Project Progress Report: Machine Learning-Based Patient Risk Identification for Mental Health and Suicide Prevention

Prepared by: Idani Singo **Mentor:** Blessing Olorunfemi **Date:** [Compilation of dates from 28 May 2025 to 4 July 2025]

Introduction This report summarises the progress of the Machine Learning (ML)-based patient risk identification project for mental health and suicide prevention. It outlines the project's objectives, the iterative process of dataset identification and preparation, the development of the ML model, and the challenges encountered during deployment, along with the successful implementation of a Minimum Viable Product (MVP).

1. Project Context and Objectives The project aims to develop a **data-driven, machine learning-based tool to proactively identify individuals at risk of mental health deterioration or suicide**, providing early intervention support to medical professionals. This initiative is driven by both academic interest and personal experience with mental health challenges, aiming to enhance diagnostic accuracy, inform clinical decision-making, and support early interventions.

- **Problem Statement**: Traditional risk assessment methods are largely subjective, reactive, and reliant on self-reported symptoms or clinical judgment, potentially failing to identify high-risk individuals in a timely manner.
- **Target Audience**: The primary target audience for this tool is **medical professionals** (psychologists, psychiatrists, general practitioners). Secondary audiences include researchers, policymakers, institutions, government departments, and organisations interested in mental health advocacy.
- **Geographic Focus**: The project specifically focuses on a **South African context**.
- **Approach**: The approach involves using machine learning algorithms for classification or predictive modelling, with a consideration for incorporating Large Language Models (LLMs) for data augmentation, symptom pattern analysis, or early-warning insights based on clinical notes or unstructured text. Due to time constraints, the project was agreed to utilise secondary data sources. The tool is designed as a **diagnostic support tool for medical professionals**, explicitly avoiding direct-to-patient implementation to mitigate risks of misuse or misinterpretation.

2. Dataset Search and Exploration

The initial phase involved an extensive search for a suitable dataset, primarily focused on South African mental health data, to train and test the ML model.

- **Initial Search Strategy**: The search began using Google Dataset Search with terms like “South Africa mental health dataset,” “Depression anxiety dataset South Africa,” and “National Income Dynamics Study (NIDS)”. DataFirst, a local South African repository, was also explored.
- **Challenges in Dataset Identification**: It was challenging to find datasets with rich mental health-related features, particularly in the South African context, where data is

often scarce, fragmented, or not labelled usefully for machine learning. Health-related datasets often do not use "mental health" as a direct searchable label.
- **Refined Search Strategy**: The strategy evolved to include searching for variables like “depression,” “psychological distress,” “emotional well-being,” and “self-reported health” in larger national datasets such as NIDS, GHS, SANHANES, and surveys archived by SAMRC.
- **Discovery of Qualitative Dataset**: A significant discovery was a qualitative dataset titled **"Female undergraduate students’ individual experiences of mental illness stigma"** from the University of Pretoria Research Data Repository, authored by Maluleke, Mallory (Mahlori Faith Maluleke). This dataset resonated personally with the researcher due to its focus on mental health stigma, cultural influences, religion, and coping mechanisms among female undergraduate students in South Africa.
  - **Insights from Maluleke's Dissertation**: The dataset originates from an MA mini-dissertation (July 2023) using a descriptive phenomenological approach with semi-structured interviews of nine Black African female undergraduate students. Key themes included mental illness being perceived as "craziness and stupidity," the "pressure to feel strong," the idea that "others have it worse," and the prioritisation of "academics first, mental health second." The **role of culture in forbidding help-seeking** was a consistent theme.
  - **Utility of Qualitative Data**: While this qualitative dataset may not directly support predictive modelling, it holds significant contextual value for informing feature selection, model framing, or hypothesis generation, especially if future data collection or augmentation is pursued. However, the raw data (interview transcripts) is not publicly accessible, necessitating outreach to the author for access or discussion.
- **AI-Aided Summarisation**: To better understand and synthesise thesis material, AI tools were used. ChatGPT was employed to craft a structured prompt for NotebookLM, which then analysed documents and presented findings in a condensed, research-style format. This demonstrated the value of AI-supported research in extracting meaning from complex source materials.
- **Selected Dataset**: A publicly available **"Student Mental Health Dataset" from Kaggle** was identified and selected for the project's core ML model development. This dataset includes multiple features like sleep, academic stress, and depressive symptoms, and was downloaded in Excel format (Student Mental health.csv.xlsx). The refined project scope targeted depression and anxiety prediction among students, ideally with South African-aligned data, but open to broader student samples.

3. Data Preparation and Model Development

Upon securing the "Student Mental Health Dataset," the project proceeded with data preparation and model training.

- **Data Inspection and Cleaning**: The dataset contained 100 student records and 11 columns. Only one missing value was found and subsequently dropped, indicating the dataset was mostly clean.
- **Feature Selection and Risk Labelling**: Key mental health features selected included: "Do you have Depression?", "Do you have Anxiety?", "Do you have Panic

attack?", and "Did you seek any specialist for a treatment?" These were mapped from categorical Yes/No to binary 1/0 variables. A composite `Risk_Label` was created to identify students likely at risk, defined as having any of depression, anxiety, or panic attack AND not having sought treatment. The distribution showed 58 students at risk and 42 not at risk.
- **Model Training and Evaluation**:
  - A **Random Forest classifier** was initially trained, achieving 100% accuracy, which was likely due to overfitting the small dataset.
  - Subsequently, a **Logistic Regression model** was trained, yielding an **85% accuracy with good recall** (catching all at-risk students), deemed more realistic and trustworthy. This model serves as a reliable baseline for risk prediction.

4. Deployment Efforts and Challenges

The project transitioned to deploying the developed model, encountering significant challenges with initial platforms.

- **Initial Deployment Plan (Streamlit via GitHub)**: The initial strategy was to build an interactive Streamlit web application and host it on Streamlit Cloud, which requires the app to be linked to a GitHub repository.
- **GitHub Integration Issues**: Persistent issues were faced when attempting to connect the local project folder (C:\Users\idani\OneDrive\Desktop\DSA) to a newly created GitHub repository (`patient-risk-identifier`). Problems included Git not being initially recognised (resolved by installing Git) and subsequent remote path issues, preventing a successful push of files to GitHub. Despite following standard Git commands, no files appeared on the GitHub repository.
- **Streamlit Cloud Deployment Blocked**: As Streamlit Cloud deployment is contingent on GitHub integration, the inability to push files to GitHub directly blocked deployment to Streamlit Cloud.
- **Pivot to Replit (Temporary Workaround)**: Due to these unresolved GitHub and Streamlit issues, a **temporary workaround was proposed and implemented: switching to Replit**. Replit allows web-based execution without direct GitHub integration, providing a more immediate and accessible platform for launching and testing an MVP.
- **Successful MVP Deployment on Replit**: A **Minimum Viable Product (MVP)** version of the application was successfully developed using **Flask for the backend and simple HTML for the frontend**, replacing the earlier Streamlit plan. This pivot allowed for better control and easier deployment on Replit.
  - **Project Structure**: The Replit project structure includes `main.py` (Flask backend handling prediction requests), `templates/index.html` (frontend web form), `model/suicide_risk_model.pkl` (the trained ML model), `requirements.txt` (listing necessary packages), and `.replit` (for Replit configuration).
  - **Functionality**: All files were uploaded to a Replit project, dependencies installed, and the app successfully loaded and ran without errors. The frontend form collects four binary inputs (Depression, Anxiety, Panic Attacks,

        and Sought Treatment), which are fed into the loaded model by the backend, and the prediction is clearly displayed on the webpage.
    - **Current Status**: The app is now fully functional and accessible via a Replit URL, demonstrating an end-to-end working MVP.

5. Conclusion and Next Steps

The project has made significant strides from initial problem definition and dataset exploration to developing and deploying a functional MVP for patient risk identification in mental health. The iterative process of dataset identification, including leveraging qualitative insights and selecting a suitable quantitative dataset, has been crucial.

- **Current Achievement**: A working ML model (Logistic Regression) has been developed, and a Flask-based web application is successfully deployed on Replit, demonstrating core functionality for patient risk prediction based on student mental health data.
- **Immediate Next Steps**:
    - Continue debugging the Git push issue for eventual GitHub integration and Streamlit Cloud deployment for the full version.
- **Future Improvements**: Future work includes expanding the number of features, utilising a larger and more diverse dataset, applying rigorous model evaluation techniques (such as cross-validation), and improving the frontend design for a better user experience. The insights from the qualitative dataset can continue to inform feature selection or hypothesis generation for future data collection or augmentation.

Amendment: Development Process and Application Framework Update

During the development of the Suicide Risk Predictor project, the initial application prototype was implemented using **Streamlit** for rapid interface development. The model employed is a logistic regression classifier trained using **scikit-learn** and saved with `joblib` in the file `suicide_risk_model.pkl`.

To meet the project submission requirements, the application was adapted from Streamlit to a **Flask-based web application**. This involved:

- Rewriting the frontend interface as a traditional HTML form served by Flask.

- Implementing Flask routes to handle form display and prediction requests.

- Ensuring the `suicide_risk_model.pkl` file is loaded correctly within the Flask app for inference.

This migration enables easier integration with standard web deployment environments and aligns with the submission guidelines.

The model file remains in the `.pkl` format as this is the standard and appropriate serialization format for scikit-learn models. Conversion to `.pth` (PyTorch) or `.h5` (Keras) was deemed unnecessary since the model is not a deep learning model.

The final project package includes:

- The **Flask application source code** (`app.py`)
- The **HTML template** (`templates/index.html`)
- The **trained logistic regression model** (`suicide_risk_model.pkl`)
- The **dataset or a link to the dataset** (as per project instructions)
- The **original Jupyter notebook and exported HTML report**

This structured approach ensures that all functional requirements are met, and the project is packaged consistently for evaluation.