

```
# ===== FULL CODE FOR COLAB =====
```

```
# 📌 Upload CSV file
```

```
from google.colab import files
```

```
uploaded = files.upload()
```

```
# 📌 Import libraries
```

```
import numpy as np
```

```
import pandas as pd
```

```
from sklearn import svm
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns; sns.set(font_scale=1.2)
```

```
# 📌 Load dataset
```

```
file_name = list(uploaded.keys())[0]
```

```
recipes = pd.read_csv(file_name)
```

```
print("✅ Dataset Loaded Successfully\n")
```

```
print(recipes.head())
```

```
print("\nShape:", recipes.shape)
```

```
# 📊 Scatter plot of data
```

```
sns.lmplot(x='Sugar', y='Flour', data=recipes, hue='Type',
```

```
          palette='Set1', fit_reg=False, scatter_kws={"s":70})
```

```
plt.title("Muffins vs Cupcakes - Ingredients Distribution")
```

```
plt.show()
```

```
# ----- SVM Training -----
```

```
sugar_flour = recipes[['Sugar','Flour']].values
```

```
type_label = np.where(recipes['Type']=='Muffin', 0, 1)
```

```

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

model.fit(sugar_flour, type_label)

# Hyperplane calculation
w = model.coef_[0]
a = -w[0] / w[1]
xx = np.linspace(min(sugar_flour[:,0]) - 2, max(sugar_flour[:,0]) + 2)
yy = a * xx - (model.intercept_[0] / w[1])

# Support vectors
b = model.support_vectors_[0]
yy_down = a * xx + (b[1] - a * b[0])
b = model.support_vectors_[-1]
yy_up = a * xx + (b[1] - a * b[0])

# 🎨 Plot SVM boundary + support vectors
sns.lmplot(x='Sugar', y='Flour', data=recipes, hue='Type',
           palette='Set1', fit_reg=False, scatter_kws={"s":70})
plt.plot(xx, yy, linewidth=2, color='black')
plt.plot(xx, yy_down, 'k--')
plt.plot(xx, yy_up, 'k--')
plt.scatter(model.support_vectors_[0], model.support_vectors_[1],
            s=120, facecolors='none', edgecolors='k')
plt.title("SVM Separation Boundary (Muffins vs Cupcakes)")
plt.show()

# ----- Model Evaluation -----
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(sugar_flour, type_label,

```

```
test_size=0.2, random_state=42)
```

```
model1 = svm.SVC(kernel='linear')
```

```
model1.fit(x_train, y_train)
```

```
pred = model1.predict(x_test)
```

```
print("\n 💎 Predictions:", pred)
```

```
print("\n 🇮🇹 Confusion Matrix:\n", confusion_matrix(y_test, pred))
```

```
print("\n 📊 Classification Report:\n", classification_report(y_test, pred))
```

```
print("✅ Experiment completed!")
```

Choose Files recipes_muffins_cupcakes.csv

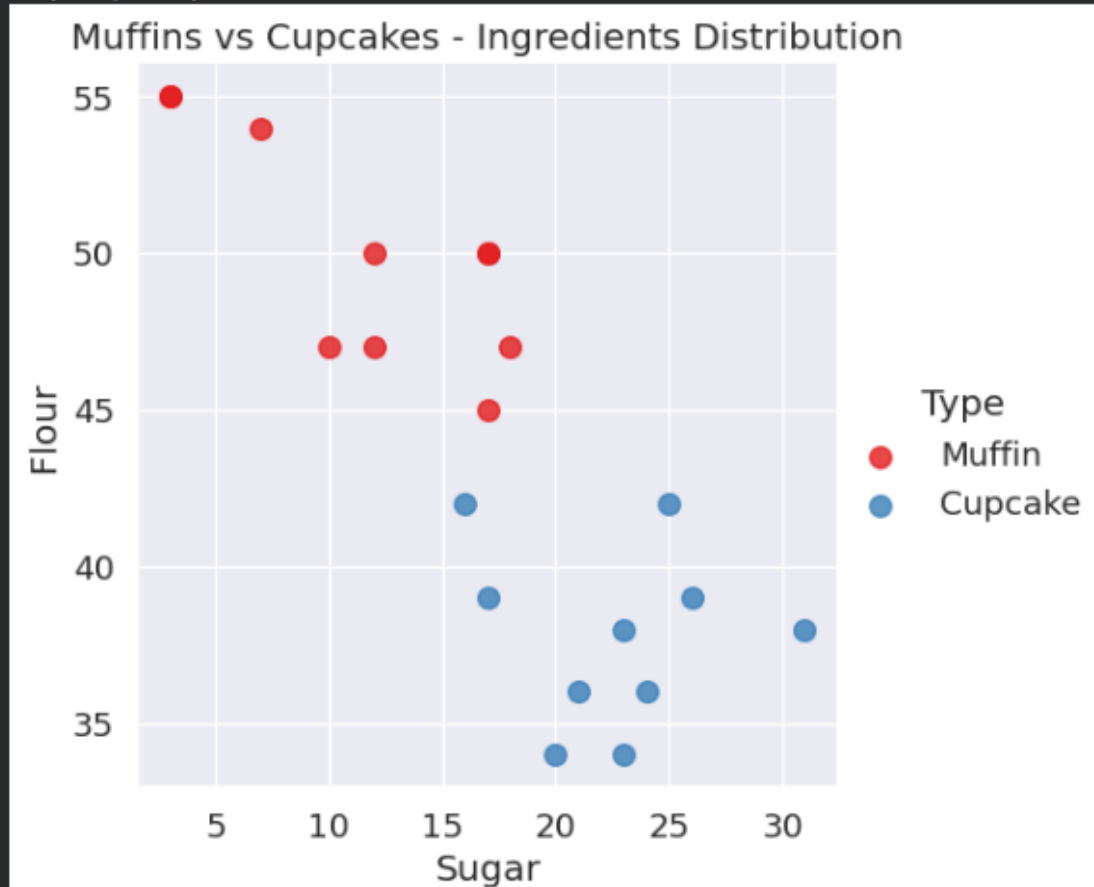
recipes_muffins_cupcakes.csv(text/csv) - 622 bytes, last modified: 11/5/2025 - 100% done

Saving recipes_muffins_cupcakes.csv to recipes_muffins_cupcakes.csv

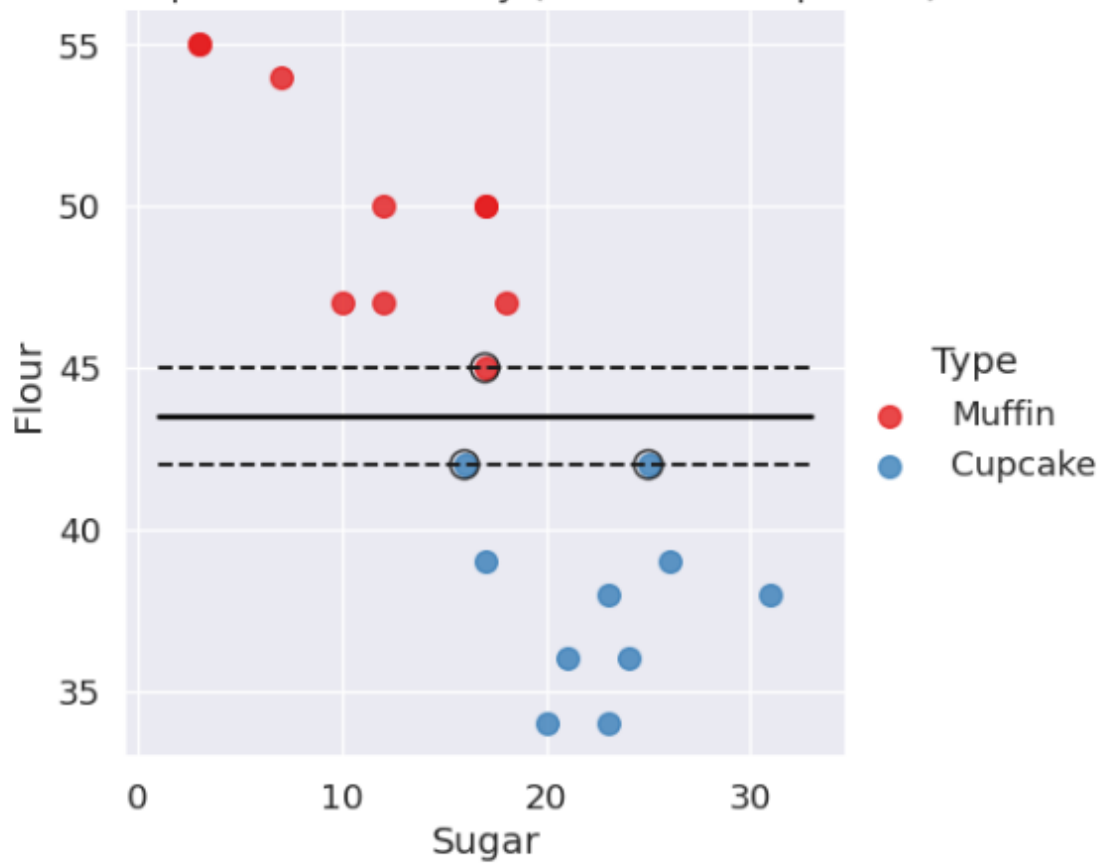
✓ Dataset Loaded Successfully

	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: (20, 9)



SVM Separation Boundary (Muffins vs Cupcakes)



◆ Predictions: [0 1 0 0]

📊 Confusion Matrix:

```
[[2 0]
 [1 1]]
```