ITAI 1371

Ashton Smith
Emmanuel ABABIO
Erwing Cheng
Khuliso Mukwevho

Midterm

Emmanuel: In this project, I worked with the dataset **“Datasetapproved.csv”**, which contains medical records from **1,025 patients**. My goal was to clean, prepare, and explore the dataset using Python modules such as **NumPy**, **Pandas**, and **Scikit-learn**. Through this project, I learned how these libraries help handle data efficiently and transform raw data into a form suitable for analysis or machine learning. I discovered that **Pandas** was very helpful for loading, inspecting, and cleaning the dataset. I learned how to remove duplicates, handle missing values, and create new columns. **NumPy** was useful for mathematical operations, especially when defining numeric conditions and filling missing values with medians. **Scikit-learn** provided tools for preprocessing, encoding categorical variables, and scaling data using techniques like *StandardScaler* and *MinMaxScaler*.

Although I faced some struggles when coding the part that was supposed to clean the data, especially when trying to properly handle missing values and column encoding, these challenges helped me better understand how data preprocessing works in practice. Through this experience, I learned how to use **Pandas** to clean and organize data, apply **NumPy** for numerical processing, and build **Scikit-learn** pipelines for data transformation. The final result is a cleaned and well-prepared dataset that can now be used for further analysis or predictive modeling.

Khuliso: The dataset underwent a comprehensive cleaning and preprocessing workflow, which involved addressing missing values using appropriate imputation strategies, eliminating duplicates, and engineering new features such as age categories. Categorical features were transformed using one hot encoding, while numerical features were scaled and normalized to prepare the data for machine learning algorithms. These operations were implemented using Python’s key data science libraries, pandas enabled efficient data manipulation, inspection, and feature creation; **NumPy** provided high performance numerical computations for feature interactions and array based operations, and scikit-learn allowed for a reproducible and modular preprocessing pipeline, including imputing missing values, encoding features, scaling and normalization of numeric variables, and integrating all steps into cohesive pipelines. An initial evaluation using a Logistic Regression model produced a perfect accuracy of

100%. While this result may initially appear ideal, such flawless performance is highly improbable in real world datasets, strongly suggesting the possibility of data leakage or an error in the train test split. Therefore, a careful review of the entire data preparation and modeling pipeline is essential to identify and rectify any issues before proceeding with further model development or interpretation.

Ashton: In this experiment, I learned the difference between looking at an unclean dataset versus a cleaned one. For this assignment, I helped prepare and analyze the heart disease dataset for this project. There weren't any obvious missing values, but there were rows that turned out to be duplicates that didn't seem like duplicates at first because of the wording, along with some redundant and irrelevant information that needed to be reduced. Using Pandas and NumPy made it easier to organize the data, drop unnecessary columns, and make sure everything stayed consistent. We also used Scikit-learn for some processing and scaling data. The charts and graphs helped us see how variables like maximum heart rate, chest pain type, age, and exercise-induced angina related to heart disease risk. Cleaning the data sounds simple, but it gets tricky fast because you have to decide what's actually useful and what just adds noise. Once we trimmed it down to the key features, the dataset looked cleaner, made more sense, and felt ready for accurate predictions.

In conclusion, when we ran the Logistic regression model, it showed perfect accuracy, which I'd say is a red flag. Data leaks or other such things between training and testing. It's just a nice reminder that high accuracy doesn't always mean perfection. It means we need to pay more attention to how the data was handled.

Erwin: Throughout this project, We focused on transforming an unstructured and inconsistent dataset into a clean, well-organized, and analysis-ready resource. The dataset contained several health-related variables, but it was initially messy, with missing values, inconsistent data types, and unnecessary columns that made it difficult to work with. Our goal was to enhance the data's quality and usability so it could support accurate analysis, visualization, and predictive modeling.

The first step we took was to assess the dataset's overall condition using exploratory analysis tools like info(), describe(), and head() in Python's pandas library. This allowed us to identify missing data, redundant columns, and formatting inconsistencies. From this analysis, We noticed that some columns, such as fasting_blood_sugar, rest_ecg, *and* vessels_colored_by_flourosopy, were not providing much value to the overall analysis. We made the decision to remove these columns to streamline the dataset and focus on features that had real analytical importance.

Once we identified the problem areas, We began cleaning and transforming the data systematically. We handled missing values by filling them with appropriate averages or most frequent values, ensuring no gaps would distort future results. Then, we standardized numerical features using StandardScaler and MinMaxScaler so all variables could be compared fairly. For categorical data, We applied OneHotEncoder to convert text labels into numerical values that machine learning models could interpret. We also built a preprocessing pipeline using scikit-learn's tools, which automated these steps and made the process repeatable for future data updates.

By the end of the project, We had successfully turned a messy dataset into a reliable, structured, and well-balanced version that was ready for analysis. The cleaned data is now easier to visualize, interpret, and use for predictive modeling, particularly in identifying health risk patterns. This project not only improved the dataset's technical quality but also deepened my understanding of real-world data cleaning, feature engineering, and preprocessing. All of our contributions ensured that the dataset can now support meaningful insights and data-driven decision-making.