Final Project Report

Title:
Early Identification of Distress Signals for Students for Promoting of Mental Health Using Random Forests Classification

1. **(5pts) Honor Code and LLM Usage for this Report.**
   We have not given or received unauthorized assistance on this assignment.

   The usage of large language models in making this proposal was minimal, only being used to improve the legibility of text and grammar.

2. **(10pts) Learning Objectives:**
   List the learning objectives from your proposal. In your own words explain whether you met those objectives and how (50-100 words each objective). Also describe if you learned something different than expected or anything additional.

   - **Conceptual Understanding:** Learn about random forest classification as a classic machine learning algorithm
     - We met the objective of learning about random forest classification. We first investigated decision trees to gain a deeper understanding of how the random forest classifies. We used scikit-learn to implement the random forest classifier, and we were able to test results using a random state and other scikit-learn functions. This includes accuracy, confusion, and other statistical metrics.
   - **Conceptual Understanding:** Understand the inference and predictability of machine learning models.
     - We met the objective of understanding the inference and predictability of machine learning models. We used a public dataset to train multiple models using different features. Our first model has an average accuracy of around 80%. This gives the model a higher chance of a correct classification. Our second model has a lower average accuracy of around 50%. This model was tasked with a harder classification, and so the model has a lower chance of a correct classification.
   - **Conceptual Understanding:** Understand the impact of the dataset properties into the problem resolution.
     - We met the objective of understanding the dataset properties in relation to the problem. We initially chose a dataset through Kaggle. This dataset contained 18 columns of data ranging from location, age, gender, profession, and degree. However, after further investigation we noticed

the data demographics were mainly from cities in India. Some label categories, such as the degrees, were specific to India. Additionally, the dataset demographics affects who would benefit from the classification algorithm trained on it. To address this, we chose a city-agnostic dataset with more relevant column labels.

- **Skill development:** Develop proficiency in the application of machine learning algorithms into real world datasets.
    - We met the objective of developing proficiency to use real world datasets in machine learning algorithms. In the project, we developed a random forest classifier algorithm using scikit-learn. We used a real-world dataset available online. Before we trained the model, we preprocessed the data using pandas. This step allowed us to use categorical data with a numerical model.
- **Skill development:** Acquire experience working with machine learning libraries such as scikit-learn in Python.
    - We met the objective of acquiring experience working with machine learning libraries. For our use case, scikit-learn was an appropriate choice for a traditional machine learning approach. We implemented a random forest classifier through scikit-learn, utilizing built-in functions to train and test the model. We also used the pandas library to process and store our data.
- **Skill development:** Achieve the ability to critically evaluate the performance of a machine learning model
    - We met the objective of achieving the ability to critically assess a machine learning model's performance. After training the model for the first time, we got accuracy between 85% and 95%. We experimented with different numbers of trees in the model and noticed that there was not a large difference between 1000 trees and up. After removing one of the initial features, we obtained lower accuracy. This feature may have more of an impact on model performance than we thought.
- **Skill development:** Gain insight into the process of passing from code to a working website.
    - The process of developing the website gave our team a first introduction in the making of websites through Python libraries like streamlit and gradio, achieving a functional hosted website. It also introduced basic CSS usage and website structuring concepts. As such, we consider the objective of gaining insight into the website development process as met.

3. **(10pts) Timeline:**
   Outline how you spent time on your project. Break down the time into specific tasks or milestones. Here is an adjustable schedule to get you started. Actual Details should be 50-100 words each and should compare or reflect on differences from your proposal.

Time	**Task**	**Expected Details from Proposal**	**Actual Details**
Hour 1-2	Research and gather resources	Reviewing classification trees and random forest slides to understand key concepts. Explore tutorials and templates for building websites, focusing on AI project deployment. Collect relevant research papers to understand best practices and relevant developments. Identify datasets and external resources necessary for the project implementation. This stage ensures a solid foundation and references materials to guide coding, data analysis, and deployment efficiently.	We reviewed the classification and random forest slides during a team meeting when deciding what algorithm to use. We referenced online tutorials for building the algorithm and creating the website. We did not end up collecting research papers as presented in our proposal. We considered a couple of datasets on Kaggle before deciding on the final one.
Hour 3-4	Design the project structure and plan	Plan the project workflow, including data, analysis, model training, and evaluation. Outline milestones for each step to maintain	We held a team meeting after gathering resources to structure the plan. We assigned tasks to each member and a goal to reach by the next meeting. We did not draft a

Time	Task	Expected Details from Proposal	Actual Details
		clear progress. Draft an AI deployment strategy for website, considering backend and frontend integration. Decide on tools and frameworks such as Python with Scikit-learn, Node.js and React. This planning sets a structured roadmap to reduce errors and improve efficiency during coding and implementation.	specific AI deployment strategy, but at this point, we were considering how to deploy the website. We decided on using scikit-learn and pandas at this stage.
Hour 5-6	Start coding the basic functionalities	Building the main project features. Write initial scripts to explore the data. Train basic models using Scikit-learn, making sure the results are easy to understand. Create a GitHub repository and set up Node.js React for the web interface. Focus on making a working example, even very simple, to serve as a base for adding more features later.	We followed most of the details initially proposed in this section. During our meeting to review progress, we showed our initial scripts. We developed a random forest classifier with statistics and comments to understand the code. We created a working local version of the website in gradio, and we had code to present personalized advice to users. During this meeting, we also set up a GitHub repository.
Hour 7-8	Test and debug the initial version	Improve the website by adding visuals like graphs and charts.	We achieved a functional working local prototype of the website in streamlit with the basic expected interface

Time	Task	Expected Details from Proposal	Actual Details
		Show how well the model works and presents the results clearly on the site. Connect the frontend and backends smoothly and include AI deployment features. Use user feedback to make the models and code better. This step makes the project easier to use and more functional.	characteristics detailed in the proposal. Later on, a connection between the models and the views would be implemented. The app performance was also improved by storing the ML models into cached .pkl files. Finally, the advice model was refined by removing unnecessary flask dependencies.
Hour 9-10	Refine and add advanced features	Test the website and AI functionalities for bugs and performance issues. Document code and results clearly. Ensure that GitHub repository is organized and includes instructions for reproduction. Make final adjustments to visualizations, layout, and deployment scripts. This final stage prepares the project for submission.	The team improved existing code by adding relevant information to the readme, updating the repository structure, and adding comments. On the website side, improved replacement charts were introduced. The Advice module of the app was also implemented to give polled people advice on how to start changing their habits. A new model was also introduced to predict suicidal thoughts of risk. (though with lesser accuracy than before for depression)

4. **(55pts) Final Product Description:**
   Include your proposed MVP, Target, and Reach versions.
   i. Minimum Viable Product (MVP): The most basic version of the project will be a simple website (hosted by a third-party service) offering a short survey for

students of different ages. Once finished, the website will show a simple true – false result showing if the user is suffering depression or not based on the dataset, and where to find help if needed

ii. Target Product: For this version we aim to achieve a prettified user-friendly interface, offering visitors an extended survey. The questions will be classified into different sections, and transitions between sections will be added. Once finished, the website will show the probability of suffering depression, anxiety, or mental problems based on the answers. A chart explaining the relation of the user in relation to the data will also be included

iii. Reach Version: This version aims to offer personalized guidance based on the survey using openai API models and guardrails. The visitors will be able to respond to multiple surveys of their choice, targeted to different audiences and conditions. Upon completion, they will have a deep dive into the data of similar profiles thanks to the dataset usage and RAG reinforced LLM models.

iv. (20pts) Description of final product including target audience, user story, problem statement, key features, technical details and technologies used. (100 – 150 words).

The final product consists of a running website hosted through the Streamlit community cloud. The website collects diagnostic data from the user to predict depression and suicidal thoughts, and personalized feedback is given. The target audience is students aged 18-35 who want an early detection and/or advice for the predictions above. Our problem statement is to aid in addressing declining mental health in students. The user interacting with the product receives personalized advice to improve their mental state and additional resources to contact if needed. Our key features include depression prediction, suicidal thought prediction, and tailored advice. We utilized machine learning libraries such as scikit-learn and pandas and application libraries such as streamlit, flask, and gradio.

v. (20pts) Link to a video demo of product. This should be 1-2 minutes and narrated. Level of difficulty and detail of the project should be reasonable for 10 hours of work with LLM support. The project should not be something an LLM can solve without significant effort by the developer. (Be sure to have someone else test that your link is working).

https://youtu.be/kceowfLu7RA

vi. (15pts) Any input files, coding files, test files should be uploaded. Provide a list here of file names and purposes, or any links to live sites or artifacts. Remember code should also be commented. A README file should be created and uploaded so that we have the option to follow your instructions to run your project.

Project Repository & Code Submission Details:
Project Repository: https://code.vt.edu/gracee/vgc_group11

- **code/** – All source code files for the project (organized by component).
- **data/** – Input files, datasets, or configuration files used by the program.
- **tests/** – Test scripts or files demonstrating how the code was verified.
- **docs/** – Supporting materials such as screenshots, reports, or documentation.
- **report/** – This final report document.
- **README.md** – A detailed file describing:
  - Project overview and purpose
  - Video link of your project
  - Installation and setup instructions
  - How to run the program and reproduce results
  - Technologies or libraries used
  - Author(s) and contribution summary

Required:

- Maintain a logical directory structure, do not store all files at the root level.
- Include comments in your code to explain logic and design decisions.
- Keep your repository **private** until grades are released, then you may make it public.

Share access with the following personnel (Add them as collaborators):

GTA Name/Prof. email	Section	Professor
Yue Shen	83485	O. Emebo
Abdullah Al Noman	83486	O. Emebo
onyeka@vt.edu	All(VT)	O. Emebo
driofrioa@usfq.edu.ec	USFQ	Daniel Riofrío

5. **(10pts) Consultation and Use of LLMs:**
   Each student must create a unique project but is allowed to consult with other people and use Large Language Models (LLMs). Describe how you incorporated these resources into your project:

- **Consultation Description:**
  Describe how you ended up seeking advice or feedback from peers, mentors, or online communities.

  During the development of this project, our team worked together to divide the tasks and combine our parts. We had weekly 2-3 meetings to share our progress, talk about the next step, fix problems, and make sure all our parts worked well together. We gave each other feedback by looking at each other's works and we discussed the suggestion to improve the project. For guidance, we used course materials and lecture slides which helped us to understand the main method that we used. We also used the Depression Student Dataset from Kaggle (https://www.kaggle.com/datasets/ikynahidwin/depression-student-dataset) for our model. Through the project, we regularly communicated often, shared ideas, and helped each other to complete our parts.

- **Use of LLMs:**
  Explain how you ended up utilizing LLMs to assist with coding, debugging, learning technologies and concepts, or generating ideas

  During the preparation of this assignment, Carlos Flores used Google Antigravity in the implementation of interface parts and improvement of the code to speed up the implementation and debugging of the website. After using this tool, I reviewed and edited the content as needed to ensure its accuracy and take full responsibility for the content in relation to grading.

  During the preparation of this assignment, Grace Marrone used Microsoft Copilot when implementing the machine learning algorithm to determine if and how the data should be encoded. After using this tool, I reviewed and edited the content as needed to ensure its accuracy and take full responsibility for the content in relation to grading.

  During the preparation of this assignment, Esin Karapinar used ChatGPT in backend development, learning Flask, debugging, to understand the tools better and solving technical problems. After using this tool, I/we reviewed and edited the content as needed to ensure its accuracy and take full responsibility for the content in relation to grading.

6. **(10 pts) Ethical Considerations:**

Ethical considerations by documenting *data provenance and consent (1)*: specify what data you used, ownership, permissions to use/share, presence of personal data, consent obtained, anonymization/pseudonymization, and retention policy. Explain *privacy and security risks (2)*: which sensitive attributes might be exposed and how you protect data and artifacts (access controls, secret management, redaction). Assess *fairness and bias (3)*: identify groups that could be disadvantaged and describe checks and mitigations (sampling reviews, baselines, group metrics, error analysis). Evaluate *misuse and safety (4)*: outline potential dual-use harms (surveillance, harassment, fraud) and safeguards (restrictions, rate limits, filters, warnings, scope limits). Provide *transparency and accountability (5)*: what users can know (sources, limitations, failure modes), include a Model Card and Data Datasheet, and name maintenance responsibility. Consider *cross-cultural and accessibility (6)* factors across the U.S. and Ecuador (language, bandwidth, norms, laws), and inclusive design. Clarify *intellectual property, licensing, compliance, and attributions (7)*. Optionally, address compute and sustainability: model size, costs, and impacts.

i. Completeness & Specificity (4 pts): Addresses at least four areas with project-specific details (not generic statements).
ii. Risk Analysis & Mitigation (4 pts): Identifies realistic risks and documents at least one implemented mitigation with evidence.
iii. Cross-Cultural Insight (2 pts): Explicitly compares implications in U.S. vs. Ecuador and notes accessibility choices.

During this project, our team considered ethical issues related to data, privacy, fairness, and safety. We sourced our data from the Depression Student Dataset on Kaggle, an open resource that contains no personal information. Because mental health attributes is naturally sensitive information, we considered privacy and security risks and were careful to ensure no identifiable information was collected or stored. Sensitive attributes include depression indicators, sleep habits, and lifestyle patterns. They are protected through HTTPS encryption, in memory processing, and secure secret management.

We considered fairness and bias in our dataset. We originally chose a dataset including city data. After further investigation, we realized this dataset was local to India and included location specific features. We ethically considered if this data could be used to predict depression in people from the United States and Ecuador. We determined that there are numerous lifestyle differences between these two locations. Factors that may point to depression in India are not the same across the globe. We decided to use city agnostic data with features relevant to our audience. Overall, this occurrence made us realize the limitations and ethical challenges of trying to globalize LM models, as their

predictions and realized inferences rely on the data they were trained on. As such, careful consideration must be put when defining the target audience, as a depression detector miss calibrated for a region can be very harmful.

We aim to be transparent and accountable through various measures. Mental health and its prediction is a sensitive task. We provide disclaimers throughout the website notifying the user that our model was trained with AI and does not replace professional diagnosis. All our code is open source. Users can see how the model was trained, what parameters were used, and how advice was generated. By doing this, we hope to provide transparency and accountability.

Despite transparency and accountability measures in place, we ethically consider how and if they are effective. Users may not have knowledge of AI techniques or how a model is trained, even if we inform the user that machine learning was used. Users may not have the technical ability to access and view the code, even if it is open source. Users may feel confused about how predictions are being made if they don't see the formulas behind them. We keep our code open source knowing this, and we provide disclaimers throughout the website to remind the user of the limitations of the product.

User survey data was processed in memory only and not saved, so no name or other personal details were stored. Users gave consent by choosing to complete the survey, and disclaimers made clear that the tool is for only educational purposes.

Cross-Cultural Insight

During the realization of the project, we also realized how the cultural differences of both countries heavily influence the reliability of our website depression predictor: At a certain point in the project Carlos, the Ecuadorian member of the team, started writing comments in Spanish by accident. This resulted in a difficult time for the rest of the team trying to understand what the code was and how it worked. At the end the comments were quickly translated, and all remained as a funny accident, but this situation highlighted how linguistic differences between the countries can lead to a lack of understanding of the other side.

This same realization can be applied to the implemented project, and not only on a linguistic level, but a cultural one as well. After all, terms like "Academical pressure" "Economical pressure" and "Academical Satisfaction" can be perceived very differently between countries, as they have significantly different views. In trying to mitigate this, we used country agnostic data to make our model as plural as possible, but ultimately, it's impossible to create a completely bias-free dataset.

The main conclusion of all this is that depression and suicidal risk should be diagnosed and treated in the context of the respective location, and that a globally trained ML

predictor model, while potentially useful for a first impression, is not suitable for making autonomous, reliable classifications of real patients.