#Accuracy Comparison with various models using LSHWGAN

import matplotlib.pyplot as plt

 # Example data for three variables

modules = ['UCSD PED-1', 'Shanghai Tech', 'Custom Data Set']

LSHWGAN = [0, 89, 95.67]

LSGAN = [0, 80, 92.46]

DCGAN = [0, 70, 85.67]

SGAN = [0, 70, 84.45]

 # Plotting plt.figure(figsize=(10, 6))

plt.plot(modules, LSHWGAN, marker='o', color='green', linestyle='-', label='LSHWGAN')

plt.plot(modules, LSGAN, marker='o', color='blue', linestyle='-', label='LSGAN')

plt.plot(modules, DCGAN, marker='o', color='red', linestyle='-', label='DCGAN')

plt.plot(modules, SGAN, marker='o', color='red', linestyle='-', label='SGAN')

 # Adding labels and title

plt.xlabel('Data Sets')

plt.ylabel('Accuracy (%)')

plt.title('Accuracy of Different Models on Various Datasets')

 # Set efficiency range from 0 to 100

plt.ylim(0, 100)

# Adding a legend

plt.legend()

 # Display the plot

plt.show()

#Accuracy Comparison with various models (LSGAN, DCGAN, SGAN, HDCNN, DCNN) using LSHWGAN

#Accuracy Comparison with various models using LSHWGAN

import matplotlib.pyplot as plt

 # Example data for three variables

modules = ['UCSD PED-1', 'Shanghai Tech', 'Custom Data Set']

LSHWGAN = [0, 89, 95.67]

LSGAN = [0, 80, 92.46]

DCGAN = [0, 70, 85.67]

SGAN = [0, 70, 84.45]

HDCNN = [0, 68, 80.02]

DCNN = [0, 66, 78.12]

 # Plotting plt.figure(figsize=(10, 6))

plt.plot(modules, LSHWGAN, marker='o', color='green', linestyle='-', label='LSHWGAN')

plt.plot(modules, LSGAN, marker='o', color='blue', linestyle='-', label='LSGAN')

plt.plot(modules, DCGAN, marker='o', color='red', linestyle='-', label='DCGAN')

plt.plot(modules, SGAN, marker='o', color='red', linestyle='-', label='SGAN')

plt.plot(modules, HDCNN, marker='o', color='Yellow', linestyle='-', label='HDCNN')

plt.plot(modules, DCNN, marker='o', color='magenta', linestyle='-', label='DCNN')

 # Adding labels and title

plt.xlabel('Data Sets')

plt.ylabel('Accuracy (%)')

plt.title('Accuracy of Different Models on Various Datasets')

 # Set efficiency range from 0 to 100

plt.ylim(0, 100)

# Adding a legend

plt.legend()

 # Display the plot

plt.show(