from sklearn.datasets import load\_iris

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

from sklearn.preprocessing import StandardScaler

from sklearn.svm import SVC

from sklearn.metrics import accuracy\_score

import matplotlib.pyplot as plt

# Load and split data

X, y = load\_iris(return\_X\_y=True)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Proper preprocessing

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Evaluate different kernels

kernels = ['rbf', 'poly', 'sigmoid']

accuracies = []

for k in kernels:

model = SVC(kernel=k, degree=3, random\_state=42)

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

acc = accuracy\_score(y\_test, model.predict(X\_test))

accuracies.append(acc)

print(f"{k.capitalize()} Kernel Accuracy: {acc:.2%}")

# Plot

plt.bar(['RBF', 'Polynomial', 'Sigmoid'], accuracies, color=['blue', 'green', 'red'])

plt.title('SVM Kernel Accuracy Comparison')

plt.ylabel('Accuracy')

plt.show()