Diabetes Prediction Report

Phase 3: Development Part 1

In this phase of the project, we began building a machine learning model for diabetes prediction using IBM Cloud Watson Studio. The primary objective was to define the predictive use case, select a relevant dataset, import the dataset, preprocess the data, select features, and train a machine learning model.

Project Overview

- **Predictive Use Case**: Diabetes Prediction

Data Preparation

- **Dataset**: We used the diabetes dataset from an external source.

diabetes_dataset = pd.read_csv('diabetes.csv')

- **Exploratory Data Analysis**:
    - Dataset Shape: The dataset contains 768 rows and 9 columns.
    - Statistical Measures: We explored the statistical measures of the data using **.describe()**.
    - Class Distribution: The target variable 'Outcome' is binary (0 or 1) and represents whether a person is diabetic (1) or not (0).
    - Mean Values by Outcome: We calculated the mean values of features based on the 'Outcome' category to understand how the features correlate with diabetes.

Data Preprocessing

- **Data Standardization**: We standardized the feature data using the **StandardScaler** to ensure that all features have the same scale.

pythonCopy code

scaler = StandardScaler() scaler.fit(X) standardized_data = scaler.transform(X)

Model Building

- **Data Splitting**: We split the data into training and testing sets with a 80/20 ratio while ensuring that the class distribution is maintained.

pythonCopy code

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

- **Model Selection**: We chose the Support Vector Machine (SVM) classifier with a linear kernel as our machine learning algorithm.

pythonCopy code

classifier = svm.SVC(kernel='linear')

- **Model Training**: We trained the SVM classifier on the training data.

pythonCopy code

classifier.fit(X_train, Y_train)

Model Evaluation

- **Training Accuracy**: We calculated the accuracy of the model on the training data.

pythonCopy code

X_train_prediction = classifier.predict(X_train) training_data_accuracy = accuracy_score(X_train_prediction, Y_train)

- **Testing Accuracy**: We calculated the accuracy of the model on the test data.

pythonCopy code

X_test_prediction = classifier.predict(X_test) test_data_accuracy = accuracy_score(X_test_prediction, Y_test)

The accuracy score of the test data was: **Accuracy score of the test data: <test_data_accuracy>**

Prediction

- **Sample Prediction**: We made a sample prediction using a test data point.

pythonCopy code

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

After standardizing the input data and predicting, the result was: **The person is not diabetic** or **The person is diabetic**