# Machine Learning for Sustainable Development Goal 3: Good Health and Well-being

# 1. Introduction

Project Objective: An ML model to analyze patient data and predict the likelihood of developing chronic diseases such as diabetes, heart disease, or cancer. The solution should also provide personalized health recommendations to improve patient outcomes and well-being

Motivation: to detect diabetes stems from the rising number of people affected by the disease and the challenges of traditional screening methods, which can delay diagnoses. By analyzing data with machine learning, we can identify at-risk patients earlier, enabling personalized care and timely interventions. This approach aims to improve healthcare access and outcomes for individuals living with diabetes.

## 2. Data Collection

Data Source: Kaggle Dataset ( “diabetes prediction Dataset”)

Dataset Description:  
- Features: pregnancies, Glucose, bloodpressure, Skinthickness, insulin, BMI, Age, diabetspedigreefunction, outcome  
- Size: 768 rows by 9 columns  
- Target Variable: outcome (binary)

## 3. Exploratory Data Analysis (EDA)

Summary Statistics: Mean, median, and distribution of each feature.  
Visualizations:  
- Correlation heatmap to understand relationships between variables.

A screenshot of a graph

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a cell phone

Description automatically generated

- Boxplots for outlier detection.

A white background with black text

Description automatically generatedA graph of a graph with blue squares

Description automatically generated with medium confidence

## 4. Data Preprocessing

Handling Missing Values: Used median imputation for features with missing values.

Pregnancies 0

Glucose 0

BloodPressure 0

SkinThickness 0

Insulin 0

BMI 0

DiabetesPedigreeFunction 0

Age 0

Outcome 0

dtype: int64

Feature Scaling: Standardized features using `StandardScaler` for better performance in machine learning models.

## 5. Machine Learning Model Selection

Model Choices:  
- Logistic Regression (for binary classification).  
Why Scikit-Learn:

* Scikit-Learn provides a straightforward interface for implementing logistic regression, making it accessible for users of varying experience levels.
* **Diverse Algorithms**: The library offers a wide range of machine learning algorithms, allowing for easy experimentation with different models if needed in the future.
* **Comprehensive Performance Metrics**: Scikit-Learn includes built-in functions to evaluate model performance effectively, ensuring that we can measure accuracy and other relevant metrics.

Evaluation Metric: Accuracy, Precision, Recall, and F1-Score due to the critical nature of accurately identifying diabetes.

## 6. Model Implementation

Data Splitting: Split dataset into 80% training and 20% testing sets using `train\_test\_split` from Scikit-Learn.

Code Example:

from sklearn.model\_selection import train\_test\_split x\_train,x\_test,y\_train,y\_test=train\_test\_split(scaled\_data,y,test\_size=.2,random\_state=100)

from sklearn.linear\_model import LogisticRegression model=LogisticRegression() model.fit(x\_train,y\_train)

y\_pred = best\_model.predict(X\_test)  
print(classification\_report(y\_test, y\_pred))

## 7. Results and Evaluation

Model Performance:  
- Random Forest achieved an accuracy of X%, F1-score of Y%, and precision/recall values

trian\_accuracy=model.score(x\_train,y\_train) print("train\_accuracy :",trian\_accuracy) test\_accuracy=model.score(x\_test,y\_test) print("train\_accuracy :",test\_accuracy)

indicatintrain\_accuracy : 0.7879282218597063

train\_accuracy : 0.7467532467532467

the model’s strength in predicting contamination risk.

#recall score  
from sklearn.metrics import f1\_score , recall\_score

y\_pred=model.predict(x\_test)

print(f1\_score(y\_test,y\_pred))

0.6213592233009708

## 8. Conclusion and Future Work

Key Takeaways: Machine learning models effectively predict diabetis on the condition of the patient . The project demonstrates potential for real-time monitoring and resource allocation.

## 9. References

- Kaggle Dataset  
- Scikit-Learn Documentation