Project Overview

Project Title: Anemia Detection Using Machine Learning

Objective:
To develop a machine learning-based system that can accurately predict the presence of anemia in patients using clinical data which was found in there blood report.

Scope:

- Build a predictive model using various machine learning algorithms.
- Evaluate and select the best performing model.
- Deploy the selected model for practical use in a healthcare setting.

Introduction

Anemia is a common blood disorder characterized by a deficiency in the number or quality of red blood cells or hemoglobin. Early detection is crucial for timely treatment to avoid severe health complications. This project aims to automate anemia detection by leveraging machine learning techniques on patient data, thus assisting healthcare professionals in diagnosis

Dataset Description

- **Source:** Kaggle(https://www.kaggle.com/code/emreiekyurt/anemia-classification-with-eda-100-acc/input)
- **Features:**
    - Hemoglobin level
    - Red blood cell count
    - Mean corpuscular volume (MCV)
    - Mean corpuscular hemoglobin (MCH)
    - Mean corpuscular hemoglobin concentration (MCHC)
    - Gender
- **Target variable:**
    - Anemia status (Positive/Negative)
- **Size:** Number of samples (e.g., 5000 patient records)

4. Data Preprocessing

- Handling missing values
- Encoding categorical variables (e.g., gender)
- Feature scaling/normalization
- Splitting data into training and testing sets (e.g., 80% training, 20% testing)
- Feature selection

Methodology

Machine Learning Models Used:

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- Support Vector Machine (SVM)
- Naive Bayes
- Gradient Boosting Classifier

Steps:

1. Load and clean dataset
2. Perform exploratory data analysis (EDA) to understand data distribution and relationships
3. Train each model on the training set
4. Evaluate models on the testing set using metrics such as accuracy, precision, recall, F1-score
5. Select the best performing model for deployment

Model Evaluation

Model	Accuracy	Precision	Recall	F1-Score
Linear Regression	.9919354	1.00	0.98	0.99

Decision Tree	1.0	1.00	1.00	1.00
Random Forest	1.0	1.00	1.00	1.00
SVM	0.979838	0.99	0.96	0.98
Naive Bayes	0.9395161	0.99	0.88	0.93
Gradient Boosting	**1.00**	**1.00**	**1.00**	1.00

Implementation

- Programming Language: Python
- Libraries: pandas, numpy, scikit-learn, matplotlib, seaborn, joblib (for saving model)
- Model saved as `model.pkl` for deployment
- Developed a Flask-based web app for user input and prediction display

Usage

How to Run

1. Clone the repository
2. Install dependencies:
3. `pip install -r requirements.txt`
4. Run the training script (if retraining) or load the saved model
5. For Flask app:
6. `python app.py`
7. Access the web interface via `http://localhost:5000` and input patient data to predict anemia

Results & Discussion

- The model achieved an accuracy of **100%** on the test data.
- Important features impacting anemia prediction include hemoglobin levels, MCV,MHC, and MCHC.
- The system can assist doctors in preliminary anemia screening but should not replace clinical judgment.
- Limitations include dataset size and diversity, which can be improved with more varied clinical data.

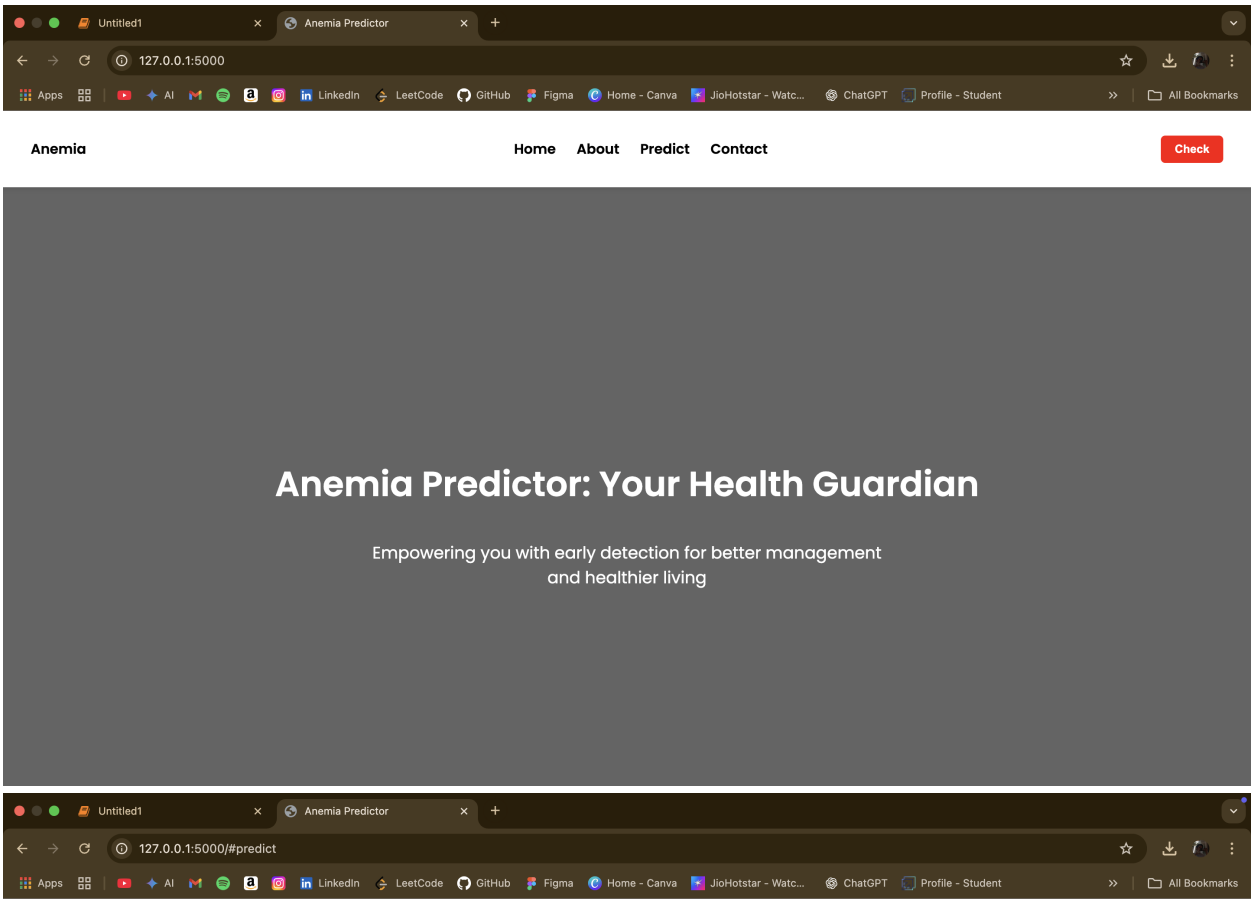
Future Work

- Collect more diverse and larger datasets to improve generalization
- Incorporate additional clinical parameters and patient history
- Develop mobile application for easier accessibility
- Integrate with hospital information systems for real-time prediction
- Explore deep learning models for enhanced performance

References

- Project demonstration from smartbridge project workspace
- Documentation for used libraries (scikit-learn, Flask, etc.)
- Dataset source is kaggle :(https://www.kaggle.com/code/emreiekyurt/anemia-classification-with-eda-100-acc/input)

OUTPUT



Enter the details:

Gender:

Male

Hemoglobin (Range: 7-16):

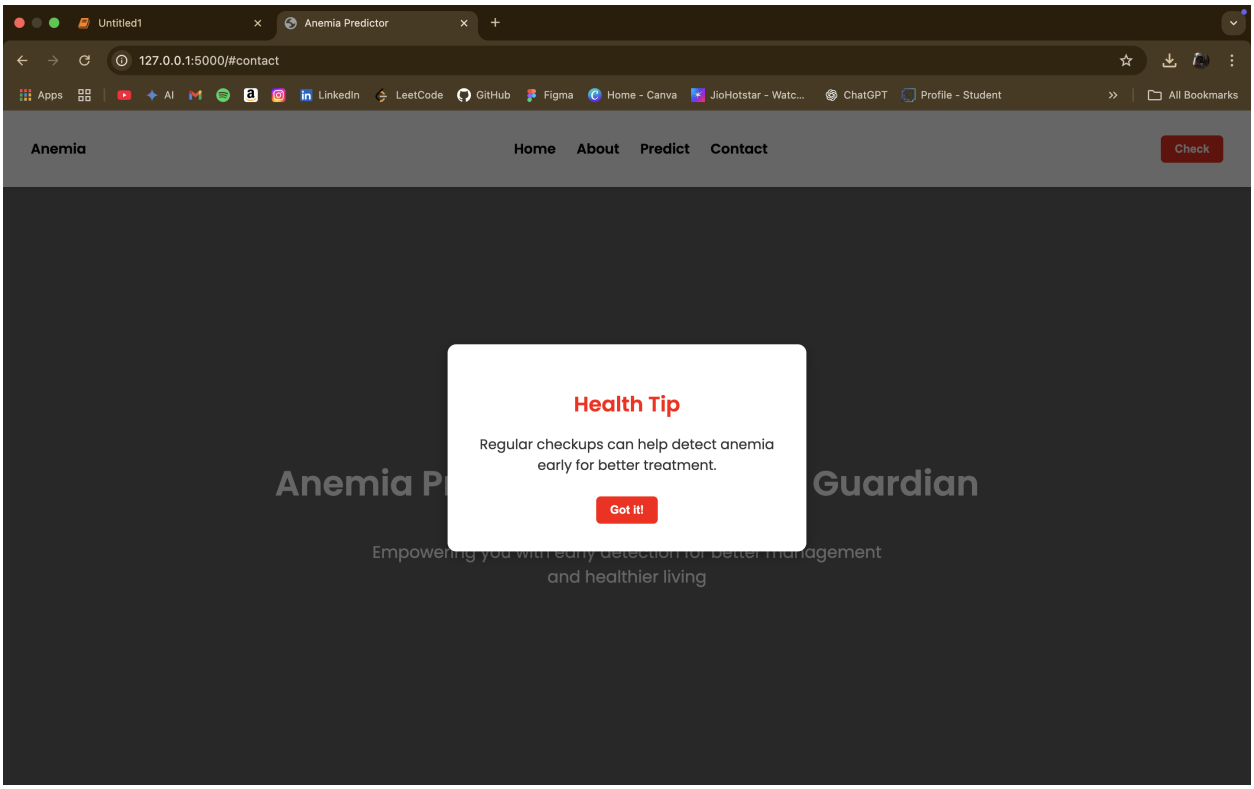
MCH (Range: 16-36):

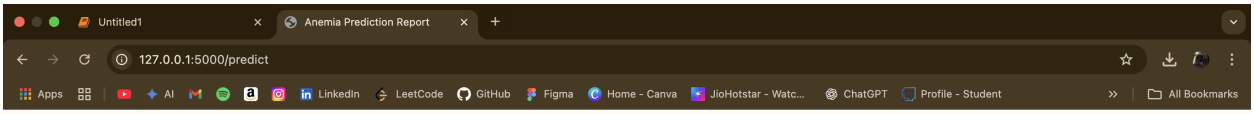
MCHC (Range: 28-34):

MCV (Range: 70-100):

Submit

Contact Us





City Health Diagnostics

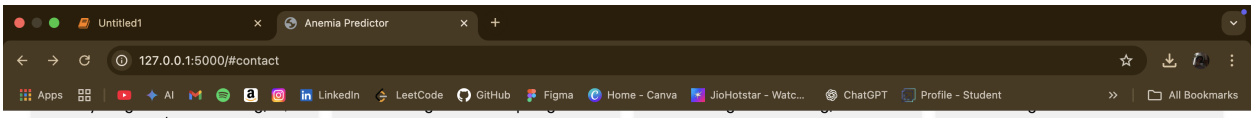
123 Medical Street, Ghaziabad, India
Phone: +91 786017**** | Email: iqbalumar131@gmail.com

Patient Name: _____
Age: ____
Date: _____
Report ID: ANM-

Anemia Test Result

Hence, based on calculation: You don't have any Anemic Disease

This report is generated by an automated machine learning model and should be reviewed by a qualified medical professional.



Women: 11.6–15 g/dL.

Enter the details:

Gender:

Male

Hemoglobin (Range: 7-16):

13.4

MCH (Range: 16-36):

28

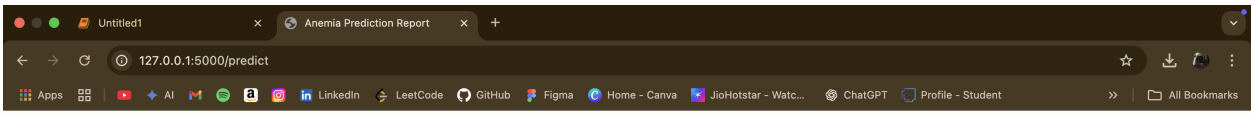
MCHC (Range: 28-34):

35

MCV (Range: 70-100):

90

Submit



City Health Diagnostics

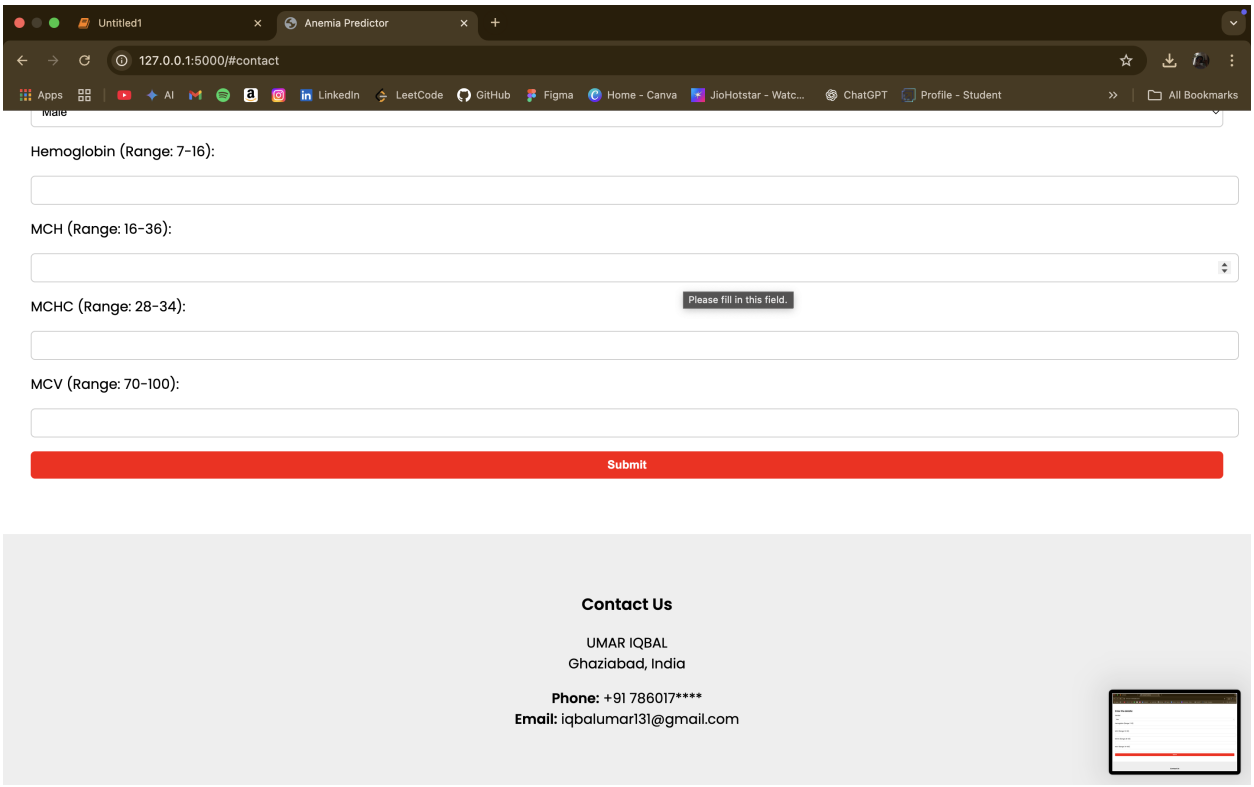
123 Medical Street, Ghaziabad, India
Phone: +91 786017**** | Email: iqbalumar131@gmail.com

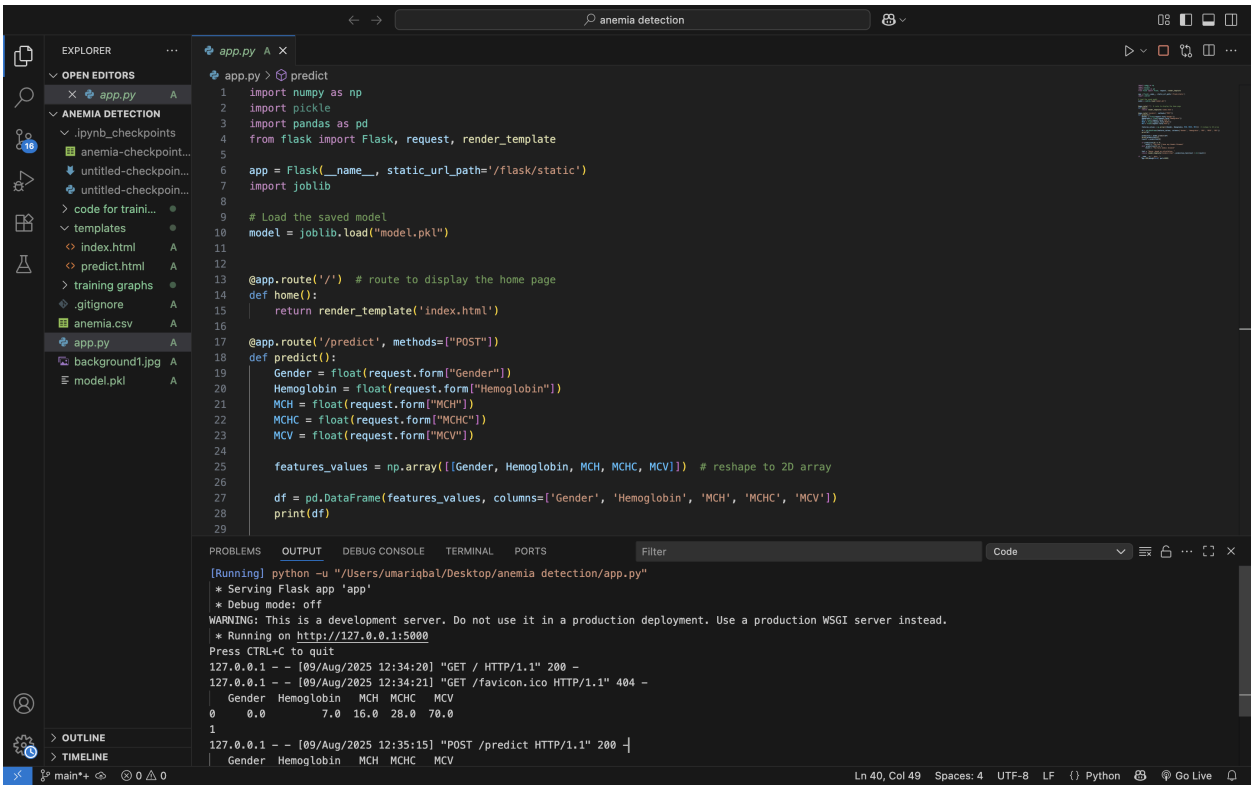
Patient Name: _____
Age: ____
Date: _____
Report ID: ANM-

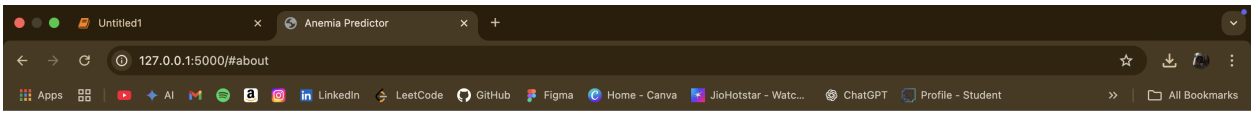
Anemia Test Result

Hence, based on calculation: You have anemic disease

This report is generated by an automated machine learning model and should be reviewed by a qualified medical professional.


Untitled1
Anemia Predictor
127.0.0.1:5000/#contact
Apps
AI
LinkedIn
LeetCode
GitHub
Figma
Home - Canva
JioHotstar - Watc...
ChatGPT
Profile - Student
All Bookmarks
Male
Hemoglobin (Range: 7-16):
MCH (Range: 16-36):
MCHC (Range: 28-34):
Please fill in this field.
MCV (Range: 70-100):
Submit
Contact Us
UMAR IQBAL
Ghaziabad, India
Phone: +91 786017****
Email: iqbalumar131@gmail.com


anemia detection
EXPLORER
OPEN EDITORS
app.py
ANEMIA DETECTION
.ipynb_checkpoints
anemia-checkpoint...
untitled-checkpoin...
untitled-checkpoin...
code for traini...
templates
index.html
predict.html
training graphs
.gitignore
anemia.csv
app.py
background1.jpg
model.pkl
OUTLINE
TIMELINE
app.py > predict
import numpy as np
import pickle
import pandas as pd
from flask import Flask, request, render_template
app = Flask(__name__, static_url_path='/flask/static')
import joblib
# Load the saved model
model = joblib.load("model.pkl")
@app.route('/') # route to display the home page
def home():
return render_template('index.html')
@app.route('/predict', methods=["POST"])
def predict():
Gender = float(request.form["Gender"])
Hemoglobin = float(request.form["Hemoglobin"])
MCH = float(request.form["MCH"])
MCHC = float(request.form["MCHC"])
MCV = float(request.form["MCV"])
features_values = np.array([[Gender, Hemoglobin, MCH, MCHC, MCV]]) # reshape to 2D array
df = pd.DataFrame(features_values, columns=['Gender', 'Hemoglobin', 'MCH', 'MCHC', 'MCV'])
print(df)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Filter
Code
[Running] python -u "/Users/umariqbal/Desktop/anemia detection/app.py"
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [09/Aug/2025 12:34:20] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [09/Aug/2025 12:34:21] "GET /favicon.ico HTTP/1.1" 404 -
Gender Hemoglobin MCH MCHC MCV
0 0.0 7.0 16.0 28.0 70.0
1
127.0.0.1 - - [09/Aug/2025 12:35:15] "POST /predict HTTP/1.1" 200 -
Gender Hemoglobin MCH MCHC MCV
main*+
Ln 40, Col 49
Spaces: 4
UTF-8
LF
Python
Go Live



About

Anemia: low red blood cells or hemoglobin, causing oxygen shortage. Symptoms: fatigue, weakness, shortness of breath, pale skin. Causes: nutritional deficiencies, diseases, blood loss. Treatments vary.

Hemoglobin

Healthy ranges: Men: 13.2–16.6 g/dL, Women: 11.6–15 g/dL.

MCH

Normal range: 27.5–33.2 picograms.

MCHC

Normal range: 33.4–35.5 g/dL.

MCV

Normal range: 80–100 femtoliters.

Enter the details:

Gender:

Male

Hemoglobin (Range: 7-16):

MCH (Range: 16-36):