Artificial Intelligence

Lab Tasks # 12

Name: Samreen Bibi

Sap Id: 46484

Batch: BSCS-6th semester

Lab Instructor: Mam Ayesha Akram

Task # 01

```python
# Import libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv('C:/Users/Lenovo/Downloads/svm_dataset.csv')
data.columns = [col.strip() for col in data.columns]  # Clean extra spaces

# Feature and label
X = data[['Feature1 (X1)', 'Feature2 (X2)']].values
y = data['Label (Y)'].values

# Split into training and testing data (70% train, 30% test)
X_train, X_test, y_train, y_test = train_test_split( *arrays: X, y, test_size=0.3, random_state=42)

# Create a pipeline that includes scaling and the SVM model
pipeline = make_pipeline( *steps: StandardScaler(), SVC(kernel='linear'))
```

```python
# Train the model
pipeline.fit(X_train, y_train)

# Predict and evaluate accuracy
accuracy = pipeline.score(X_test, y_test)
print("Accuracy:", accuracy)

# Plotting
plt.scatter(X[:, 0], X[:, 1], c=y, cmap='coolwarm')
ax = plt.gca()
xlim = ax.get_xlim()
ylim = ax.get_ylim()

# Create grid to evaluate model
xx = np.linspace(xlim[0], xlim[1])
yy = np.linspace(ylim[0], ylim[1])
YY, XX = np.meshgrid( *xi: yy, xx)
xy = np.vstack([XX.ravel(), YY.ravel()]).T
Z = pipeline.decision_function(xy).reshape(XX.shape)

# Plot decision boundary and margins
ax.contour( *args: XX, YY, Z, colors='k', levels=[-1, 0, 1], alpha=0.5,
           linestyles=['--', '-', '--'])
```

```
plt.title('SVM Decision Boundary with Pipeline')
plt.xlabel('Feature 1 (X1)')
plt.ylabel('Feature 2 (X2)')
plt.show()
```

Output:

```
Accuracy: 1.0
```