**Diagnosify**

Vansh Saxena(E22CSEU1194)

Yushita Kalra(E22CSEU1223)

Kashish Khurana(E22CSEU1204)

Rajdeep Mohanty(E22CSEU1202)

A report submitted in part fulfilment of the degree of

**BTech in Computer Science**

**Supervisor:** Dr. Satyam Omar



Department of Computer Science

Bennett University

**Motivation**

* Early Intervention and Prevention:

The primary motivation for developing an app that predicts diseases based on symptoms is to enable early intervention and preventive measures. By identifying potential health issues at an early stage, users can take proactive steps to manage their health and prevent the progression of certain diseases.

* Health Awareness:

The app aims to increase health awareness among users by providing them with insights into their symptoms and potential underlying conditions. This knowledge empowers individuals to make informed decisions about their health and encourages a proactive approach to well-being.

* Reducing Healthcare Costs:

Early detection and home-based management of certain diseases can contribute to a reduction in healthcare costs. By offering users a tool to assess their symptoms and receive initial guidance on at-home care, the app seeks to minimize the need for unnecessary hospital visits and medical expenses.

* Accessibility to Information:

The app aims to make healthcare information more accessible to a wider audience. By leveraging technology to analyze symptoms and provide relevant information, the app democratizes access to healthcare insights, particularly for individuals who may face barriers to traditional healthcare services.

**Objective**

* Accurate Symptom Analysis:

Develop algorithms and a user-friendly interface that accurately analyzes user-inputted symptoms. The app should provide reliable predictions and descriptions of potential diseases based on the symptoms presented.

* Educational Content:

Offer detailed and easily understandable descriptions of identified diseases. Include information about symptoms, causes, and potential complications to enhance user understanding of their health conditions.

* Home-based Care Recommendations:

Provide practical and actionable suggestions for at-home care based on the predicted diseases. This may include lifestyle changes, dietary recommendations, and home remedies, fostering a sense of empowerment and self-care among users.

* Privacy and Security:

Prioritize user privacy and data security. Implement robust measures to protect user data and ensure compliance with relevant healthcare privacy regulations.

* Continuous Improvement:

Regularly update the app with the latest medical research and advancements to enhance the accuracy of disease predictions and improve the overall user experience.

**Methodology**

* Data Collection:
  + Gather a diverse and comprehensive dataset of symptoms and corresponding medical conditions from reputable sources, medical literature, and healthcare professionals.
  + Implement machine learning algorithms to analyze patterns and correlations within the dataset for accurate disease predictions.
* Algorithm Refinement:
  + Utilize a combination of supervised and unsupervised learning techniques to train the app's prediction model.
  + Regularly update the algorithm to incorporate new medical research findings and enhance prediction accuracy.
* User-Feedback Integration:
  + Implement user feedback mechanisms to continuously improve the app's performance and enhance the accuracy of disease predictions.
  + Encourage users to report the outcomes of their healthcare journeys for ongoing refinement of the prediction model.
* Medical Expert Collaboration:
  + Collaborate with healthcare professionals to validate the accuracy of the app's predictions and ensure alignment with current medical standards.
  + Establish a feedback loop with medical experts for ongoing refinement and improvement of the app's methodology.
* User Interface Design:
  + Develop an intuitive and user-friendly interface that facilitates easy input of symptoms and provides clear, understandable information about predicted diseases.
  + Conduct usability testing to ensure the app is accessible to a wide range of users, including those with varying levels of health literacy.
* Privacy Measures:
  + Implement robust security measures to protect user data and ensure compliance with healthcare privacy regulations.
  + Clearly communicate the app's privacy policies to users and obtain informed consent for data usage.

**Tools to achieve the goal**

* + **Streamlit:** It's a popular Python library for creating web applications for data analysis and machine learning models. In this code, Streamlit is used to build an interactive web app for disease prediction.
  + **NumPy and Pandas:** NumPy is used for numerical operations in Python, and Pandas is used for data manipulation and analysis. They are used here for handling datasets, data manipulation, and splitting data into training and testing sets.

**Scikit-learn (sklearn):** It's a powerful machine learning library in Python. In this code, various scikit-learn modules are used for:

Model selection (train\_test\_split, cross\_val\_score) to split the data and evaluate model performance.

* + Classification models (SVC, GaussianNB, RandomForestClassifier) for disease prediction.
  + Metrics (accuracy\_score, confusion\_matrix) to evaluate model performance.
  + **Matplotlib and Seaborn:** These are visualization libraries in Python. Matplotlib and Seaborn are used for creating visualizations such as bar plots, heatmaps, and confusion matrices to understand data distribution and model performance.
  + **Counter from collections:** The Counter class from the collections module is used to count occurrences of predictions and find the most common prediction among multiple models.
  + **Symptom Index Dictionary:** It's used to encode input symptoms into numerical form for model input.
  + Prediction Function (predictDisease): This function takes symptoms as input, processes them, generates predictions using trained machine learning models (SVM, Naive Bayes, Random Forest), and combines predictions to provide the final disease prediction.

**Model**

**Support Vector Machine (SVM):**

Model: SVC() from scikit-learn.

Explanation: SVM is a powerful classification algorithm that finds a hyperplane in an N-dimensional space (where N is the number of features) that distinctly classifies data points into different classes.

Naive Bayes:

Model: GaussianNB() from scikit-learn.

Explanation: Naive Bayes is a probabilistic classifier based on Bayes' theorem with an assumption of independence between features. Despite its simple assumption, it often performs well and is computationally efficient.

Random Forest:

Model: RandomForestClassifier() from scikit-learn.

Explanation: Random Forest is an ensemble learning method that constructs multiple decision trees during training and combines their predictions to improve accuracy and control overfitting.

K-Nearest Neighbors (KNN):

Model: KNeighborsClassifier() from scikit-learn (not explicitly used in the provided code but commonly used for similar classification tasks).

Explanation: KNN is a simple and effective classification algorithm that works based on the similarity of feature vectors. It assigns a class label to a sample by a majority vote of its neighbors.

Logistic Regression:

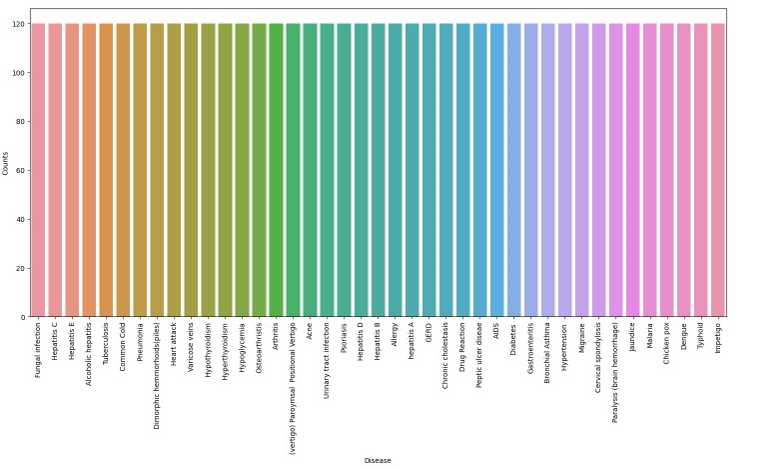
Model: LogisticRegression() from scikit-learn (not explicitly used in the provided code but often used for binary classification tasks).

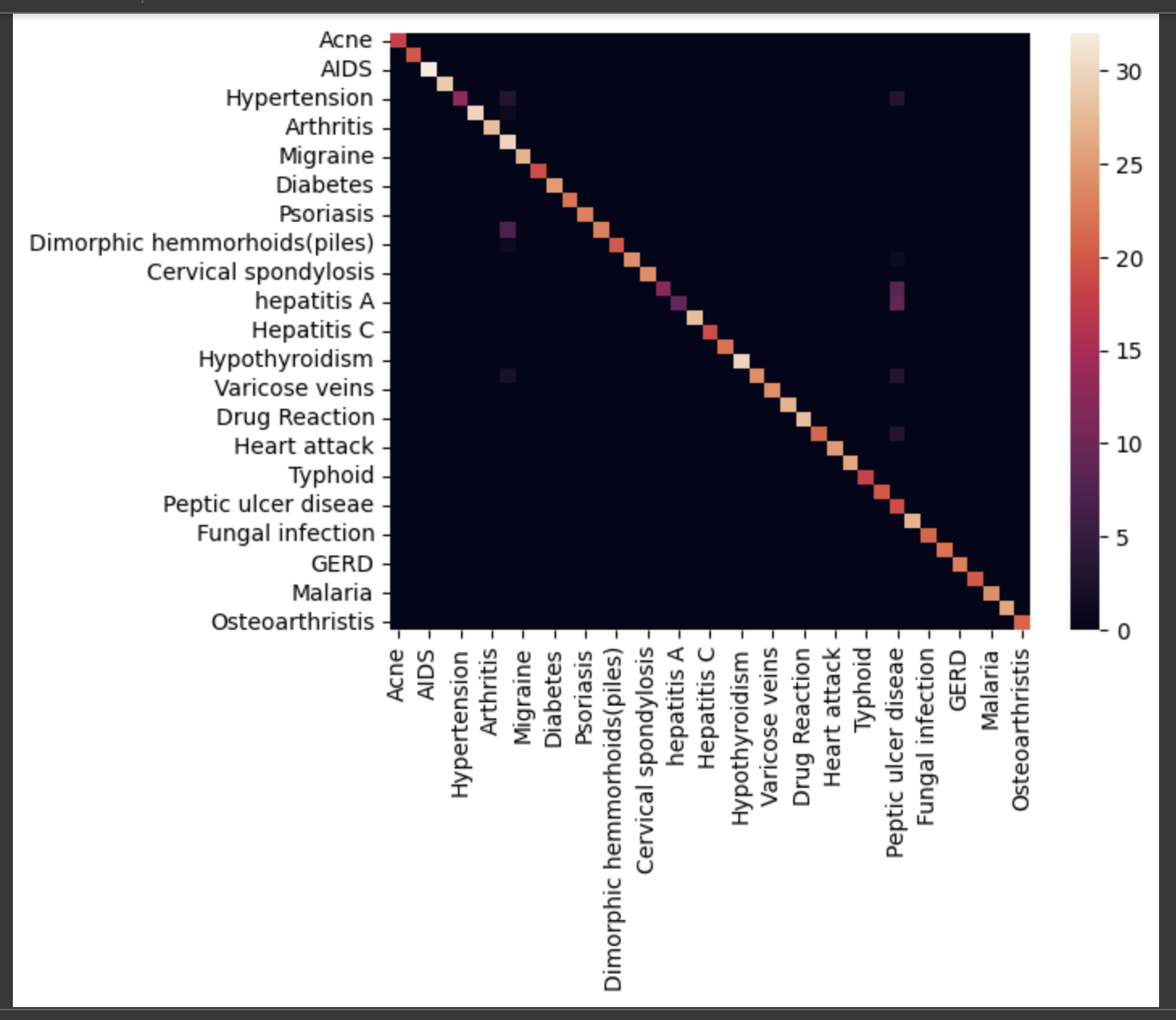
Explanation: Despite its name, logistic regression is a linear model for classification that predicts the probability of a sample belonging to a particular class.

**Code**

https://colab.research.google.com/drive/1pg4LUsQLqgaoaj1erghMusjhNv5BFL8g?usp=sharing

**Analysis and graphical representation**

****

****

**Limitation**

* Accuracy Constraints:
  + Despite continuous refinement, the app's predictions may not be 100% accurate. Users should be encouraged to consult healthcare professionals for a definitive diagnosis.
* Dependency on User-Provided Data:
  + The accuracy of predictions relies on the completeness and accuracy of user-inputted symptoms. Incomplete or inaccurate information may lead to less precise predictions.
* Scope of Predictions:
  + The app may not cover all possible medical conditions, and certain rare or complex diseases may not be accurately predicted. Users should be aware of the app's limitations and seek professional medical advice for unusual or severe symptoms.
* Internet Dependency:
  + The app's functionality may be limited without a stable internet connection, potentially hindering accessibility in certain situations.
* Not a Substitute for Professional Medical Advice:
  + The app is designed to provide preliminary information and suggestions for at-home care but is not a substitute for professional medical advice. Users should consult healthcare professionals for a thorough diagnosis and treatment plan.

**Conclusion**

In conclusion, the development of this disease prediction app represents a step towards democratizing healthcare information and empowering individuals to take control of their well-being. While the app serves as a valuable tool for early intervention and home-based care, its limitations emphasize the importance of seeking professional medical guidance for comprehensive healthcare management. Through ongoing collaboration with medical experts, user feedback integration, and adherence to stringent privacy measures, the app aims to contribute to a more informed and proactive approach to personal health management.