

IBM NAAN MUDHALVAN

ARTIFICIAL INTELLIGENCE-GROUP 2

PHASE - 4 : DEVELOPMENT PART 2

PROBLEM STATEMENT:

The problem is to build an **AI-powered diabetes prediction system** that uses machine learning algorithms to analyze medical data and predict the likelihood of an individual developing diabetes. The system aims to provide early risk assessment and personalized preventive measures, allowing individuals to take proactive actions to manage their health.

EXPLANATION:

An AI-based diabetes prediction system is a computer program or application that uses artificial intelligence techniques, such as machine learning algorithms, to analyze data related to an individual's health, lifestyle, and medical history in order to predict the likelihood of them developing diabetes in the future. This system can help identify individuals at higher risk of diabetes, allowing for early intervention and preventive measures to manage or mitigate the disease.

INTRODUCTION

- Diabetes is a chronic metabolic disorder that affects millions of people worldwide, with potentially severe consequences for health and quality of life. Early detection and proactive management of diabetes are essential for improving patient outcomes. One promising approach to address this challenge is the development of AI-based diabetes prediction systems.
- An AI-based diabetes prediction system leverages the power of artificial intelligence and machine learning to analyze data and make accurate predictions about an individual's risk of developing diabetes or the

progression of the disease. Such a system can be a valuable tool for healthcare providers, researchers, and individuals at risk of diabetes.

- These systems typically use a variety of data sources, including medical records, patient demographics, clinical measurements (such as blood glucose levels, BMI, and blood pressure), and lifestyle information to generate predictions. By processing and analyzing this data, AI models can identify patterns, relationships, and risk factors that may not be apparent through traditional diagnostic methods.
- The key components of an AI-based diabetes prediction system include data collection, preprocessing, feature engineering, model training, and evaluation. These systems can provide several benefits:
 - ❖ **Early Detection:** AI models can identify potential diabetes cases at an early stage, enabling timely intervention and management, which can help prevent complications.
 - ❖ **Personalized Care:** By considering individual characteristics and health histories, these systems can offer personalized risk assessments and recommendations for lifestyle changes or medical interventions.
 - ❖ **Efficiency:** Automation of the prediction process streamlines the workflow for healthcare providers, making it easier to identify at-risk individuals and tailor treatment plans.
 - ❖ **Research Insights:** AI-based diabetes prediction systems can also be valuable tools for researchers, aiding in the discovery of new risk factors, biomarkers, and treatment strategies.
 - ❖ **Improved Outcomes:** Ultimately, the goal is to improve patient outcomes and quality of life by enabling early intervention and informed decision-making.

PREVIOUS PHASE-3:

Data preprocessing is a crucial step in building AI-based diabetes prediction systems (or any machine learning system). It involves cleaning, transforming, and organizing your data to make it suitable for model training.

In previous phase we have done the data preprocessing to enhance the data for the AI based diabetes prediction system for development part 1 and we have already documented the data preprocessing for development part 1.

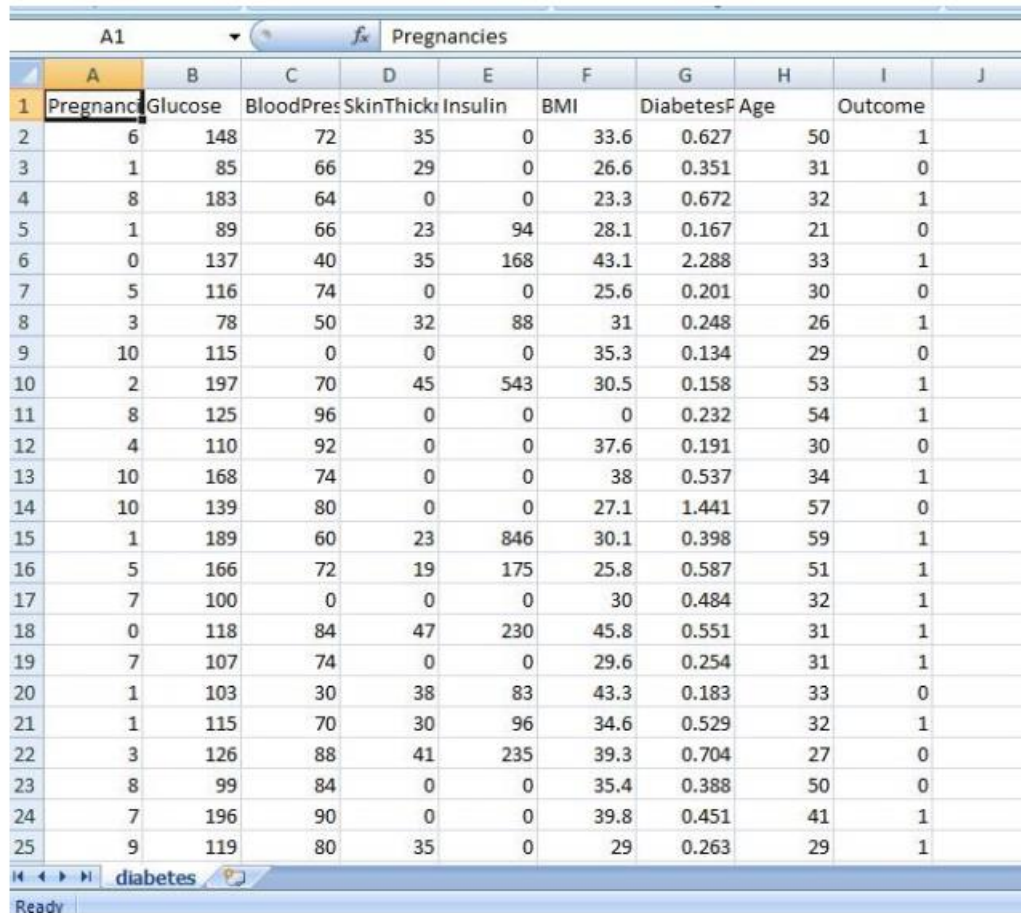
DATASET DETAILS:

Context: This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective is to predict based on diagnostic measurements whether a patient has diabetes. Content: Several constraints were placed on the selection of these instances from a larger database.

In particular, all patients here are females at least 21 years old of Pima Indian heritage.

- **Pregnancies:** Number of times pregnant
- **Glucose:** Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- **BloodPressure:** Diastolic blood pressure (mm Hg)
- **SkinThickness:** Triceps skin fold thickness (mm)
- **Insulin:** 2-Hour serum insulin (mu U/ml)
- **BMI:** Body mass index (weight in kg/(height in m)²)
- **DiabetesPedigreeFunction:** Diabetes pedigree function
- **Age:** Age (years)
- **Outcome:** Class variable (0 or 1)

GIVEN DATA SET:



	A	B	C	D	E	F	G	H	I	J
1	Pregnancies	Glucose	BloodPres	SkinThick	Insulin	BMI	DiabetesF	Age	Outcome	
2	6	148	72	35	0	33.6	0.627	50	1	
3	1	85	66	29	0	26.6	0.351	31	0	
4	8	183	64	0	0	23.3	0.672	32	1	
5	1	89	66	23	94	28.1	0.167	21	0	
6	0	137	40	35	168	43.1	2.288	33	1	
7	5	116	74	0	0	25.6	0.201	30	0	
8	3	78	50	32	88	31	0.248	26	1	
9	10	115	0	0	0	35.3	0.134	29	0	
10	2	197	70	45	543	30.5	0.158	53	1	
11	8	125	96	0	0	0	0.232	54	1	
12	4	110	92	0	0	37.6	0.191	30	0	
13	10	168	74	0	0	38	0.537	34	1	
14	10	139	80	0	0	27.1	1.441	57	0	
15	1	189	60	23	846	30.1	0.398	59	1	
16	5	166	72	19	175	25.8	0.587	51	1	
17	7	100	0	0	0	30	0.484	32	1	
18	0	118	84	47	230	45.8	0.551	31	1	
19	7	107	74	0	0	29.6	0.254	31	1	
20	1	103	30	38	83	43.3	0.183	33	0	
21	1	115	70	30	96	34.6	0.529	32	1	
22	3	126	88	41	235	39.3	0.704	27	0	
23	8	99	84	0	0	35.4	0.388	50	0	
24	7	196	90	0	0	39.8	0.451	41	1	
25	9	119	80	35	0	29	0.263	29	1	

Phase 4: Development Part 2

In this part we will continue building our project. In this phase, we'll continue building the diabetes prediction system by:

- ✓ Selecting a machine learning algorithm.
- ✓ Training the model.
- ✓ Evaluating its performance.

SELECTING MACHINE LEARNING ALGORITHM:

Creating an AI-based diabetes prediction system involves various steps, including data collection, preprocessing, feature engineering, model selection, training, and evaluation. There are several machine learning algorithms you can use for this task.

CODE:

```
# separating the data and labels
X = diabetes_dataset.drop(columns = 'Outcome', axis=1)
Y = diabetes_dataset['Outcome']
print(X)
```

OUTPUT:

	Pregnancies	Glucose	BloodPressure	...	BMI	DiabetesPedigreeFunction	Age
0	6	148	72 ...	33.6	0.627	50	
1	1	85	66 ...	26.6	0.351	31	
2	8	183	64 ...	23.3	0.672	32	
3	1	89	66 ...	28.1	0.167	21	
4	0	137	40 ...	43.1	2.288	33	
..	
763	10	101	76 ...	32.9	0.171	63	
764	2	122	70 ...	36.8	0.340	27	
765	5	121	72 ...	26.2	0.245	30	
766	1	126	60 ...	30.1	0.349	47	
767	1	93	70 ...	30.4	0.315	23	

[768 rows x 8 columns]

CODE:

```
print(Y)
```

OUTPUT:

```
0    1
```

```
1    0
```

```
2    1
```

```
3    0
```

```
4    1
```

```
..
```

```
763  0
```

```
764  0
```

```
765  0
```

```
766  1
```

```
767  0
```

```
Name: Outcome, Length: 768, dtype: int64
```

Data Standardization:

Data standardization, also known as feature scaling, is a critical data preprocessing step in machine learning, including for AI-based diabetes prediction systems. It involves transforming numerical features into a common scale to ensure that no single feature dominates the learning process due to differences in their magnitudes.

CODE:

```
scaler = StandardScaler()  
scaler.fit(X)
```

OUTPUT:

```
StandardScaler(copy=True, with_mean=True, with_std=True)
```

CODE:

```
standardized_data = scaler.transform(X)  
print(standardized_data)
```

OUTPUT:

```
[[ 0.63994726  0.84832379  0.14964075 ...  0.20401277  0.46849198  
   1.4259954 ]  
 [-0.84488505 -1.12339636 -0.16054575 ... -0.68442195 -0.36506078  
  -0.19067191]  
 [ 1.23388019  1.94372388 -0.26394125 ... -1.10325546  0.60439732  
  -0.10558415]  
 ...  
 [ 0.3429808  0.00330087  0.14964075 ... -0.73518964 -0.68519336  
  -0.27575966]  
 [-0.84488505  0.1597866  -0.47073225 ... -0.24020459 -0.37110101  
   1.17073215]  
 [-0.84488505 -0.8730192  0.04624525 ... -0.20212881 -0.47378505  
  -0.87137393]]
```

CODE:

```
X = standardized_data
Y = diabetes_dataset['Outcome']

print(X)

print(Y)
```

OUTPUT:

```
[[ 0.63994726  0.84832379  0.14964075 ...  0.20401277  0.46849198
   1.4259954 ]
 [-0.84488505 -1.12339636 -0.16054575 ... -0.68442195 -0.36506078
  -0.19067191]
 [ 1.23388019  1.94372388 -0.26394125 ... -1.10325546  0.60439732
  -0.10558415]
 ...
 [ 0.3429808  0.00330087  0.14964075 ... -0.73518964 -0.68519336
  -0.27575966]
 [-0.84488505  0.1597866  -0.47073225 ... -0.24020459 -0.37110101
   1.17073215]
 [-0.84488505 -0.8730192  0.04624525 ... -0.20212881 -0.47378505
  -0.87137393]]

0    1
1    0
2    1
3    0
4    1
```



```
..  
763  0  
764  0  
765  0  
766  1  
767  0
```

Name: Outcome, Length: 768, dtype: int64

Train Test Split:

In an AI-based diabetes prediction system, the train-test split is a critical step in the data preprocessing phase. This split allows you to assess the model's performance on unseen data, ensuring that it can generalize well to new patient data. The train-test split divides the dataset into two sets: one for training the model and one for testing its performance.

CODE:

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = 0.2,  
stratify=Y, random_state=2)  
  
print(X.shape, X_train.shape, X_test.shape)
```

OUTPUT:

```
(768, 8) (614, 8) (154, 8)
```

Training the Model:

Training the model for an AI-based diabetes prediction system involves several steps, and it's essential to follow a systematic approach to ensure the best possible results.

CODE:

```
classifier = svm.SVC(kernel='linear')  
  
#training the support vector Machine Classifier  
classifier.fit(X_train, Y_train)
```

OUTPUT:

```
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,  
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='linear',  
    max_iter=-1, probability=False, random_state=None, shrinking=True,  
    tol=0.001, verbose=False)
```

Model Evaluation:

Model evaluation is a crucial step in assessing the performance and reliability of an AI-based diabetes prediction system. You need to ensure that your model is accurate, generalizes well to unseen data, and aligns with your specific project objectives.

Accuracy Score :

Accuracy is a commonly used evaluation metric for classification tasks, including AI-based diabetes prediction systems. It measures the proportion of correctly predicted instances (both true positives and true negatives) to the total number of instances in your dataset.

CODE:

```
# accuracy score on the training data  
X_train_prediction = classifier.predict(X_train)
```

```
training_data_accuracy = accuracy_score(X_train_prediction,  
Y_train)  
  
print('Accuracy score of the training data : ', training_data_accuracy)
```

OUTPUT:

Accuracy score of the training data : 0.7866449511400652

CODE:

```
# accuracy score on the test data  
  
X_test_prediction = classifier.predict(X_test)  
  
test_data_accuracy = accuracy_score(X_test_prediction, Y_test)  
  
  
print('Accuracy score of the test data : ', test_data_accuracy)
```

OUTPUT:

Accuracy score of the test data : 0.7727272727272727

Making a Predictive System:

A predictive system, often referred to as a predictive analytics system, is a technology or software solution that uses data analysis and statistical algorithms to make predictions or forecasts about future events, trends, or outcomes. These systems are widely used across various domains, including healthcare, finance, marketing, and manufacturing, to assist in decision-making, improve efficiency, and gain insights.

CODE:

```
input_data = (5,166,72,19,175,25.8,0.587,51)
```

```
# changing the input_data to numpy array
input_data_as_numpy_array = np.asarray(input_data)

# reshape the array as we are predicting for one instance
input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

# standardize the input data
std_data = scaler.transform(input_data_reshaped)
print(std_data)

prediction = classifier.predict(std_data)
print(prediction)

if (prediction[0] == 0):
    print('THE PERSSON IS NOT DIABETIC')
else:
    print('THE PERSON IS DIABETIC')
```

OUTPUT:

```
[[ 0.3429808  1.41167241  0.14964075 -0.09637905  0.82661621 -0.78595734
  0.34768723  1.51108316]]
[1]          THE PERSON IS DIABETIC
```

ANOTHER EXAMPLE:

In this example we have predicted the model using with different input data to show the various predicting outcomes such as,

CODE:

```
input_data = (6,146,82,69,125,15.8,0.507,31)

# changing the input_data to numpy array
input_data_as_numpy_array = np.asarray(input_data)

# reshape the array as we are predicting for one instance
input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

# standardize the input data
std_data = scaler.transform(input_data_reshaped)
print(std_data)

prediction = classifier.predict(std_data)
print(prediction)

if (prediction[0] == 0):
    print('THE PERSON IS NOT DIABETIC')
else:
    print('THE PERSON IS DIABETIC')
```

OUTPUT:

```
[[ 0.63994726  0.7857295  0.66661825  3.040024  0.39247142 -2.0551498
  0.10607774 -0.19067191]]

[0]          THE PERSON IS NOT DIABETIC
```

DIFFICULTIES OF BUILDING AN AI-BASED DIABETES PREDICTION SYSTEM:

Building an AI-based diabetes prediction system presents several challenges and difficulties that need to be addressed for the system to be effective, reliable, and ethically sound. Here are some of the key challenges and difficulties in developing such a system:

- Data Quality and Availability
- Data Privacy and Security
- Data Imbalance
- Feature Engineering
- Model Selection and Tuning
- Interpretability
- Bias and Fairness
- Generalization
- Continuous Monitoring and Updating
- Integration with Healthcare Workflow
- Communication and Education
- Robustness and Security
- Ethical and Legal Considerations
- Scalability

ADVANTAGES OF AI-BASED DIABETES PREDICTION SYSTEM:

AI-based diabetes prediction systems offer numerous advantages that can significantly impact healthcare and patient outcomes. Here are some key advantages of such systems:

- Early Detection
- Personalized Risk Assessment
- Efficiency
- Improved Patient Outcomes
- Data-Driven Insights
- Reduction of Healthcare Costs
- Support for Healthcare Providers
- Continuous Monitoring
- Patient Empowerment

- Research Advancements
- Scalability
- Reduction of Diagnostic Delays
- Population Health Management
- Remote Monitoring

DISADVANTAGES OF AI-BASED DIABETES PREDICTION SYSTEM:

AI-based diabetes prediction systems offer numerous advantages, as mentioned in previous responses. However, they also come with certain disadvantages and challenges that need to be carefully considered. Here are some of the disadvantages of AI-based diabetes prediction systems:

- Bias and Fairness
- Overfitting and Underfitting
- Lack of Context
- Ethical Concerns
- Complexity and Cost
- Maintenance and Updating
- Integration with Healthcare Workflow
- Patient Education
- Legal and Regulatory Compliance
- Dependency on Technology
- Limited Predictive Power

CONCLUSION:

In conclusion, the development of an AI-based diabetes prediction system is a complex and multidisciplinary endeavor that holds significant promise for improving healthcare and patient outcomes.

In summary, AI-based diabetes prediction systems have the potential to transform healthcare by facilitating early detection, personalized care, and improved patient outcomes. However, they also introduce ethical, privacy, and data quality challenges that demand careful attention. By addressing these challenges with diligence, collaboration, and a commitment to ethical and responsible use, AI-based diabetes prediction systems can offer significant benefits to both healthcare providers and patients.