

Phase 3:Development Part 1

AI Based Diabetes Prediction System

Introduction:

In this initial phase of our project, we will focus on preparing the data for building a diabetes prediction system. The primary goal is to load the dataset, understand its structure, and preprocess it to make it suitable for training a machine learning model. Additionally, we will explore feature selection to determine the most relevant attributes for prediction.

Dataset Description:

We will be working with a dataset containing various health-related features, demographic information, and an indicator of whether an individual has diabetes or not. This dataset is crucial for developing an accurate diabetes prediction system.

Dataset Link:

[**https://www.kaggle.com/datasets/mathchi/diabetes-data-set**](https://www.kaggle.com/datasets/mathchi/diabetes-data-set)

Loading the Kaggle Dataset:

The first step is to access and import the dataset from Kaggle. Loading the dataset is essential as it provides the data we need for model development.

Data Preprocessing:

Data preprocessing is a crucial part of any machine learning project. This step involves cleaning, transforming, and preparing the data to make it suitable for analysis and modeling. It may include handling missing values, scaling features, encoding categorical variables, and more. For a diabetes prediction system, this ensures the data is in a format that the model can work with effectively.

Selecting Relevant Features:

Feature selection is the process of identifying and choosing the most important variables (features) in the dataset that are likely to have a significant impact on the prediction of diabetes. This is an essential step to reduce noise, improve model performance, and decrease computational complexity. It might involve statistical tests, domain knowledge, or model-based techniques to determine feature importance.

Program:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import f_classif

# Step 1: Load the dataset
data = pd.read_csv("diabetes.csv")

# Step 2: Data Preprocessing
X = data.drop("Outcome", axis=1) # Features
y = data["Outcome"] # Target variable

# Step 3: Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Initialize the feature selector
feature_selector = SelectKBest(score_func=f_classif, k=5)

# Fit the selector to the training data
X_train_selected = feature_selector.fit_transform(X_train, y_train)
X_test_selected = feature_selector.transform(X_test)

# Step 5: Create a RandomForestClassifier for model development
model = RandomForestClassifier(random_state=42)

# Step 6: Train the model on the selected features
model.fit(X_train_selected, y_train)
```

Step 7: Make predictions and evaluate the model

```
y_pred = model.predict(X_test_selected)
```

Output to confirm data preprocessing

```
print("Data Preprocessing Completed:")
```

```
print(f"Number of training samples: {len(X_train)}")
```

```
print(f"Number of testing samples: {len(X_test)}")
```

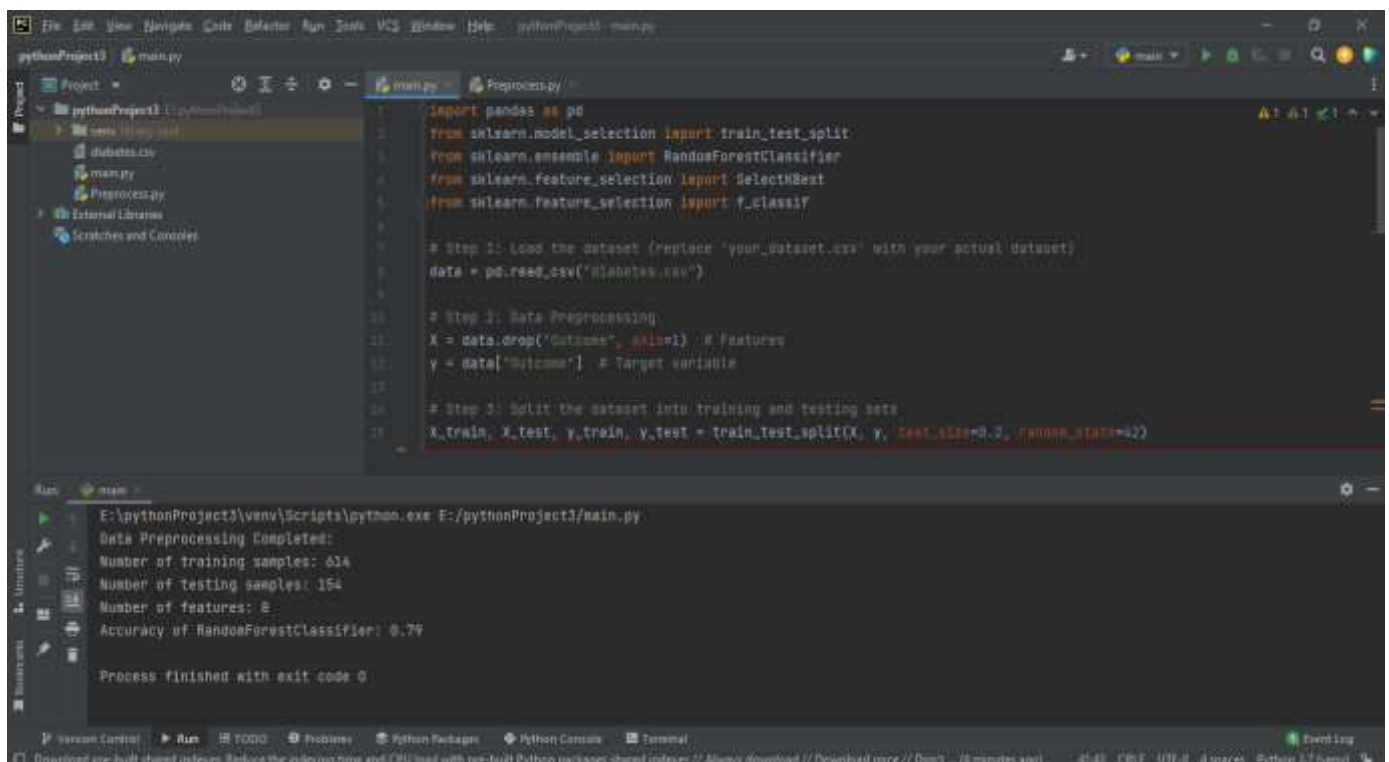
```
print(f"Number of features: {X_train.shape[1]}")
```

Step 8: Model evaluation

```
accuracy = (y_pred == y_test).mean()
```

```
print(f"Accuracy of RandomForestClassifier: {accuracy:.2f}")
```

Output:



```
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.feature_selection import SelectKBest
5 from sklearn.feature_selection import f_classif
6
7 # Step 1: Load the dataset (replace 'your_dataset.csv' with your actual dataset)
8 data = pd.read_csv('diabetes.csv')
9
10 # Step 2: Data Preprocessing
11 X = data.drop('Outcome', axis=1) # Features
12 y = data['Outcome'] # Target variable
13
14 # Step 3: Split the dataset into training and testing sets
15 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
16
17 # Step 4: Feature Selection
18 selector = SelectKBest(score_func=f_classif, k=5)
19 X_train_selected = selector.fit_transform(X_train, y_train)
20 X_test_selected = selector.transform(X_test)
21
22 # Step 5: Model Training
23 model = RandomForestClassifier()
24 model.fit(X_train_selected, y_train)
25
26 # Step 6: Model Evaluation
27 y_pred = model.predict(X_test_selected)
28 accuracy = (y_pred == y_test).mean()
29 print(f"Accuracy of RandomForestClassifier: {accuracy:.2f}")
```

Run: main

E:\pythonProject3\venv\Scripts\python.exe E:/pythonProject3/main.py

Data Preprocessing Completed:
Number of training samples: 614
Number of testing samples: 124
Number of features: 8
Accuracy of RandomForestClassifier: 0.79

Process finished with exit code 0

Conclusion:

This phase provides a high-level overview of the initial steps to be taken in the project, with an emphasis on data preparation and the selection of relevant features. It is a foundational phase that sets the stage for the subsequent steps in building the diabetes prediction system, such as model development, training, evaluation, and deployment.