IBM NAAN MUDHALVAN AI – GROUP 3

AI BASED DIABETES PREDICTION SYSTEM

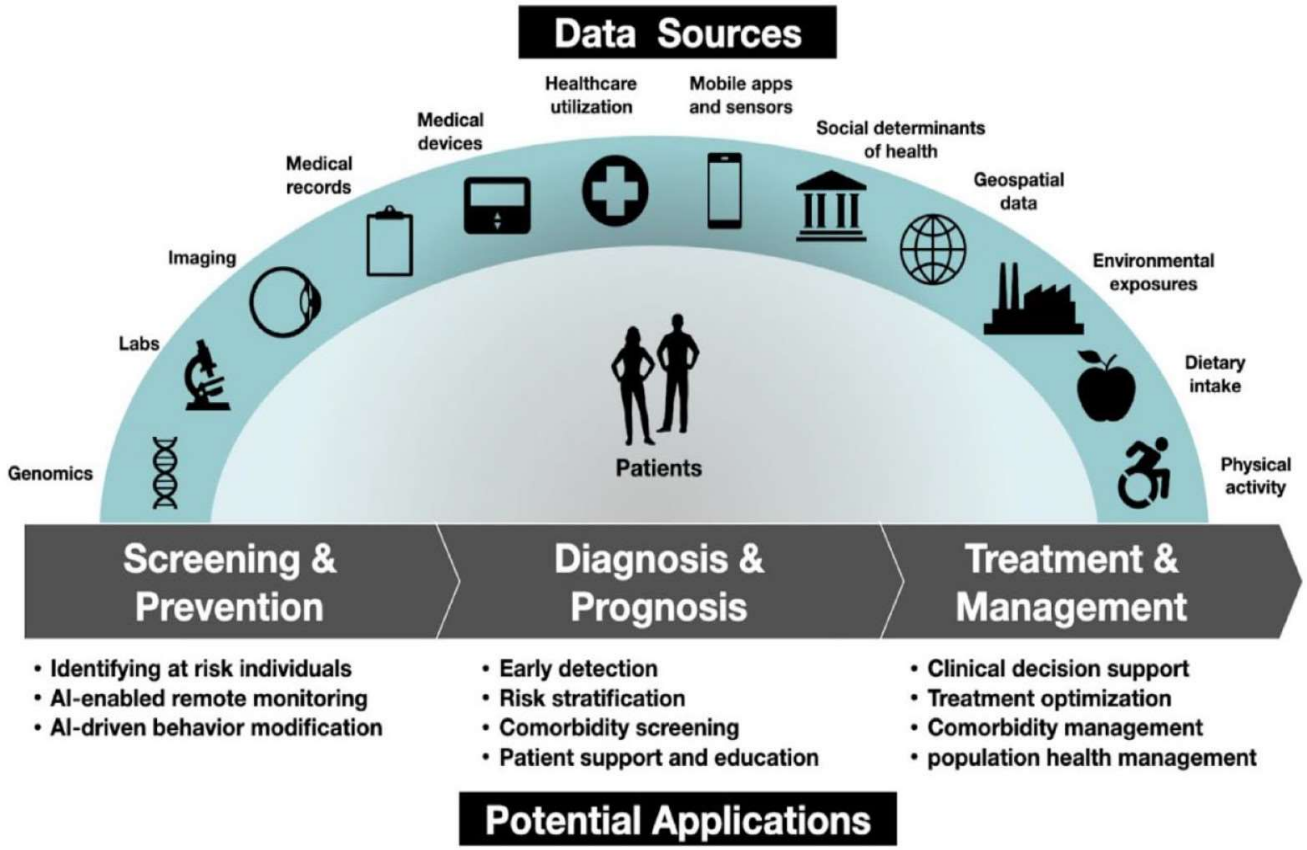
TEAM MEMBER

NAME: Muthu R

REGISTER NO: 721921243072

Project Title: AI Based Diabetes Prediction System

PHASE-3 : DEVELOPMENT PART- I



INTRODUCTION:

An AI-based diabetes prediction system is a sophisticated healthcare solution that leverages artificial intelligence and machine learning techniques to assess an

individual's risk of developing diabetes. It analyzes various factors, such as medical history, lifestyle choices, genetic predisposition, and clinical data, to make predictions about the likelihood of diabetes onset. This system aims to provide early detection and personalized preventive measures, helping individuals and healthcare professionals manage and mitigate the impact of diabetes.

ALGORITHM:

STEP 1: Import the library packages and modules

STEP 2: Load the Datasets

STEP 3: Exploring the Datasets

STEP 4: Split the dataset into training and testing sets

STEP 5: Predicting models using dataset

STEP 6: Visualising models and maintenance STEP 6:Print the results

Data Set Link:

https://www.kaggle.com/datasets/mathchi/diabetes-data-set

PROGRAM:

```
# Import necessary libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

# Load your dataset (You would typically have a larger, more comprehensive dataset)
data = pd.read_csv('diabetes_data.csv')

# Split the data into features (X) and the target variable (y)
X = data.drop('diabetes', axis=1)
y = data['diabetes']
```

```
# Split the dataset into a training and testing set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Initialize and train a machine learning model (Random Forest, in this case)
model = RandomForestClassifier()
model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = model.predict(X_test)

# Evaluate the model's accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Model Accuracy: {accuracy}")

# You can now use this model for diabetes prediction.
```

Developing a complete AI-based diabetes prediction system requires a substantial amount of code and resources. Here's a simplified example using Python and the scikit-learn library to create a basic diabetes prediction model. Please note that a real-world system would be significantly more complex and require large datasets, feature engineering, and careful handling of sensitive medical data.

Libraries and Modules used:

Developing an AI-based diabetes prediction system requires a variety of libraries and modules, depending on the programming language you choose and the specific components of the system. Here are some commonly used libraries and modules for such a project:

1. Python: Python is a popular language for AI and machine learning, and many libraries are available for data analysis, modeling, and deployment.
    - NumPy: For numerical operations and array manipulation.Pandas: For data manipulation and analysis.
    - Scikit-learn: A machine learning library for building and evaluating predictive models.
    - TensorFlow or PyTorch: For deep learning and neural network modeling.

- Matplotlib or Seaborn: For data visualization.
- Flask or Django: For building a web-based user interface.

2. Data Management and Analysis:
   - SQL/NoSQL databases: For data storage and retrieval.pymysql, psycopg2, pymongo: Python modules for interacting with databases.
   - SQLAlchemy: For database management and SQL query generation.
3. Data Preprocessing:
   - Scikit-learn's preprocessing module: For scaling, encoding, and transforming data.
4. Feature Engineering:
   - Scikit-learn or feature-engine: Libraries to create and engineer features.
5. Model Development:
   - Scikit-learn: For traditional machine learning models.
   - TensorFlow or PyTorch: For deep learning and neural network models.

CONCLUSION:

An AI-based diabetes prediction system holds significant promise in the field of healthcare. By leveraging machine learning algorithms and vast datasets, it can accurately assess an individual's risk of developing diabetes, enabling early intervention and personalized care. However, it's essential to continually refine and validate these systems, ensure patient privacy, and integrate them effectively into healthcare practices to realize their full potential in preventing and managing diabetes.

THANKING YOU