A Machine Learning-Based Clinical Decision Support System for Mental Health Risk Profiling

Juan Francisco Paoli*, Parag Chatterjee and María F. Pollo-Cattaneo

GEMIS-GEMISBA, Facultad Regional Buenos Aires, Universidad Tecnológica Nacional, Buenos Aires, Argentina

Abstract

Clinical Decision Support Systems (CDSS) are increasingly being adopted to enhance healthcare delivery, particularly in mental health. This paper presents the design and implementation of a CDSS framework tailored for mental health-related data, focusing on predictive risk profiling and supporting healthcare professionals in data-driven decision-making. The system integrates machine learning algorithms for both classification and regression tasks, facilitating personalized risk assessments and treatment recommendations. It features a modular architecture, consisting of a data processing pipeline, machine learning engine, and an intuitive user interface, allowing for efficient handling of diverse datasets and seamless integration with existing clinical workflows. The system was tested on multiple open datasets, each requiring varying levels of preprocessing and data cleaning. Key results include the performance of models like Random Forest, Gradient Boosting, and K-Nearest Neighbors, and the significant impact of feature complexity over patient volume on processing times. Despite being deployed on mid-range hardware, the system achieved fast response times, highlighting its feasibility for real-time clinical use. The work underscores the importance of usability, performance efficiency, and interoperability in developing CDSS solutions, paving the way for broader adoption in mental health contexts.

Keywords

Clinical Decision Support System, Mental Health, Machine Learning, Risk profiling, Predictive Analytics

1. Introduction

A Clinical Decision Support System (CDSS) is an electronic tool designed to assess patient-specific information and provide treatment recommendations for healthcare professionals to consider during the clinical decision-making process [1]. Based on the level of user interaction and automation, CDSSs are categorized into three types: active, semi-active, and passive systems. Active CDSSs are automatically triggered and can occasionally make decisions autonomously without user input, whereas passive CDSSs require explicit user initiation.

In the context of mental health, the primary users of CDSSs are healthcare professionals, including therapists, psychologists, and psychiatrists. Patients, whose mental health status is being assessed, are indirect beneficiaries of these systems. Various studies have demonstrated that both clinicians and patients recognize the value of CDSSs and express interest in their continued development and potential future applications [2, 3, 4, 5].

Research into CDSSs for mental health has been ongoing since at least 1987 [1], highlighting a longstanding demand for tools that assist healthcare providers in clinical decision-making. Despite this demonstrated interest, efforts to advance CDSS technology for mental health have been insufficient, as existing systems remain underdeveloped and are not yet widely implemented in clinical practice. While publications on mental health CDSS are limited, those available have proven to be effective in supporting professionals in their decision-making processes [1, 2, 3, 6, 7].

A typical CDSS consists of three key components:

1. A database or knowledge base,
2. a server or semantic reasoner, where decision-support logic, such as data analysis and artificial intelligence, is executed, and

ICAIW 2024: Workshops at the 7th International Conference on Applied Informatics 2024, October 24–26, 2024, Viña del Mar, Chile

*Corresponding author.

jpaoli@frba.utn.edu.ar (J. F. Paoli); parag@frba.utn.edu.ar (P. Chatterjee); flo.pollo@gmail.com (M. F. Pollo-Cattaneo)

0009-0005-5077-4332 (J. F. Paoli); 0000-0001-6760-4704 (P. Chatterjee); 0000-0003-4197-3880 (M. F. Pollo-Cattaneo)

3. a graphical user interface (GUI), typically a web-based or desktop application.

When integrated with an Electronic Health Record (EHR) system—which digitally stores patient clinical records accessible to authorized users—this is often achieved through Application Programming Interfaces (APIs) [8].

Despite ongoing research, significant opportunities for improvement remain in CDSSs for mental health. These systems have yet to be deployed at scale, and much work is needed to enhance usability and treatment recommendation capabilities. This can be achieved by incorporating co-design methodologies with key stakeholders [1], fully integrating CDSSs with EHR systems [2, 9], analyzing comorbidities to predict mental health conditions [6], and enabling patient history tracking for improved clinical insights [6]. Additionally, there is a need for a standardized, user-friendly interface to promote broader research and application [1].

The current challenges in CDSS development for mental health include:

- A lack of standardized frameworks for system development and implementation.
- Gaps in the full development lifecycle of CDSSs, often resulting in poor alignment with user needs or system failure during implementation.
- Limited co-design methodologies, restricting user involvement in system development and reducing the clarity of CDSS objectives [1].

The aim of this work is to propose a comprehensive framework for a CDSS specifically designed for mental health applications. The primary objectives include the integration of the core CDSS components (GUI, server, and database), the ability to accommodate dynamic data during system operation, and the incorporation of machine learning models for data analysis. A key feature of this framework is a user-friendly interface that facilitates interaction with the system.

The system pipeline, representing a lower level of abstraction beneath the overarching three-component architecture, outlines how these functional elements interact to achieve the system's goals. This framework is designed to be as generic and modular as possible, supporting various database types and workflows, while maintaining flexibility for adaptation to different contexts. Ultimately, this system aims to contribute to the evolving research landscape of mental health CDSS, promoting broader adoption and facilitating further advancements in this nascent field.

2. Methods

2.1. Environment

To accomplish the objectives of this study, the platform was developed within a Jupyter notebook environment, described as "a server-client application that allows editing and running notebook documents via a web browser" [10], utilizing Python version 3.11.4. This environment was selected for several key reasons. First, Python offers extensive library support for data analysis and web development, making it highly versatile for handling complex computational tasks. Second, Jupyter notebooks offer enhanced flexibility in code execution compared to a standard Python interpreter, facilitating seamless integration of annotations, data visualization, and structured code blocks.

The primary libraries employed include:

- pandas and NumPy for data retrieval and preprocessing,
- YData Profiling for generating data summaries,
- matplotlib and Seaborn for data visualization,
- Flask for the web interface, and
- scikit-learn for implementing machine learning models.

The development and execution were carried out on a laptop computer equipped with an Intel Core i3-1005G1 (2x1.2GHz, 4 threads) CPU, Intel UHD Graphics (300MHz), and 8064MiB DDR4 2666MHz RAM Memory, running the Linux Mint 21.1 x86_64 operating system, with kernel version 5.15.0-83-generic.

2.2. Components

The CDSS was designed around three primary components, as outlined in the previous section:

1. **Dataset:** The patient data is stored in a .csv or .xlsx file, with each row representing an individual patient and each column representing a specific feature. The dataset is provided by the clinician, who acts as the system's direct user, and the data may be modified during system operation.
2. **Server:** The notebook is responsible for generating all output data and graphical user interface (GUI) views. Although these processes occur on the same physical server, they are logically distinct, as described further in the "Pipeline" section.
3. **GUI:** The GUI is a web-based interface, generated by the server within the notebook using Python libraries. It serves as the primary interaction point for the clinician. Detailed functionality of the GUI is also explained in the "Pipeline" section.

2.3. Pipeline

The pipeline describes the server's operation and the user's interaction with the GUI, serving as the backbone of the system. It offers flexibility in customization based on the specific requirements of the clinic utilizing the CDSS. The pipeline encompasses five main stages: data retrieval, initial data processing, server initialization, system output, and dynamic updates. Each of these stages is elaborated in the following subsections.

2.3.1. Data retrieval

The dataset is loaded into the server's memory using Python libraries that support both .csv and .xlsx formats. This process is initialized via a specified file path, making the data available for further processing.

2.3.2. Initial data processing

Upon retrieval, the data undergoes preprocessing, which includes removing unnecessary columns (e.g., patient IDs, timestamps, or any other irrelevant information). Subsequently, a data summary is generated and saved as a html file for further analysis. The summary can be accessed by the clinician through the web-based interface.

2.3.3. Server initialization

Following the completion of data processing, the server is initialized, making the web interface accessible via a standard HTTP protocol [11]. This setup allows the user to interact with the CDSS through a web browser. The system operates under a thin-client architecture, wherein the server handles all data processing, GUI rendering, and output generation, while the client (the web browser) is tasked solely with retrieving updated output files as necessary.

2.3.4. System output

The user interacts with the system via a web interface composed of three distinct sections accessible by scrolling. The system provides two primary outputs that support the clinical decision-making process:

- Exploratory Data Analysis (EDA): This segment aids in identifying patterns within the dataset by generating a comprehensive summary of all features, examining correlations between variables, and presenting data comparisons using various plot types tailored to specific data types.
- Machine Learning Models: The primary purpose of this section is to assist in clinical decision-making by leveraging AI models trained on the dataset to predict patient behavior and outcomes.

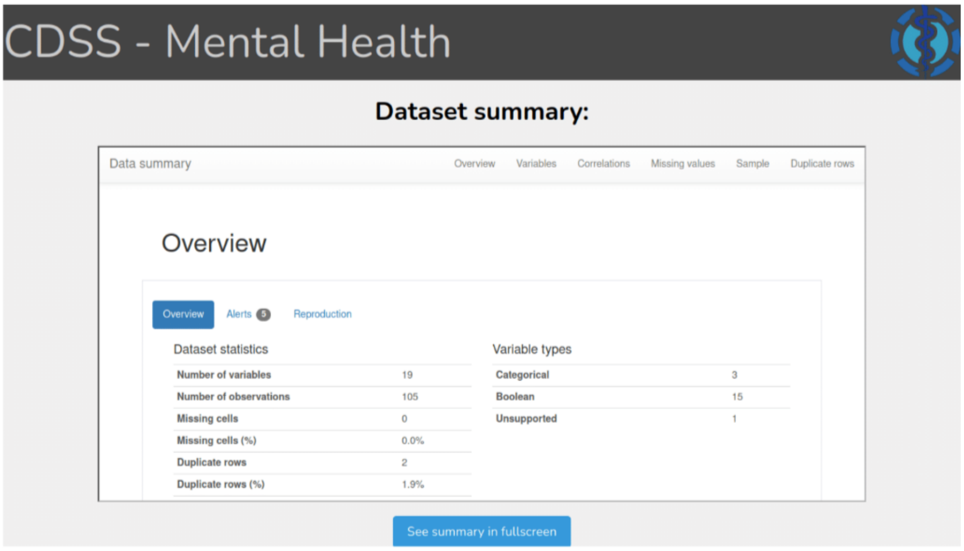

Figure 1: GUI through webpage, dataset summary section.

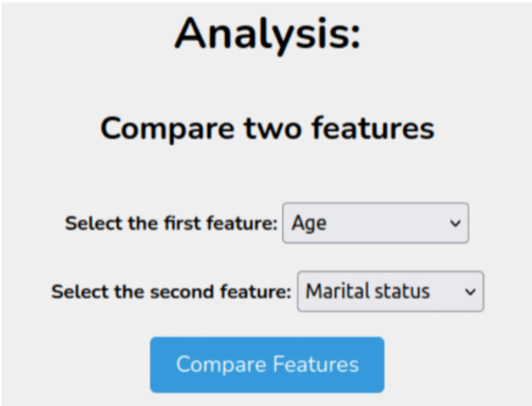

Figure 2: GUI through webpage, plotting section for dataset [12]

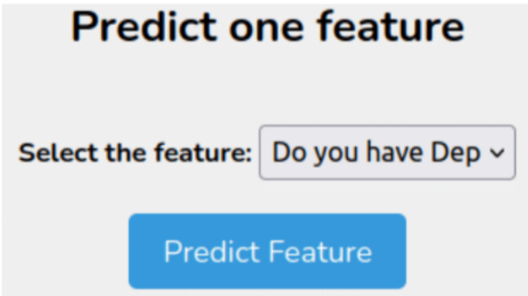

Figure 3: GUI through webpage, prediction section for dataset [12]

Since the summary generated is a .html file, it uses an HTML tag called "iframe" that allows the generated webpage inside the main one. It also has a button that takes the user to the path containing the summary in full size for better readability and usability. The plotting section allows users to select two features between the ones in the dataset, and after pressing the "Compare Features" button it takes them to a page that generates the plot for the features selected. On the other hand, the prediction section allows selecting a feature between the ones in the dataset, and after pressing the "Predict Feature" button it trains the model and takes them to a page for the prediction of the feature in new patients.

2.3.5. Dynamic updates

Due to the system's nature, the dataset may undergo modifications during runtime, necessitating dynamic updates to maintain the accuracy of decision-making. The modifications accounted for include:

- The addition of new patients' data (i.e., new rows in the dataset).
- Changes to existing data, which may involve updating one or more columns for a specific patient or multiple patients.

When such changes occur, the system responds by updating the summary and deleting any previously generated plots involving the modified columns. In the case of row additions or deletions, all previously saved plots are cleared.

If a user is viewing a plot that has been deleted due to dataset changes, the page will automatically refresh, prompting a new request to generate and display an updated version of the plot. This is achieved through an embedded script within the webpage that continuously monitors for deleted plots and resets the display as necessary.

Likewise, if the user is on the main page displaying the data summary, the page will refresh to present the updated summary. This mechanism ensures that the clinician always has access to the most current version of the data.

Machine learning models are also re-trained automatically whenever the page refreshes, ensuring that the models' predictions remain aligned with the updated dataset. This continuous retraining is essential for maintaining the validity of predictions, as the models' conclusions may evolve with the inclusion of new or modified data.

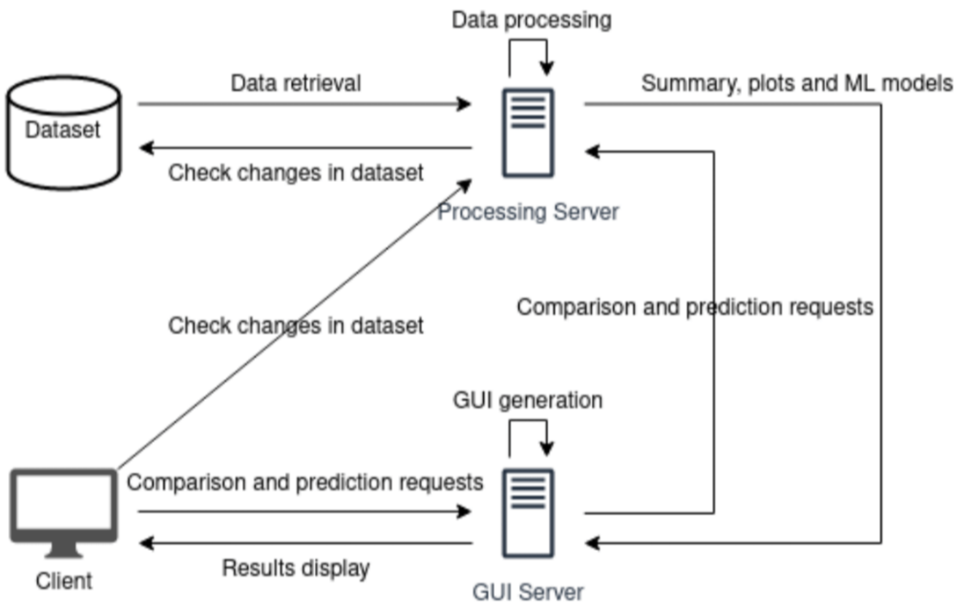

Figure 4: The system's pipeline, including data retrieval, server initialization, data processing (initial and ML models training), system output, and dynamic updates.

2.4. Exploratory data analysis

This section of the system output is designed to assist users in understanding the dataset and identifying relationships between features, thus supporting decision-making and improving patient diagnosis. To achieve this, the system provides both a data summary and plotting capabilities. Users can visualize the following:

- **General Overview of the Dataset:** The system displays the total number of features (variables) and observations (patients). It also shows samples from the first and last ten rows of the dataset.

Additionally, it identifies potential data issues, such as missing values, highly correlated features, and duplicated rows, through highlighted alerts.

- **Characteristics of Single Features:** For each feature, the system provides details such as the data type, categories (for categorical variables), mean value (for numerical features), and the range (minimum and maximum) of values. The distribution of feature values is also visualized.
- **Correlations Between Features:** The system displays correlations between features using both a heatmap and a table with the correlation coefficients for each feature pair. Additionally, users can compare two features through visual plots, offering more insight into the relationships between variables than a correlation coefficient alone. Depending on the data types, the system automatically selects the most appropriate plot type. For instance, if both features are categorical (e.g., binary features or features with multiple categories), a bar plot (or column chart) is generated. If one feature is numerical and the other categorical, a box plot is used to show the distribution of the numerical feature within each category.

2.5. Machine Learning models

As defined by [13], machine learning refers to the "aspect of artificial intelligence that competently performs automation in the process of building analytical models that allow machines to adapt independently to new scenarios, enabling software to successfully predict and react to the deployment of scenarios based on past results.". In the context of the proposed CDSS, supervised machine learning models are employed, where the system learns from a provided dataset to make predictions about future outcomes (i.e., the value of a target feature). The goal is to uncover associations between features that may not be immediately evident through exploratory data analysis alone, supporting healthcare professionals in predicting target feature values to enhance decision-making.

Depending on the data type of the target feature, the system categorizes the problem as either classification or regression:

- **Classification:** This method is used when the target feature is categorical. Classification models establish relationships between features through mathematical functions, predicting discrete values based on input features. Algorithms like Logistic Regression, Gradient Boosting, K-Nearest Neighbors, and Support Vector Machines are commonly applied in this context [14].
- **Regression:** This method is used when the target feature is numerical, and the goal is to predict continuous values. While regression is less frequently employed in mental health diagnostics—where many target features are categorical (e.g., "Does the patient have anxiety?")—it can be useful in predicting continuous indicators. For example, a Deep Neural Network with Multiple Regression could predict a set of continuous values indicative of depression, aiding in early diagnosis [15].

The steps for using the CDSS machine learning models are as follows:

1. The user selects a feature to predict.
2. The system trains multiple machine learning models on the selected feature.
3. The trained models are evaluated based on performance metrics.
4. The system presents the best-performing models to the user. For classification models, metrics such as F1 Score and accuracy are used, while for regression models, the mean absolute percentage error (MAPE [16]) is reported.
5. The user selects one of the models for future predictions.
6. For new patients, the system uses the trained model to predict the value of the selected feature based on the other feature values in the dataset.

2.6. Performance metrics

To assess the system's performance across datasets with varying numbers of features (columns) and patients (rows), several performance metrics were recorded, aimed at understanding how changes in dataset size affect user experience.

For each dataset, the following metrics were recorded:

- Number of features (columns) and number of patients (rows).
- Mean generation time for the data summary.
- Mean generation time for feature comparison plots.
- Mean training time for each machine learning algorithm.
- Accuracy (or other relevant metrics) for the machine learning models.

These performance values were automatically logged into a .json file after each system execution to facilitate further analysis.

3. Results

The system was executed on five publicly available datasets [12, 17, 18, 19, 20], all related to mental health in various contexts. Dataset [18] required some minor preprocessing before it could be processed by the automated pipeline. Specifically, the "comments" feature was removed, as analyzing individual strings would not yield meaningful correlations with other features. Additionally, standard data cleaning procedures were applied to correct inconsistencies in the dataset.

3.1. Exploratory data analysis

The data summary generated by the system produced varying results depending on the dataset. Below are examples of the output for different datasets:

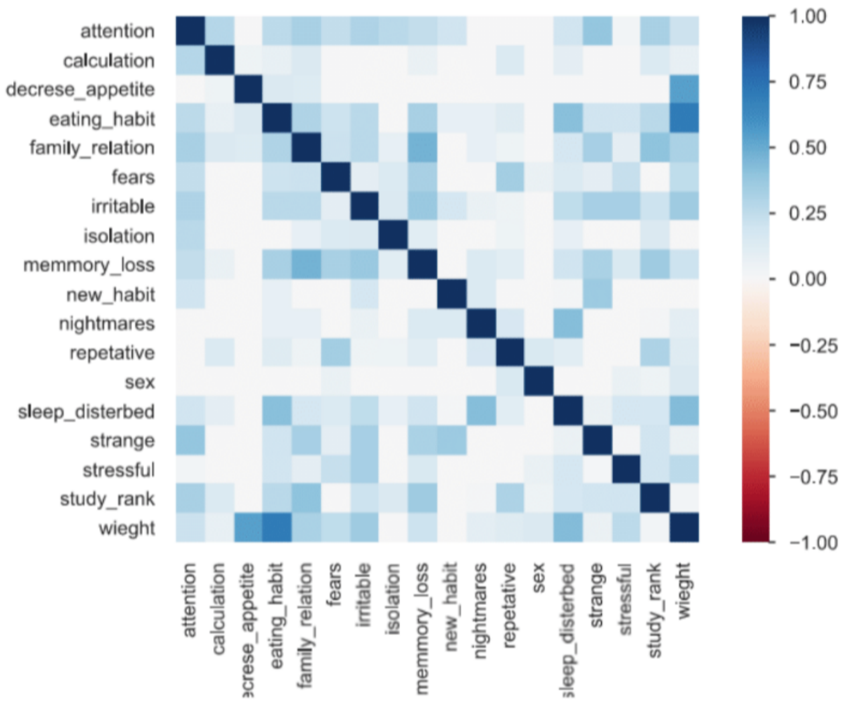

Figure 5: A heatmap displaying the correlation between features in dataset [19]. High correlations were found between "eating habit" and "weight," as well as between "nightmares" and "sleep disturbed," among other features.

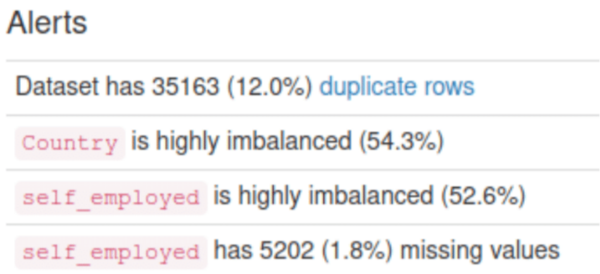

Figure 6: Alerts from dataset [17], highlighting highly imbalanced features, duplicated rows, and missing values. The feature "Country" was flagged as highly imbalanced because the survey data was predominantly from respondents in the USA.

In accordance with section 2.3.4, the system generated various types of plots based on the data types of the features. The following are examples of generated plots from different datasets.

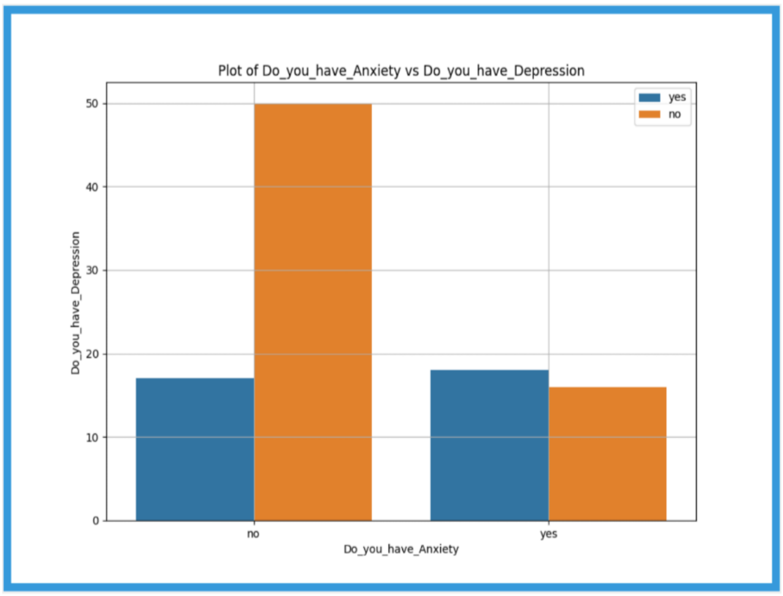

Figure 7: A bar plot showing the relationship between the features "Do you have anxiety?" and "Do you have depression?" from dataset [12].

From this plot, it is evident that individuals who suffer from anxiety are more prone to also suffer from depression, as the proportion of patients with depression is significantly higher among those with anxiety.

Figure 8 presents a plot that reveals a subtle association between the age of respondents and whether or not they have sought help.

Figure 9 presents a plot that clearly indicates a strong relationship between feeling burned out and becoming less interested in studies, demonstrating a strong correlation between these two variables.

3.2. Machine Learning models

The implementation of machine learning algorithms constitutes a crucial part of the CDSS pipeline, especially given the diversity and complexity of the datasets. The nature of mental health data often involves intricate patterns, making it challenging to rely on a predetermined set of models. For classification tasks, the models implemented in this study were Random Forest (RF), Gradient Boosting (GB), and K- Nearest Neighbors (KNN). For regression tasks, Linear Regression was selected as an initial approach within this CDSS framework.

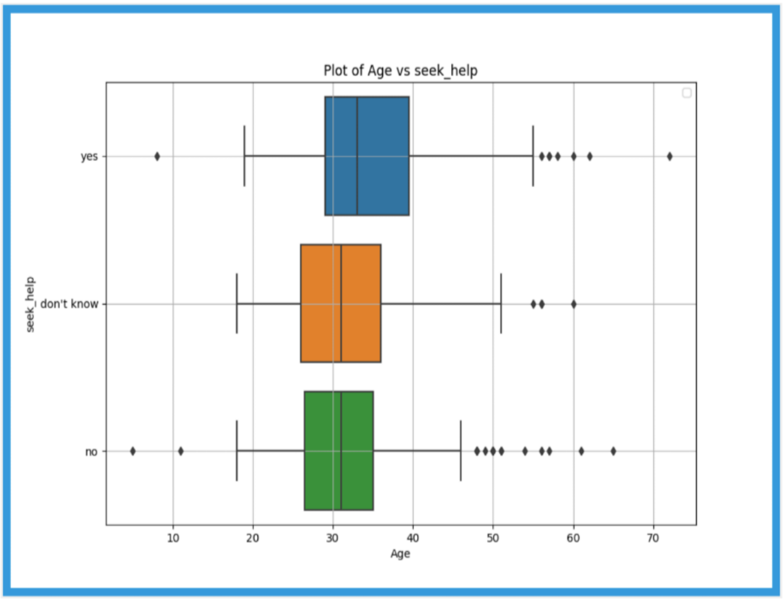

Figure 8: A box plot illustrating the relationship between the features "Age" (numeric) and "Seek help" (categorical) from dataset [18].

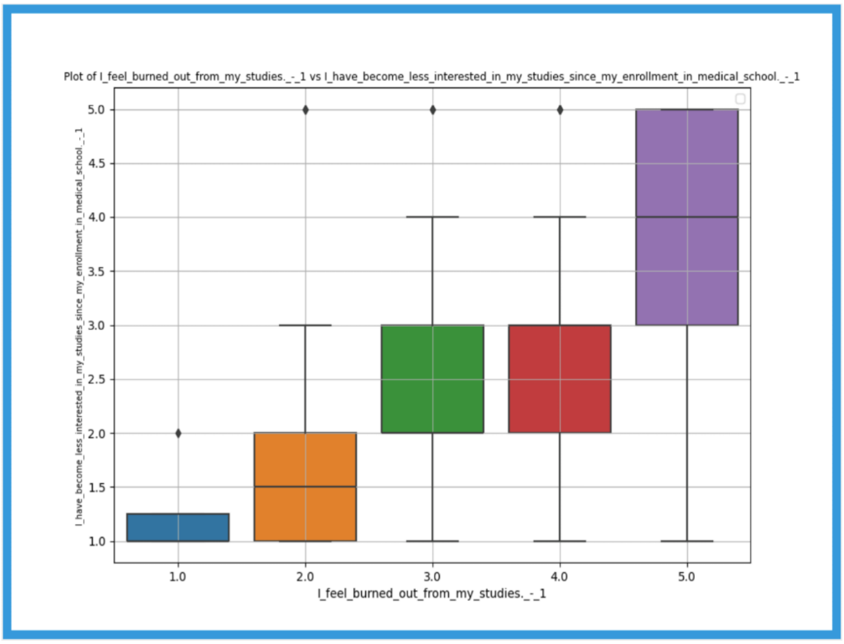

Figure 9: A box plot comparing the features "I feel burned out from my studies" and "I have become less interested in my studies since my enrollment in medical school," both from dataset [20] and both numeric features.

The page provides details about the selected machine learning algorithm and its associated accuracy score. It also allows users to input values for the other features in the dataset, which are then used to predict the target feature. If the target feature is categorical, the available values in the dataset are shown as options. For numerical features, users are allowed to manually input numbers (see Figure 10).

3.3. Performance metrics

The system's performance was evaluated by recording the generation times of various outputs, such as data summaries and plots, as well as the training times of machine learning models. These tests were conducted across all datasets to obtain consistent and reliable metrics.

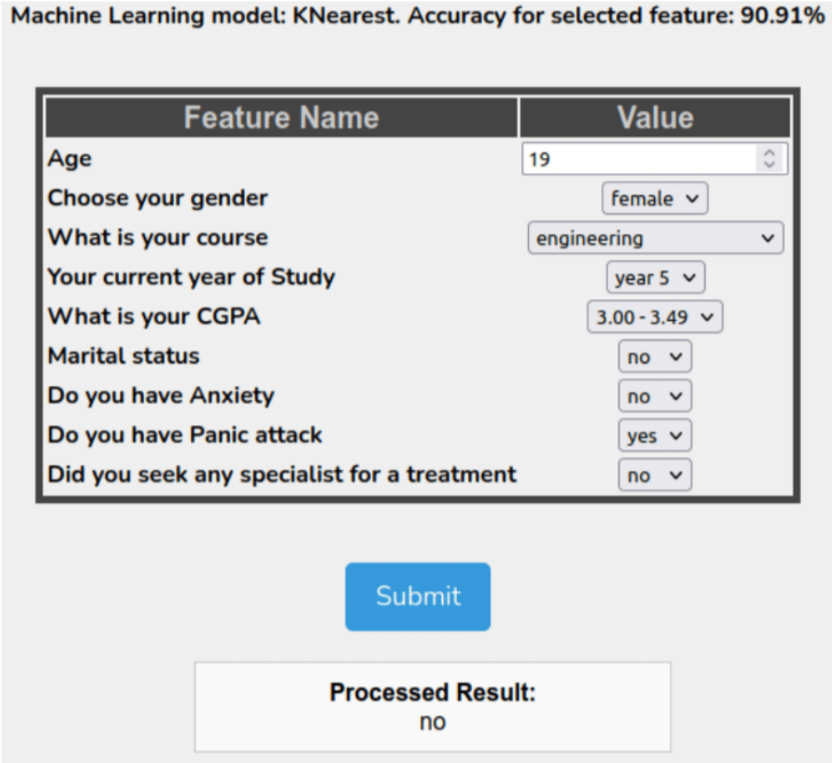

Figure 10: The GUI for Machine Learning predictions. This page is displayed after the user selects "Predict Feature" on the main interface.

- **Data Summary Generation:** For each dataset, the summary was generated 10 times to compute an average time for the generation process.
- **Plot Generation:** Different plots were also generated at least 10 times in total. The mean generation time of the plots was calculated by averaging the generation times for each dataset, without considering the specific frequency of each plot type.
- **Machine Learning Models:** Each machine learning model was trained 10 times on each dataset. Although the F1 Score was implemented and tested, the registered metric for performance comparison was accuracy. The overall system performance was largely dependent on the size and complexity of the datasets used.

After examining the results shown in Tables 1 and 2, several critical observations emerge:

- Impact of Features vs. Patients on Summary Generation Time: The number of features in the dataset appears to have a much greater influence on the summary generation time than the number of patients. This pattern is evident in datasets [17] and [20] (see Table 1).
- Impact of Patients on Training Time: On the other hand, the number of patients has a strong correlation with the training time of the machine learning models, demonstrating that datasets with more patients tend to significantly increase the model training time (see Table 2).
- Effect of Data Types on Plot Generation: The type of data, whether numerical or categorical, influences the plot generation time more than the number of features or patients. This is an important observation, as it indicates that the complexity of visualizations depends on data structure, not just dataset size (see Table 1).
- Algorithm Performance Consistency: No single machine learning algorithm consistently outperforms the others in either accuracy or training time across all datasets. This variability suggests that the choice of model should be dataset-specific, and a comparison tool within the system would be useful for selecting the most suitable algorithm based on specific performance metrics (see Table 2).

- Usability and Hardware Considerations: It's crucial to note that the system's response times were fast enough to enable efficient decision-making, which is a key feature for real-world applications. Moreover, the fact that the tests were performed on hardware without high-end specifications is promising, as it suggests that healthcare professionals should not face major hardware-related barriers when implementing this system, whether on local machines or central servers (see Tables 1 and 2).

Table 1
Registered sizes and exploratory data analysis execution times for each dataset.

Dataset	No. of features	No. of patients	Mean summary generation time (sec)	Mean plots generation time (sec)
[12]	10	101	4.54799	0.30589
[17]	16	292364	18.54937	0.58081
[18]	25	1253	11.51516	0.89916
[19]	19	105	5.93595	0.25060
[20]	79	242	47.52413	0.41671

Table 2
Registered sizes for each dataset and training times and accuracy for each algorithm.

Dataset and target	No. of features	No. of patients	ML model	Mean training time (sec)	Accuracy (%)
	10	101	RF	0.26105	81.81818
[12] "Do you have Depression?"			GB	0.06628	54.54545
			KNN	0.05086	90.90909
	16	292364	RF	21.48912	98.51216
[17] "Growing Stress"			GB	24.41467	99.04915
			KNN	167.61352	98.42323
	25	1253	RF	0.30493	62.69841
[18] "mental health consequence"			GB	1.84071	57.14286
			KNN	0.11396	50
	19	105	RF	0.12949	90.90909
[19] "irritable"			GB	0.09895	63.63636
			KNN	0.03609	72.72727

4. Conclusion

This work introduces a preliminary framework for a Clinical Decision Support System (CDSS) aimed at mental health, with the objective of providing predictive risk profiling and supporting data-driven decision-making for healthcare professionals. To effectively implement the system into daily clinical workflows, it is essential to collaborate closely with healthcare professionals and other key stakeholders. This collaboration will help define critical architectural aspects of the system, such as where the server will be hosted (for both data processing and the GUI) and the source of the database. These decisions must be customized for each clinic, as they are highly dependent on the available infrastructure. Furthermore, considerations around software quality, as specified by the ISO 25010 standard, must be addressed. Key factors include security, interoperability, and performance efficiency, all of which are vital for ensuring the system functions effectively in a clinical environment.

Future work could focus on expanding the types of data supported by the system beyond just numerical and categorical variables. In clinical contexts, multimodal models that integrate data types such as text, images, and audio could greatly enhance the decision-making process. Research has shown the value of incorporating such data for richer and more informed predictions. The ultimate goal is to enable healthcare professionals to access a unified database containing comprehensive patient data, significantly improving the system's usability and effectiveness.

In conclusion, this system, together with its successful primary implementation, highlights the critical role that data-driven insights can play in enhancing clinical decision-making, especially in the context of mental health. By leveraging such technologies, healthcare professionals may be better equipped to provide accurate and timely diagnoses, ultimately improving patient outcomes.

5. Acknowledgements

The authors acknowledge Cloudgenia for their technical assistance in this work.

References

[1] Y. Gu, A. E. Andargoli, J. L. Mackelprang, D. Meyer, Design and implementation of clinical decision support systems in mental health helpline services: A systematic review, International Journal of Medical Informatics (2024) 105416.
[2] C. Clausen, B. Leventhal, Ø. Nytrø, R. Koposov, T. B. Røst, O. S. Westbye, K. Koochakpour, T. Frodl, L. Stien, N. Skokauskas, Usability of the iddeas prototype in child and adolescent mental health services: A qualitative study for clinical decision support system development, Frontiers in Psychiatry 14 (2023) 1033724.
[3] C. Popescu, G. Golden, D. Benrimoh, M. Tanguay-Sela, D. Slowey, E. Lundrigan, J. Williams, B. Desormeau, D. Kardani, T. Perez, et al., Evaluating the clinical feasibility of an artificial intelligence–powered, web-based clinical decision support system for the treatment of depression in adults: Longitudinal feasibility study, JMIR formative research 5 (2021) e31862.
[4] M. Gunlicks-Stoessel, Y. Liu, C. Parkhill, N. Morrell, M. Choy-Brown, C. Mehus, J. Hetler, G. August, Adolescent, parent, and provider attitudes toward a machine learning based clinical decision support system for selecting treatment for youth depression, BMC medical informatics and decision making 24 (2024) 4.
[5] F. Velosa, H. Florez, Edge solution with machine learning and open data to interpret signs for people with visual disability., CEUR Workshop Proceedings 2714 (2020) 15–26.
[6] T. B. Røst, C. Clausen, Ø. Nytrø, R. Koposov, B. Leventhal, O. S. Westbye, V. Bakken, L. H. K. Flygel, K. Koochakpour, N. Skokauskas, Local, early, and precise: designing a clinical decision support system for child and adolescent mental health services, Frontiers in Psychiatry 11 (2020) 564205.
[7] P. Mortier, F. Amigo, M. Bhargav, S. Conde, M. Ferrer, O. Flygare, B. Kizilaslan, L. Latorre Moreno, A. Leis, M. A. Mayer, et al., Developing a clinical decision support system software prototype that assists in the management of patients with self-harm in the emergency department: protocol of the permanens project, BMC psychiatry 24 (2024) 220.
[8] S. A. Ajagbe, H. Florez, J. B. Awotunde, Aesrsa: a new cryptography key for electronic health record security, in: International Conference on Applied Informatics, Springer, 2022, pp. 237–251.
[9] J. B. Awotunde, S. A. Ajagbe, H. Florez, Internet of things with wearable devices and artificial intelligence for elderly uninterrupted healthcare monitoring systems, in: International Conference on Applied Informatics, Springer, 2022, pp. 278–291.
[10] A. Ingargiola, What is the jupyter notebook?, 2015.
[11] M. contributors, Http, 2023.
[12] M. S. Islam, Student mental health [dataset], 2023.
[13] R. P. França, A. C. B. Monteiro, R. Arthur, Y. Iano, An overview of deep learning in big data, image, and signal processing in the modern digital age, Trends in deep learning methodologies (2021) 63–87.
[14] J. Chung, J. Teo, Single classifier vs. ensemble machine learning approaches for mental health prediction, Brain informatics 10 (2023) 1.

[15] J.-W. Baek, K. Chung, Context deep neural network model for predicting depression risk using multiple regression, IEEE Access 8 (2020) 18171–18181.
[16] A. De Myttenaere, B. Golden, B. Le Grand, F. Rossi, Mean absolute percentage error for regression models, Neurocomputing 192 (2016) 38–48.
[17] B. Jikadara, Mental health dataset [dataset], 2024.
[18] E. Kim, Mental health in tech survey [dataset], 2024.
[19] I. Faeck Jaafar, S. Ahmed Mehdi, A.-H. Fawzi, Covid19 and student stressful [dataset], 2022.
[20] M. Jestin, Copy of mental health survey [dataset], 2021.