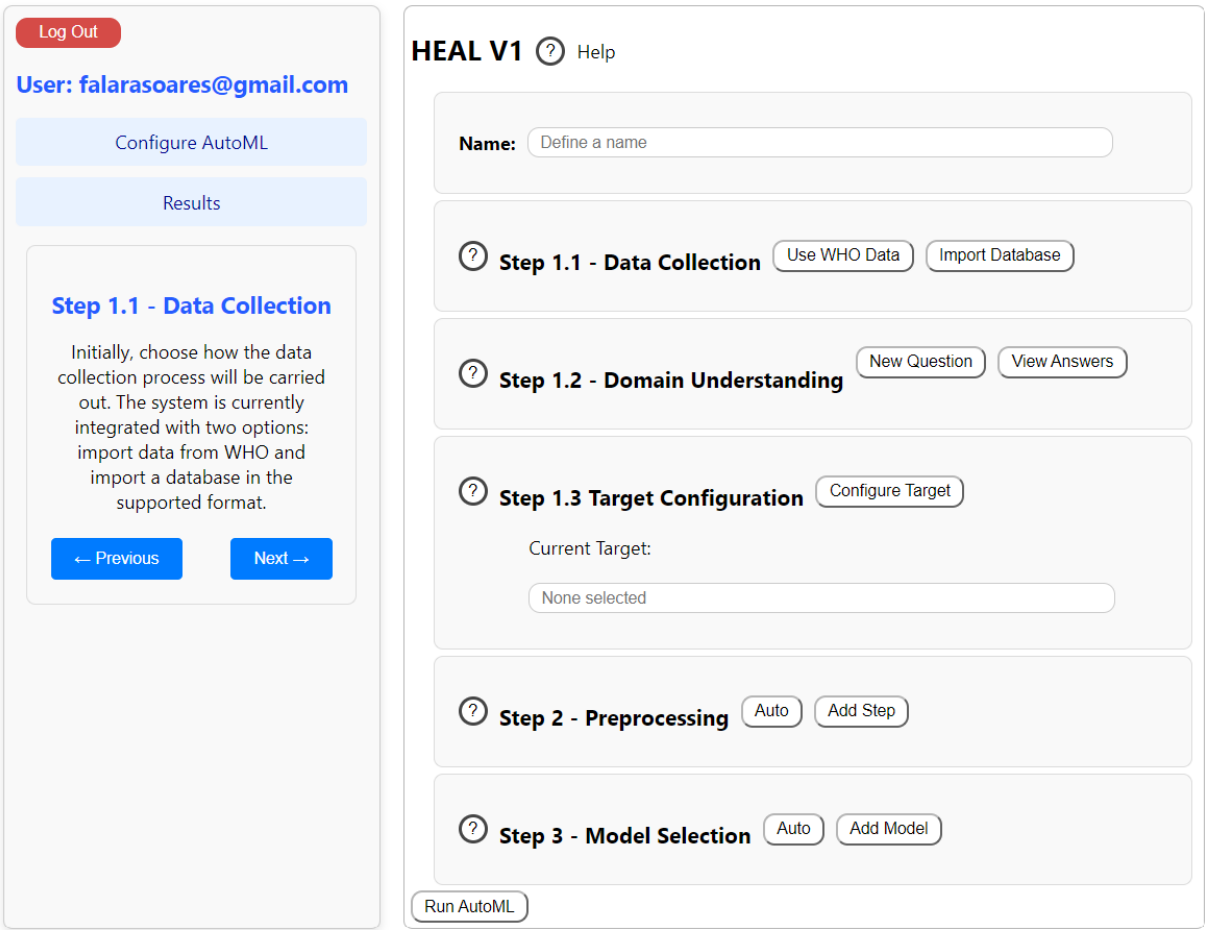
Figure 30 – Screens - Login and Register



Source: Elaborated by the Author.

Upon logging in, the user has access to the homepage, as shown in Figure 31. On this screen, the user can perform all the necessary configurations in HEAL, that is, steps 1.1 Data Collection, 1.2 Domain Understanding, 1.3 Target Configuration, 2 Preprocessing and 3 Model Selection. On the left side, the user has a tutorial explaining each step of HEAL, in addition to being able to navigate to the results screen.

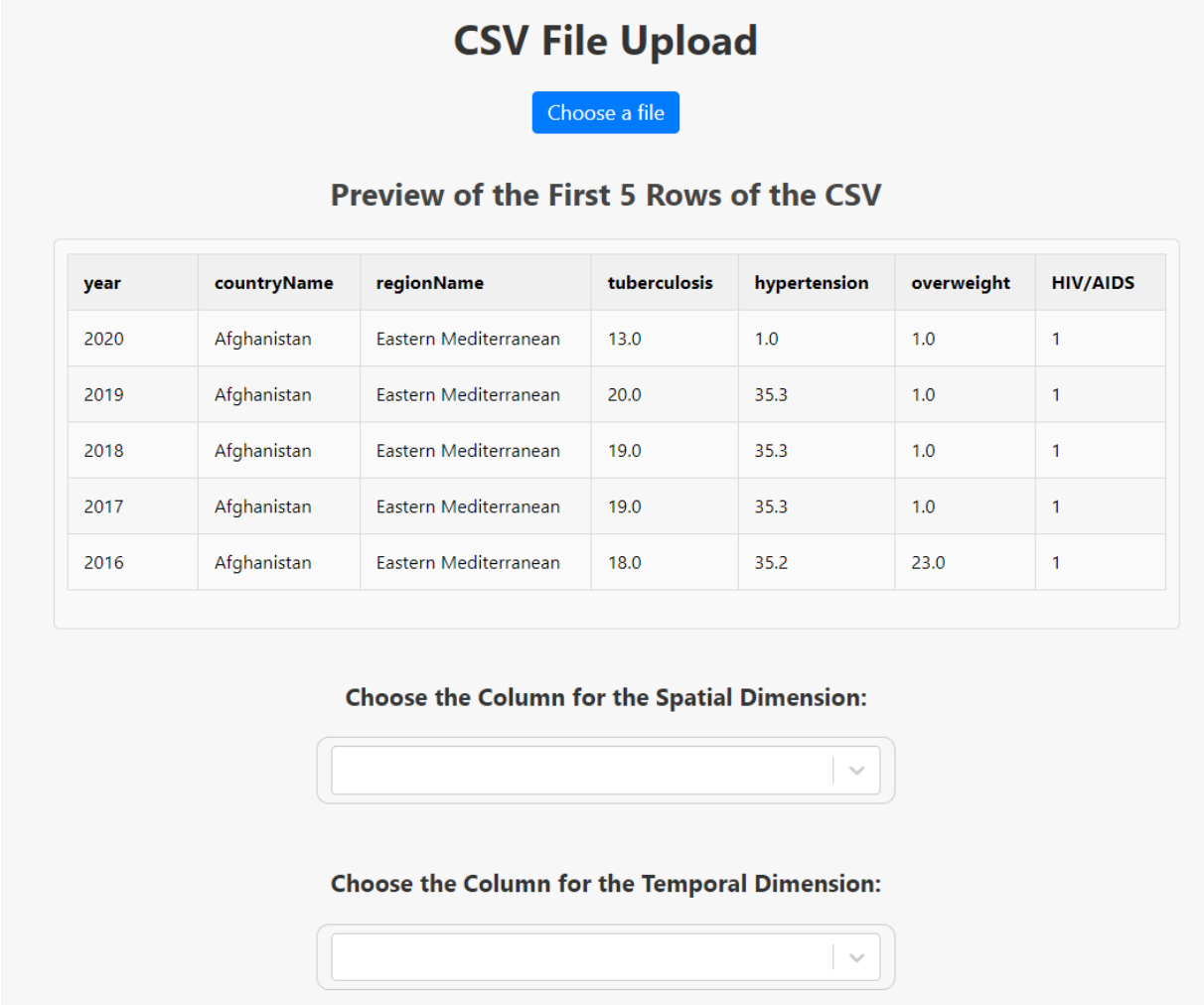
Figure 31 – Screen - Homepage



Source: Elaborated by the Author.

In step 1.1 Data Collection, the user can choose two database options: use data from the World Health Organization (Use WHO Data) or import their own database (Import Database). By clicking on the second option, the user is redirected to a screen (as illustrated in Figure 32), where they can import their database in a data panel structure, viewing the first 5 lines of the same for confirmation and selecting the temporal and spatial dimensions.

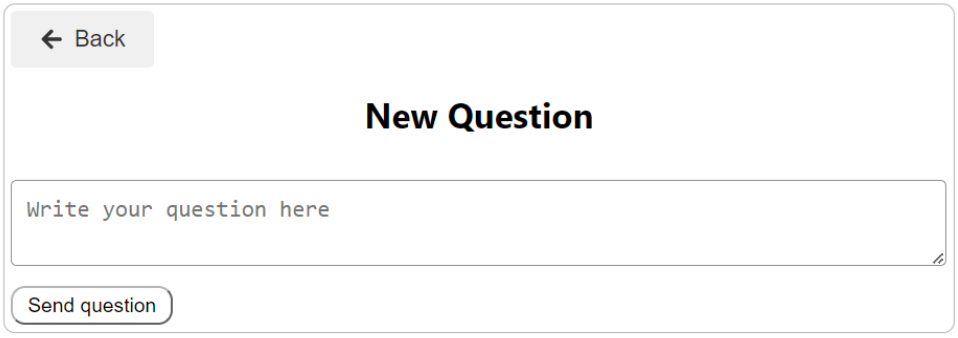
Figure 32 – Screen - File Upload



Source: Elaborated by the Author.

When clicking on the New Question option on the Home Page, the user is redirected to the screen shown in Figure 33. On this screen, the user can ask a question about the database that will be answered by HEAL, which helps and promotes understanding of the domain.

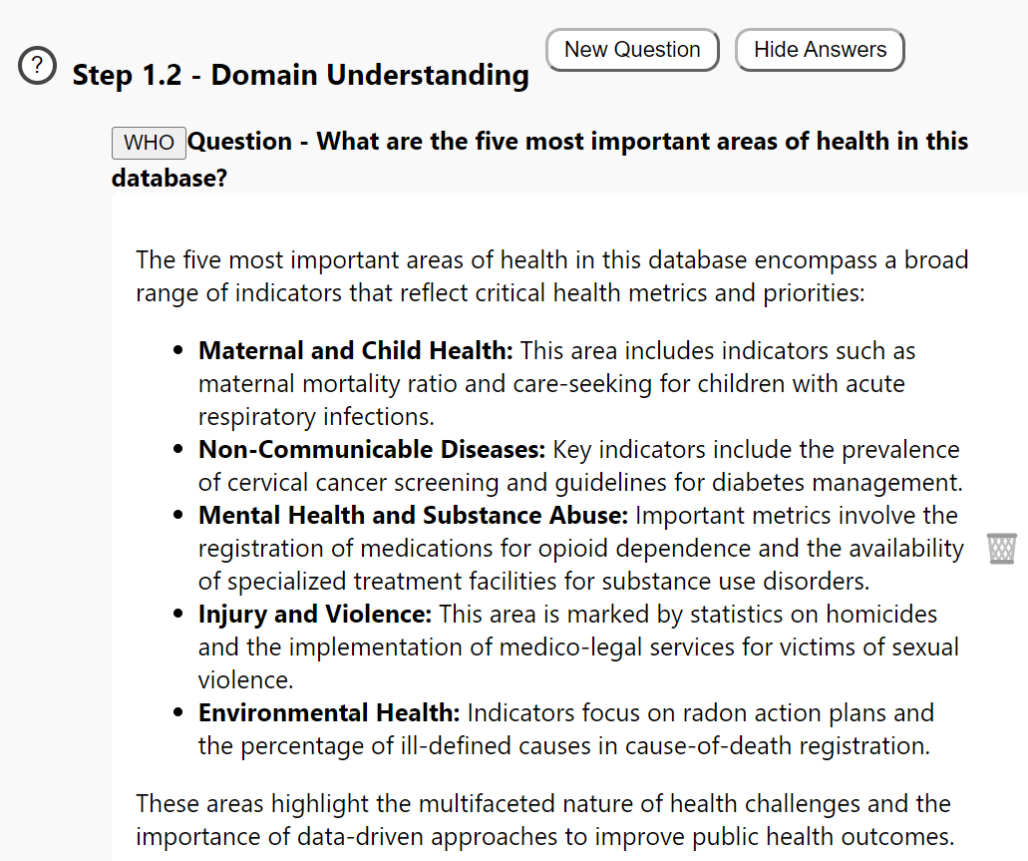
Figure 33 – Screen - New Question



Source: Elaborated by the Author.

Examples of questions and answers for understanding the domain are found in Figure 34.

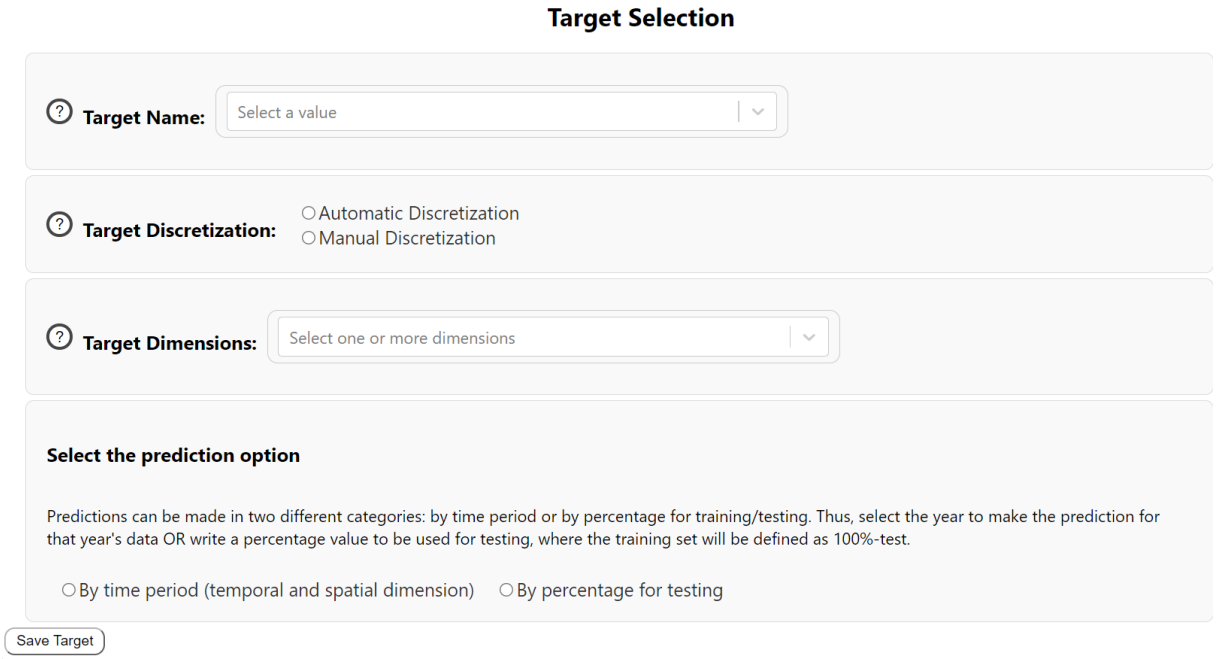
Figure 34 – Screen - Questions and Answers



Source: Elaborated by the Author.

When clicking on the Configure Target option on Homepage, the user is redirected to the screen shown in Figure 35. On this screen, the user can configure the target for AutoML execution, as well as its other characteristics, such as discretization, dimensions, percentage for testing or time period to perform the prediction.

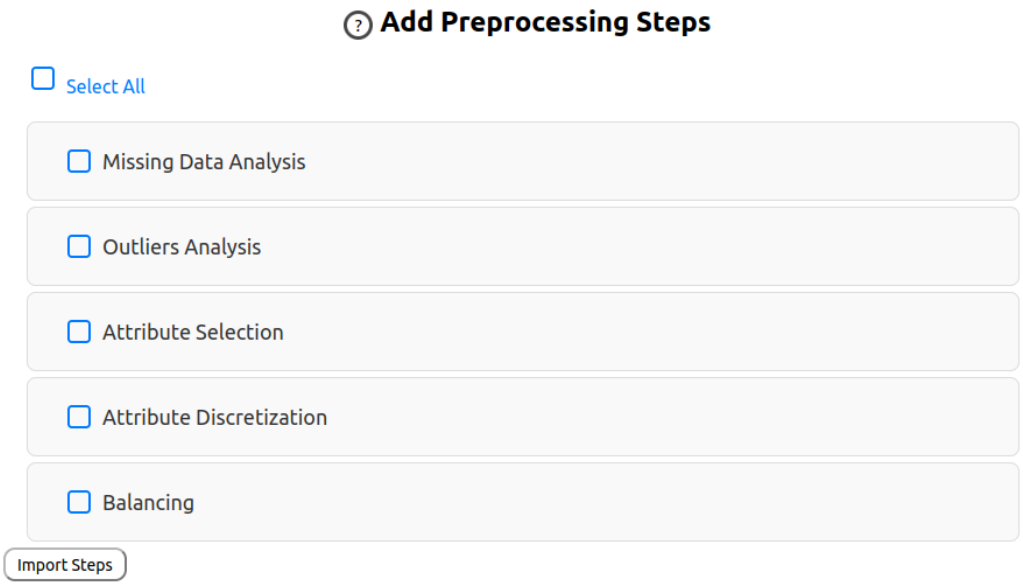
Figure 35 – Screen - Target Selection



Source: Elaborated by the Author.

When clicking on the Add Step option in Step 2 - Pre-processing on the Home Page, the user is redirected to the screen illustrated in Figure 36. On this screen, the user can select which pre-processing steps they want to apply when running AutoML. It is also possible to select the automatic option, in which all steps will be executed automatically as defined by the system.

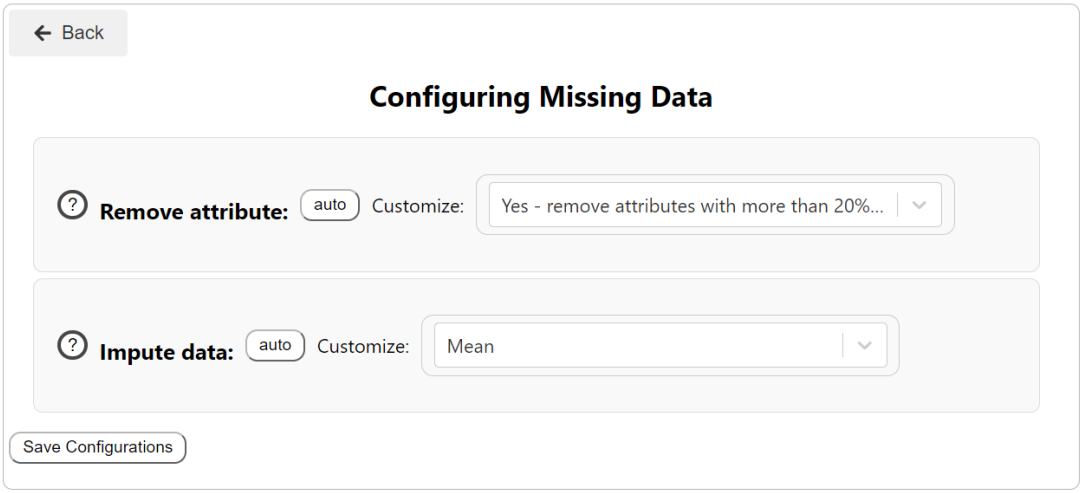
Figure 36 – Screen - Add Pre-processing Steps



Source: Elaborated by the Author.

Figure 37 shows the missing data settings options screen, where the user can configure attribute removal or data import.

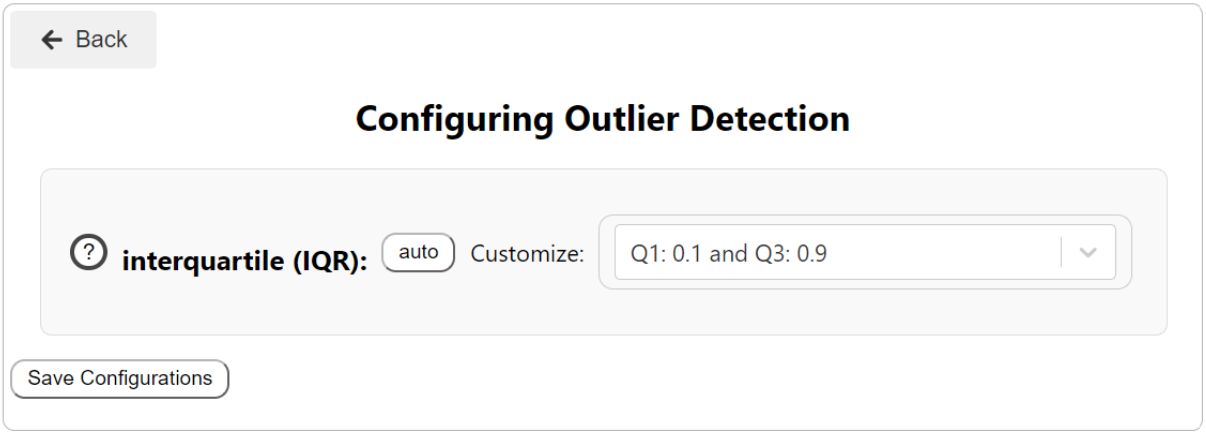
Figure 37 – Screen - Configuring Missing Data



Source: Elaborated by the Author.

Figure 38 shows the outlier detection settings options screen, where the user can configure the application of interquartile range (IQR).

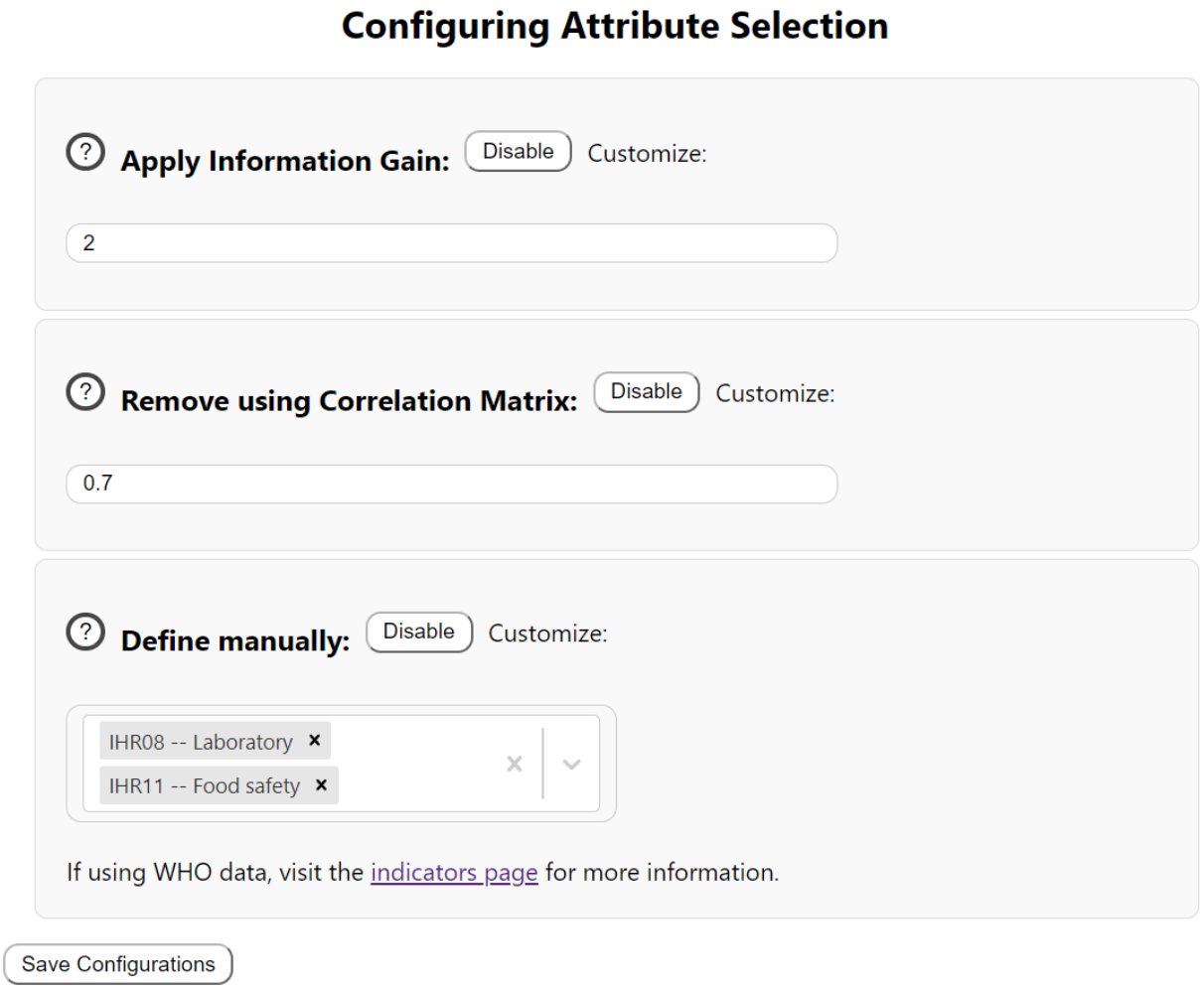
Figure 38 – Screen - Configuring Outlier Detection



Source: Elaborated by the Author.

Figure 39 shows the attribute selection settings options screen, where the user can configure the application of information gain, correlation matrix or manual definition for the selection of attributes used during the AutoML process.

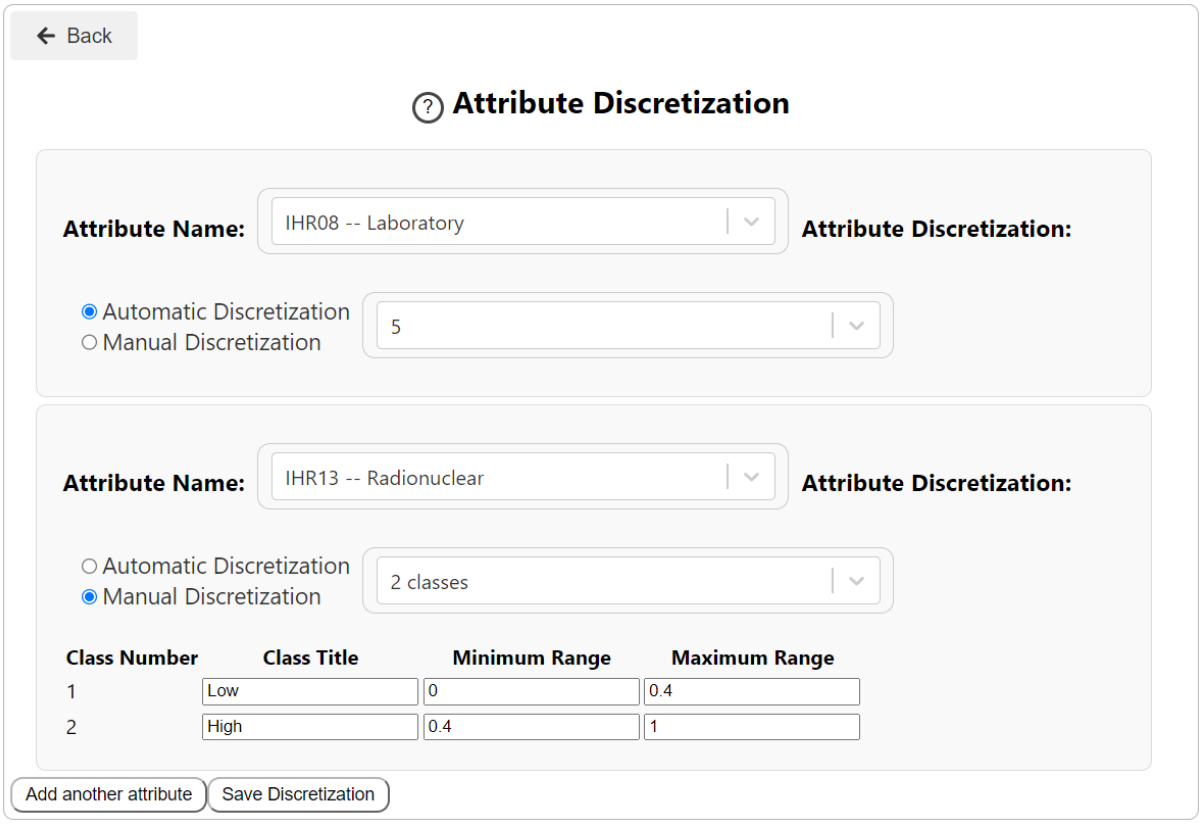
Figure 39 – Screen - Configuring Attribute Selection



Source: Elaborated by the Author.

Figure 40 shows the attribute discretization settings options screen, which the user can use to apply automatic and manual discretization to as many attributes as desired.

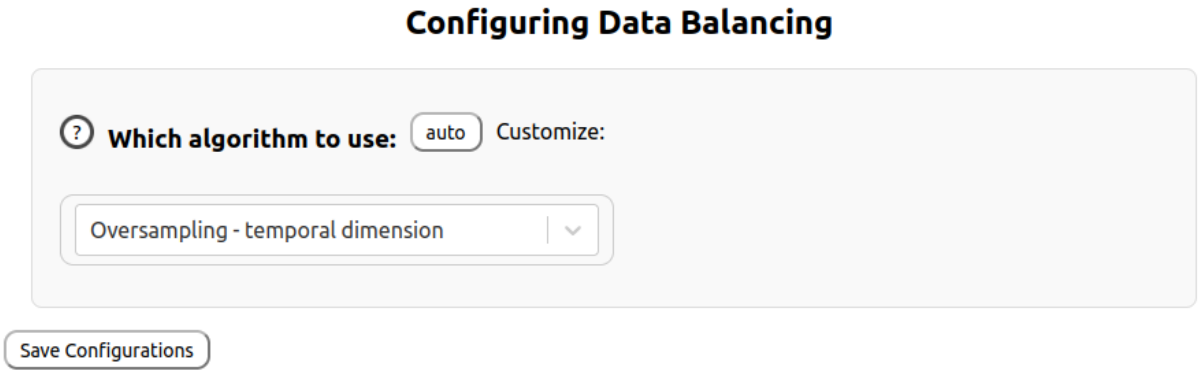
Figure 40 – Screen - Attribute Discretization



Source: Elaborated by the Author.

Finally, Figure 41 shows the data balancing configuration options screen, where the user can configure which algorithm will be applied for balancing or choose not to apply this step.

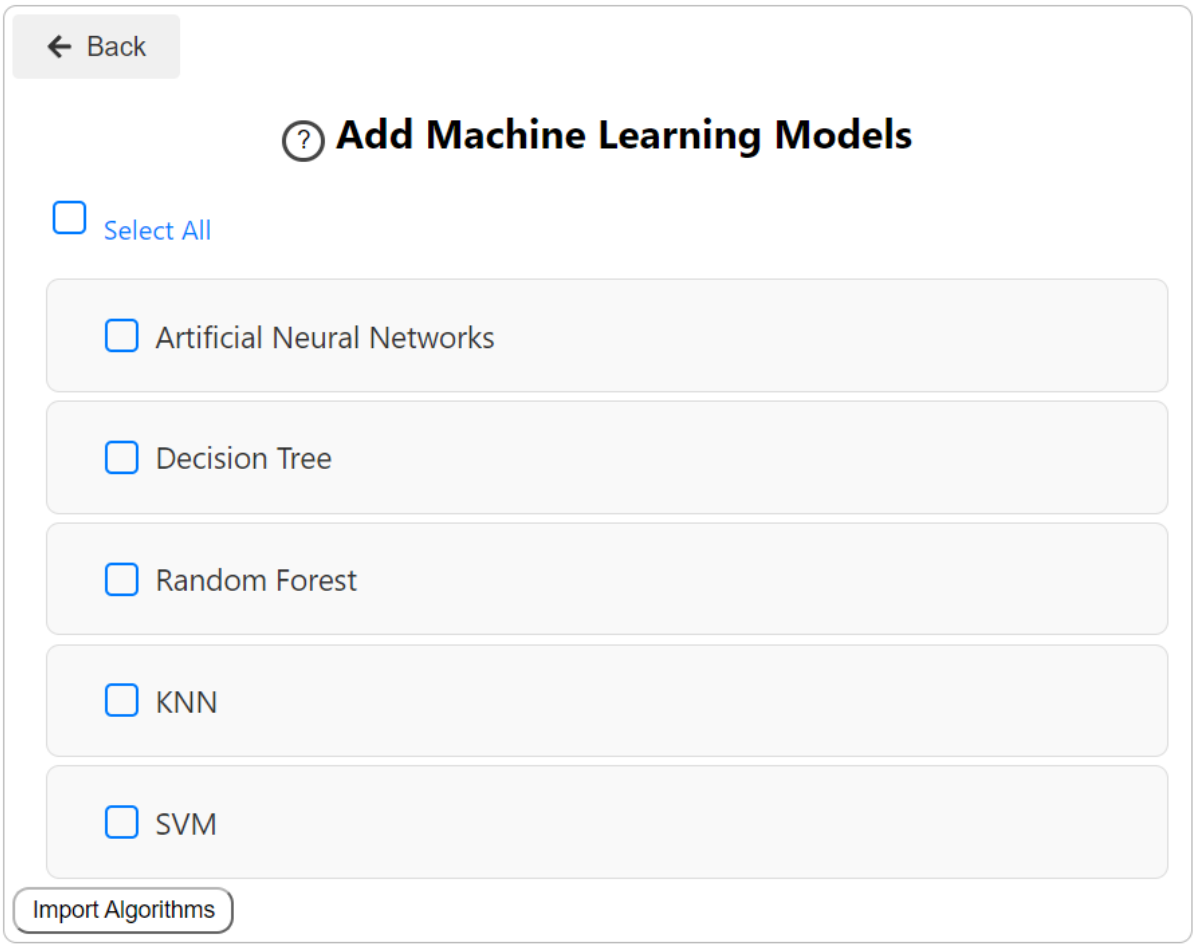
Figure 41 – Screen - Configuring Data Balancing



Source: Elaborated by the Author.

When clicking on the Add Step option in Step 3 - Model Selection on the Home Page, the user is redirected to the screen illustrated in Figure 42. On this screen, the user can select which Machine Learning model they would like to apply when running AutoML. It is also possible to select the auto option, in which all the steps will be executed automatically by the system definition.

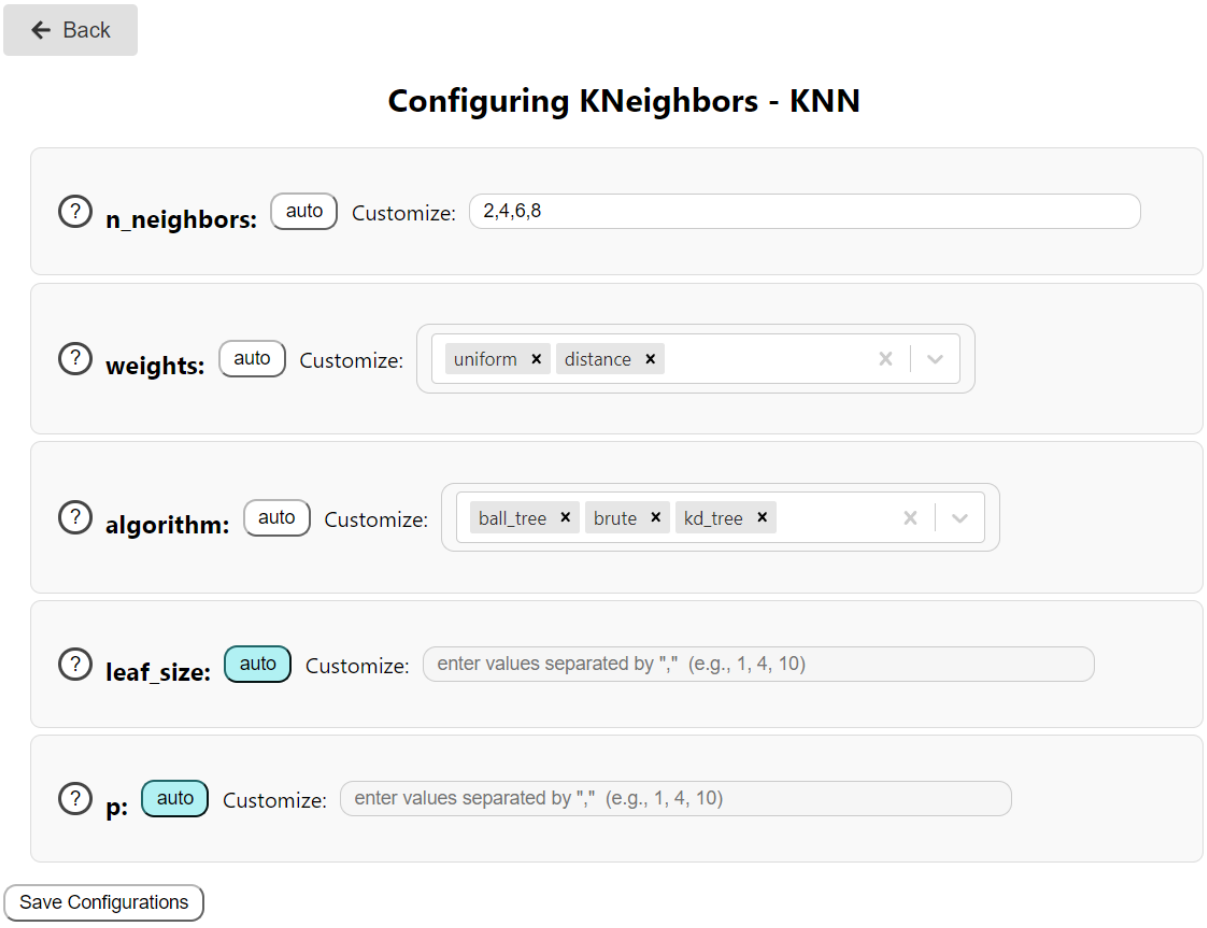
Figure 42 – Screen - Add Machine Learning Models



Source: Elaborated by the Author.

Figure 43 shows an example screen of machine learning algorithm configuration options. A customized screen was developed for each model, aiming to customize the hyperparameters of each one. Since all the screens are similar, changing only the hyperparameters, the screen shown in this figure represents all the others.

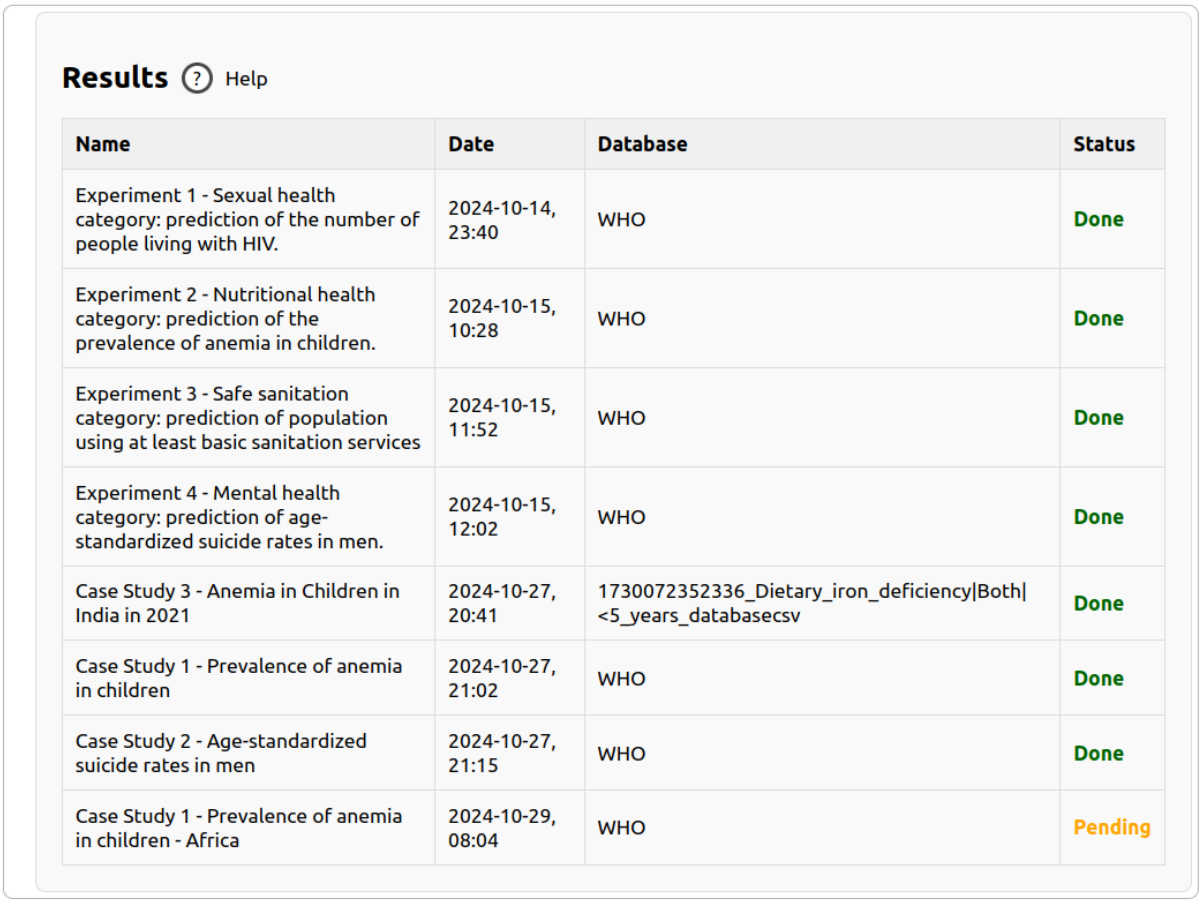
Figure 43 – Screen - Machine Learning Model Configuration Example



Source: Elaborated by the Author.

Figure 44 illustrates the screen that displays all the executions requested by the user, as well as the corresponding status, which can be: pending, completed or error.

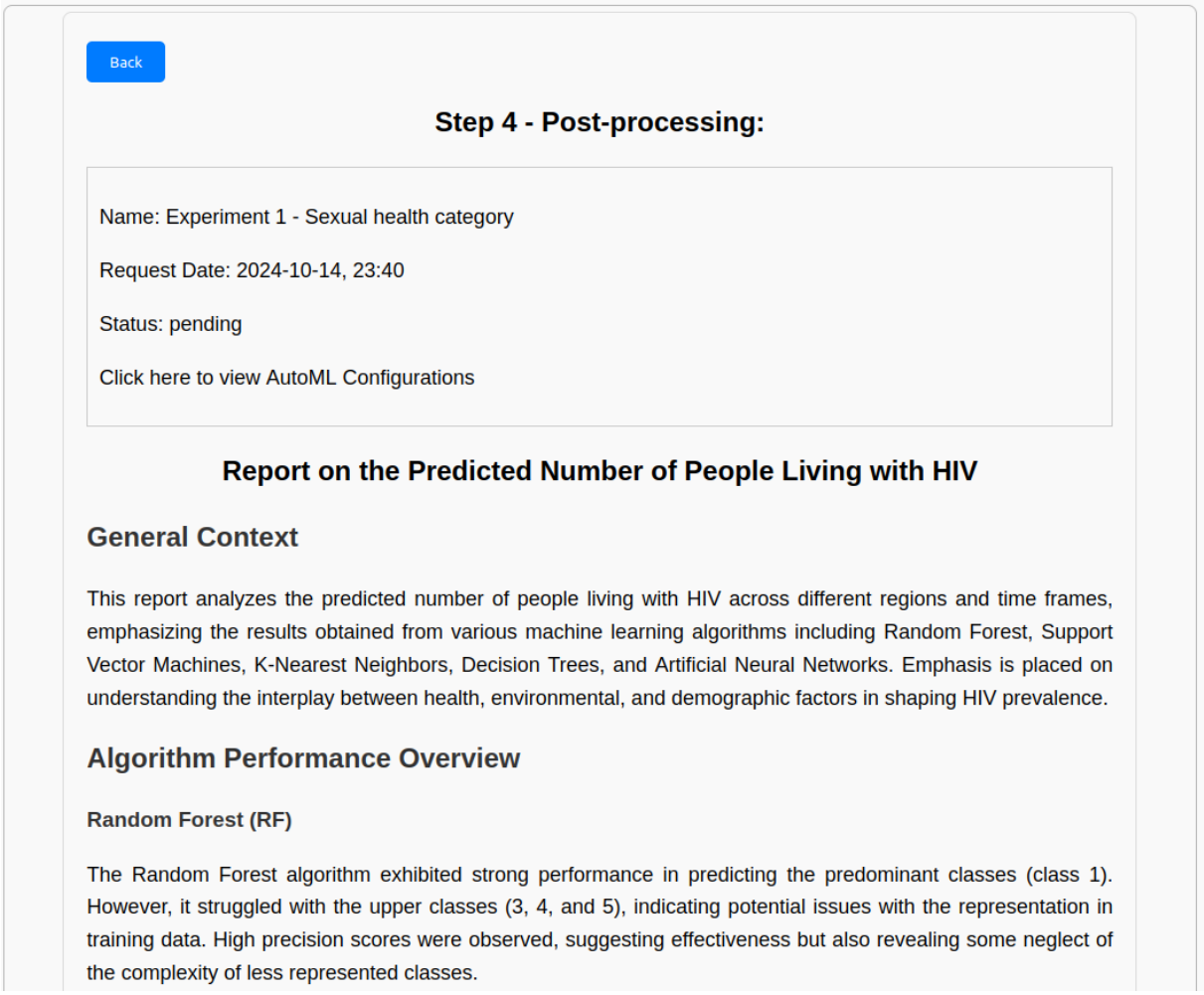
Figure 44 – Screen - Results



Source: Elaborated by the Author.

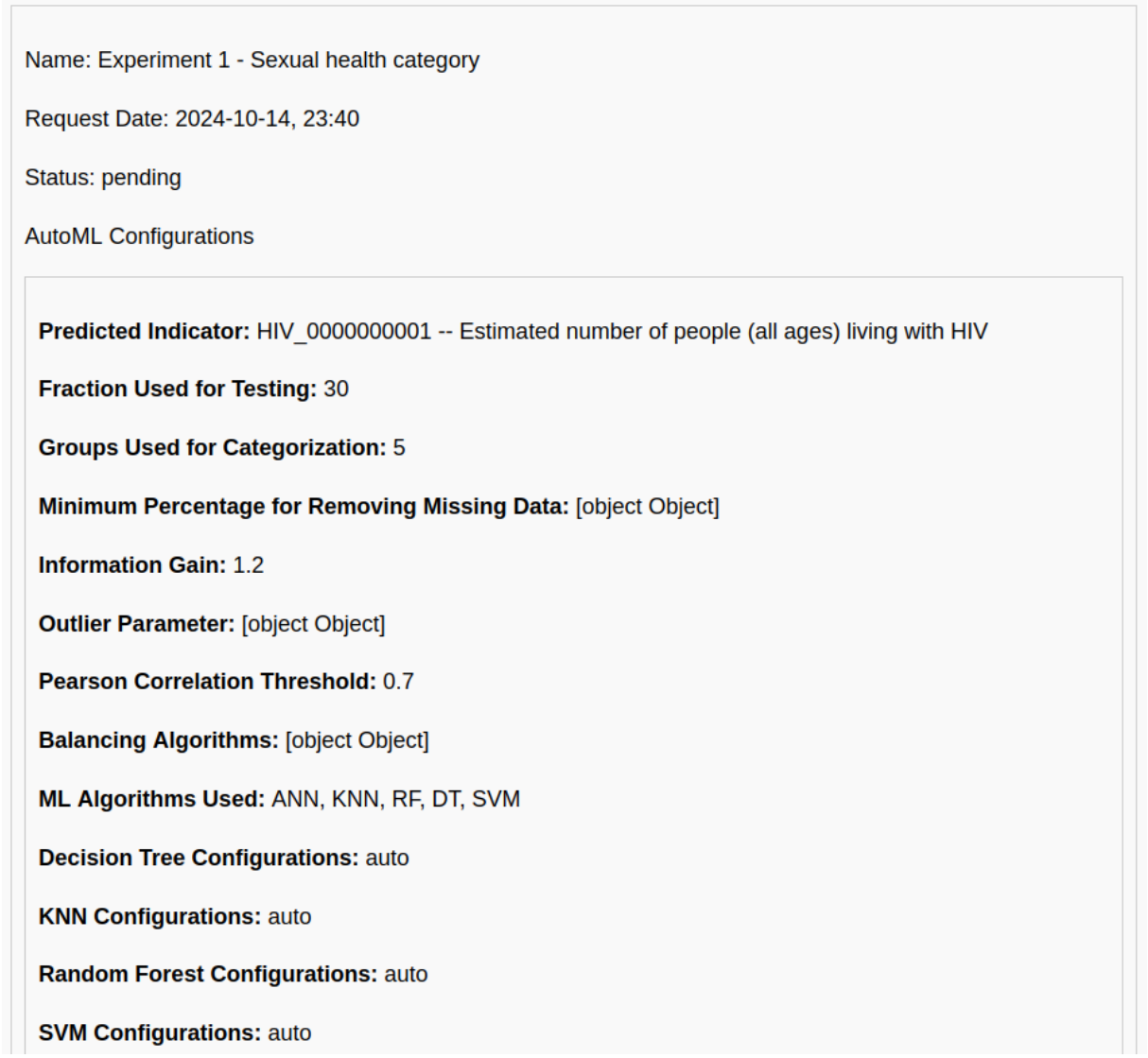
Finally, Figures 45, 46 and 47 show the HEAL results presentation screen, where it is possible to view step 4 - Post-processing with the generation of reports, the configurations performed, as well as the logs of each step of the AutoML process.

Figure 45 – Screen - Post-processing



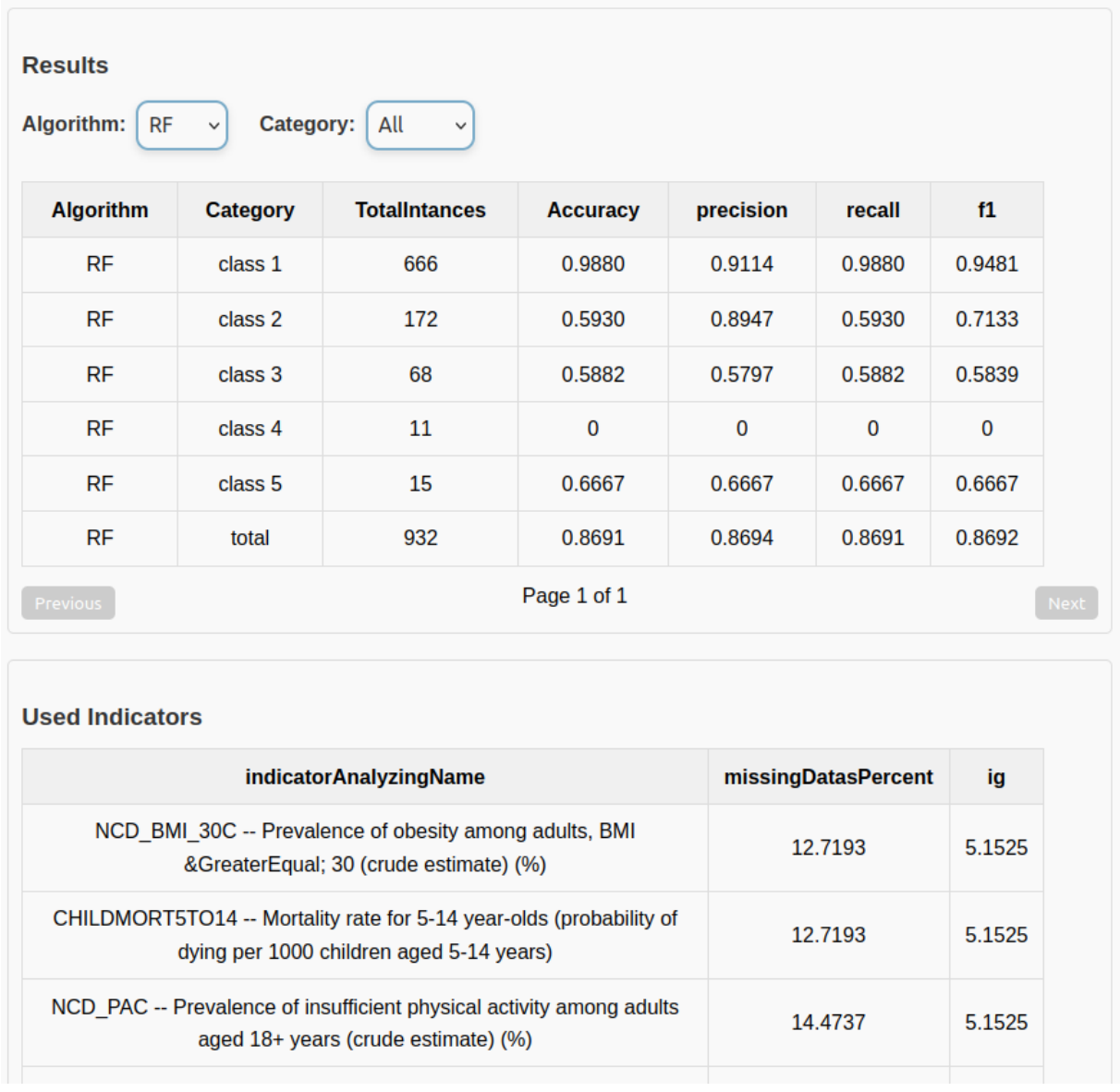
Source: Elaborated by the Author.

Figure 46 – Screen - Log of Configurations Performed



Source: Elaborated by the Author.

Figure 47 – Screen - Presentation of the Results Found



Results

Algorithm: RF Category: All

Algorithm	Category	TotalIntances	Accuracy	precision	recall	f1
RF	class 1	666	0.9880	0.9114	0.9880	0.9481
RF	class 2	172	0.5930	0.8947	0.5930	0.7133
RF	class 3	68	0.5882	0.5797	0.5882	0.5839
RF	class 4	11	0	0	0	0
RF	class 5	15	0.6667	0.6667	0.6667	0.6667
RF	total	932	0.8691	0.8694	0.8691	0.8692

Previous Page 1 of 1 Next

Used Indicators

indicatorAnalyzingName	missingDatasPercent	ig
NCD_BMI_30C -- Prevalence of obesity among adults, BMI ≥ 30 (crude estimate) (%)	12.7193	5.1525
CHILDMORT5TO14 -- Mortality rate for 5-14 year-olds (probability of dying per 1000 children aged 5-14 years)	12.7193	5.1525
NCD_PAC -- Prevalence of insufficient physical activity among adults aged 18+ years (crude estimate) (%)	14.4737	5.1525

Source: Elaborated by the Author.

APPENDIX B – HEAL POST-PROCESSING

This appendix presents all post-processing results generated by HEAL for the three case studies performed in this work. Thus, Sections B.1, B.2 and B.3 present this step for case studies 1, 2 and 3, respectively.

B.1 Post-Processing - Case Study 1

Important: The following report was generated by a tool that uses LLM. Before applying the information contained, it is highly recommended to seek the opinion of an expert.

B.1.1 Analysis of Anemia Prevalence Prediction Results (2019)

Introduction - This report presents an analysis of the predicted prevalence of anemia in children aged 6-59 months in the Americas for the year 2019. The predictions were made using various machine learning algorithms, each evaluated based on their performance metrics.

Interpretation of Results - The results indicate a distinct variability in performance across different machine learning algorithms implemented, namely Artificial Neural Networks (ANN), Decision Trees (DT), Random Forest (RF), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM).

The Decision Trees (DT) and Random Forest (RF) demonstrated the strongest overall performance, achieving high accuracy and robust metrics across all classes. Notably, the RF model maintained an impressive f1 score of approximately 0.964, indicating excellent precision and recall in classifying the majority of the instances accurately.

K-Nearest Neighbors (KNN) achieved perfect metrics with an f1 score of 1.0, suggesting that this model was capable of classifying all instances correctly. However, such high performance might suggest overfitting, and further validation is necessary to ensure generalizability beyond this dataset.

In contrast, the Artificial Neural Networks (ANN) exhibited the weakest performance, particularly with low precision and recall for certain classes, which indicates challenges in accurately classifying instances of anemia. This raises concerns regarding the model's suitability for practical applications in health interventions.

The Support Vector Machine (SVM) achieved competent results but lacked the consistency displayed by DT and RF, particularly in class 3 where it fell short in precision and recall.

Conclusions and Recommendations - Based on the analysis of the predictive models, it is recommended to prioritize the use of Decision Trees and Random Forest algorithms for decision-making processes pertaining to anemia prevalence in children. These models not only achieved the highest overall accuracy but also displayed well-balanced precision and recall across various classes. Further exploration of hyperparameter tuning and feature selection may enhance model performance even more.

K-Nearest Neighbors, while exhibiting perfect results here, should be approached cautiously to prevent potential overfitting. Future evaluations on unseen data will be crucial in validating the robustness of this model.

It is essential to continually integrate real-world data and monitor outcomes, tailoring interventions based on the findings to effectively combat anemia in vulnerable populations.

B.1.2 Analysis Report on Outliers in Anaemia Predication

Introduction - This report aims to analyze the outliers identified in the dataset concerning the prevalence of anaemia in children aged 6-59 months across the Americas for the year 2019. The focus is on unearthing significant insights and correlations that might not be immediately evident.

B.1.2.1 Key Findings from the Outlier Analysis

Upon reviewing the outliers, the following non-obvious discoveries have been made:

1. Regional Disparities and Population Impact - Countries like Sao Tome and Principe and Niue exhibit extraordinarily high anaemia prevalence rates in children compared to their peers in the Americas. There appears to be a systemic issue related to nutrition and healthcare accessibility that demands attention. The high prevalence of anaemia may reflect broader socioeconomic challenges such as poverty levels, access to healthcare, and nutrition education.

2. Correlation with Other Health Indicators - Several outliers also show a significant correlation with child mortality rates and maternal health statistics. For example, regions with high anaemia prevalence often coincide with higher infant and under-five mortality rates. This correlation indicates a potential cyclical nature of health indicators—where poor maternal health leads to higher childhood anaemia and, subse-

quently, increased child mortality.

3. Importance of Contextual Data - The contextual factors, such as food security and public health initiatives, may illuminate the reasons behind these outliers. For instance, the data shows that Cook Islands and Marshall Islands, despite being developed nations, still face challenges with anaemia potentially due to high obesity rates and related chronic diseases affecting the population's overall health.

4. Potential for Intervention Strategies - The identification of these outliers presents an opportunity for targeted intervention. Countries exhibiting high rates of anaemia should implement robust nutrition programs, enhance maternal education on health, and improve healthcare access to mitigate the prevalence of these health issues.

Conclusion - The analysis of outlier data has unveiled critical relationships between anaemia prevalence in children and broader health determinants. Addressing the underlying socio-economic and health disparities will be essential in reducing the prevalence of anaemia and improving child health outcomes across the Americas.

B.1.3 Correlation Analysis Report on Anaemia Prevalence in Children

This report presents an analysis of the Pearson correlation matrix among various health-related attributes with a focus on the Prevalence of Anaemia in Children Aged 6-59 Months (%). The goal is to uncover non-obvious insights from the data, highlighting important relationships that may affect anaemia prevalence.

Key Findings - High correlations with mortality rates: the analysis reveals a striking high correlation (approximately 0.99) between the prevalence of anaemia in children and various mortality rates, including Infant Mortality Rate and Under-Five Mortality Rate. This suggests that regions with higher mortality rates among infants and children also tend to have increased prevalence of anaemia, indicating a potential underlying issue in healthcare quality or access in these areas.

Negative correlation with the number of women with anaemia: there is a noteworthy negative correlation (around -0.71) between the prevalence of anaemia in children and the prevalence of anaemia in adolescent women (aged 10-19). This could imply that as the prevalence of anaemia in young women decreases, the prevalence in children might increase, leading to the hypothesis that young women may have a protective role in maternal health, affecting child health outcomes.

Nutrition-related factors - The analysis shows a moderate positive correlation (approximately 0.51) between the prevalence of anaemia in children and the prevalence of anaemia in non-pregnant women (aged 15-49). This suggests that nutritional deficiencies affecting women could be reflective of similar deficiencies in children, underlying dietary

patterns that contribute to anaemia across these demographics.

Physical Activity and Anaemia - Interestingly, while many attributes related to mortality have high correlations, the prevalence of insufficient physical activity in adults shows only a weak correlation (about 0.08) with childhood anaemia prevalence. This indicates that factors directly linked to health outcomes, such as nutrition and healthcare access, may be much more significant than lifestyle factors like physical activity when it comes to childhood anaemia.

Potential discrepancies in data - The very low correlations found between anaemia prevalence and neonatal mortality (about -0.11) suggest the need for careful examination of data distribution and collection methods. These discrepancies might indicate misclassification or reporting errors in the data regarding how anaemia is perceived or documented across healthcare systems.

Conclusion - In conclusion, this correlation analysis highlights complex interdependencies impacting the prevalence of anaemia in children aged 6-59 months. The findings emphasize the role of mortality rates and maternal health on child anaemia prevalence and suggest that concerted efforts are needed to address nutrition-related factors in both women and children to improve health outcomes.

Further investigations should integrate qualitative analysis to deepen understanding and refine programs targeting anaemia reduction in vulnerable populations.

B.1.4 Feature Importance Analysis for Anaemia Prevalence Prediction

This report summarizes the findings from the feature importance analysis conducted using four machine learning algorithms: Random Forest, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Neural Networks. The objective was to identify key predictors of anaemia prevalence in children aged 6-59 months in the Americas for 2019.

B.1.4.1 Key Findings

Commonly Important Features Across Algorithms - Prevalence of Anaemia in Non-Pregnant Women (Aged 15-49) (%): this feature consistently appears among the top contributors in both Random Forest and SVM analyses, suggesting a significant relationship between the health of non-pregnant women and children's anaemia prevalence.

Number of Non-Pregnant Women (Aged 15-49 Years) with Anaemia (Thousands) - This feature is also prominently highlighted in the Random Forest model, which indicates that increased anaemia rates in women of reproductive age may correlate

with higher rates in children.

Neonatal Mortality Rate - This variable emerges in both KNN and SVM analyses and points to a concerning association between initial health outcomes in infants and their subsequent risk of developing anaemia.

Prevalence of Insufficient Physical Activity (Adults) - Identified as significant in both KNN and SVM approaches, this may imply a broader public health concern where lifestyle factors in parents or guardians contribute to children's nutritional and health status.

B.1.4.2 Unique Insights from Different Algorithms

Random Forest - Random Forest highlights various mortality rates, such as under-five and stillbirth rates, which are not prioritized in other algorithms. This suggests that these mortality metrics may have indirect associations with anaemia through systemic health status in the population.

KNN - KNN emphasizes the role of insufficient physical activity and its direct correlation with anaemia in children. However, the lack of substantial contribution from the anaemia rates in women indicates this algorithm is sensitive to more localized data patterns.

SVM - SVM analyses reveal that overweight and obesity rates among adults are critical in predisposing children to dietary deficiencies. This finding emphasizes the importance of maternal health and nutrition in the home environment.

Neural Networks - Neural Networks, though not specified in detail here, typically seek complex patterns in data, which may uncover additional nonlinear relationships among these features. The integration of demographic variables alongside health phenomena is crucial for understanding the multifaceted nature of child health.

Conclusions - The analyses from various algorithms indicate a multifactorial relationship between maternal health indicators and children's anaemia prevalence. Notably, while some features were universally acknowledged as significant, unique perspectives emerged from each algorithm that warrant further investigation. These insights should guide targeted interventions aimed at reducing anaemia rates in vulnerable populations.

Recommendations for Further Study - Future research could focus on longitudinal studies to better understand causative relationships. Investigating the community-level factors that influence both maternal and child health may also provide valuable insights for public health policies.

B.1.5 Analysis of Predicted Prevalence of Anaemia in Children Aged 6-59 Months

Overview - This report analyzes the results obtained from multiple machine learning algorithms applied to predict the prevalence of anaemia in children within the Americas for the year 2019. The performance of the models was evaluated via confusion matrices generated from various classification techniques, including Artificial Neural Networks, Decision Trees, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM).

B.1.5.1 Key Findings

Artificial Neural Networks (ANN) - The confusion matrix indicates that while the model performed reasonably well in classifying categories 1 and 2, its ability to distinguish categories 3, 4, and 5 was considerably limited. Specifically, the model failed to achieve any predictions for category 5, possibly indicating a significant gap in the training data for severe anaemia. This deficiency highlights the necessity for more robust data collection efforts, concentrating on underrepresented classes.

Decision Trees - The Decision Tree achieved high accuracy for category 1 but struggled with classes 3, 4, and 5. Remarkably, only a single instance was predicted for category 2. The model's rigid partitioning characteristic may have hindered its effectiveness in capturing the interactions between features that could correlate with the various categories of anaemia. This suggests a potential for overfitting, reinforcing the need for a more flexible method or additional tuning.

Random Forest - Random Forest showed great variance in its performance, with an interesting pattern where category 5 was predicted perfectly while category 4 was neglected altogether. This model, typically known for its ability to handle diverse data, may require enhanced feature engineering or hyperparameter optimization to obtain a more balanced classification across all categories, especially to manage medium prevalence rates.

K-Nearest Neighbors (KNN) - The KNN algorithm essentially mirrored the Random Forest results, indicating a strength in classifying category 1 yet showing a stark limitation in recognizing categories 3, 4, and 5. The lack of adaptability in capturing the complexities of data distributions underlines that, although effective, KNN might not be the most suitable approach in this domain without further refinement or additional relevant features.

Support Vector Machines (SVM) - The SVM results were notably similar to KNN and Random Forest, particularly the patterns across categories. This method

showed reasonable performance for categories 1 and 2 yet presented challenges for categories 3, 4, and 5. The absence of predictions for severe anaemia also raises concerns regarding the representation of data in training, suggesting the need for a more stratified sampling approach to improve overall model performance.

Conclusions - Across all models, a clear pattern emerges indicating a significant underrepresentation of classes, particularly for severe anaemia. Addressing this imbalance should be prioritized in future iterations of model development. Additionally, the results suggest that ensemble methods may yield better performance when leveraged with careful feature selection, enhanced data preprocessing, and potentially incorporating additional sources of data to cater to the nuances associated with anaemia prevalence. Future work should thus focus on systematic data enrichment and exploring more complex models that can adaptively deal with such imbalances.

B.1.6 Analysis of Predicted Results for Anaemia Prevalence in Children (2019)

Introduction - This report analyzes the predictive performance of various machine learning algorithms used to estimate the prevalence of anaemia in children aged 6-59 months in the Americas region for the year 2019. The analysis includes insights into common errors and fruitful discoveries related to the datasets used.

B.1.6.1 Error Analysis

Common Prediction Errors - The algorithms exhibited similar tendencies in misclassifying specific countries, particularly in cases where true values were significantly lower than the predicted values. For instance: Peru and Colombia were predicted inaccurately across multiple algorithms, with predictions leaning toward a higher prevalence category. Argentina, Belize, and Jamaica also showed inconsistencies, with predictions often overshooting the actual levels of anaemia prevalence.

Regional Considerations - Upon examining these errors, a correlation can be observed with socio-economic factors prevalent in the Americas region during that period. Many misclassified countries are those facing political instability, economic challenges, or natural disasters, which impact healthcare systems and the availability of data. For instance: the Venezuelan crisis may have affected the availability and accuracy of health-related data, leading to mispredictions. Countries with fluctuating infrastructure levels, like Haiti and Nicaragua, may also exhibit unreliable health data reflecting on anaemia prevalence statistics.

B.1.6.2 Correct Predictions and Knowledge Discoveries

Surprising Accuracy - Interestingly, the predictive models did exceptionally well in estimating anaemia prevalence for higher-risk countries, such as Dominica and Grenada, which possess more consistent health data. This can be attributed to: a more robust health reporting systems in these countries.

B.1.6.3 Community-wide health initiatives and policies aimed at tackling malnutrition and anaemia.

Patterns in Successful Predictions - Further analysis revealed patterns in countries that achieved high prediction accuracy: countries like Trinidad and Tobago and the United States showcased effective public health campaigns, minimizing anaemia prevalence and improving accuracy in predictions.

Countries with lower socio-economic status, but with recent health investments, such as Guatemala, reflected surprising accuracy in predictions, suggesting positive effects from recent health initiatives.

Conclusion - In summary, while the predictive algorithms successfully identified many trends within anaemia prevalence among children in the Americas, significant regional discrepancies remained, primarily influenced by socio-economic conditions and the political landscape. The insights gathered from both the errors and successes provide an imperative understanding of how external factors impact healthcare and data reliability, ultimately guiding future efforts in health data collection and predictive modeling.

B.1.7 Analysis of Selected Hyperparameters for Classification of Anaemia Prevalence in Children

Introduction - This report presents an analysis of hyperparameter selections derived from various machine learning algorithms used in predicting the prevalence of anaemia in children aged 6-59 months in the Americas for the year 2019. The selected techniques include Artificial Neural Networks (ANN), Decision Trees, Random Forests, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM).

B.1.7.1 Key Findings

1. Artificial Neural Networks (ANN) - The best-performing configuration was identified with a combination of 40 neurons, a dropout rate of 0.2, the leaky_relu activation function, using SGD as the optimizer. This configuration achieved a score of approximately 89.96%. Notably, the presence of the dropout layer indicates a proactive

strategy against overfitting, which is critical in health-related predictions where model generalization is paramount.

2. Decision Trees - Several configurations yielded similar mean test scores around 70% with gini criterion and best splitter. The significance of the max depth parameter emerges here. A maximum depth of 20 offered the best balance of accuracy without excessive model complexity, implying the data's inherent simplicity in structure allowing for simpler models to perform adequately.

3. Random Forest - The Random Forest model clearly outperformed others with a mean test score of approximately 85.51% using 50 trees with a maximum depth of 20. The use of bootstrap sampling provides insight into increased robustness and handling of variability significantly associated with health data. The ranking of importance evident from the variations in the number of estimators suggests an optimal stopping point in model training, reinforcing the balancing act required in complex health datasets.

4. K-Nearest Neighbors (KNN) - This algorithm exhibited exceedingly consistent results with a top mean test score of 95.74%, where varied configurations with distance weight garnered high ranks. This reveals KNN's potential strength in capturing local patterns effectively in health data, where nuanced distinctions can be essential.

5. Support Vector Machines (SVM) - Results illustrated lesser reliability, with the highest mean test score reaching only 90.37%. Although the rbf kernel with C set to 1 managed to perform best, the overall lower scores suggest challenges in capturing the intricate patterns in the data compared to ensemble or non-parametric methods such as Random Forest and KNN.

Conclusion - Analysis of the model hyperparameters reveals significant insights about the nature of the anaemia prevalence prediction problem in children. The selection strategies applied during random search lend evidence to the idea that model simplicity and robustness are key. Moreover, the efficacy of ensemble methods like Random Forests and KNN, with high adaptability and local pattern recognition, makes them rather advantageous in this domain. Further research could delve into combining models in a stacked ensemble approach aimed at addressing potential underfitting exhibited in other models. As healthcare data is typically complex and diverse, careful tuning of hyperparameters with a focus on robust validation methods will remain essential for deriving reliable and actionable insights.

B.2 Post-Processing - Case Study 2

Important: The following report was generated by a tool that uses LLM. Before applying the information contained, it is highly recommended to seek the opinion of an

expert.

B.2.1 Class-specific Insights

Class 1 was consistently identified with high recall across all models, indicating an effective detection of cases classified under this category. Conversely, Class 2 exhibited issues, particularly using the Decision Tree model, which failed to recognize cases (zero recall). This raises concerns over potential data imbalances or inadequacies that may have influenced the algorithm's performance.

Furthermore, Classes 4 and 5 showed strong performances across all algorithms, achieving perfect F1 scores, attesting to the robustness in detecting these categories. However, increased inspection into Class 2 is crucial, as the significant drop in performance could obscure critical findings regarding suicide risks in certain populations.

Recommendations for Further Analysis - Given the results, a thorough re-evaluation of the features contributing to Class 2 should be undertaken. Perhaps a combination of feature engineering or a closer examination of the underlying data distribution could yield better performance. Balancing the dataset and applying different methods of oversampling may also prove beneficial. Ultimately, while the KNN algorithm stands out, integrating insights and leveraging ensemble methods may enhance the predictability of suicide rates further, contributing to improved mental health strategies and interventions in Africa.

Conclusion - This predictive modeling exercise has underscored the potential of machine learning in tackling urgent public health issues such as suicide prevention. Continuous refinement of these models, alongside domain expertise, is essential to foster effective policy-making and intervention strategies.

B.2.2 Outlier Analysis Report on Suicide Rates in Africa (2019)

Introduction - This report presents an analysis of outliers identified in the dataset concerning age-standardized suicide rates across various African countries in 2019. The analysis aims to uncover insights that may provide a deeper understanding of the underlying factors associated with higher suicide rates.

B.2.2.1 Analysis of Outliers

Upon analyzing the listed outliers, several key observations were made:

Eswatini's High Suicide Rates - The years 2010 and 2012 show exceptionally high suicide rates. These coincided with poor mental health indicators and significant non-

communicable diseases (NCD) mortality rates. This suggests a potential overlap between health issues and mental health crises, warranting further investigation into mental health services availability.

Lesotho's Alarming Statistics - Lesotho exhibits the highest suicide rate in 2014. Coupled with high maternal mortality rates, it indicates crucial public health challenges where mental health may not be adequately addressed in health policies. Addressing maternal health might have a dual benefit of improving mental wellness among new mothers.

Cabo Verde's Lower Suicide Rate - Despite being an outlier, Cabo Verde's suicide rate appears comparatively low. This could indicate effective local health policies, community support systems, or sociocultural factors mitigating against suicides. Investigation into these mitigating factors could help replicate similar successes in other countries.

Seychelles' Unique Position - The lower suicide rate in Seychelles amidst higher prevalence of obesity suggests that lifestyle factors may not directly correlate with mental health. Further psychological studies are needed to conclude the impact of lifestyle related to suicide risk in this region.

Need for Holistic Approach - Overall, the data supports a multifactorial approach to suicide prevention. It indicates the necessity of integrating mental health awareness with other public health domains such as maternal health, adolescent health, and NCD management.

Conclusion - The analysis of outliers in suicide rates across Africa highlights essential public health concerns that intertwine mental health with physical health indicators. These findings signal the urgent need for interventions focusing on mental health, particularly in regions with high NCD mortality and maternal health challenges. Addressing these areas could provide a pathway to reducing suicide rates effectively.

B.2.3 Correlation Analysis Report

This report highlights key insights derived from the Pearson correlation matrix conducted on various health-related attributes in Africa for the year 2019, focusing on their relationships with the Age-standardized suicide rates (per 100,000 population).

B.2.3.1 Key Discoveries:

High Positive Correlation - A notable high correlation (0.991) exists between LIFE_0000000031 (number of people left alive at age x) and WHOSIS_000007 (Healthy

life expectancy at age 60). This suggests that higher life expectancy is closely related to the longevity of populations, potentially indicating lower suicide rates.

Strong Negative Correlation - The attribute NCD_BMI_30A (prevalence of obesity among adults) displays a significant negative correlation (-0.845) with age-standardized suicide rates. This could imply that areas with higher obesity rates may experience lower suicide rates, possibly reflecting socioeconomic support or health resources available to populations with higher obesity.

Understood Risk Factors - MORT_MATERNALNUM (number of maternal deaths) and MDG_0000000026 (maternal mortality ratio) were both found to be negatively correlated with suicide rates (-0.740 and -0.765, respectively). This observation may suggest that higher maternal mortality rates might correlate with increased community distress and wider societal issues, potentially leading to higher suicide rates.

Complex Interactions - A strong positive correlation (0.998) was observed between MORTADO (adolescent mortality rate) and SDG_SH_DTH_RNCOM (number of deaths attributed to non-communicable diseases). While these may not directly correlate with suicide rates, they indicate underlying issues with health standards and the well-being of younger populations, which could influence suicide rates indirectly.

Violence and Mental Health - The number of homicides (VIOLENCE HOMICIDENUM) showcases a substantial negative correlation (-0.836) with healthy life expectancy. This correlation may suggest that regions with higher homicide rates also struggle with mental health issues and could see increased suicide rates, revealing a complex narrative of violence and mental wellness.

Conclusion - This correlation analysis underscores several pivotal relationships that can inform targeted interventions to reduce suicide rates. Understanding the interplay of health determinants such as obesity, maternal health, and violence can aid policy-makers and health professionals in addressing the multifaceted causes of suicide in Africa. Further exploratory and causal analyses are warranted to unravel the complexities at play and to develop actionable strategies tailored to the region's distinct health challenges.

B.2.4 Feature Importance Analysis Report

Overview - This report provides a comprehensive analysis of feature importance derived from various machine learning algorithms applied to predict age-standardized suicide rates in Africa for the year 2019. The algorithms examined include Random Forest, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Artificial Neural Networks (ANN).

B.2.4.1 Key Findings

The Random Forest algorithm identified Total NCD Deaths (NCD_DTH_TOT) and Number of deaths attributed to non-communicable diseases (SDG_SH_DTH_RNCOM) as the most significant features, highlighting the correlation between non-communicable diseases and suicide rates.

For KNN, the features Person-years lived above age x (LIFE_0000000034) and Neonatal mortality rate (WHOSIS_000003) exhibited notable importance, suggesting a potential connection between early life health indicators and later suicide rates.

In SVM, Healthy life expectancy at age 60 (WHOSIS_000007) was found to be critical, indicating that the quality of life at older ages could be a determinant factor for suicide risk.

B.2.5 Unique Discoveries

Some intriguing insights were observed - The prevalence of obesity and being overweight emerged as relevant features across most algorithms; however, their impact varied, indicating they may influence suicide rates differently based on the underlying health conditions related to mental health.

While several features related to violence and homicide rates were present, they showed minimal importance in KNN and SVM models, possibly suggesting that indirect indicators are more predictive of suicide rates than direct violence measures.

Notably, maternal mortality and adolescent mortality rates also appeared in several importance rankings, hinting at the long-term effects of maternal health on youth mental health outcomes.

Conclusion - This analysis underlines the complexity of factors contributing to suicide rates, emphasizing the need for a multifaceted approach in public health strategies. The findings warrant further investigation into the identified features to develop targeted interventions aimed at reducing suicide rates in the African context.

B.2.6 Analysis of Confusion Matrices for Multiple Algorithms

This report analyzes the performance of various machine learning algorithms applied to the classification of Age-standardized suicide rates in the African region for the year 2019. The confusion matrices obtained from different algorithms reveal key insights and potential areas for further investigation.

1. **Neural Networks -** The confusion matrix from the Neural Networks model

shows that the majority of predictions align with class 1 (30 correct predictions). However, classes 2 and 5 exhibit notable imbalances, with only 10 and 2 correct predictions, respectively. The absence of predicted instances for classes 3 and 4 raises concerns about model sensitivity to the less frequent categories. This suggests a need for improved handling of class imbalance, potentially through techniques such as oversampling or cost-sensitive learning.

2. Decision Trees - The Decision Tree algorithm demonstrates a strong prediction capability for classes 1 and 2, with 36 and 4 correct predictions, respectively. However, there is evident confusion when predicting class 4 with only one correct instance, despite its presence in the predictions. Class 3 is seemingly ignored. This model's performance indicates some degree of overfitting, considering the limited transitions between its predictions. The split criteria utilized might require reassessment or tuning to balance performance across all classes effectively.

3. Random Forest - The Random Forest model mirrors the performance of Decision Trees, maintaining strong accuracy for classes 1 and 2 but struggling with classes 3 and 4. It correctly identifies class 1 (36 instances) but shows confusion in distinguishing classes 2 and 4, indicating the potential for further feature importance analysis to refine model decision-making. Given that it achieved a perfect score with instance class 5, exploring features that contribute to its identification may shed light on effective predictors.

4. K-Nearest Neighbors (KNN) - KNN achieved the best understanding of class 1 with 39 correct predictions, showcasing its strength with prevalent categories. However, like the Decision Tree and Random Forest, it faces substantial challenges with the minority classes. The overlap in misclassifications hints at a lack of sufficient training instances for classes 2, 3, and 4. This finding emphasizes the impact of feature scaling in KNN's performance, suggesting that normalization might enhance accuracy across all categories.

5. Support Vector Machine (SVM) - The SVM model performed similarly to the Random Forest and Decision Tree in classifying the prevalent categories. Still, it suffers from the same issues regarding class imbalance, particularly for classes 3 and 4. Its consistent misclassification of these minor classes indicates that more sophisticated kernels or hyperparameter tuning may be necessary to enhance its discriminative power. Moreover, exploring the implementation of one-vs-all strategies could potentially improve its handling of the less frequent categories.

Conclusion - Across all models tested, a common theme emerges the substantial challenges posed by class imbalance, particularly for less frequent suicide rate categories. While models maintain accuracy for the majority class, the predictive performance for minority classes is critically limited. Future work should include exploring advanced

techniques for addressing class imbalance, alongside improved feature selection processes. Such improvements could significantly enhance model robustness and ultimately lead to more impactful insights in public health strategies to mitigate suicide rates.

B.2.7 Analysis of Prediction Results - Suicide Rates in Africa (2019)

Introduction - This report summarizes the findings from the predictions of age-standardized suicide rates across African nations in 2019 using various machine learning algorithms, including Neural Networks, Decision Trees, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM).

Overview of Predictions - The modeling exercises produced consistent predictions for the majority of the countries in the region. Notably, several countries were often categorized similarly across the different algorithms, highlighting a potential underlying pattern in the data. However, some notable discrepancies were observed, particularly in the predictions of Lesotho and São Tomé and Príncipe.

Analysis of Errors, Common Errors Across Algorithms - Lesotho, for example, was predicted as class 2 by the Decision Tree and class 5 by Random Forest, indicating a profound inconsistency among algorithms. This suggests that the feature representation for Lesotho might lack robustness; potentially an indication of missing socioeconomic or healthcare data which could significantly impact mental health and, consequently, suicide rates.

Regional Insights - Looking at the region's context in 2019, it's essential to consider factors such as political stability, healthcare access, economic conditions, and social programs aimed at mental health. The Southern African region, particularly affected by high rates of poverty and unemployment along with limited access to mental health resources, may contribute to inconsistent predictions. Countries like Lesotho with unique socioeconomic challenges may require tailored models that account for these nuances.

Temporality of Predictions - Given the year of prediction (2019), political dynamics and healthcare reforms may have led to shifts in mental health crisis management, which are not reflected in the historical training datasets used for these algorithms. These time-varying factors could have introduced discrepancies in expected outcomes.

B.2.7.1 Insights from Accurate Predictions

Positive Consistency in Predictions - The algorithms consistently identified Nigeria, Burkina Faso, Togo, and Kenya as having high suicide rates (category 1), which the societal dynamics and public health challenges corroborate. This consistency indicates

a reliable pattern that aligns with existing public health data, suggesting that such models might be useful for monitoring trends in specific regions.

Non-Obvious Discoveries - The fact that certain countries like São Tomé and Príncipe were misclassified gives rise to questions about the availability and accuracy of data. The varying accuracy in rural versus urban regions in Africa may significantly affect outcomes; thus emphasizing the need for localized data collection, as bulk datasets may obscure critical detail. This could lead to misrepresenting the mental health crisis in smaller nations or rural communities.

Conclusion - This analysis indicates that while algorithms provide a promising framework for predicting health outcomes, they require context-specific adjustments to accurately reflect the socio-political dynamism of the regions in question. Continued efforts to improve data quality and representation will enhance predictive power, potentially aiding policymakers in addressing mental health issues proactively.

B.2.8 Analysis of Hyperparameter Selection for Multi-class Classification of Age-standardized Suicide Rates in Africa (2019)

Introduction - This report provides an analysis of the hyperparameters selected through random search across various classification algorithms used to predict age standardized suicide rates per 100,000 population in Africa for the year 2019. The objective is to uncover non-obvious insights regarding the performance of the models.

B.2.8.1 Hyperparameter Analysis

1. Neural Networks - The best performing configuration for the neural network consisted of 41 neurons, a dropout rate of 0.2, utilizing the leaky_relu activation function, with the rmsprop optimizer and a learning rate of approximately 0.0013. This configuration achieved a score of 0.8351, indicating strong predictive capability.

Interestingly, it was noted that while tanh activation functions generally yield good results, in this context, leaky_relu produced significantly better performance. This suggests that the models benefit from reduced risk of vanishing gradients when learning from the dataset.

2. Decision Trees - For decision trees, the highest scores emerged with max depths of 20 and 40, suggesting a tendency for deeper trees to offer richer decision boundaries. The best performing configuration utilized the gini criterion with a max depth of 10, recording a mean test score of 0.6191, reinforcing the effectiveness of limiting depth to prevent overfitting.

It is also noteworthy that varying the max_features parameter did not significantly enhance the model performance, indicating that the model's simplicity might be adequate for this problem space.

3. Random Forest - The Random Forest model indicated that an optimal max_depth of 10, with log2 as max_features, resulted in the highest mean test score of 0.7363. Surprisingly, increasing the number of estimators beyond 100 did not lead to substantial performance gains, indicating potential diminishing returns of adding more trees to the ensemble.

This suggests the necessity of fine-tuning both depth and feature subsets rather than solely increasing tree count for performance optimization in this context.

4. K-Nearest Neighbors (KNN) - KNN configurations demonstrated that increasing the number of neighbors generally improved accuracy. The uniform weight setting with 7 neighbors yielded the highest score of 0.7917. However, using distance weighting also provided competing scores, implying that the dataset's local structure has significant implications for predictive performance.

These results suggest that attention to neighbor weighting can yield unforeseen benefits in certain contexts while also reflecting the complexity of the dataset's feature space.

5. Support Vector Machines (SVM) - The SVM model performance revealed mixed results, where the radial basis function (RBF) kernel with a complexity parameter (C) of 10 provided the best score of 0.7523. Interestingly, a polynomial kernel also indicated competitive performance with a slightly higher complexity. This leads to the observation that complex relationships in the data may prefer non-linear modeling techniques.

Furthermore, despite the linear kernel's simplicity, it had marginally lower performance, denoting that the assumption of linearity may not hold within this dataset.

Conclusion - The analysis of hyperparameters selected through random search evidenced several non-obvious findings relevant to modeling of age-standardized suicide rates. The choice of activation functions, model depths, and weighting parameters across different algorithms play a crucial role in the predictive performance, demonstrating that refined tuning and understanding of the dataset can yield significant improvements. This insight could be instrumental in refining strategies to address the public health challenge of suicide in Africa.

Recommendations - Explore further activation functions in neural networks for potentially better performance. Experiment with non-traditional max_features settings for decision trees and Random Forest models. Consider additional distance metrics for

KNN to further enhance model accuracy. Investigate the underlying data distribution to leverage SVM optimally.

B.3 Post-Processing - Case Study 3

Important: The following report was generated by a tool that uses LLM. Before applying the information contained, it is highly recommended to seek the opinion of an expert.

B.3.1 Analysis of Dietary Iron Deficiency Prediction Results

The predictive modeling effort for identifying dietary iron deficiency in children under five years in Asia for the year 2021 utilized multiple machine learning algorithms, which yielded varying accuracy and performance metrics. The following is a detailed analysis of the results obtained.

Overall Performance - Across the evaluated models, the Random Forest (RF) and K-Nearest Neighbors (KNN) algorithms demonstrated the highest overall accuracy, both achieving approximately 93%. Both models also exhibited strong precision and recall metrics, indicating their reliability in identifying true positives for the classes of interest.

Class-Specific Insights - When breaking down the results by class, it is notable that Class 4 consistently outperformed Class 3 across all algorithms. This trend suggests that the features used in the prediction modeling were more effective in capturing the characteristics leading to iron deficiency in the Class 4 category. For instance, the RF and KNN models achieved perfect precision (1.0) and very high recall values (over 0.95) for Class 4, indicating a strong capability to detect cases without generating many false positives.

In contrast, Class 3 displayed more variability in performance, with lower recall rates, particularly in the Artificial Neural Network (ANN) model, which struggled to identify Class 3 instances, achieving a recall score of only 0.5. This inconsistency warrants further investigation into the input features and data distribution related to Class 3, as it suggests potential shortcomings in feature representation or class imbalance.

Algorithm Comparison - The Decision Tree (DT) algorithm emerged as a solid performer with an overall accuracy of about 73%, which is reasonable but notably lower than RF and KNN. Nevertheless, its ability to capture Class 4 instances effectively is commendable, although improvements can be made concerning Class 3 detection.

Support Vector Machine (SVM) showed moderate performance, achieving an overall accuracy of 76.67%, yet, it fell short on the precision and recall metrics associated

with Class 3. The results indicate a need for optimization or perhaps kernel adjustments to enhance model effectiveness.

Conclusion and Recommendations - In conclusion, the RF and KNN models outperformed others in this study and should be prioritized for further application in clinical settings to assess dietary iron deficiency in children. There is a crucial need for further refinement of models regarding Class 3 instances, including conducting additional feature engineering and possibly exploring different sampling techniques to address class imbalance. Future investigations should focus on enhancing model interpretability and integrating clinical insights to ensure practical usability in healthcare scenarios.

Overall, these predictive models present a promising path toward improving dietary assessments and interventions aimed at reducing iron deficiency among vulnerable populations in Asia.

B.3.2 Correlation Analysis Report

This report presents the analysis of the Pearson correlation matrix derived from various health-related attributes in Asia for the year 2021. Notably, the target variable investigated in this analysis is "Dietary iron deficiency | Both | <5 years".

B.3.2.1 Key Findings

Strong Correlation with Tuberculosis Attributes - Several tuberculosis-related variables, such as "Tuberculosis | Male | <5 years" and "Extensively drug-resistant tuberculosis | Female | <5 years" exhibit strong positive correlations with the target variable. This suggests a critical link between iron deficiency and tuberculosis prevalence among children under five. Such insights underline the importance of addressing nutritional support as part of the therapeutic approach for tuberculosis in young patients.

Inverse Relationship with Hypertensive Heart Disease - There is a noticeable negative correlation between the dietary iron deficiency and several hypertensive heart disease attributes, especially among females aged 70 and above. An intriguing interpretation could be that dietary iron deficiency is associated with a lower incidence or reporting of hypertensive heart disease, possibly due to underlying health conditions influencing both metrics.

Impact of Respiratory Infections - The attributes related to upper and lower respiratory infections in both males and females display a weak to moderate negative correlation with dietary iron deficiency. This relationship implies that populations experiencing higher rates of respiratory infections may experience differing nutritional outcomes, potentially due to healthcare access or socioeconomic factors affecting overall health.

Gender Disparities - There are patterns indicating gender differences in the correlation with dietary deficiencies, where certain diseases such as multidrug-resistant tuberculosis appear more correlated in males than females. This discrepancy points to the possibility of gender-specific health interventions necessary for addressing iron deficiency, given its relationship with the burden of disease.

Latent Tuberculosis Infection Correlations - The data indicates multiple instances of positive correlation among latent tuberculosis infection attributes, especially in specific age groups. This finding suggests that interventions aimed at iron supplementation in young children could be vital in combatting not only iron deficiency but also improving outcomes for those with latent tuberculosis.

Overall Conclusion - The correlations present in this analysis illuminate critical intersections between dietary iron deficiency and various health attributes, particularly within the context of tuberculosis and related diseases. These relationships emphasize the necessity for integrated nutritional and medical strategies to combat health issues in vulnerable populations, especially children under five. Further research is recommended to explore causative factors and potential intervention strategies.

B.3.3 Feature Importance Analysis Report

This report provides an analysis of the feature importance derived from various machine learning algorithms applied to predict dietary iron deficiency among children under five years in Asia for the year 2021.

Algorithms Overview - Four algorithms were utilized to analyze feature importance: Random Forest, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Neural Networks. Each algorithm has its methodology, resulting in differing views on the significance of features associated with dietary iron deficiency.

B.3.3.1 Key Findings in Feature Importance

1. Common Features Across Algorithms - Upon reviewing the feature importance outputs from the various algorithms, several features emerge as significant across multiple methods:

- **Drug-susceptible tuberculosis | Female | 70+ years -** This feature was highlighted in both Random Forest and KNN, indicating that tuberculosis in older females may have a strong correlation with dietary iron deficiencies.
- **Latent tuberculosis infection -** Appeared across different age groups and genders, hinting at its broad impact on nutritional issues.

2. Unique Findings by Algorithm - While there were commonalities, specific features showed unique importance in certain models:

- **KNN -** Importance was notably assigned to Hypertensive heart disease | Female | 70+ years, which suggests that chronic conditions in older females may lead to or exacerbate dietary deficiencies.
- **SVM -** Found lower importance in features related to tuberculosis, indicating its sensitivity to imbalanced data or overfitting.

3. Non-Obvious Discoveries - Interestingly, some features showed negative importance in the SVM model. For instance, Latent tuberculosis infection | Male | 5-14 years had a negative impact, illustrating that, in this context, the presence of this feature may not align with dietary deficiencies among certain demographics.

Correlation Observations - The correlation between features indicates potential overlapping health issues affecting dietary iron deficiency. There is a notable relationship between tuberculosis cases and dietary deficiencies, highlighting the compounded risk factors that should not be overlooked. Chronic health conditions such as hypertensive heart disease indicate a vulnerability which may necessitate integrated health interventions.

Conclusions - The analysis across these algorithms reveals critical intersections between infectious diseases and nutritional deficiencies, underscoring the importance of targeted healthcare strategies. Future studies should delve deeper into these correlations, particularly focusing on demographic vulnerability.

Recommendations for Future Models - To enhance predictive accuracy, it is advisable to include interaction terms within models to explore how combined health conditions influence dietary deficiencies. Additionally, refining the data quality and ensuring balanced datasets may improve feature validation.

B.3.4 Analysis of Confusion Matrices for Dietary Iron Deficiency Prediction

This report presents the analysis of the confusion matrices obtained from the application of various machine learning algorithms for predicting dietary iron deficiency among children under 5 years of age in Asia for the year 2021.

Neural Networks - The confusion matrix for Neural Networks revealed that the model performed accurately with 3 true positives and 4 false negatives from the first class. However, it struggled significantly with the second and third classes, indicating a high number of false negatives (16) relative to the true positives (6) in the third class.

This suggests potential misclassification issues which could be attributed to insufficient training data or overlapping features among classes.

Decision Tree - The Decision Tree model showcased a more balanced performance compared to Neural Networks, particularly in the fourth class where it recorded 16 true positives. However, the presence of false negatives in other classes (notably 6 and 2) reflects a tendency towards overfitting, where the model may have memorized training data patterns instead of generalizing well to unseen data. This overfitting can lead to underperformance in real-world applications, especially in a healthcare context where accurate predictions are crucial.

Random Forest - Random Forest algorithm exhibited a robust performance, particularly in classifying the fourth category with a notable 20 true positives and only 2 false negatives. The model also demonstrated resilience against overfitting, which is often a challenge due to its ensemble nature. Nevertheless, there were false positives in the first and third classes, particularly where it misconstrued instances leading to 8 false positives in the first class. This points to a possible issue with distinguishing between similar feature sets, warranting further investigation into feature engineering and selection.

K-Nearest Neighbors (KNN) - KNN showed identical performance metrics to the Random Forest model. However, this algorithm generally relies heavily on the feature space density, which may be problematic in healthcare contexts where high dimensionality and feature scaling are essential. The 20 true positives in the fourth category indicate excellent sensitivity, but its dependency on local data points for classification raises concerns about sampling bias that could influence its performance negatively.

Support Vector Machine (SVM) - The SVM model exhibited solid performance, especially in recognizing the fourth class but still suffered from false positives in classes 2 and 3. The model's linear nature might not capture complex boundaries accurately, leading to its inability to distinguish overlapping classes effectively. The imbalance between true positives and false negatives in this instance underscores the necessity for implementing kernel functions or refining the model's parameters through hyperparameter tuning.

Conclusion - In conclusion, while each algorithm presented varying strengths and weaknesses, it was evident that the Random Forest algorithm emerged as the most reliable for this specific context. The analysis revealed critical insights, particularly the tendency for some models to misclassify certain classes, highlighting the importance of ongoing feature analysis and enhancement of model tuning to ensure valid predictions in healthcare applications. Continued exploration of ensemble methods and advanced techniques such as XGBoost or Neural Architecture Search might further enhance predictive capabilities in future iterations.

B.3.5 Analysis of AutoML Results for Dietary Iron Deficiency Predictions

Introduction - This report provides a detailed analysis of the results from the AutoML process used for predicting dietary iron deficiency among children under 5 years in various regions of Asia for the year 2021. Multiple algorithms were employed, including Neural Networks, Decision Trees, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM).

Overall Observations - Each algorithm produced varying predictions, with notable discrepancies primarily in states like Jammu & Kashmir and Ladakh, Chhattisgarh, and Tripura. While some algorithms were more successful at correctly predicting the more severe case of dietary iron deficiency (value 4), others had notable misclassifications particularly in states that are socioeconomically diverse.

B.3.5.1 Algorithm Performance and Insights

Neural Networks - While this algorithm performed well in states where health and nutrition programs are robust, it struggled with states such as Chhattisgarh and Jammu & Kashmir, where socioeconomic factors may lead to difficulties in data collection and interpretation. The prediction failures may correlate with existing health disparities in these regions, reflecting more profound systemic issues related to iron deficiency.

Decision Trees - The Decision Tree algorithm showed strong consistency across most states, but misclassified certain states like Meghalaya, where unique cultural dietary habits and food accessibility issues may not have been accounted for properly, leading to underestimations of deficiency levels.

Random Forest - This model achieved extensive accuracy; however, it mispredicted Jammu & Kashmir and Ladakh, likely due to its geographical isolation and the diverse cultural traditions affecting dietary habits. The complex interplays of these factors were not adequately captured in the model.

K-Nearest Neighbors (KNN) - KNN performed almost identically to the Random Forest regarding the overall accuracy. Its errors in Jammu & Kashmir may stem from its reliance on distance measures that could overlook local nuances in dietary practices caused by isolation and economic constraints.

Support Vector Machines (SVM) - This algorithm provided consistent results but failed markedly in Maharashtra and Tripura, potentially linking to rapid urbanization or changes in local dietary patterns that the model did not sufficiently represent. The sociopolitical dynamics could cause fluctuations in food availability that were not captured in the historical training data.

Key Findings and Considerations - The correlation between algorithmic errors and regional characteristics suggests a significant impact of socioeconomic and cultural factors on dietary iron deficiency. Regions like Jammu & Kashmir face unique challenges due to isolation, while states like Chhattisgarh and Maharashtra experience rapid changes in urbanization that may confuse models trained on historical data.

Moreover, consistent successes in states like Haryana and Uttar Pradesh indicate better infrastructure and health intervention programs, demonstrating the effectiveness of comprehensive approaches in addressing dietary deficiencies.

Conclusion - This analysis indicates that while machine learning algorithms can provide valuable predictions, understanding the underlying socio-cultural and economic factors is crucial in making these models more robust and actionable. Stakeholders should consider these dynamics to improve model performance and address dietary iron deficiency effectively.

Recommendations - Further research into localized dietary practices and more granular socioeconomic data could enhance model accuracy. Inclusion of context-aware features and establishing feedback loops with health practitioners in the regions could also provide additional insights that improve predictive success.

B.3.6 Analysis of Selected Hyperparameters for Dietary Iron Deficiency Prediction

Introduction - This report provides an analysis of hyperparameters selected for various machine learning algorithms applied to the prediction of dietary iron deficiency in children under 5 years old in Asia for the year 2021.

Summary of Results - The hyperparameter selection was performed using a random search approach across multiple algorithms including Neural Networks, Decision Trees, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM). Below is a summary of insights gained from hyperparameters crucial for improving model performance.

Neural Networks - The selected configuration with 153 neurons, a dropout rate of 0.2, and ReLU activation provided the highest score. This indicates a robust capacity of the network to prevent overfitting while maintaining expressiveness. Utilizing RMSprop as an optimizer also emphasizes the efficiency of gradient descent when adapting learning rates.

Decision Trees - A maximum depth of 10 offered the best balance of accuracy without excessive model complexity, implying the data's inherent simplicity in structure allowing for simpler models to perform adequately.

Random Forest - A notable takeaway is the high number of trees (n_estimators = 150) combined with a moderate depth and a larger number of features considered (max_features = 8). This combination is uncommon, yet it produced superior performance, suggesting that increased diversity in the forest can lead to better generalization on complex datasets.

K-Nearest Neighbors (KNN) - The results highlight that using a low number of neighbors (n_neighbors = 3) significantly improves performance. This suggests that in health-related contexts, local patterns may be more indicative of outcomes compared to the broader neighborhood, contrary to typical assumptions in classification tasks.

Support Vector Machines (SVM) - The optimum configuration utilized a radial basis function kernel with a regularization parameter (C) of 1. These results point to an important insight: while complex settings can yield remarkable performance, tuning the regularization to a lower value can help sustain model generalization and prevent overfitting.

Conclusion - Through the analysis of the selected hyperparameters across various algorithms, significant insights emerged that can provide practical guidance for future predictive modeling efforts in health-related fields. A careful selection of model complexity and feature consideration, alongside optimization strategies, proved vital in enhancing predictive accuracy for dietary iron deficiency in the specified demographic.