SVC:

import pandas as pd

import pickle

from sklearn.svm import SVC

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score

# Step 1: Load your data

df = pd.read\_csv('your\_data.csv')

X = df[['feature1', 'feature2', 'feature3']].values

y = df['target'].values

# Step 2: Train/test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Step 3: Train SVM Classifier

model = SVC(kernel='rbf') # You can change kernel: 'linear', 'poly', 'rbf', 'sigmoid'

model.fit(X\_train, y\_train)

# Step 4: Evaluate

y\_pred = model.predict(X\_test)

print(f"SVM Classification Accuracy: {accuracy\_score(y\_test, y\_pred):.4f}")

# Step 5: Save model

with open('svm\_classifier\_model.pkl', 'wb') as f:

pickle.dump(model, f)

print("SVM Classifier model saved.")

SVR:

import pandas as pd

import pickle

from sklearn.svm import SVR

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import mean\_squared\_error

# Step 1: Load your data

df = pd.read\_csv('your\_data.csv')

X = df[['feature1', 'feature2', 'feature3']].values

y = df['target'].values # continuous value for regression

# Step 2: Train/test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Step 3: Train SVR Model

model = SVR(kernel='rbf') # You can change kernel: 'linear', 'poly', 'rbf', 'sigmoid'

model.fit(X\_train, y\_train)

# Step 4: Evaluate

y\_pred = model.predict(X\_test)

mse = mean\_squared\_error(y\_test, y\_pred)

print(f"SVR Mean Squared Error: {mse:.4f}")

# Step 5: Save model

with open('svr\_model.pkl', 'wb') as f:

pickle.dump(model, f)

print("SVR model saved.")