DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
VIMAL JYOTHI ENGINEERING COLLEGE, CHEMPERI

PROJECT REPORT

GNEST305 INTRODUCTION TO ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Diabetes Detection using SVM and Decision Trees.

Submitted By: **CHRISTO BABU (VML24EC021)**

DEVAPRIYA DAS P (VML24EC022)

DRUPATH RAMESH (VML24EC023)

GAYATHRI T S (VML24EC024)

Department Of Electronics and Communication Engineering

Vimal Jyothi Engineering College, Chemperi

OCTOBER 2025

1.Introduction

➢ Objective:

The main objective of this project is to develop a machine learning model that can accurately detect the presence of diabetes in patients based on medical data. The project uses the PIMA Indians Diabetes dataset, which includes various health-related parameters such as glucose level, BMI, blood pressure, age, and insulin level.

The goal is to apply two different machine learning algorithms — Support Vector Machine (SVM) and Decision Tree — to train classification models, evaluate their accuracy, and compare their performance. By analyzing the results, the project aims to identify which algorithm performs better for diabetes prediction and how effectively machine learning can be used in medical diagnosis.

➢ Key Features:

- Uses the PIMA Indians Diabetes Dataset, a well-known medical dataset for diabetes prediction.
- Implements two popular machine learning algorithms — Support Vector Machine (SVM) and Decision Tree.
- Includes data preprocessing such as handling missing values and normalizing data for better accuracy.
- Provides visualizations like correlation heatmaps, confusion matrices, and accuracy comparison charts.
- Compares the performance and accuracy of both models to identify the most effective one.

- Demonstrates the use of Python and Scikit-learn for building and evaluating predictive models.

2.Design and Diagrams

The design of the diabetes detection system follows a simple and structured flow of data processing and model training. The system takes medical data as input, preprocesses it, and applies machine learning models to predict whether a person is diabetic or not.

Data Collection: The PIMA Indians Diabetes dataset is used as the main source of data.

Data Preprocessing: Missing or invalid values are handled, and the data is normalized for accurate model training.

Model Training: Two models — Support Vector Machine (SVM) and Decision Tree — are trained using the processed data.

Model Testing: The trained models are tested using a separate set of data to evaluate their performance.

Result Analysis: The results are analyzed using performance metrics like accuracy, confusion matrix, and visual graphs.

➢ Data Flow Diagram – Project Steps

Start
↓
Load Dataset
↓
Preprocess Data
↓
Split Data (Train/Test)
↓
Train Model (SVM and Decision Tree)
↓
Test Model
↓
Evaluate and Compare Results
↓

Display Graphs and Accuracy
↓
End

3.Implementation

The implementation of the Diabetes Detection project is carried out using the Python programming language along with machine learning libraries such as Scikit-learn, Pandas, NumPy, Matplotlib, and Seaborn. The PIMA Indians Diabetes dataset from Kaggle is used for model training and testing.

The following steps were followed during the implementation:

Data Loading:

The dataset was imported using Pandas and examined to understand its structure, number of records, and available features.

Data Preprocessing:

Missing or zero values in medical parameters such as glucose, blood pressure, and BMI were handled or replaced. The data was then normalized to improve model accuracy.

Data Splitting:

The dataset was divided into two parts — training data (80%) and testing data (20%) — using the train_test_split() function from Scikit-learn.

Model Training:

Two machine learning algorithms were applied:

- Support Vector Machine (SVM): Finds the best hyperplane that separates data points into diabetic and non-diabetic classes.

- Decision Tree: Creates a tree-like model of decisions based on different features of the dataset.

Model Evaluation:

After training, both models were tested on unseen data to check their prediction accuracy. Evaluation metrics like accuracy score, classification report, and confusion matrix were used.

Result Visualization:

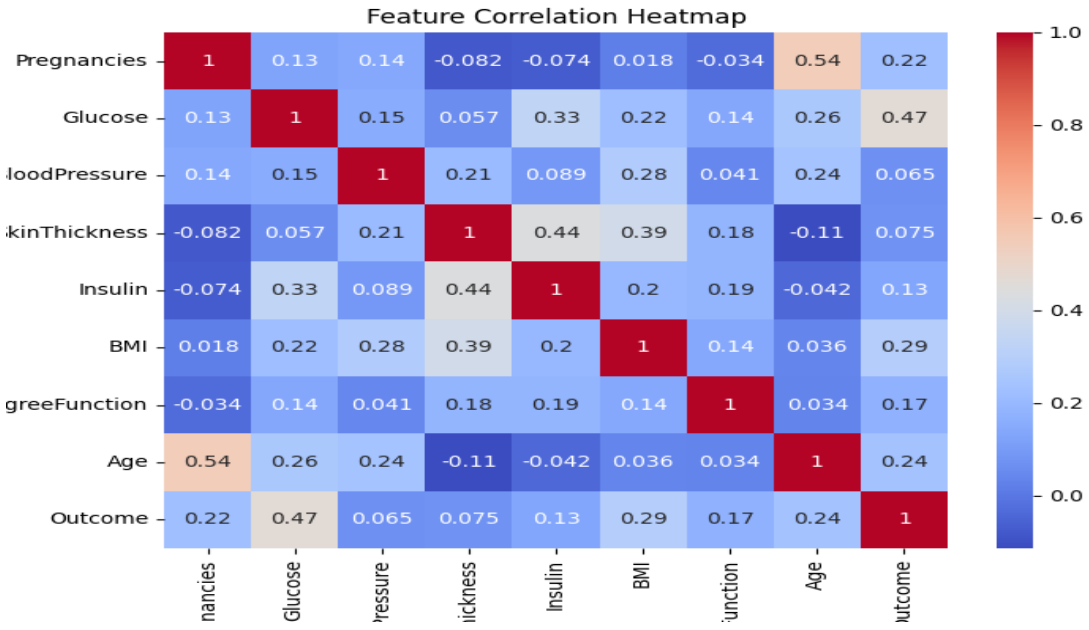
Graphs such as correlation heatmap, confusion matrices, accuracy comparison charts, and decision tree diagram were plotted using Matplotlib and Seaborn for better interpretation.

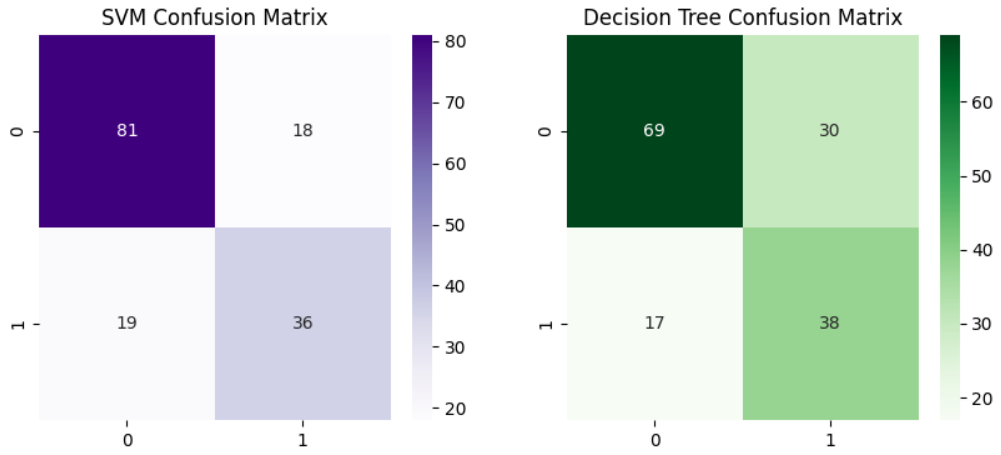
CODE:

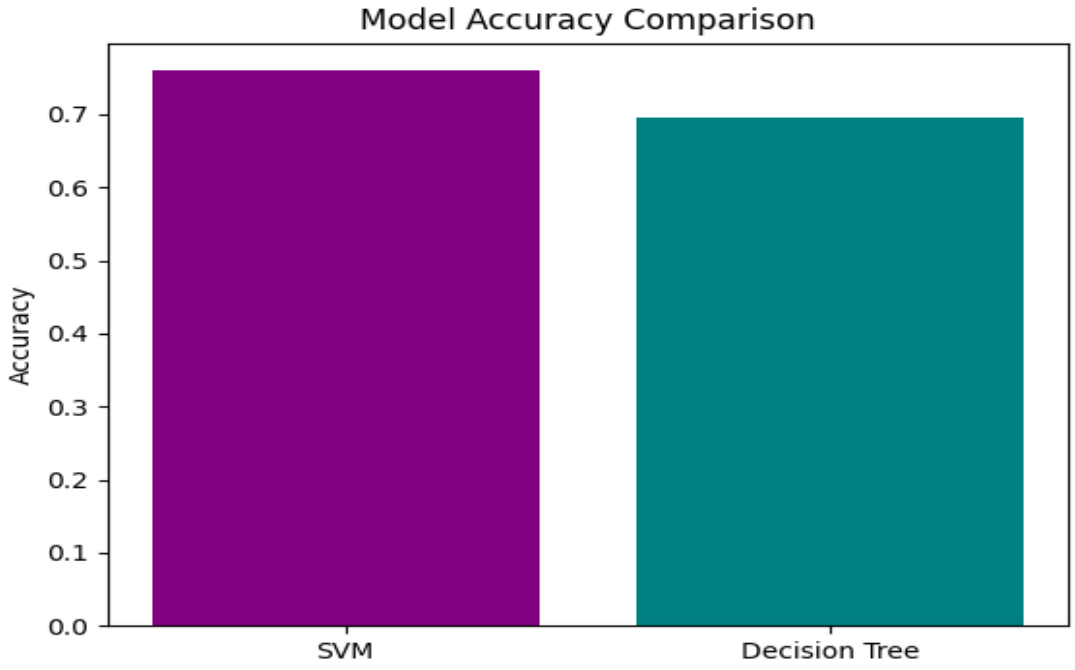
```
# Diabetes Detection using SVM and Decision Tree
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report

# Load dataset
df = pd.read_csv("diabetes.csv")

# Show dataset info
print("Dataset shape:", df.shape)
print(df.head(), "\n")

# Check for missing values
print("Missing values:\n", df.isnull().sum(), "\n")
```

```
# Heatmap
plt.figure(figsize=(8,6))
sns.heatmap(df.corr(), annot=True, cmap="coolwarm")
plt.title("Feature Correlation Heatmap")
plt.show()

# Split data
X = df.drop("Outcome", axis=1)
y = df["Outcome"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Scale data
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train models
svm_model = SVC(kernel='linear')
svm_model.fit(X_train, y_train)
svm_pred = svm_model.predict(X_test)

dt_model = DecisionTreeClassifier(random_state=42, max_depth=4)
dt_model.fit(X_train, y_train)
dt_pred = dt_model.predict(X_test)

# Evaluate
print("SVM Accuracy:", accuracy_score(y_test, svm_pred))
print("Decision Tree Accuracy:", accuracy_score(y_test, dt_pred))
print("\nSVM Report:\n", classification_report(y_test, svm_pred))
print("\nDecision Tree Report:\n", classification_report(y_test, dt_pred))

# Confusion matrices
fig, ax = plt.subplots(1, 2, figsize=(10,4))
sns.heatmap(confusion_matrix(y_test, svm_pred), annot=True, fmt='d',
cmap="Purples", ax=ax[0])
ax[0].set_title("SVM Confusion Matrix")
sns.heatmap(confusion_matrix(y_test, dt_pred), annot=True, fmt='d',
cmap="Greens", ax=ax[1])
ax[1].set_title("Decision Tree Confusion Matrix")
plt.show()

# Accuracy comparison
models = ['SVM', 'Decision Tree']
accuracy = [accuracy_score(y_test, svm_pred), accuracy_score(y_test, dt_pred)]
```

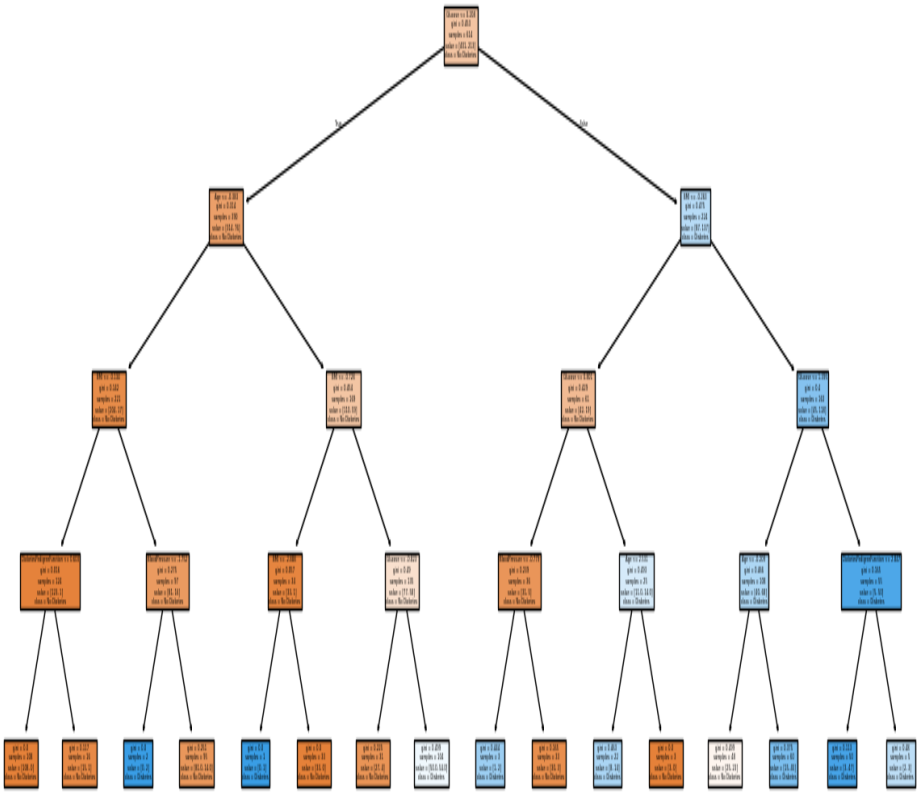
```
plt.bar(models, accuracy, color=['purple','teal'])
plt.title('Model Accuracy Comparison')
plt.ylabel('Accuracy')
plt.show()

# Visualize decision tree
plt.figure(figsize=(15,8))
plot_tree(dt_model, filled=True, feature_names=df.columns[:-1], class_names=["No
Diabetes","Diabetes"])
plt.title("Decision Tree Visualization")
plt.show()
```




SVM Confusion Matrix
81
18
19
36
0
1
80
70
60
50
40
30
20
Decision Tree Confusion Matrix
69
30
17
38
0
1
60
50
40
30
20


Model Accuracy Comparison
Accuracy
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
SVM
Decision Tree

Decision Tree Visualization



```
Dataset shape: (768, 9)
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  DiabetesPedigreeFunction  Age  Outcome
0            6      148             72             35        0  33.6                     0.627   50        1
1            1       85             66             29        0  26.6                     0.351   31        0
2            8      183             64              0        0  23.3                     0.672   32        1
3            1       89             66             23       94  28.1                     0.167   21        0
4            0      137             40             35      168  43.1                     2.288   33        1

Missing values:
 Pregnancies                 0
Glucose                      0
BloodPressure                0
SkinThickness                0
Insulin                      0
BMI                          0
DiabetesPedigreeFunction     0
Age                          0
Outcome                      0
dtype: int64
```

```
SVM Accuracy: 0.7597402597402597
Decision Tree Accuracy: 0.6948051948051948

SVM Report:
              precision    recall  f1-score   support

           0       0.81      0.82      0.81        99
           1       0.67      0.65      0.66        55

    accuracy                           0.76       154
   macro avg       0.74      0.74      0.74       154
weighted avg       0.76      0.76      0.76       154


Decision Tree Report:
              precision    recall  f1-score   support

           0       0.80      0.70      0.75        99
           1       0.56      0.69      0.62        55

    accuracy                           0.69       154
   macro avg       0.68      0.69      0.68       154
weighted avg       0.72      0.69      0.70       154
```

4.Result and Conclusion

Results:

After training and testing both models using the PIMA Indians Diabetes dataset, the results showed that both the Support Vector Machine (SVM) and Decision Tree algorithms were able to predict the presence of diabetes with good accuracy.

The SVM model achieved slightly higher accuracy compared to the Decision Tree model. This indicates that SVM was able to create a better separation between diabetic and non-diabetic cases in the given data. The Decision Tree model, on the other hand, was easier to interpret and visualize, showing the logical decision paths based on features like glucose level, BMI, and age.

The confusion matrices for both models showed that most predictions were correct, with only a few misclassifications. The accuracy comparison bar chart visually confirmed that SVM performed marginally better.

From the correlation heatmap, it was observed that glucose level and BMI had the strongest correlation with the presence of diabetes, while other parameters like skin thickness and insulin showed moderate influence.

Overall, both models performed well, demonstrating that machine learning techniques can be effectively used for medical data analysis and disease prediction.

Conclusion:

The Diabetes Detection project successfully demonstrates how machine learning techniques can be used to predict diabetes based on medical data. Using the PIMA Indians Diabetes dataset, two algorithms — Support Vector Machine (SVM) and Decision Tree — were trained and tested.
Both models were able to classify diabetic and non-diabetic cases effectively, but their performance varied slightly depending on the data distribution and parameters. The

comparison of accuracy and confusion matrices helped in understanding the strengths of each model.

Overall, the project highlights the importance of data preprocessing and model selection in achieving accurate predictions. It also shows how machine learning can assist in medical diagnosis and help in early detection of diseases like diabetes

Evaluation Rubrics

Assignment : 15 Marks

No	Parameters	Outstanding (5 Marks)	Very Good (4 Marks)	Need Improvements (< 3 Marks)
1	Model Implementation	Provides a clear and accurate implementation to develop models.	Provides a favorable implementation to develop models.	Incorrect or weak implementation
2	Result Analysis & Report	Provides correct output for model development. The report should include project objectives, project implementation and conclusion.	Provides imprecise output for model development. The report should include project objectives, project implementation and imprecise conclusion.	Provides fuzzier output for model development. The report should include project objectives, imprecise project implementation and conclusion.
3	Team Collaboration	Clearly coordinated, well-documented teamwork.	Some collaboration, uneven participation.	Poor teamwork or unclear roles.