Disease Predictor Using Machine Learning

# Abstract

The Disease Predictor Using Machine Learning is a robust and user-friendly tool designed to provide early-stage insights into common illnesses, heart disease, and COVID-19 based on user-inputted symptoms and health indicators. The project uses advanced machine learning models to make predictions, helping users assess their health risks from a variety of conditions. This document offers a comprehensive breakdown of the project’s functionalities, the technologies employed, step-by-step instructions for run...

# 1. Introduction: Project Overview and Importance

Healthcare advancements often rely on early detection of diseases, which can lead to better patient outcomes. However, access to timely and affordable medical diagnosis is not always feasible. This project, Disease Predictor Using Machine Learning, bridges that gap by offering users a preliminary health assessment based on readily available data. The project combines machine learning algorithms with intuitive user interfaces, giving individuals a way to predict health risks with a simple, non-invasive to...

## 1.1 Project Modules Overview

- Simple-Symptom Disease Predictor: Uses machine learning to predict common diseases based on general symptoms input by users.  
- Heart Disease Predictor: Analyzes various cardiovascular health indicators like cholesterol, blood pressure, and age to estimate the likelihood of heart disease.  
- COVID-19 Predictor: Evaluates the risk of COVID-19 infection using symptoms, travel history, and other relevant data points.  
Each of these predictors is powered by machine learning models trained on historical data, making the predictions both relevant and data-driven.

# 2. Technologies Used: A Breakdown of Tools and Libraries

Building this predictor required leveraging a variety of technologies and libraries, each contributing to different aspects of the project:  
  
- Python 3.x: Chosen as the main programming language due to its simplicity and widespread use in the machine learning community.  
- Tkinter: Used to build the graphical user interface (GUI).  
- Scikit-learn: Utilized to build and train the machine learning models.  
- Pandas: For data manipulation and pre-processing.  
- NumPy: For efficient storage and handling of numerical data.

# 3. How to Run the Project: Setup and Deployment Steps

Follow the steps below to clone, install, and run the project on your local machine:

## Step 1: Clone the Repository

Run the following command in your terminal:  
git clone https://github.com/your-repo/disease-predictor.git  
cd disease-predictor

## Step 2: Install Dependencies

Run the following command to install the required dependencies:  
pip install -r requirements.txt

## Step 3: Run the Application

Execute the following command to start the application:  
python main.py

# 4. Disease Predictors: Models and Code Snippets

## 4.1 Simple-Symptom Disease Predictor: Random Forest Model

This module allows users to select symptoms from a list. The underlying model predicts the most likely illness based on input symptoms.  
Model Insights:  
- Algorithm Used: Random Forest Classifier  
- Training Data: Dataset with symptom-disease correlations  
  
Code Snippet:  
  
def predict\_common\_disease(symptoms):  
 model = joblib.load('models/common\_disease\_model.pkl') # Load pre-trained model  
 prediction = model.predict([symptoms]) # Predict based on input symptoms  
 return prediction[0] # Return the most likely disease

## 4.2 Heart Disease Predictor: Logistic Regression Model

This module focuses on predicting heart disease by analyzing health metrics such as cholesterol levels and age.  
Model Insights:  
- Algorithm Used: Logistic Regression  
- Training Data: Medical indicators for heart disease prediction  
  
Code Snippet:

def predict\_heart\_disease(data):  
 model = joblib.load('models/heart\_disease\_model.pkl') # Load heart disease model  
 prediction = model.predict([data]) # Predict heart disease risk  
 return 'Positive for Heart Disease' if prediction[0] == 1 else 'Negative for Heart Disease'

## 4.3 COVID-19 Predictor: Decision Tree Classifier

The COVID-19 Predictor estimates the likelihood of infection based on reported symptoms and travel history.  
Model Insights:  
- Algorithm Used: Decision Tree Classifier  
- Training Data: Data from COVID-19 patients

Code Snippet:  
  
def predict\_covid(symptoms):  
 model = joblib.load('models/covid\_model.pkl') # Load COVID-19 prediction model  
 prediction = model.predict([symptoms]) # Predict COVID-19 likelihood  
 return 'Positive for COVID-19' if prediction[0] == 1 else 'Negative for COVID-19'

# 5. Dataset Details: Key Features and Size

The datasets used in this project include:  
- Simple Symptom Dataset: 10,000+ records for common diseases.  
- Heart Disease Dataset: 30,000+ records with medical indicators.  
- COVID-19 Dataset: 5,000+ records on COVID-19 symptoms and test results.

# 6. Future Enhancements: Expansion and Integration

Future versions will aim to:  
- Expand disease coverage  
- Improve prediction accuracy  
- Integrate real-time health data  
- Refine the user interface for a better experience

# 7. Acknowledgements: Contributors and Support

Lead Developer: Ishaan Narayan, Dhruv Jindal  
Supervisor: Dr. Achala Shakya

Project Final Snapshots

