University of Nottingham
UK | CHINA | MALAYSIA

COMP3055 Machine Learning

General Idea for Lab 5

Ying Weng

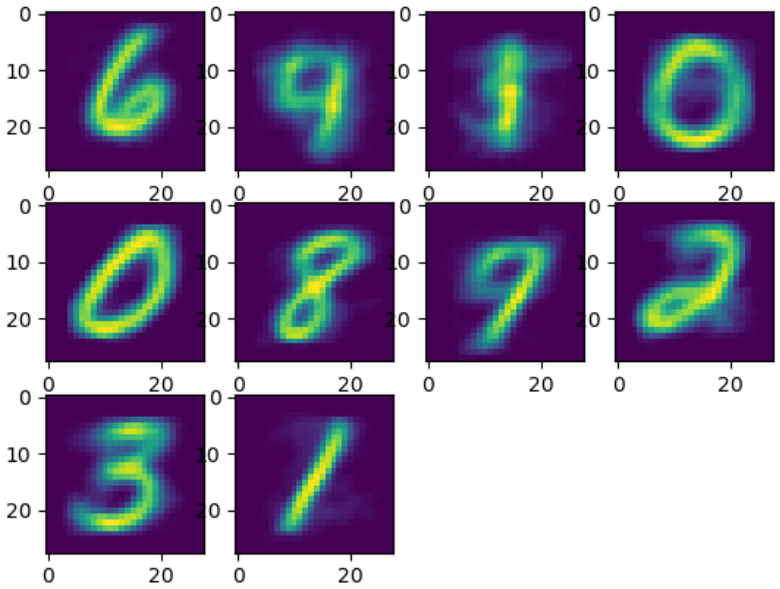
2024 Autumn

Apply K-Means

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

kmeans = KMeans(n_clusters=10, init='k-means++').fit(X_small)  # define a KMeans and train with the small training set
kmeans.labels_ = Y_small  # assign values to kmeans.labels_
centers = np.reshape(kmeans.cluster_centers_, (10, 28, 28))  # extract the trained 10 cluster centers and reshape to the image size for visualization

plt.figure()  # define a figure
# display images with for loop
for i in range(10):
    plt.subplot(3, 4, i + 1)
    plt.imshow(centers[i])
plt.savefig("cluster_centers.png")  # save the figure to 'cluster_centers.png
plt.show()  # display the figure
```

Apply K-Means

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

kmeans = KMeans(n_clusters=10, init='k-means++').fit(X_small)  # define a KMeans and train with the small training set
kmeans.labels_ = Y_small  # assign values to kmeans.labels_
centers = np.reshape(kmeans.cluster_centers_, (10, 28, 28))  # extract the trained 10 cluster centers and reshape to the image size for visualization

plt.figure()  # define a figure
# display images with for loop
for i in range(10):
    plt.subplot(3, 4, i + 1)
    plt.imshow(centers[i])
plt.savefig("cluster_centers.png")  # save
plt.show()  # display the figure
```

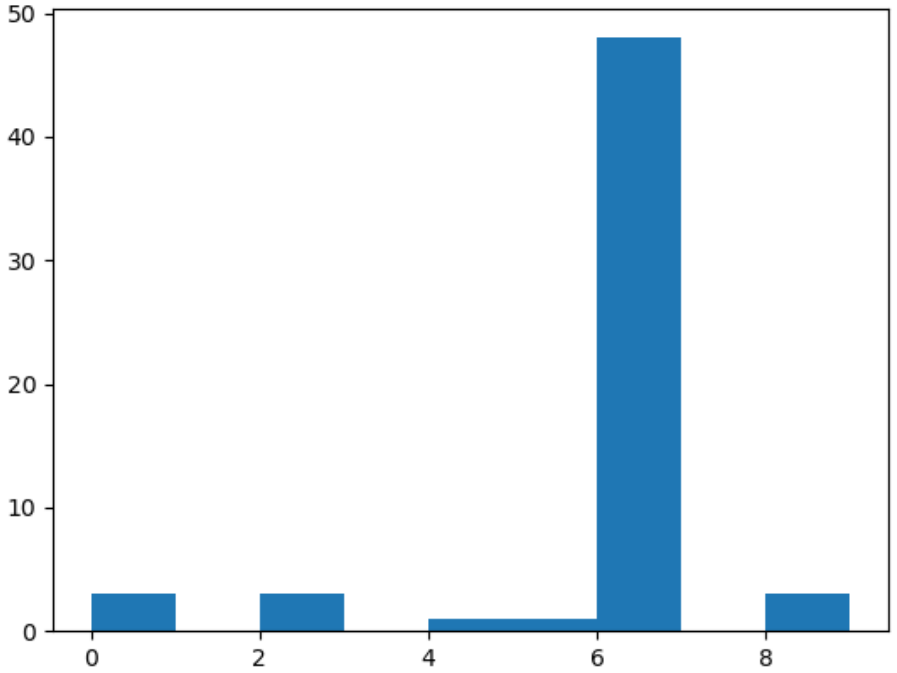


visualize label histogram of one cluster

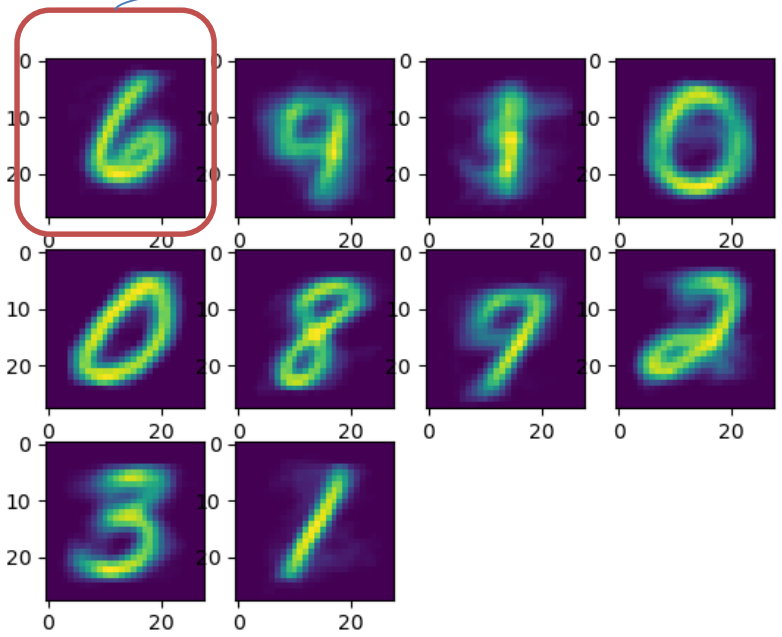
```
'''
visualize label histogram of one cluster
'''
y_cluster = kmeans.predict(X_test)      # prediction on testing data
y_cluster1 = []         # define an empty list y_cluster1
# iterate through the prediction and add the index of predictions belongs to first cluster to
y_cluster1
for i in range(1000):
    if y_cluster[i] == 0:
        y_cluster1.append(Y_test[i])

plt.hist(y_cluster1, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9])    # plot histogram of the ground truth label for
predictions belongs to first cluster
plt.savefig("label_histogram_of_one_cluster.png")      # save the figure to
'label_histogram_of_one_cluster.png
plt.show()            # display the figure
```

visualize label histogram of one cluster

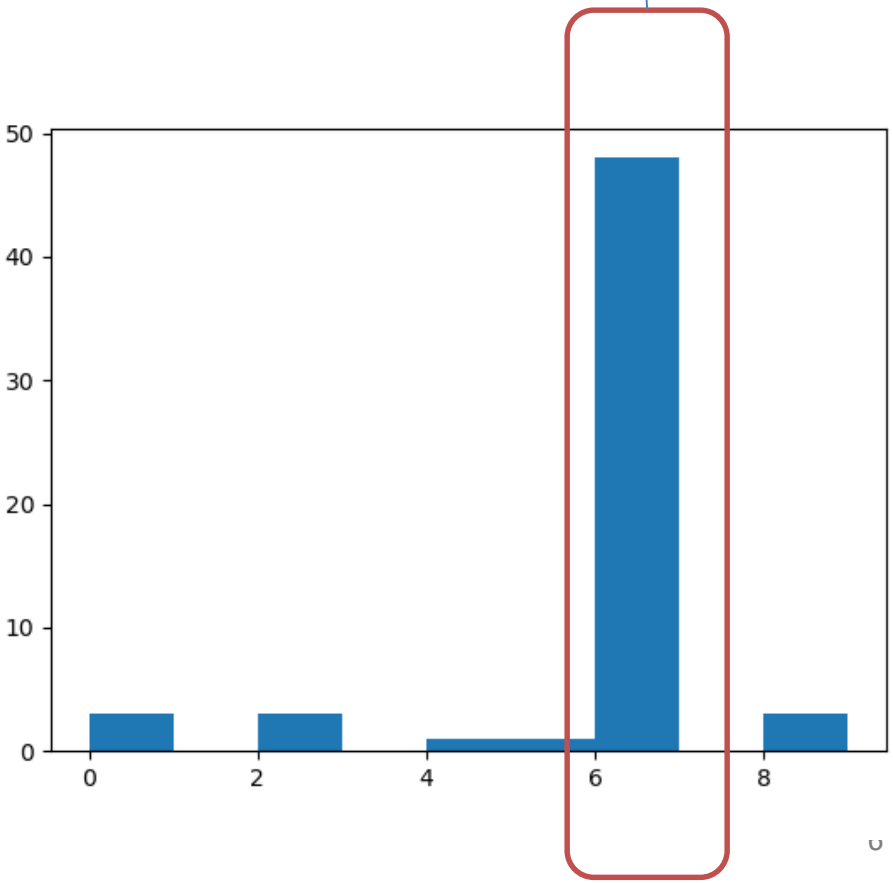
```
'''
visualize label histogram of one cluster
'''
y_cluster = kmeans.predict(X_test)      # prediction on testing data
y_cluster1 = []         # define an empty list y_cluster1
# iterate through the prediction and add the index of predictions belongs to first cluster to
y_cluster1
for i in range(1000):
    if y_cluster[i] == 0:
        y_cluster1.append(Y_test[i])

plt.hist(y_cluster1, [0, 1, 2, 3, 4, 5, 6
predictions belongs to first cluster
plt.savefig("label_histogram_of_on
'label_histogram_of_one_cluster.pr
plt.show()              # display the figu
```



Visualizations



Mostly matches



Any Questions?