Artificial Intelligence

Assignment 3

Presented by:
Ayman Ahmed Abdelaziz

CONTENTS

Summary

Implementing:

1. Loading the dataset
2. Initializing the clusters
3. Initializing the centroids
4. Distortion measure
5. Compute accuracy

Language Used:

- Python

Libraries used:

1. numpy
2. matplotlib
3. gzip

Dataset used:

Handwritten digits by Yan LeCunn

Code

Load data:

```
def load_data(num_images=5000):
    f = gzip.open('./train/train-images-idx3-ubyte.gz','r')
    g=gzip.open('./train/train-labels-idx1-ubyte.gz','r')
    image_size = 28
    f.read(100)
    g.read(8)
    buf = f.read(image_size * image_size * num_images)
    buf2 = g.read(num_images)
    data = np.frombuffer(buf, dtype=np.uint8).astype(np.float32)
    data = data.reshape(num_images, image_size*image_size)
    data = data/255.
    labels=np.frombuffer(buf2, dtype=np.uint8)
    return data,labels
```

Initialize Centroids:

```
def initialize_centroids(k,data):
    centroids=[]
    for i in range(k):
        centroids.append(data[np.random.randint(0,data.shape[0]),:-1])
        centroids[i]=centroids[i].reshape((784,1))
    return centroids
```

Code

Initialize Clusters:

```
def initialize_clusters(k):
    clusters={}
    for i in range(k):
        clusters[str(i)]=np.zeros((785,1))
    return clusters
```

Cluster:

```
def cluster(data,k=10):
    changed=True
    it=0
    distortion_array=[]
    accuracy_array=[]
    centroids=initialize_centroids(k,data)

    while(changed):
        clusters=initialize_clusters(k)
        print("Iteration: "+str(it))
        it+=1
        changed=False
        for i in range (data.shape[0]):
```

Code

```
            minimum=100000
            index=0
            data_numpy=data[i].reshape(785,1)
            for j in range(k):
                temp=abs(np.linalg.norm(data_numpy[:-1,:]-centroids[j]))
                if temp<minimum:
                    index=j
                    minimum=temp
            clusters[str(index)]=np.append(clusters[str(index)],data_numpy,axis=1)
        mean=[]
        for i in range(k):
            curr_mean=np.mean(clusters[str(i)][:-1,:],axis=1)
            curr_mean=curr_mean.reshape((784,1))
            if sum(abs(centroids[i]-curr_mean))>0.0001:
                centroids[i]=curr_mean
                changed=True
            mean.append(curr_mean)
        distortion_var=distortion_measure(clusters,centroids)
        distortion_array.append(distortion_var)
        accuracy_var=compute_accuracy(clusters,k)
        accuracy_array.append(accuracy_var)
        print("Distortion: "+str(distortion_var))
        print("Accuracy: "+str(accuracy_var))
    return clusters,centroids,distortion_array,mean,accuracy_array
```

Code

Compute accuracy:

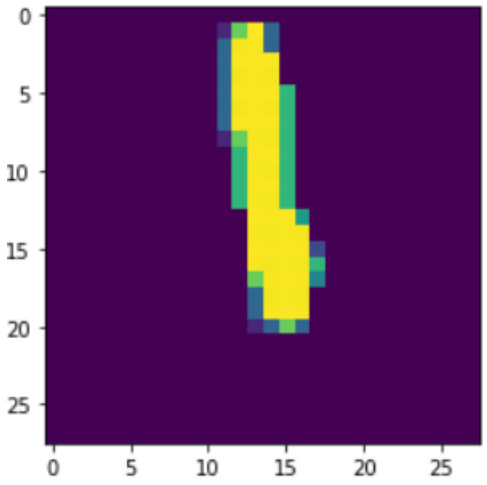
```
def compute_accuracy(clusters,k):
    labels=[]
    accuracy=0
    for i in range(k):
        labels.append(clusters[str(i)][-1,1:])
    for label in labels:
        accuracies=[]
        accuracies.append(0)
        if len(label)>0:
            for i in range(k):
                accuracies.append((sum(label==i))/len(label))
        accuracy+=max(accuracies)
    return accuracy/k
```

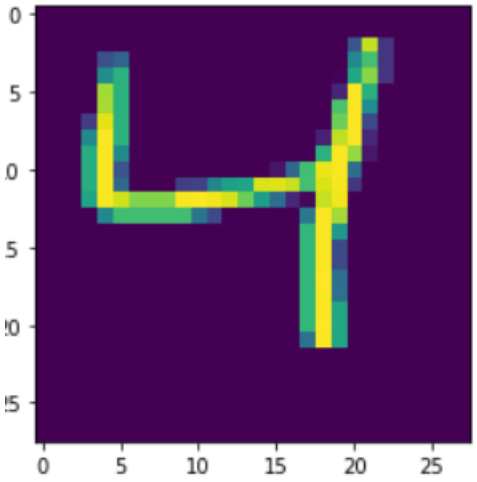
Distortion Measure:

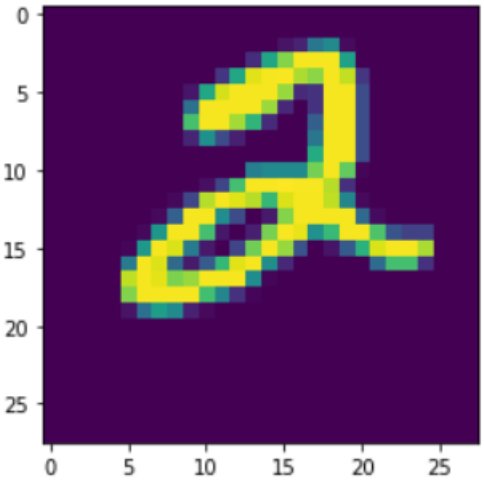
```
def distortion_measure(clusters,centroids):
    mse=0
    for i in range(len(centroids)):
        mse+=np.sum(np.square(clusters[str(i)][:-1,:]-centroids[i]))
    return mse
```

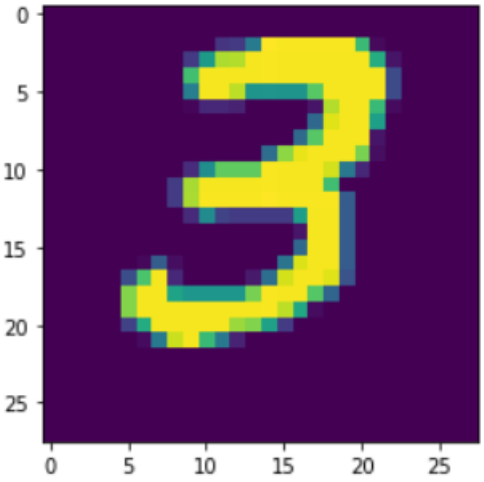
Samples from Dataset

Code



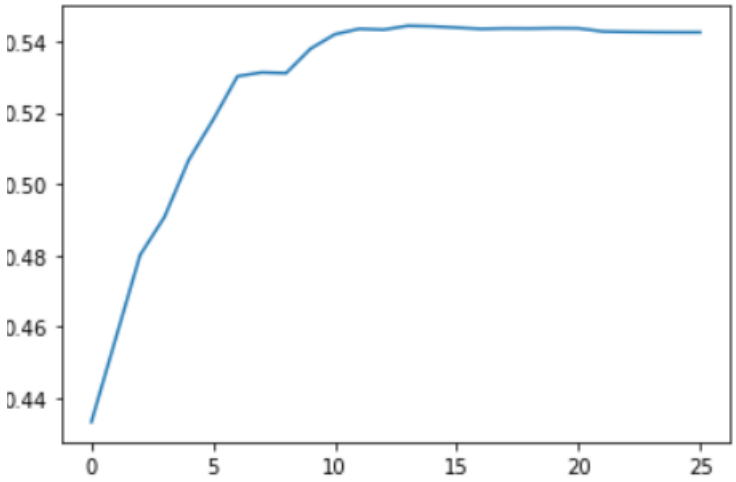




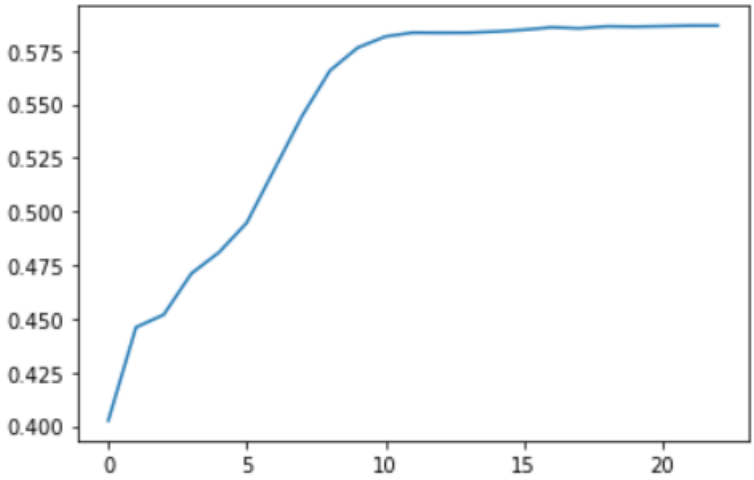


Accuracies

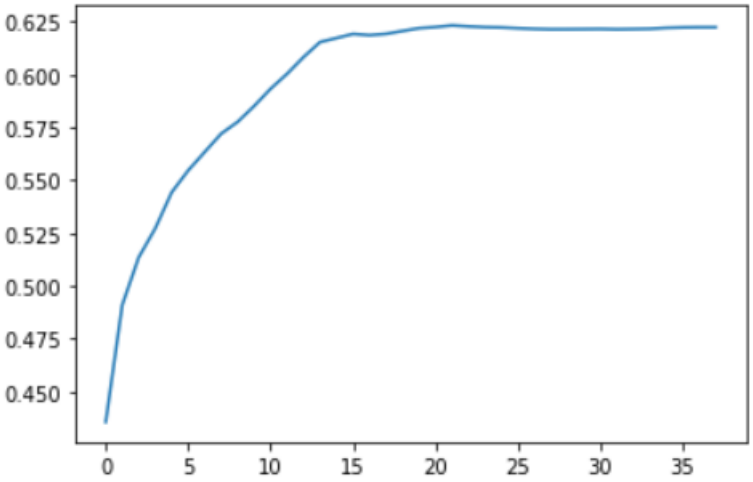
At k=10
Accuracy: 0.5425468542551343



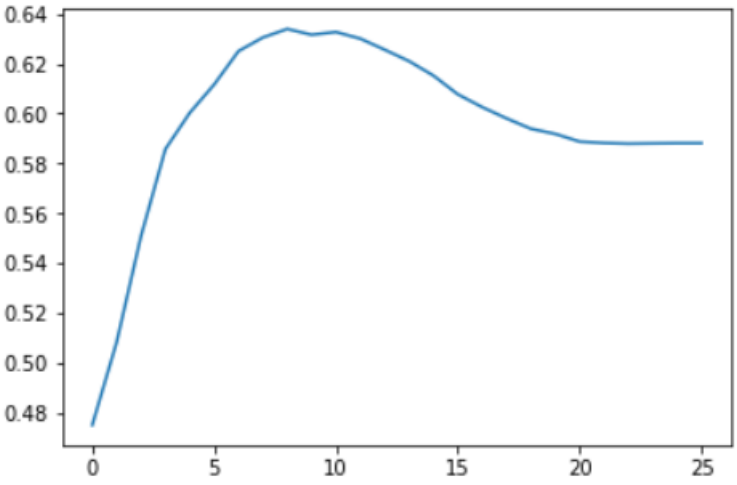
At k=7
Accuracy=0.5867339814511039



At k=12
Accuracy=0.6223558490143932

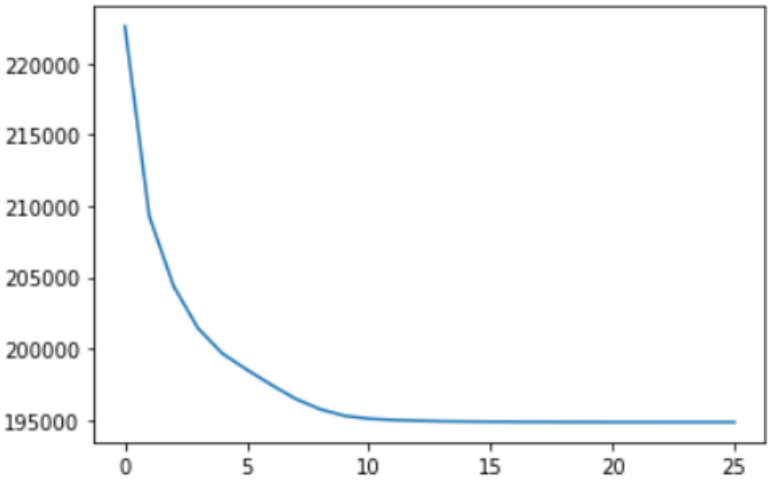


At k=8
Accuracy=0.5881000394588035

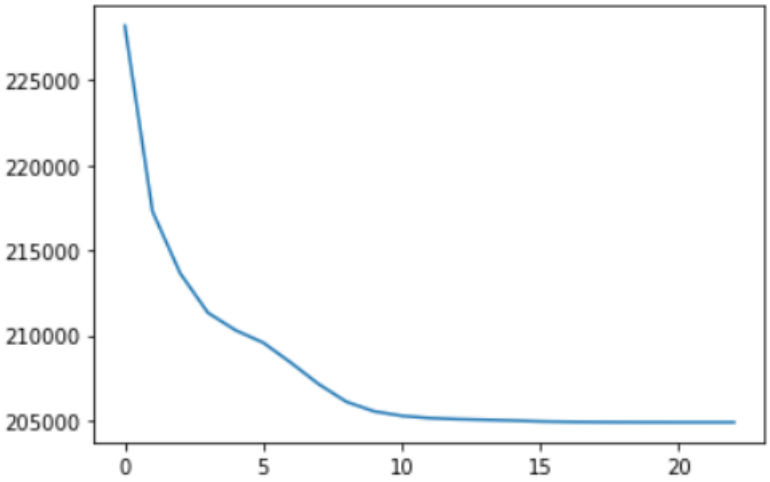


Distortion

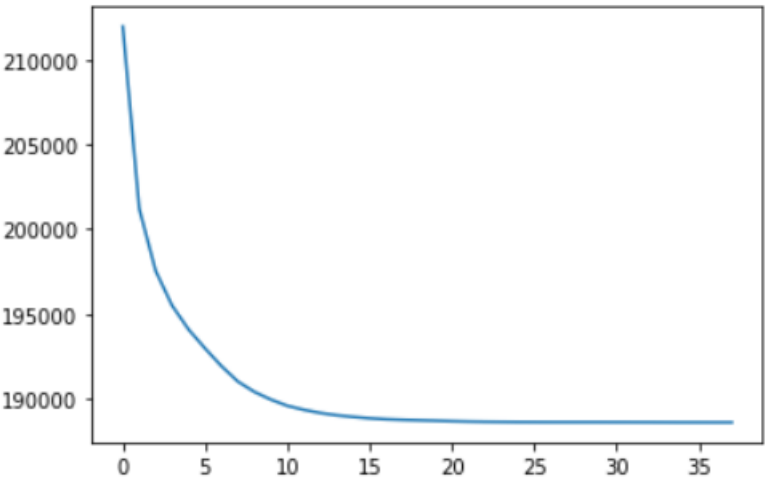
At k=10
Distortion=194867.47507179723



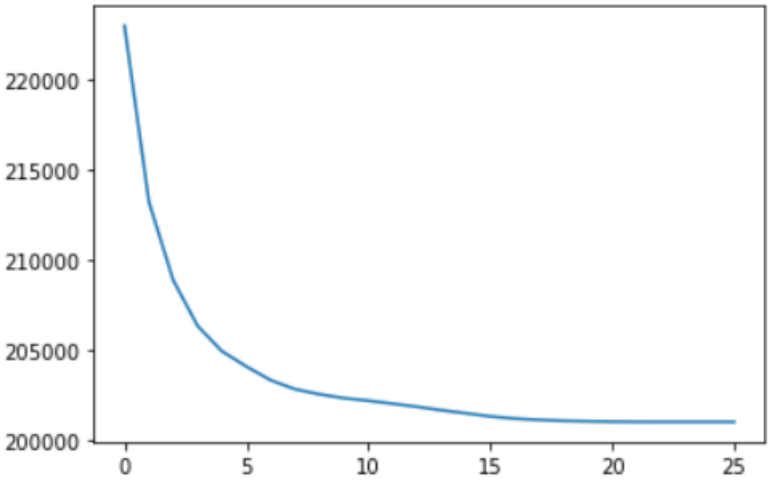
At k=7
Distortion=



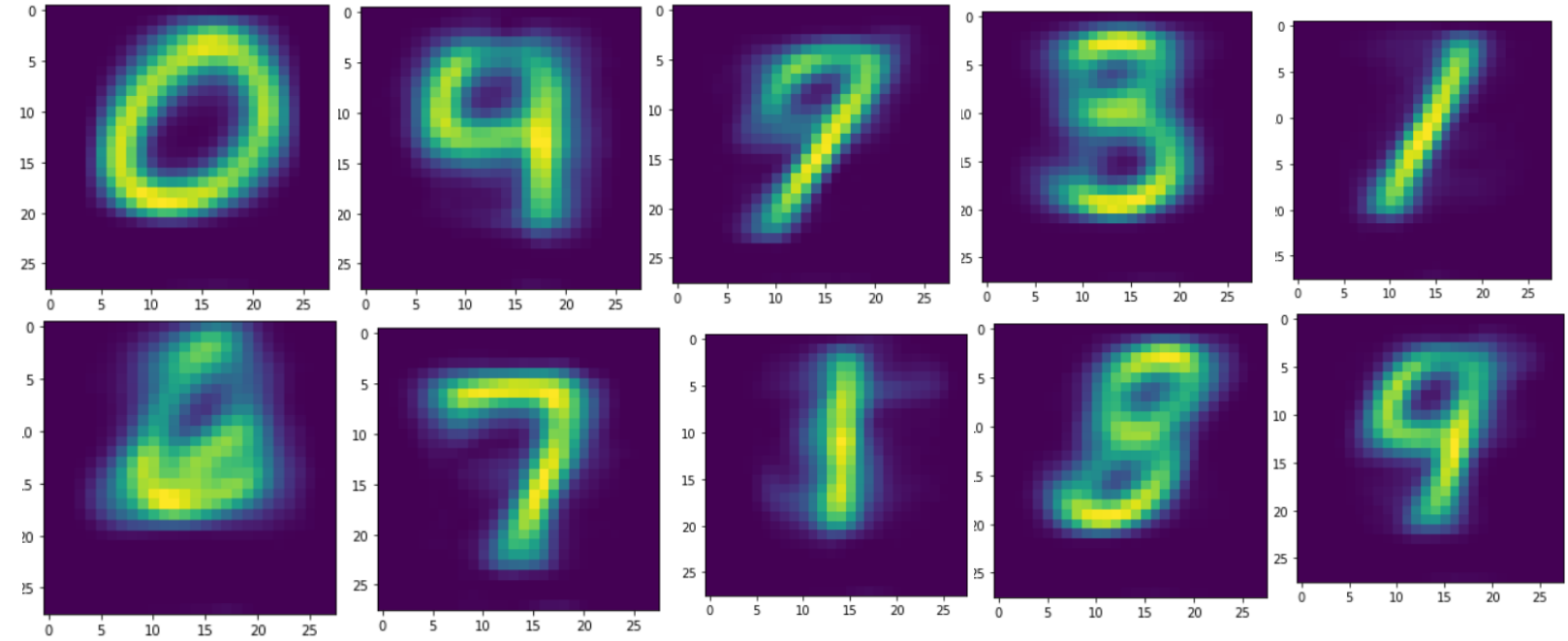
At k=12
Distortion= 188609.7251541138



At k=8
Distortion=200998.57300837908

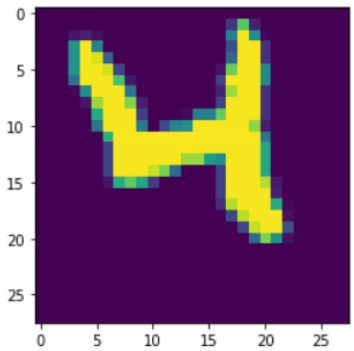


Mean Images At K=10

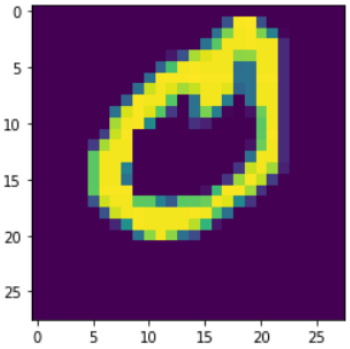


Representative Images from each cluster at K=10

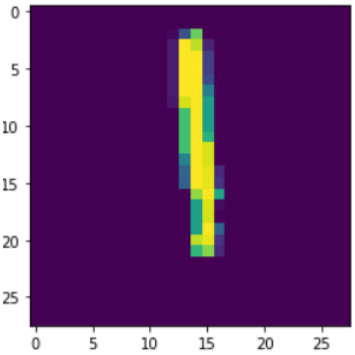
Cluster 9, most labels=4



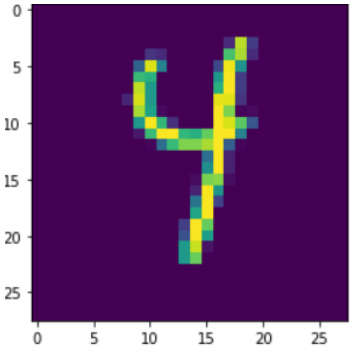
Cluster 0, most labels=0



Cluster 1, most labels=1



Cluster 9
most labels =4



Conclusion

K-Means clustering, groups the images similar to each other in clusters, using the minimum distance to the cluster's centroid, the centroids are chosen randomly from the data as an initial point so the results are different from each restart, and the results differ for each value of K, the K represent the number of clusters, and the number of centroids.
The distortion function measure the variance in each cluster, it never increases as the centroids move from a position to the mean of the cluster.
The Accuracy can decrease over time but not drastically.