EXPERIMENT-6

A python program to do face recognition using SVM classifier.

AIM:

To implement a python program to do face recognition using SVM classifier.

CODE:

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn import svm

from sklearn.metrics import confusion_matrix,
classification_report

from sklearn.model_selection import train_test_split


# Set Seaborn style

sns.set(font_scale=1.2)


# Load dataset
```

```
recipes =
pd.read_csv('/content/recipes_muffins_cupcakes.csv')


# Display first few rows

print(recipes.head())


# Show shape

print("Shape of dataset:", recipes.shape)
```

OUTPUT

	Type	Flour	Milk	Sugar	Butter	Egg	Baking Powder	Vanilla	Salt
0	Muffin	55	28	3	7	5	2	0	0
1	Muffin	47	24	12	6	9	1	0	0
2	Muffin	47	23	18	6	4	1	0	0
3	Muffin	45	11	17	17	8	1	0	0
4	Muffin	50	25	12	6	5	2	1	0

```
Shape of dataset: (20, 9)
```

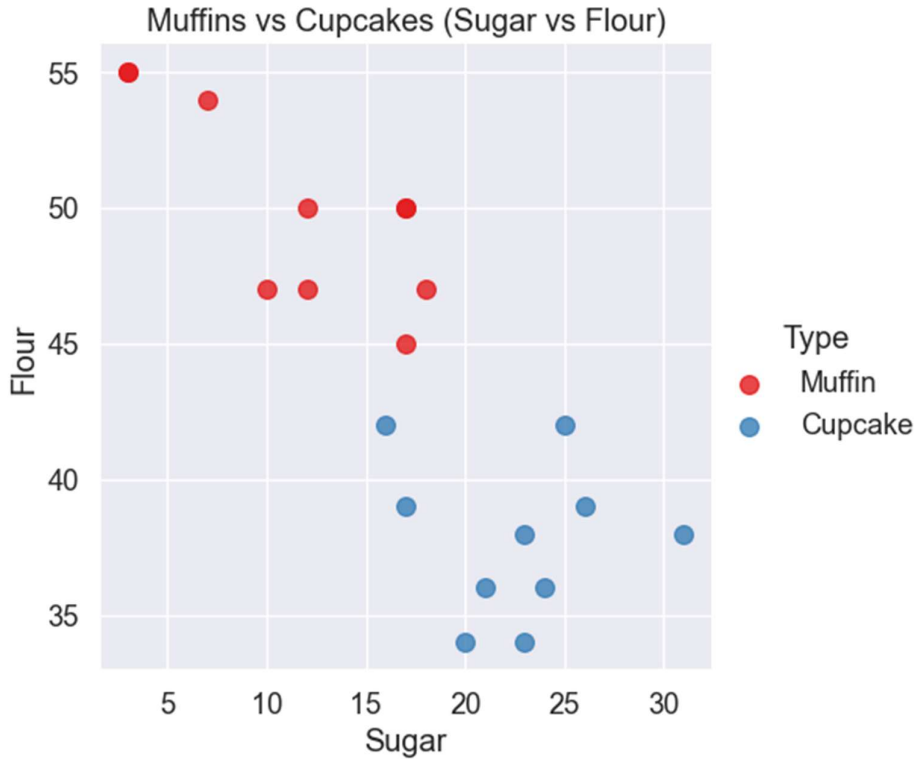
CODE:

```
sns.lmplot(x='Sugar', y='Flour', data=recipes, hue='Type',
        palette='Set1', fit_reg=False, scatter_kws={"s":70})

plt.title("Muffins vs Cupcakes (Sugar vs Flour)")
```

plt.show()

OUTPUT:



CODE:

```
# Select features and labels
sugar_flour = recipes[['Sugar', 'Flour']].values
type_label = np.where(recipes['Type'] == 'Muffin', 0, 1)

# Create and train SVM model
model = svm.SVC(kernel='linear')
model.fit(sugar_flour, type_label)
```

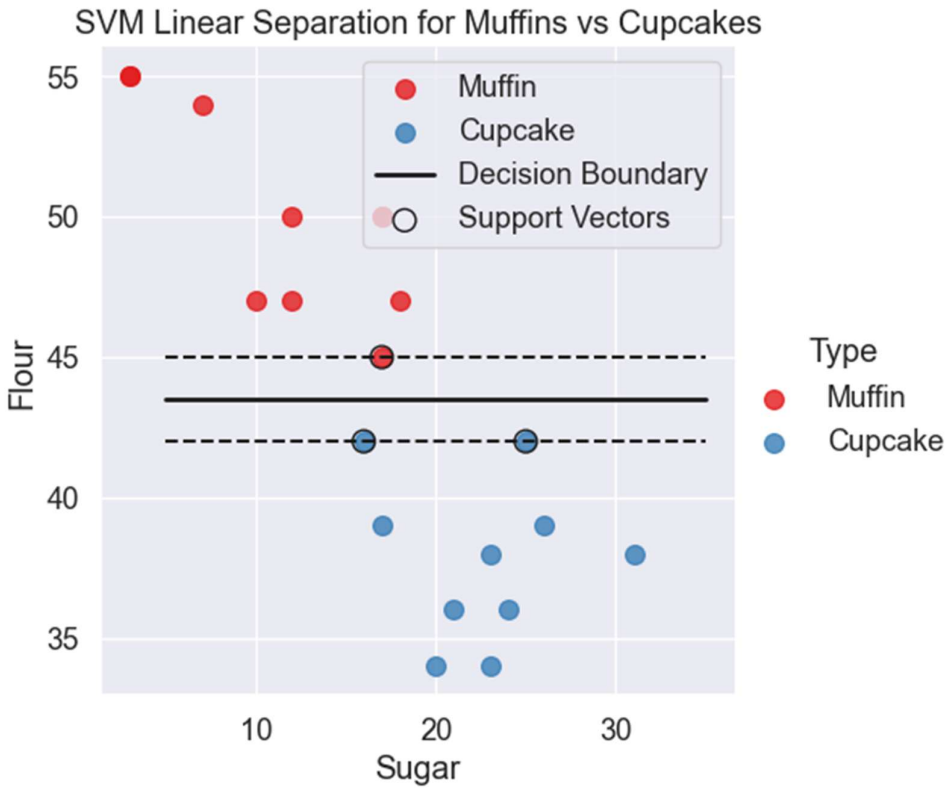
```
# Extract hyperplane parameters
w = model.coef_[0]
a = -w[0] / w[1]
xx = np.linspace(5, 35)
yy = a * xx - (model.intercept_[0] / w[1])

# Calculate margins
b_down = model.support_vectors_[0]
yy_down = a * xx + (b_down[1] - a * b_down[0])
b_up = model.support_vectors_[-1]
yy_up = a * xx + (b_up[1] - a * b_up[0])

# Plot decision boundary
sns.lmplot(x='Sugar', y='Flour', data=recipes, hue='Type',
palette='Set1',
        fit_reg=False, scatter_kws={"s":70})
plt.plot(xx, yy, 'k-', linewidth=2, label="Decision Boundary")
plt.plot(xx, yy_down, 'k--')
plt.plot(xx, yy_up, 'k--')
plt.scatter(model.support_vectors_[:, 0],
         model.support_vectors_[:, 1],
         s=100, facecolors='none', edgecolors='k', label="Support
Vectors")
```

```
plt.title("SVM Linear Separation for Muffins vs Cupcakes")
plt.legend()
plt.show()
```

OUTPUT:



CODE:

```
# Split dataset
x_train, x_test, y_train, y_test = train_test_split(
    sugar_flour, type_label, test_size=0.2, random_state=42)


# Train model
```

```
model1 = svm.SVC(kernel='linear')
model1.fit(x_train, y_train)

# Make predictions
pred = model1.predict(x_test)
print("Predicted Labels:", pred)

# Confusion Matrix
print("\nConfusion Matrix:")
print(confusion_matrix(y_test, pred))

# Classification Report
print("\nClassification Report:")
print(classification_report(y_test, pred,
target_names=['Muffin', 'Cupcake']))
```

OUTPUT:

```
Predicted Labels: [1 0 1 0]

Confusion Matrix:
[[2 0]
 [0 2]]
```

Classification Report:

	precision	recall	f1-score	support
Muffin	1.00	1.00	1.00	2
Cupcake	1.00	1.00	1.00	2
accuracy			1.00	4
macro avg	1.00	1.00	1.00	4
weighted avg	1.00	1.00	1.00	4

RESULT:

Thus a python program to do face recognition using SVM classifier is written and the output is verified successfully.