**Early Prediction for Chronic Kidney Disease (CKD): A Progressive Approach to Health Management – Aayush Bhalerao (aayushbhlerao@gmail.com)**

**Objective**

To develop a complete **end-to-end machine learning system** capable of predicting whether a patient is likely to suffer from Chronic Kidney Disease (CKD), using clinical parameters and a user-friendly web interface.

This project aims to support early diagnosis and intervention, potentially improving patient outcomes and reducing the burden on healthcare systems.

**Background & Motivation**

Chronic Kidney Disease (CKD) is a major global health issue. It often remains undiagnosed in early stages due to subtle or non-specific symptoms. Early detection can help delay or even prevent progression to kidney failure, but requires accurate and accessible screening tools.

Machine learning provides a data-driven approach to predict CKD using existing medical data. This project leverages real-world patient data to build a predictive tool that can be used interactively by healthcare professionals or integrated into clinical systems.

**Scope**

 Use a real clinical dataset (kidney\_disease.csv)

 Perform pre-processing and Exploratory Data Analysis (EDA)

 Train and evaluate multiple machine learning models

 Select and deploy the best model via a Flask web application

 Provide real-time predictions based on user input

**Dataset Description**

 **File:** kidney\_disease.csv

 **Source:** Publicly available clinical dataset

 **Target Variable:** classification (ckd / notckd)

**Key Features**

 **Numerical**: age, bp (blood pressure), hemo (hemoglobin), sc (serum creatinine), bgr (blood glucose)

 **Categorical**: rbc, pc, pcc, ba, htn, dm, cad, appet, pe, ane

**Data Pre-processing**

 **Handling Missing Values**:

* Numerical: Imputed using mean/median.
* Categorical: Imputed using mode.

 **Encoding**:

* Categorical variables label-encoded for model compatibility.

 **Feature Scaling**:

* Normalization applied to improve convergence in ANN.

**Exploratory Data Analysis (EDA)**

 **Correlation heatmaps** to identify key influencing features

 **Pair plots** and **histograms** to study distributions and relationships

 Identified features such as sc, hemo, al, and bp as highly correlated with CKD.

**Model** **Development**

*Models Implemented:*

1. Logistic Regression
2. Decision Tree
3. Random Forest
4. Artificial Neural Network (ANN) using Keras

### *Evaluation Metrics:*

1. Accuracy
2. Confusion Matrix
3. Loss (for ANN)

### *Best Model:*

* The **Artificial Neural Network (ANN)** showed the highest performance after tuning.
* It was serialized using **pickle** for deployment.

**Web Deployment**

 The **Flask app (**app.py**)** provides a clean, form-based UI.

 Users input clinical features manually.

 Model processes the input and returns:

* "CKD" (Chronic Kidney Disease)
* "Not CKD"

### User Flow

1. Visit the web page.
2. Fill out clinical data.
3. Submit form.
4. View CKD prediction result.

**Team Roles**

*Solo Project* — All parts of the project were handled by the developer:

* Data collection and cleaning
* Model development
* Evaluation and optimization
* Deployment using Flask
* Documentation

## **Results**

* The project demonstrates the feasibility of using machine learning for early detection of CKD.
* The ANN model achieved the highest performance with strong generalization.
* The web app adds usability and real-world applicability.

**Project Structure**

Chronic-Kidney-Disease-Detection/

│

├── Flask/ # Main application directory

│ ├── app.py # Flask backend script for handling routes and predictions

│ ├── Flask app for CKD.ipynb # Jupyter notebook for data analysis and model training

│ ├── CKD.pkl # Serialized (pickled) machine learning model

│ │

│ ├── templates/ # HTML templates for the Flask app

│ │ ├── index.html # Homepage with form inputs for prediction

│ │ └── home.html # (Optional) additional or result display page

│ │

│ └── static/ # Static files like CSS, JS, or images

│ └── style.css # Stylesheet for the web interface

│

├── Dataset/ # Folder for storing the dataset(s)

│ └── kidney\_disease.csv # Main dataset used for training and evaluation

│

└── README.md # Project overview, usage, and instructions

## **14. Future Improvements**

* Collect a larger, more diverse dataset
* Add patient data upload (CSV/API input)
* Improve UI design with Bootstrap or React
* Deploy on a public cloud platform (e.g., Heroku, AWS)
* Add patient risk explanation using **SHAP** or **LIME** (model interpretability)

**Technologies & Tools**

|  |  |
| --- | --- |
| *Programming Language:* | Python |

|  |  |
| --- | --- |
| *Data Analysis:* | Pandas, NumPy |

|  |  |
| --- | --- |
| *Visualization:* | Matplotlib, Seaborn |

|  |  |
| --- | --- |
| *Machine Learning:* | Scikit-learn, Keras, TensorFlow |

|  |  |
| --- | --- |
| *Web Framework:* | Flask |

|  |  |
| --- | --- |
| *Model Persistence:* | Pickle |

|  |  |
| --- | --- |
| *Development Environment:* | Jupyter Notebook, PyCharm |