**KMeans Clustering**

**MNIST digits Dataset**

**Report By**

**Aliaa Mohamed Ali Abbas 3747**

**Instructor**

**Dr/Marwan el Torki**

**Objective:**

In this project we apply KMeans clustering algorithm to cluster the MNIST digit dataset.

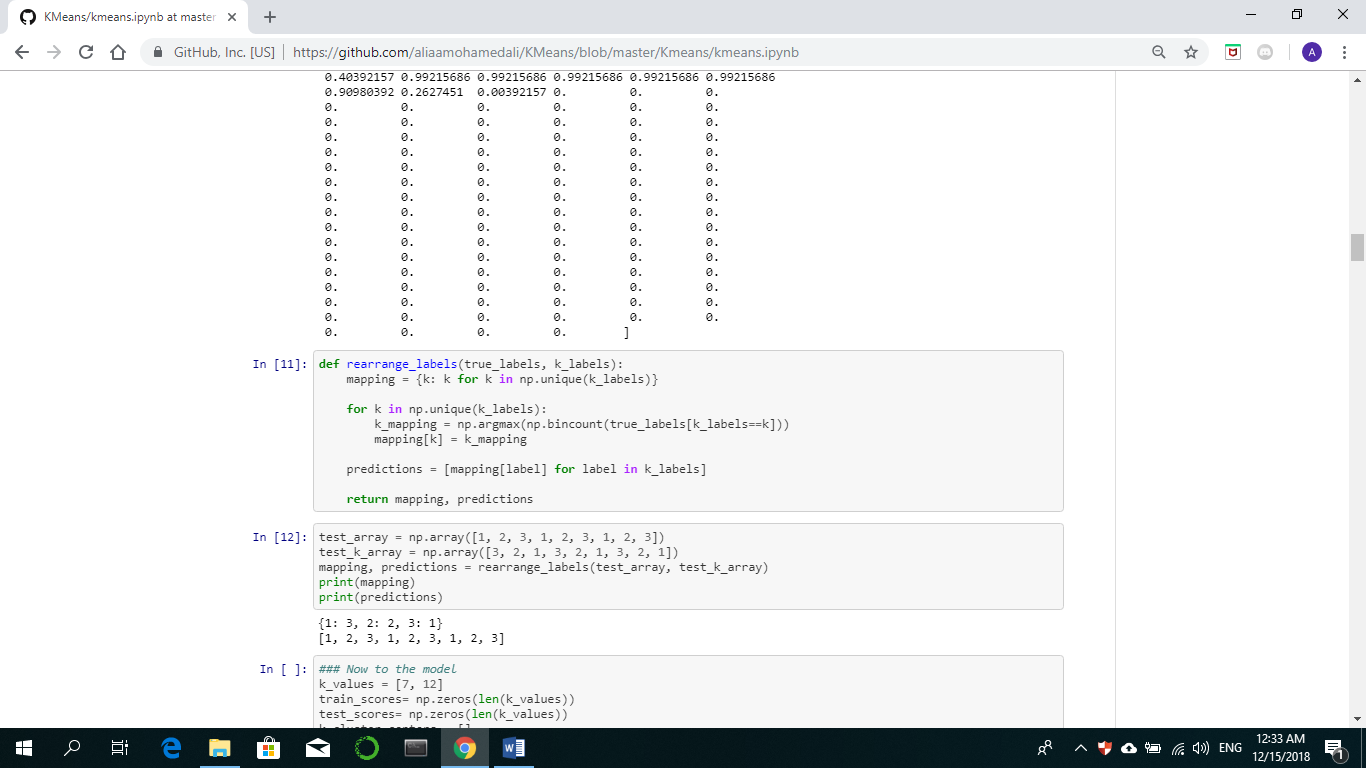
**Introduction:**

Originally the dataset is meant for classification purposes and is so labeled, as clustering by definition is performed on unlabeled datasets we don’t use labels during training at all. As will be shown in results section, the ‘mean images’ obtained by reshaping the cluster centers into the image dimensions show that the centers appear to have taken the pattern of the digits.

To make sure that the centers have done this we test them with the labels (Again, the labels are not used to train the model only to discover what sort of clusters the model found). We primarily test quality using the KMeans inertia (Objective function)

We also map each of our clusters to a label, by counting the max number of images of a certain label in the cluster and assigning that label to it.

The code snippet to map the labels is below, using this we can map a cluster ‘i’ to a label ‘k’ and see if the mean image of the cluster represents the label so we can have an intuition as to the patterns the model recognized:



**Parameters:**

We run the program for values of K (5, 7, 10, 12)

*We re-initialize the centroids 5 times for each value of K*

*We run the program for a maximum number of iterations = 300.*

*The KMeans algorithm aims to minimize inertia, that is distance between points and their cluster centroids.*

**Results:**

For k = 5:

Run1: Max= 2005877.525, Min= 1824012.219

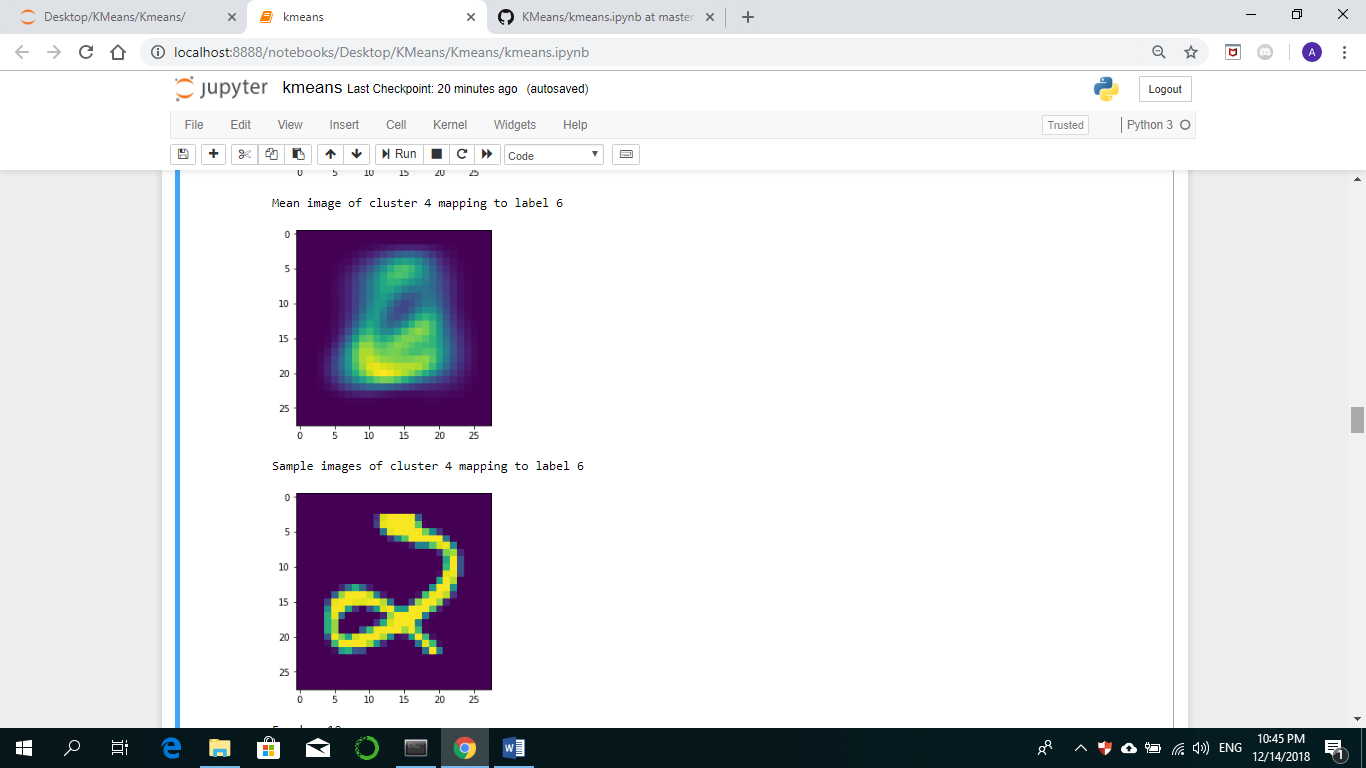
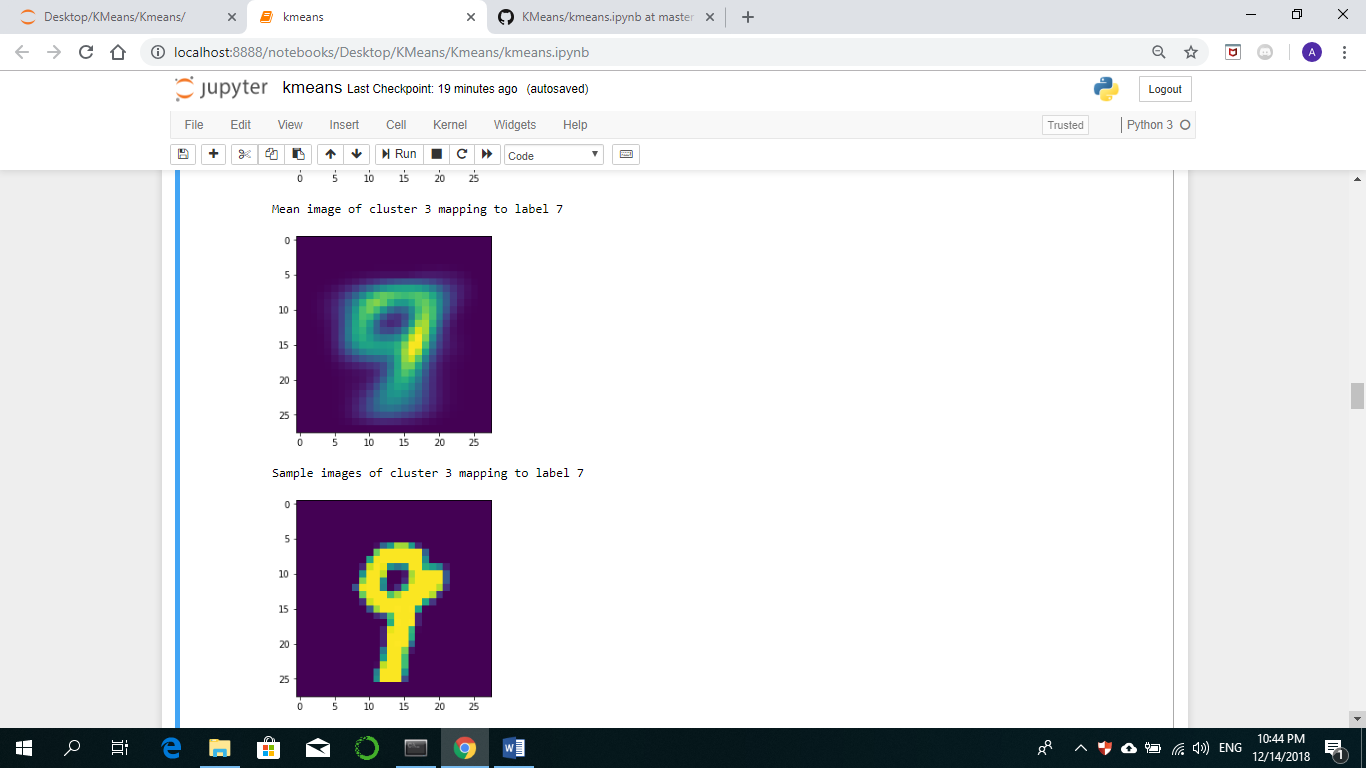
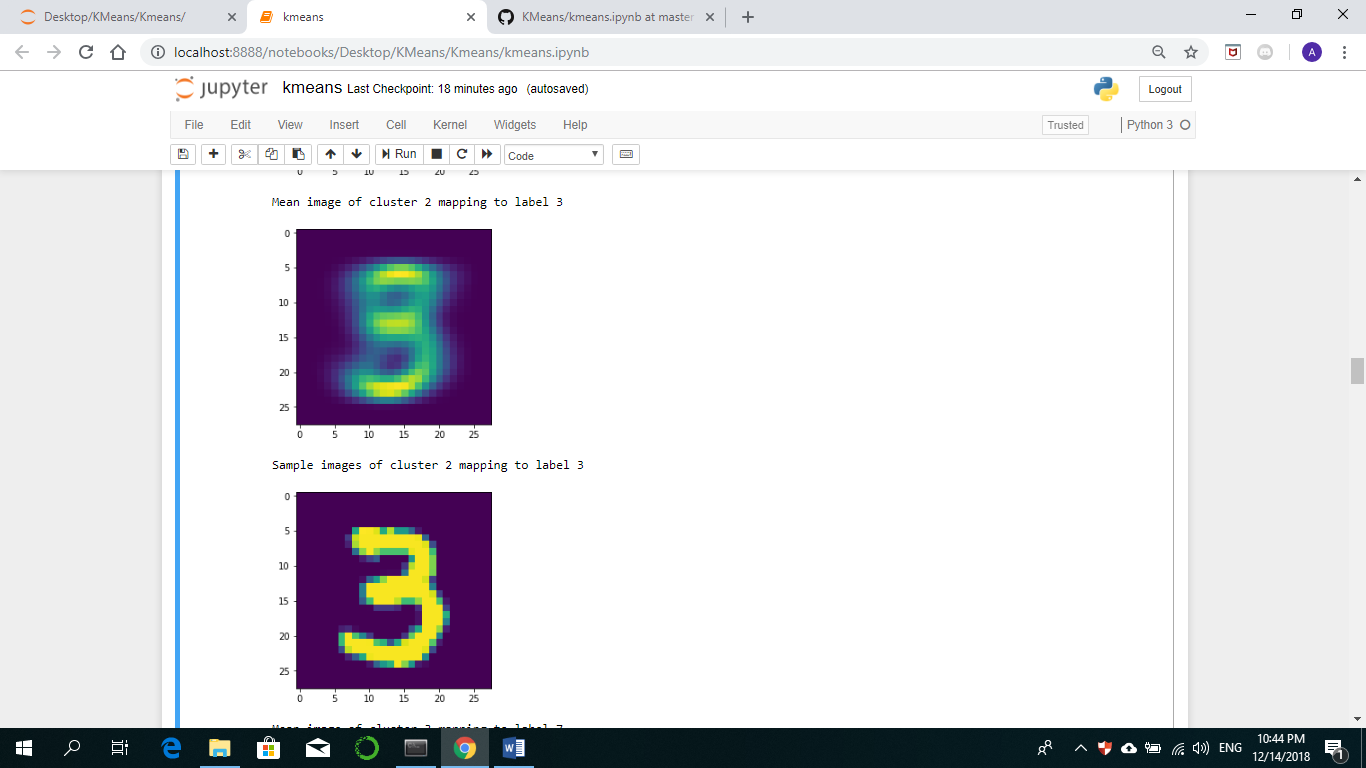
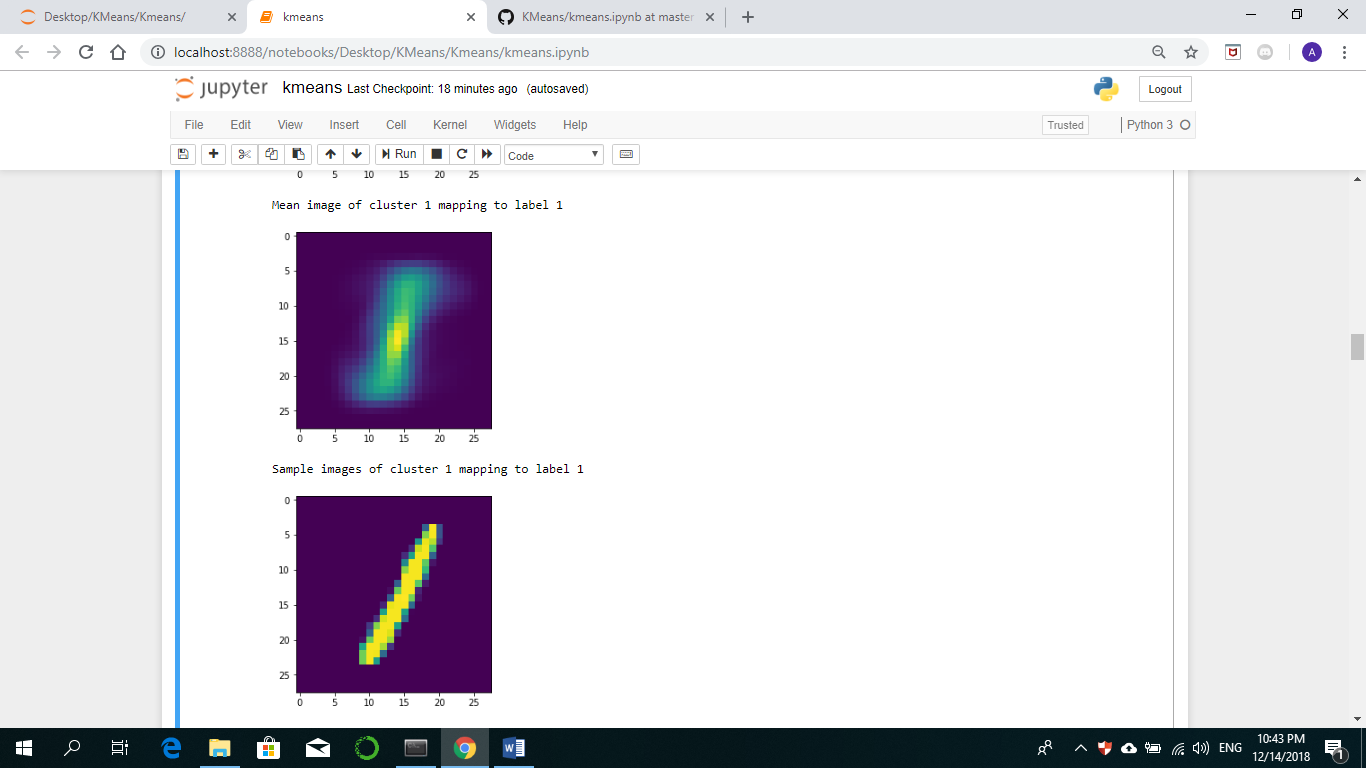
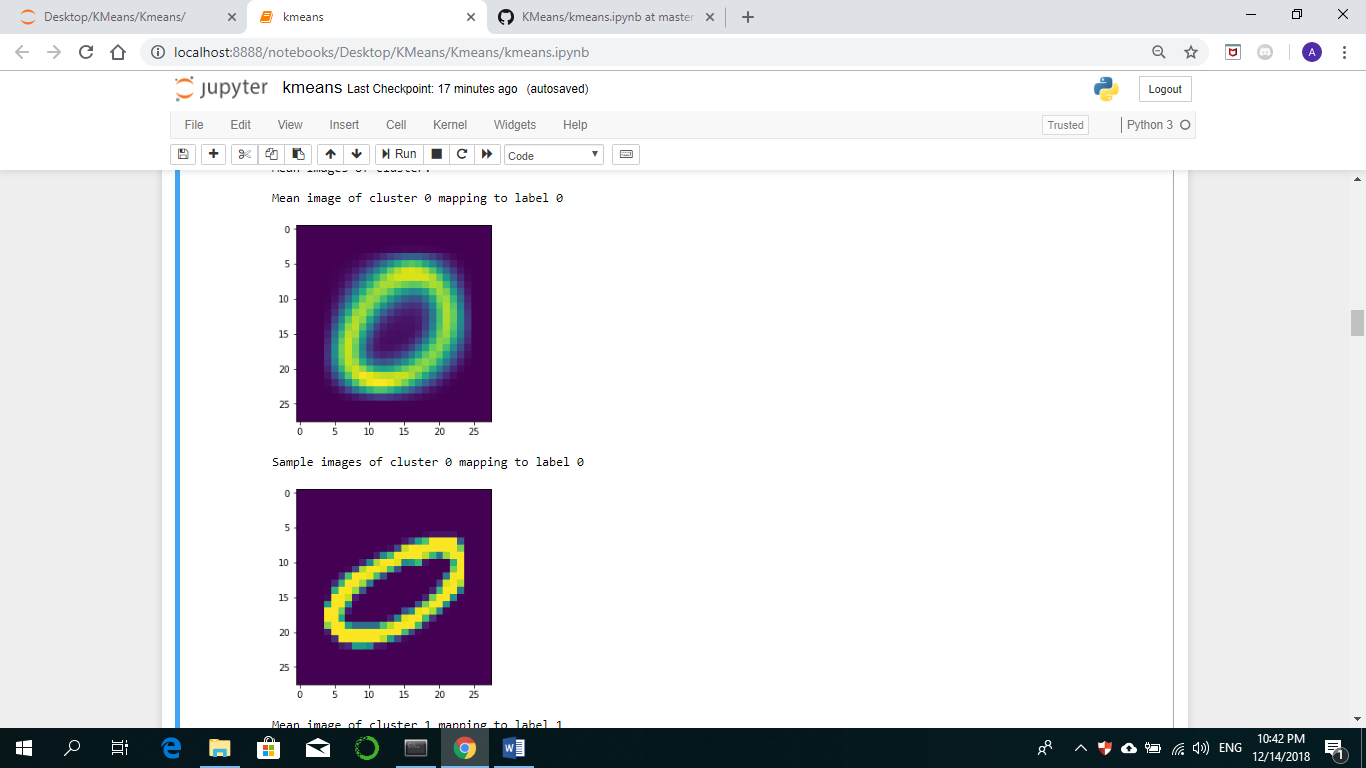
Run2: Max= 1985790.222, Min= 1824030.471

Run3: Max= 2058536.885, Min= 1824011.109

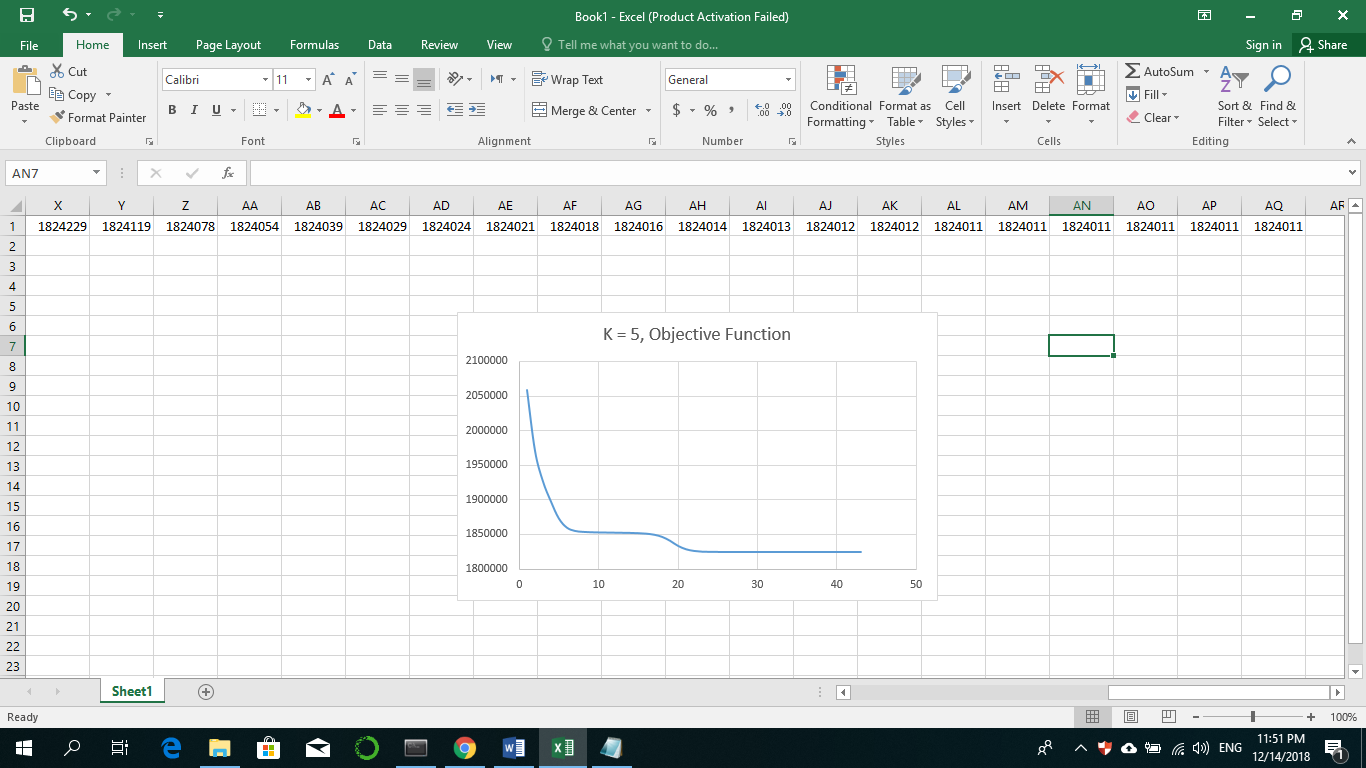
Run4: Max= 2017129.383, Min= 1824011.315

Run5: Max= 1957515.944, Min= 1824012.14

Train Accuracy = 0.4511, Test Accuracy = 0.4537



Mappings 0->0, 1->1, 2->3, 3->7, 4->6



For k = 7:

Run1: Max= 1867412.635848724, Min= 1735097.3918885968

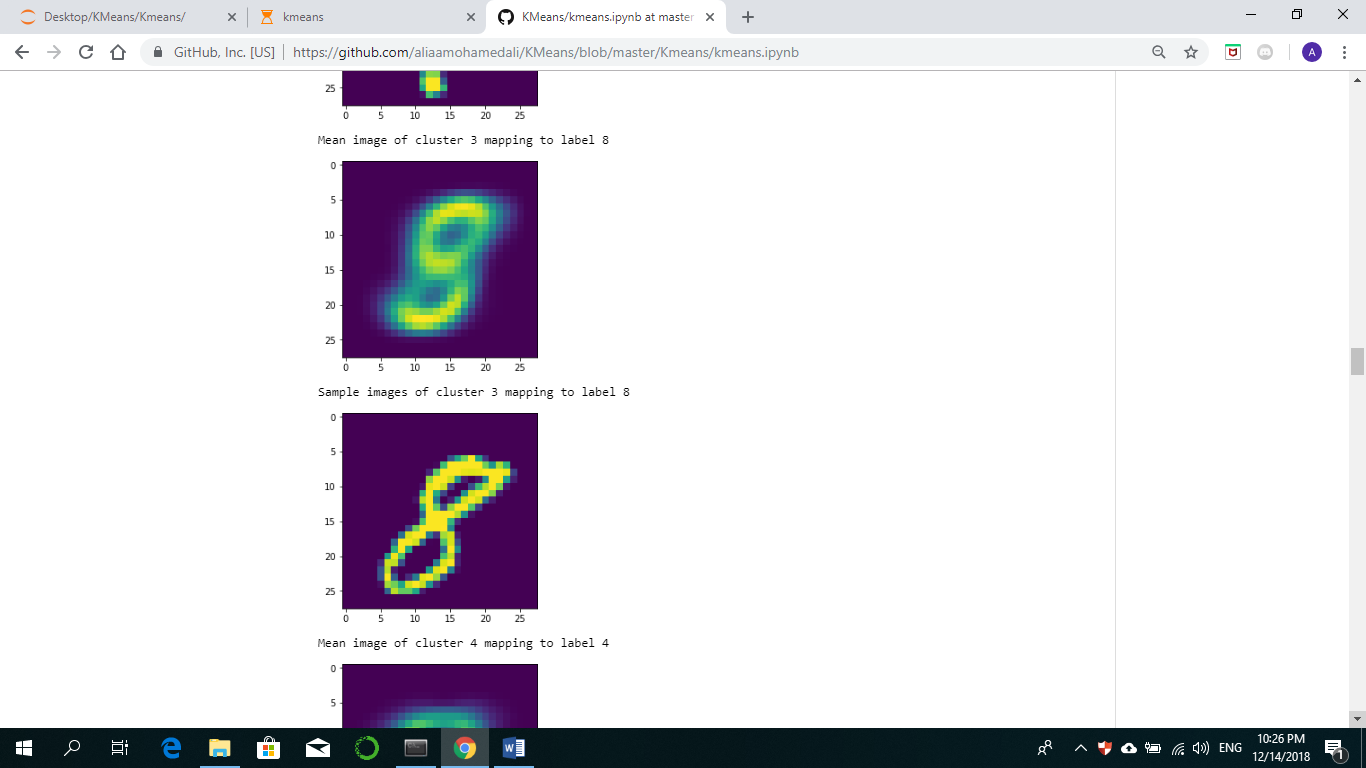
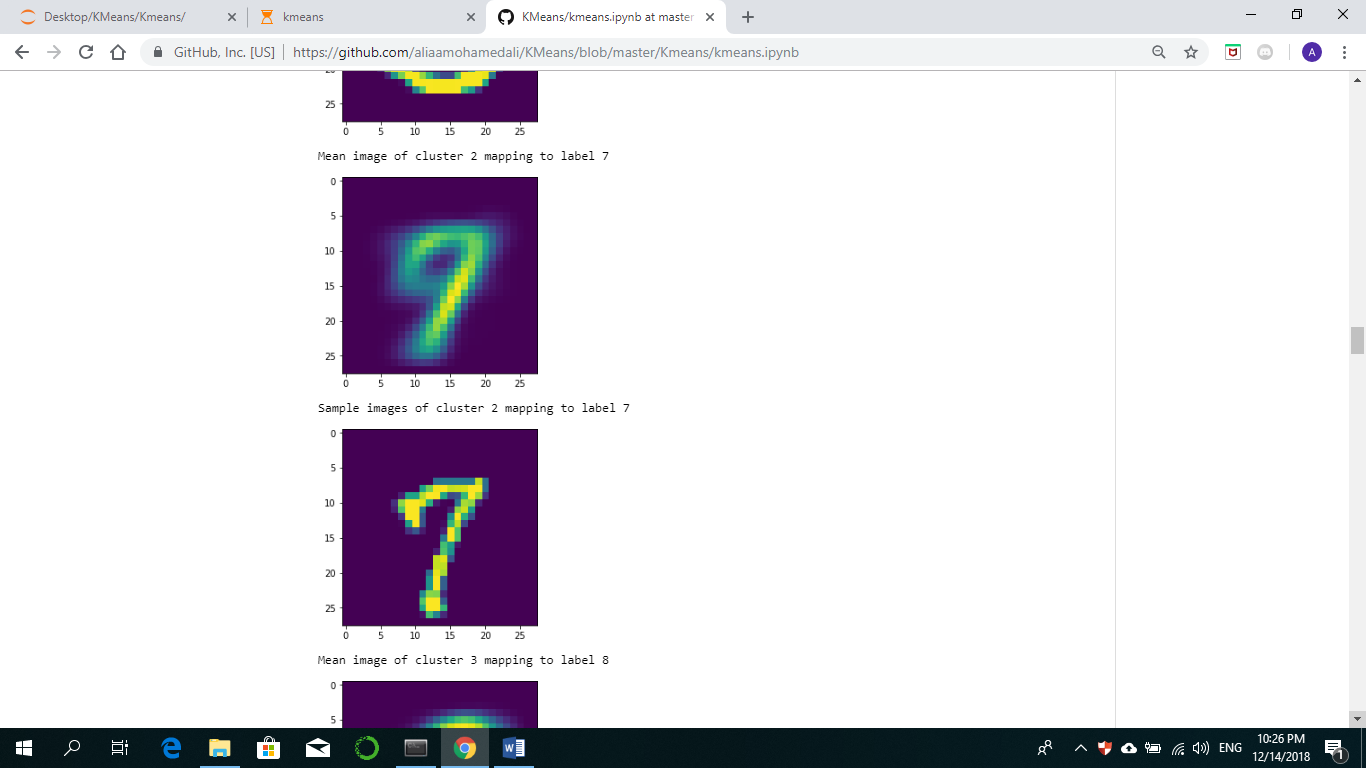
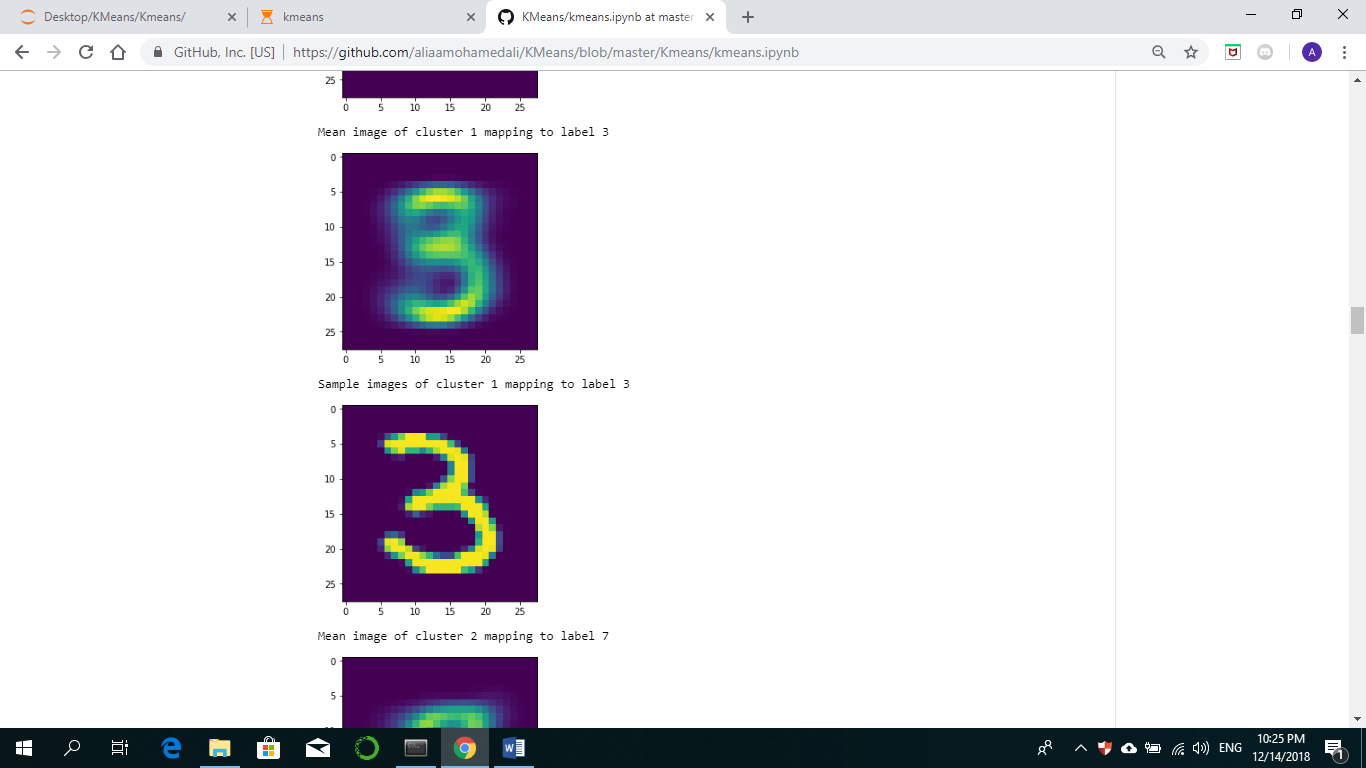
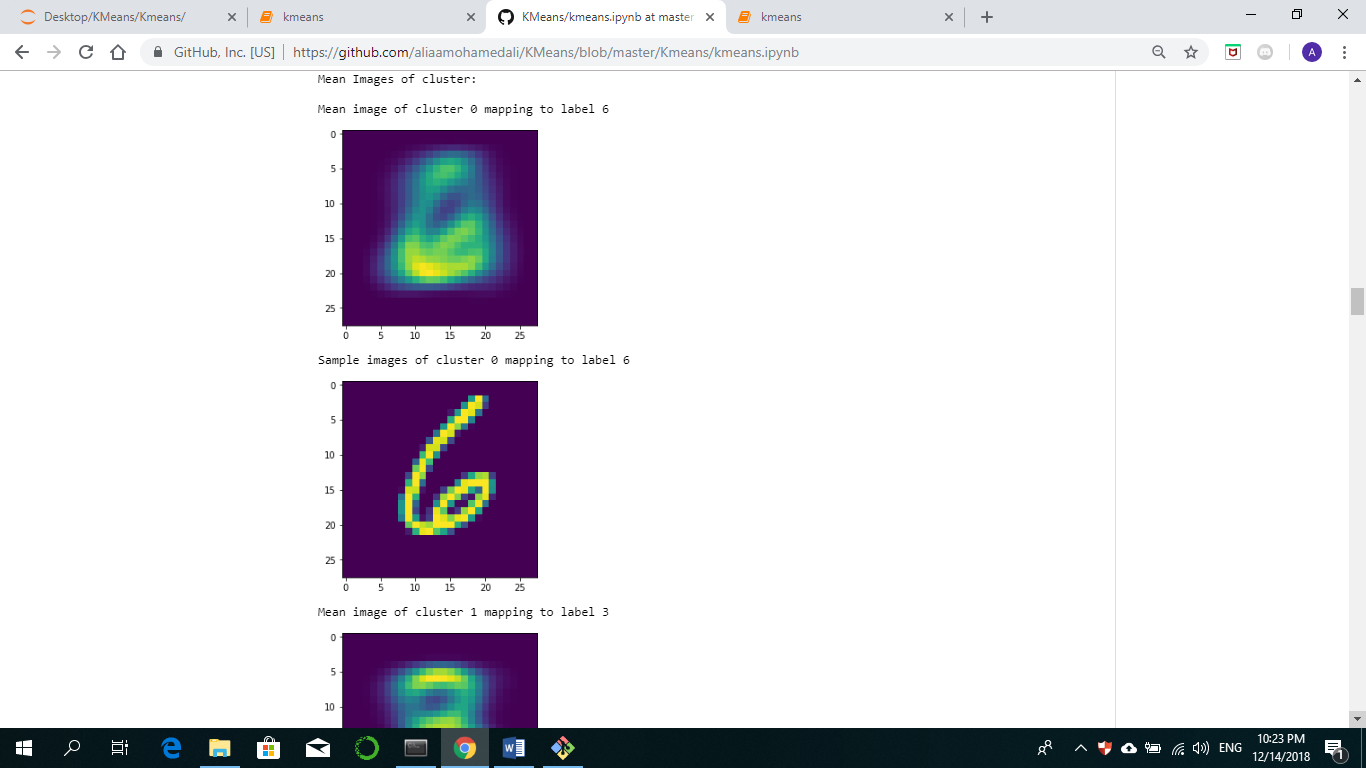
Run2: Max= 1834291.6736012741, Min= 1735097.9146356802

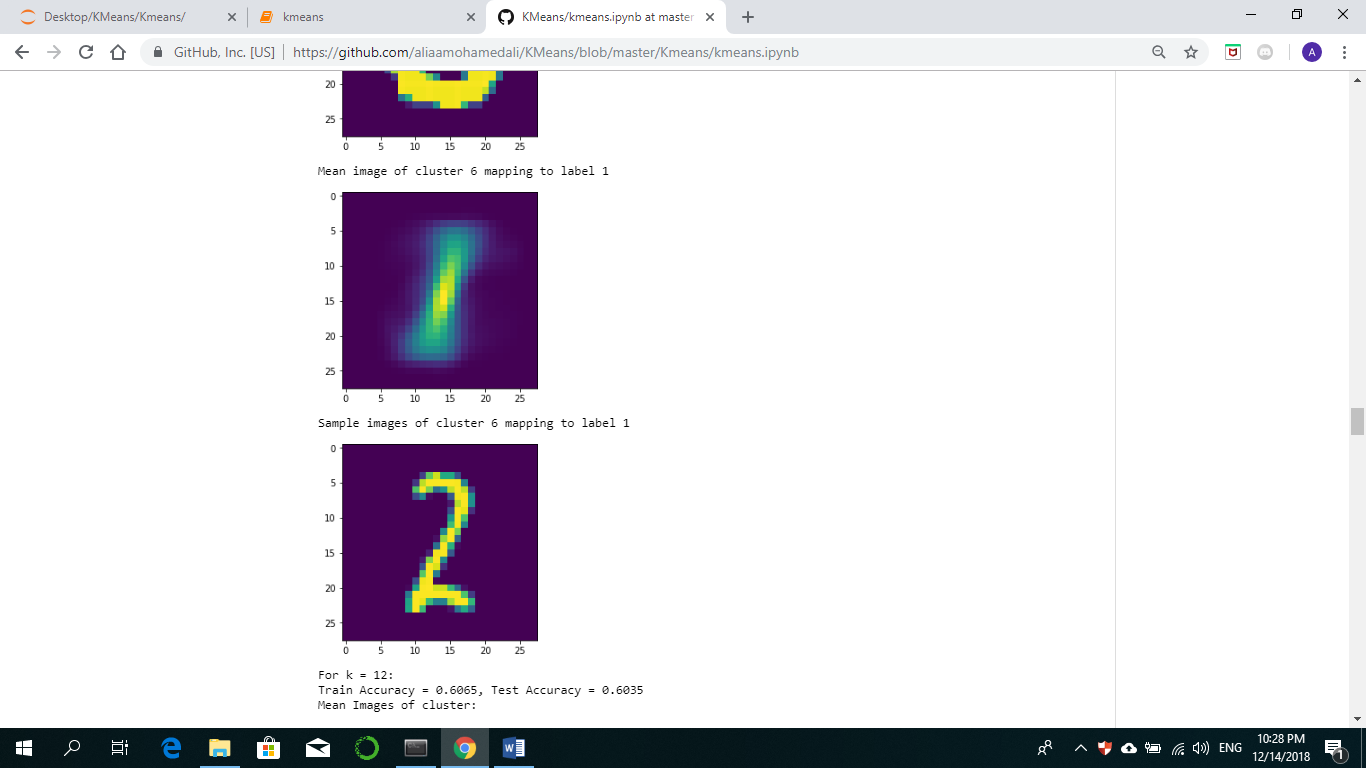
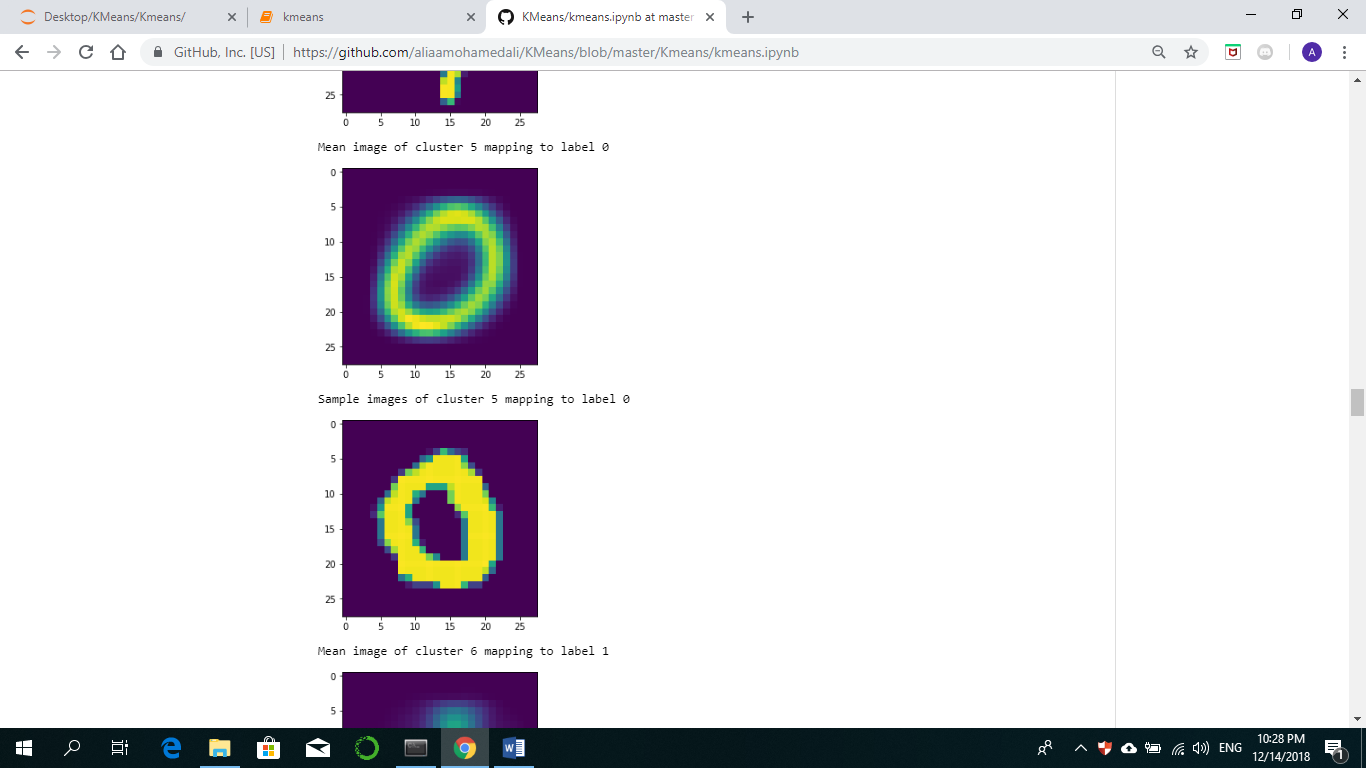
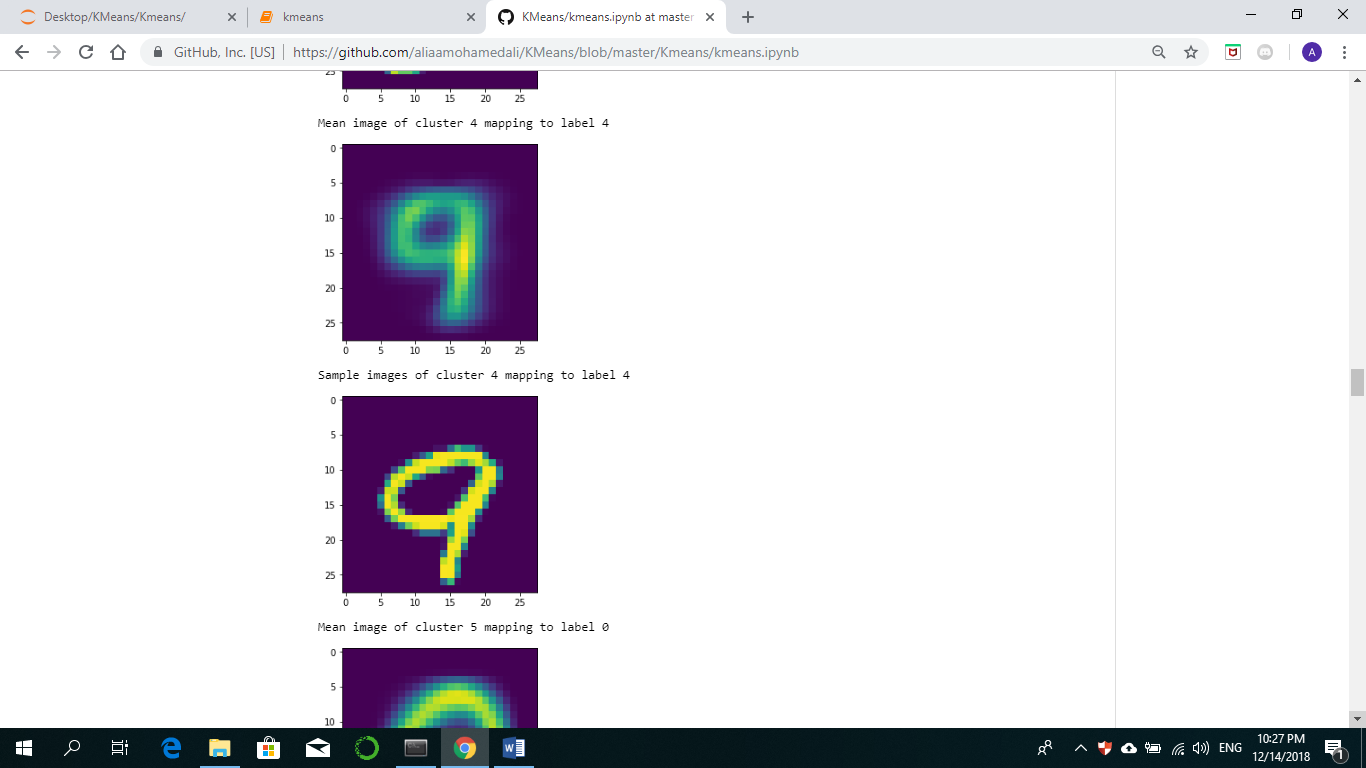
Run3: Max= 1792639.742930492, Min= 1735097.4671158127

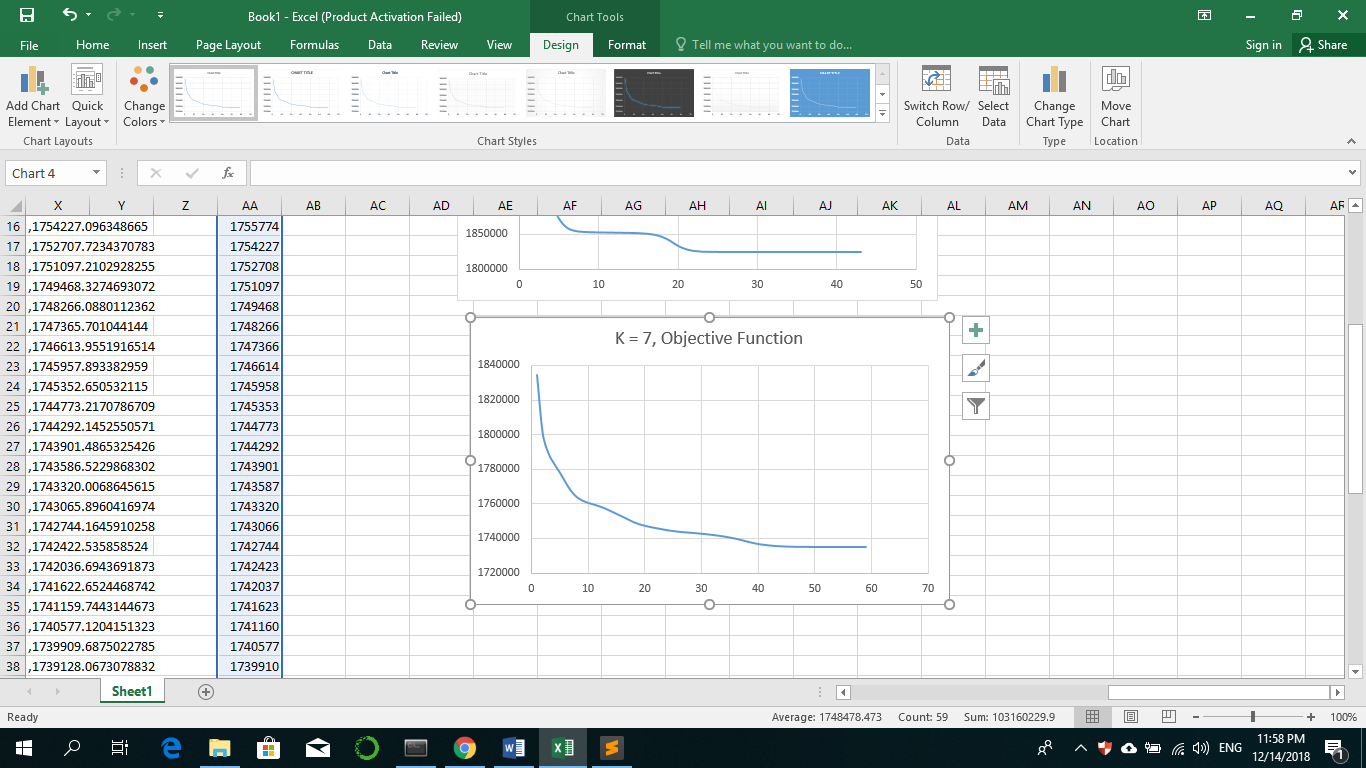
Run4: Max= 1804453.0941065643, Min= 1735097.9146356802

Run5: Max= 1834414.5604952567, Min= 1750063.3164397313

Train Accuracy = 0.5083, Test Accuracy = 0.5064







For k = 10:

Run1: Max= 1842558.399211303, Min= 1651141.5429655006

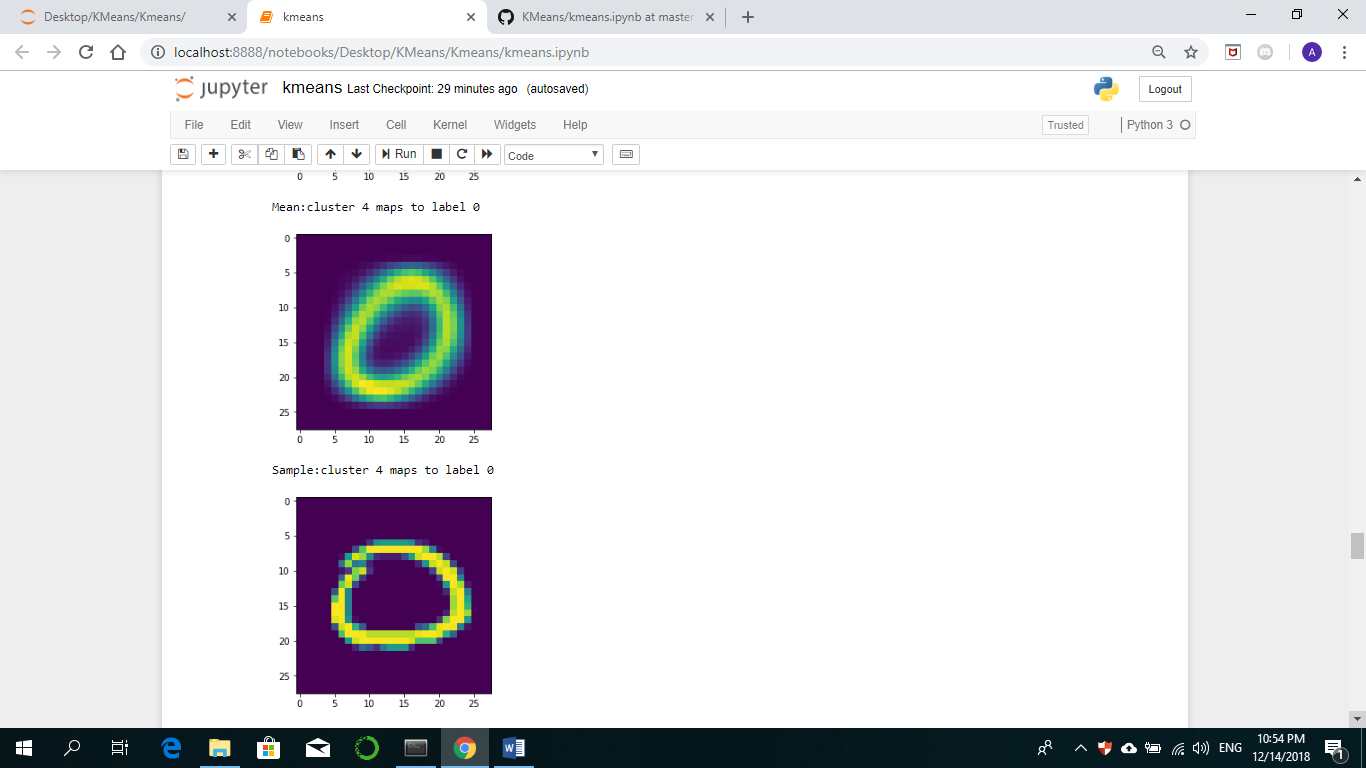
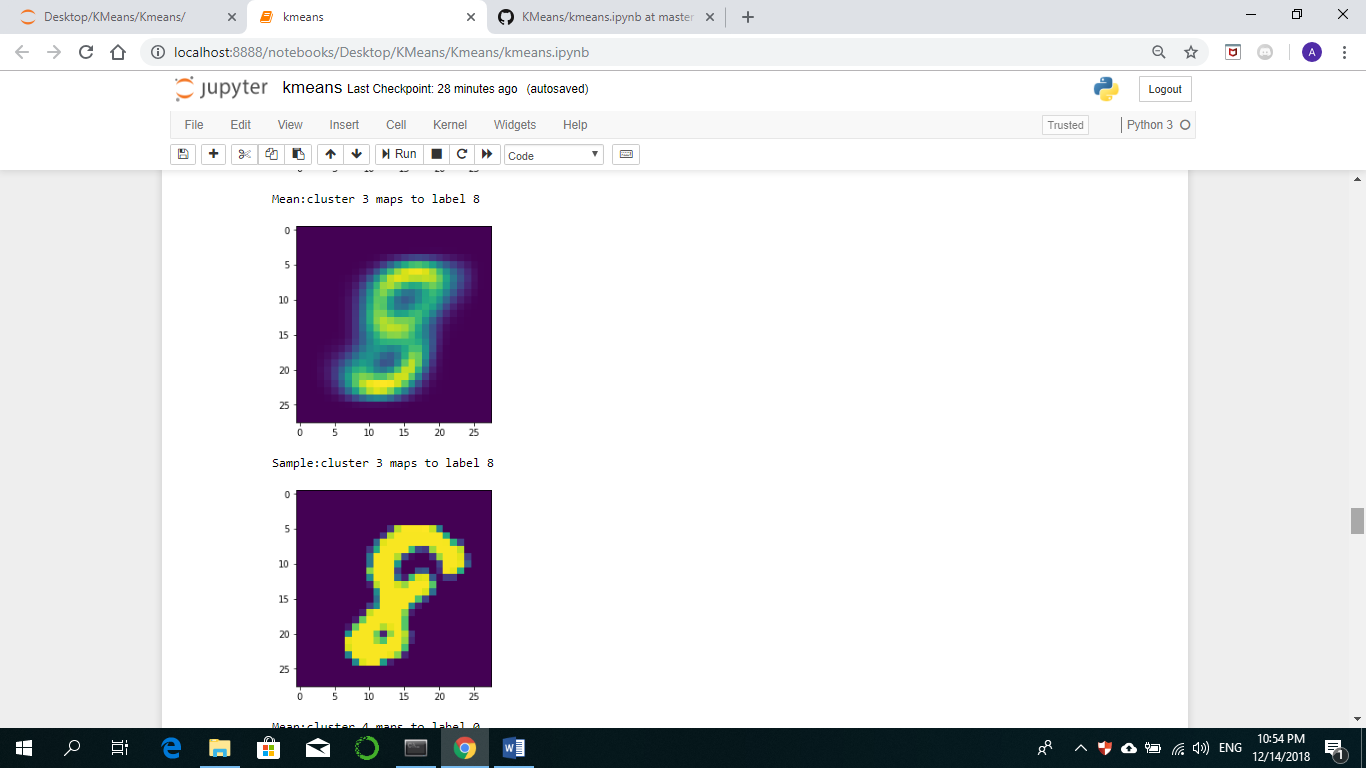
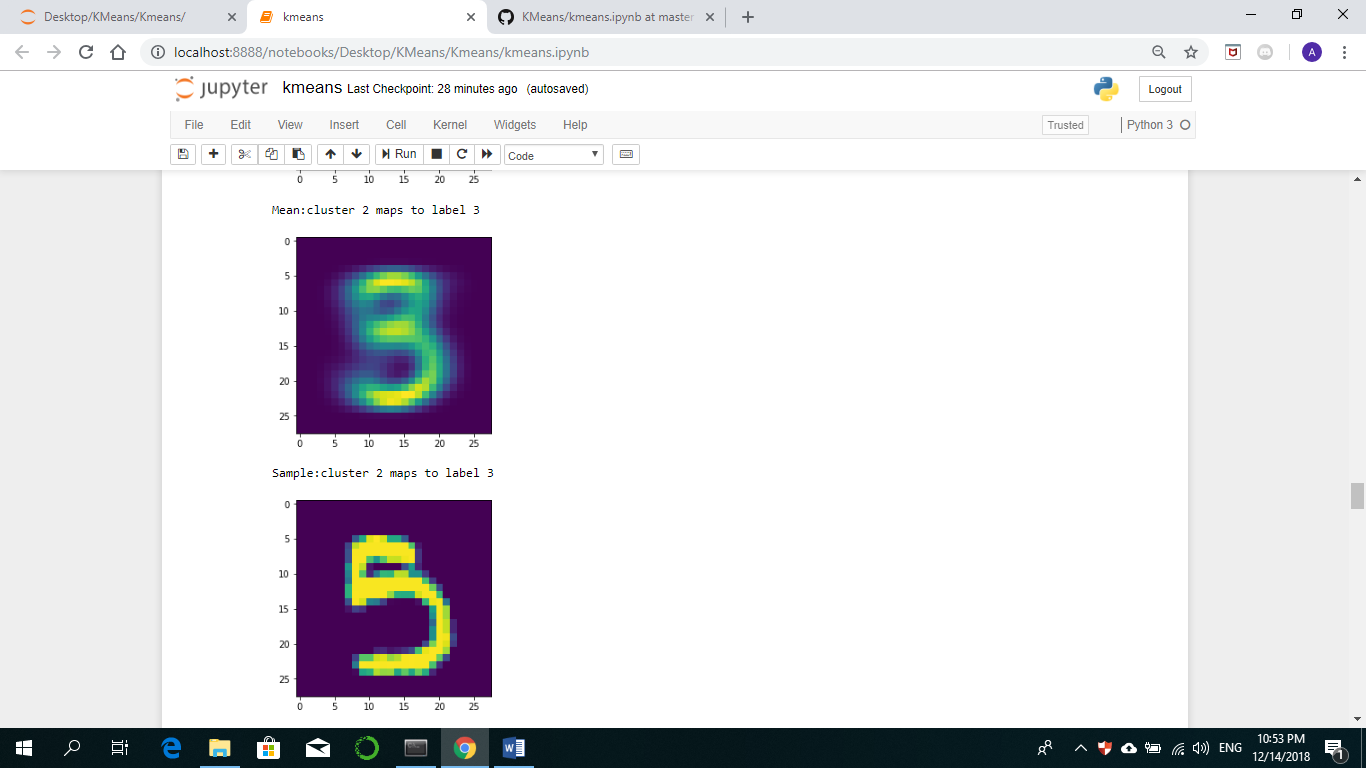
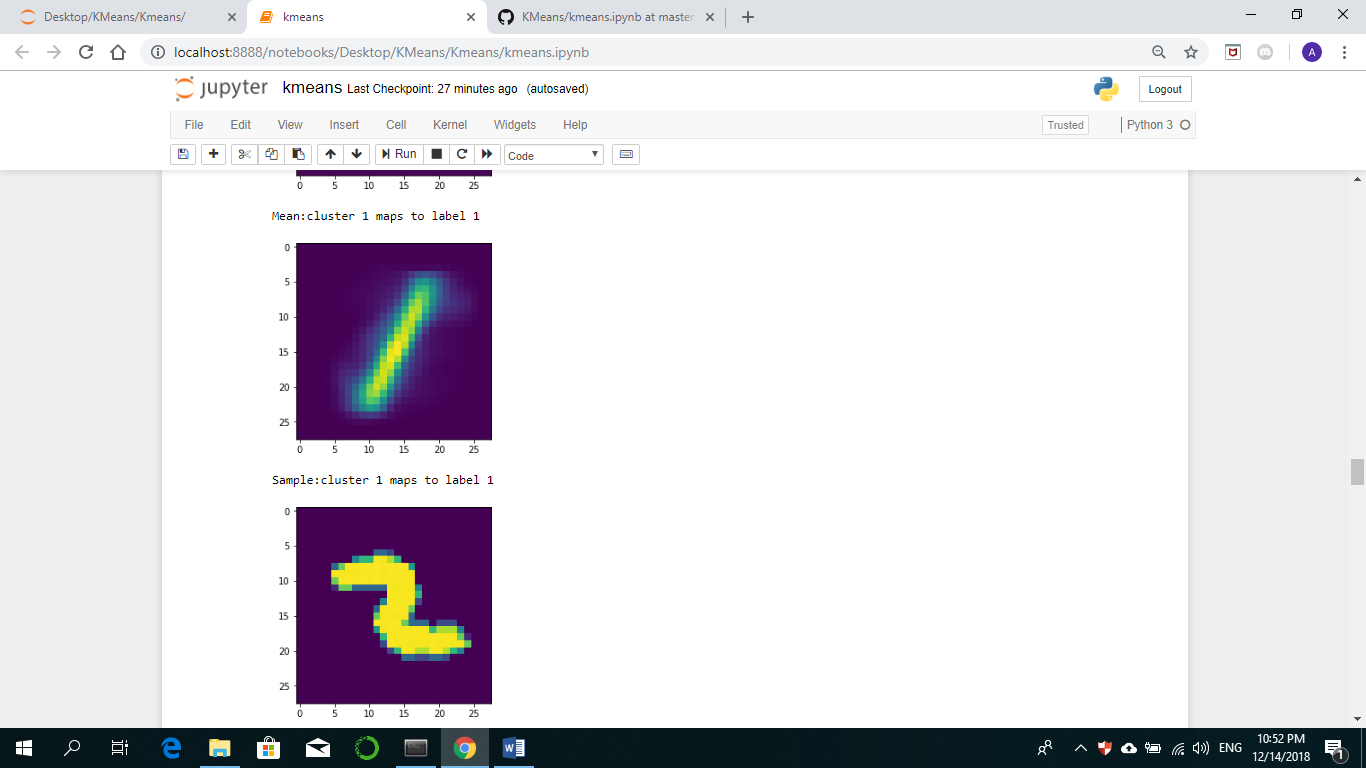
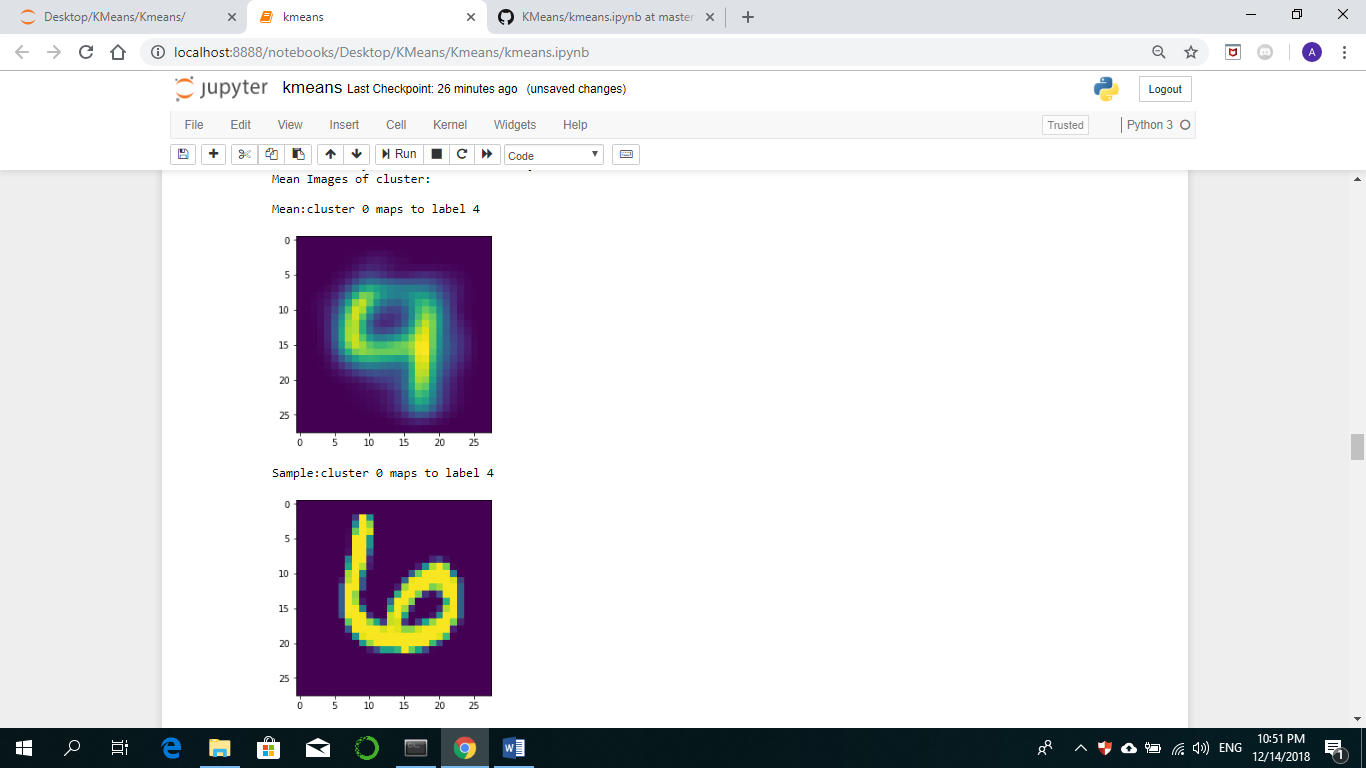
Run2: Max= 1868784.6883395873, Min= 1646039.2485595671

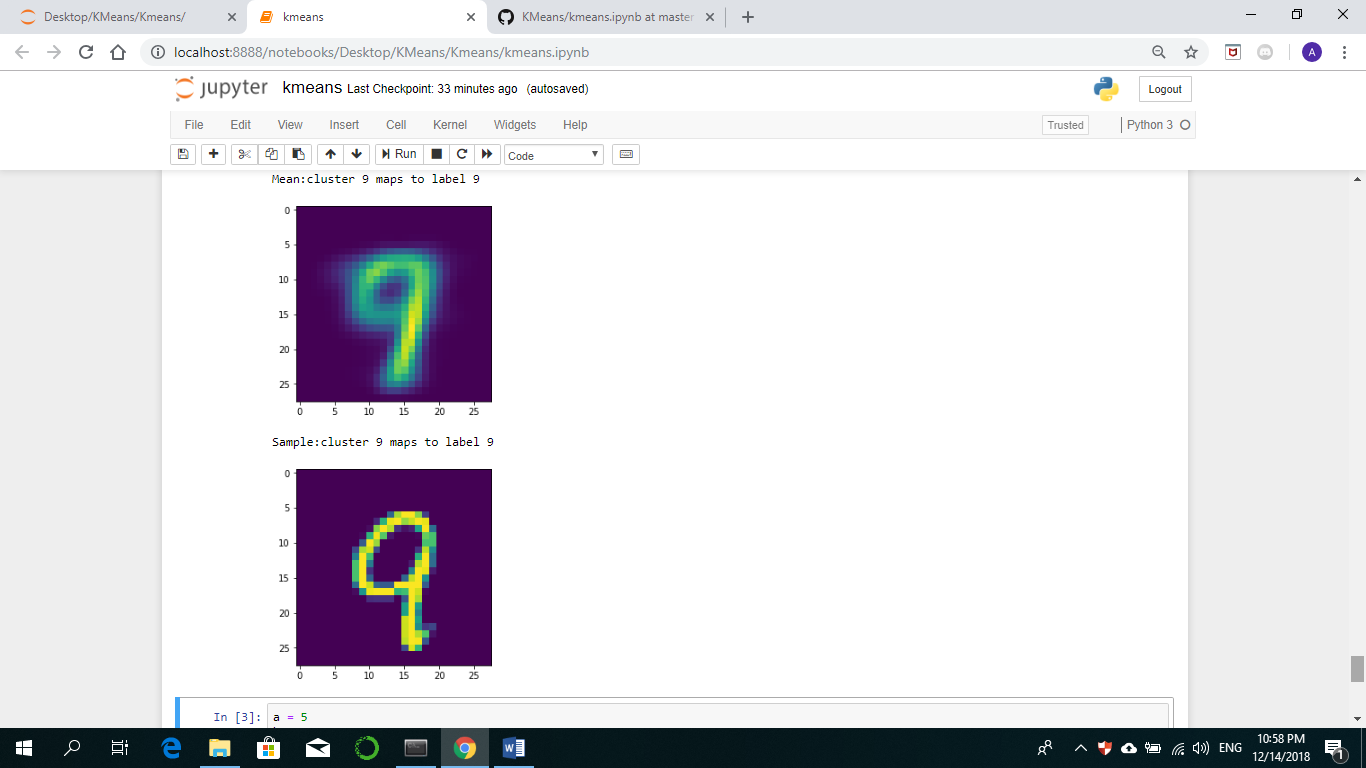
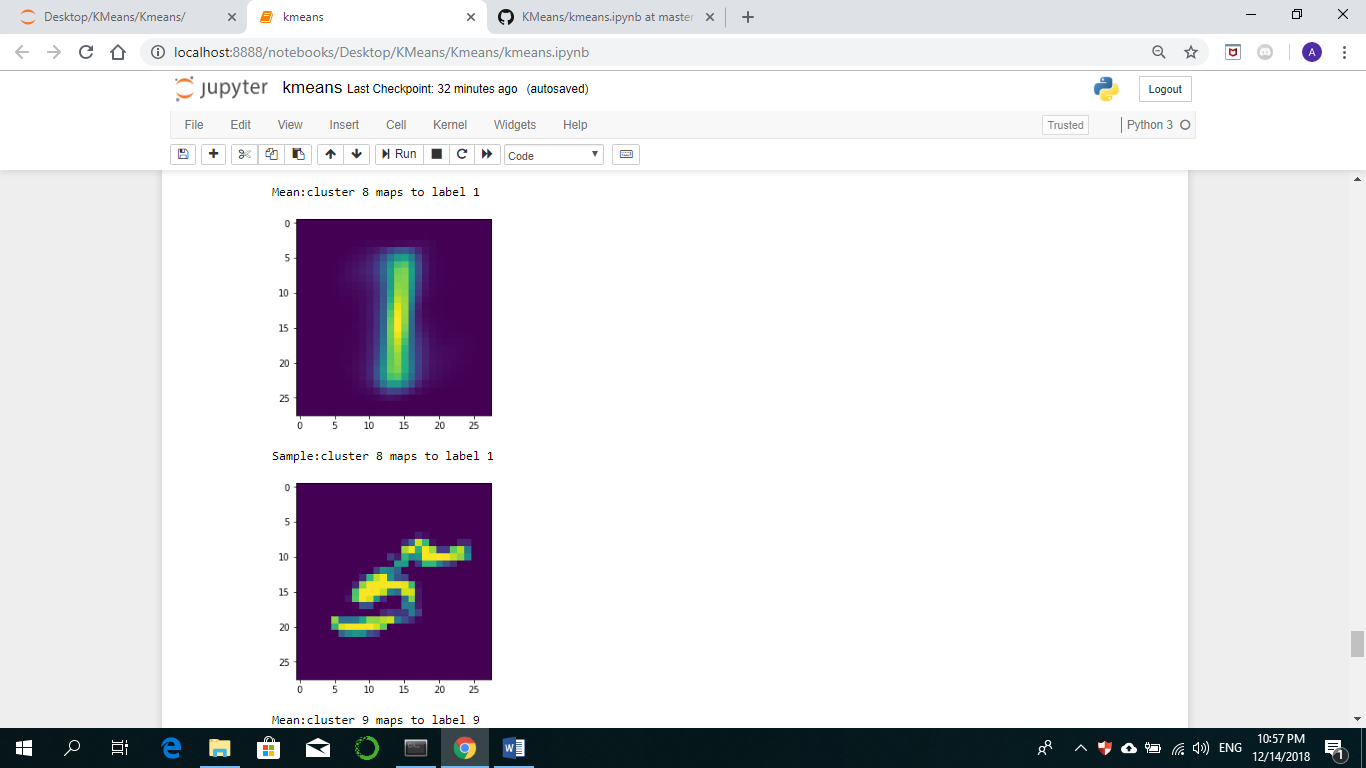
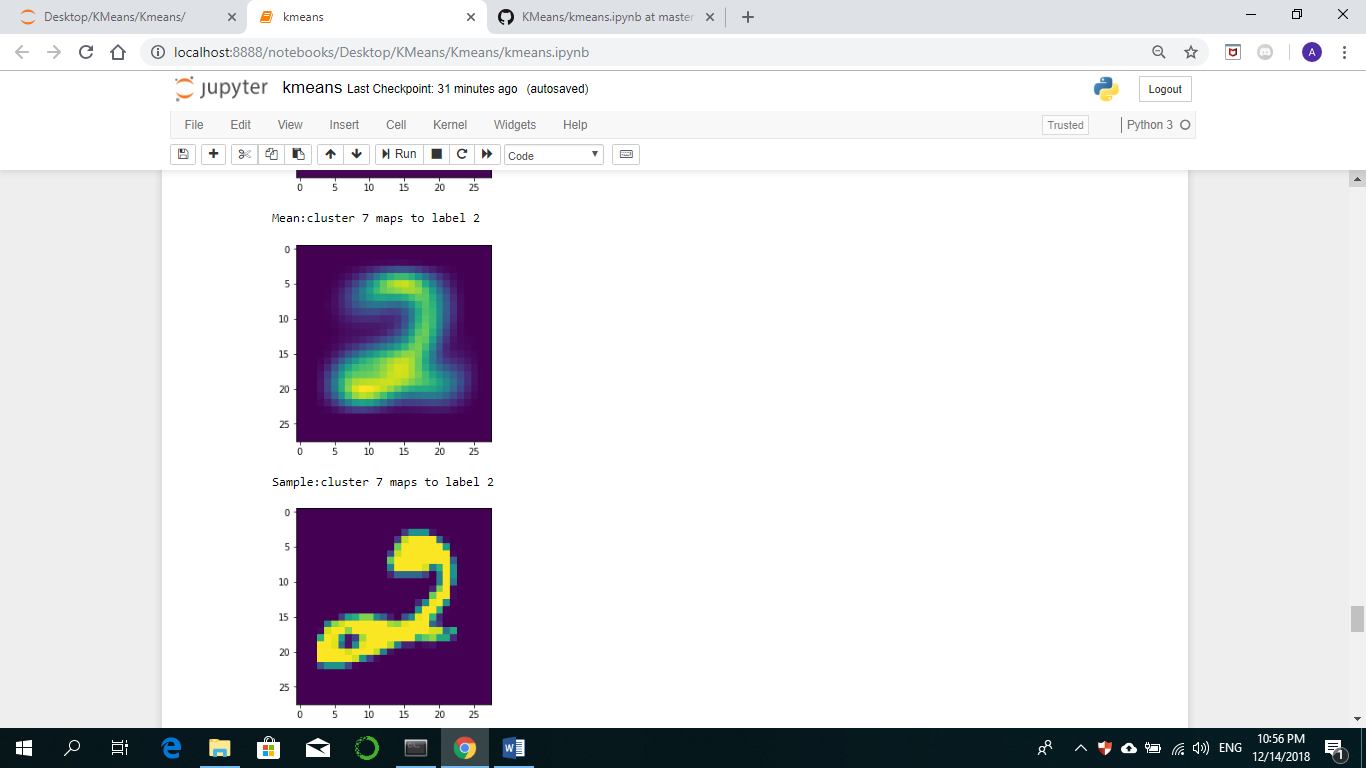
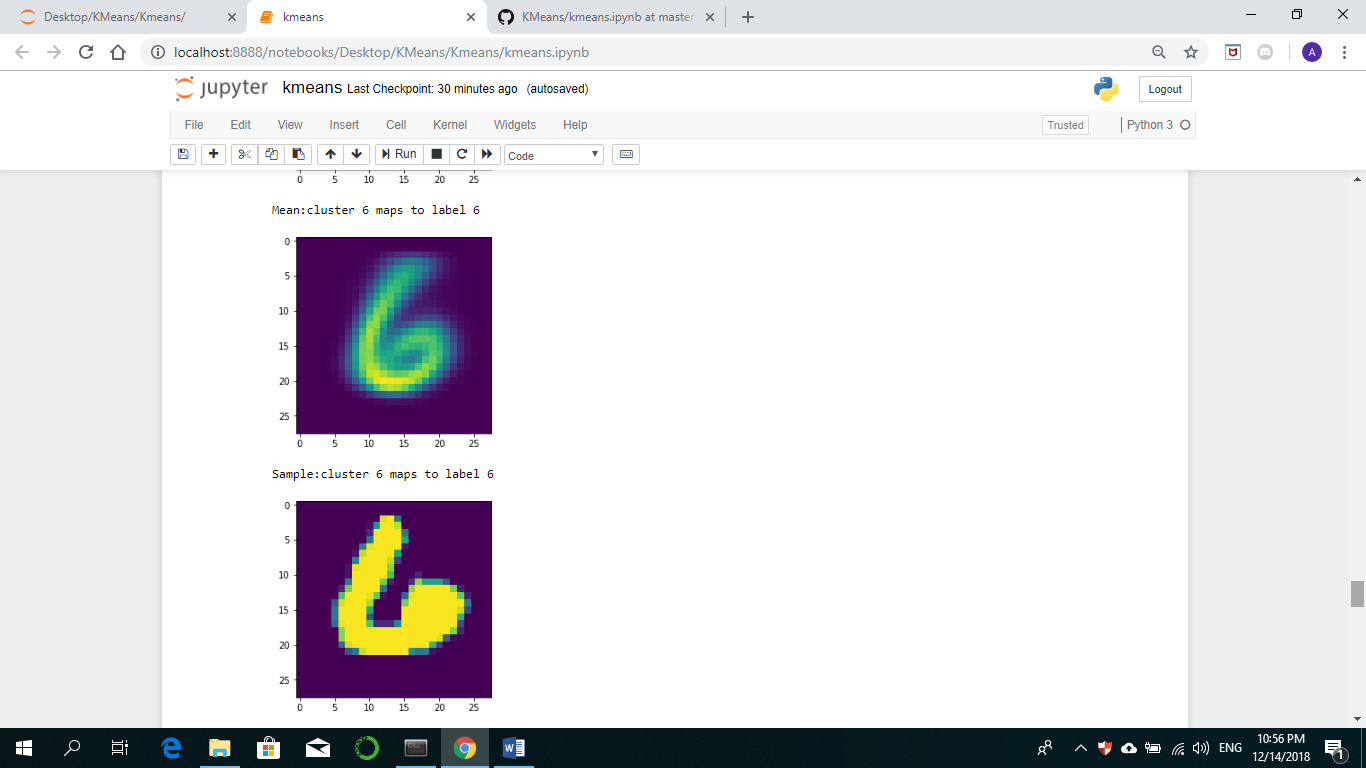
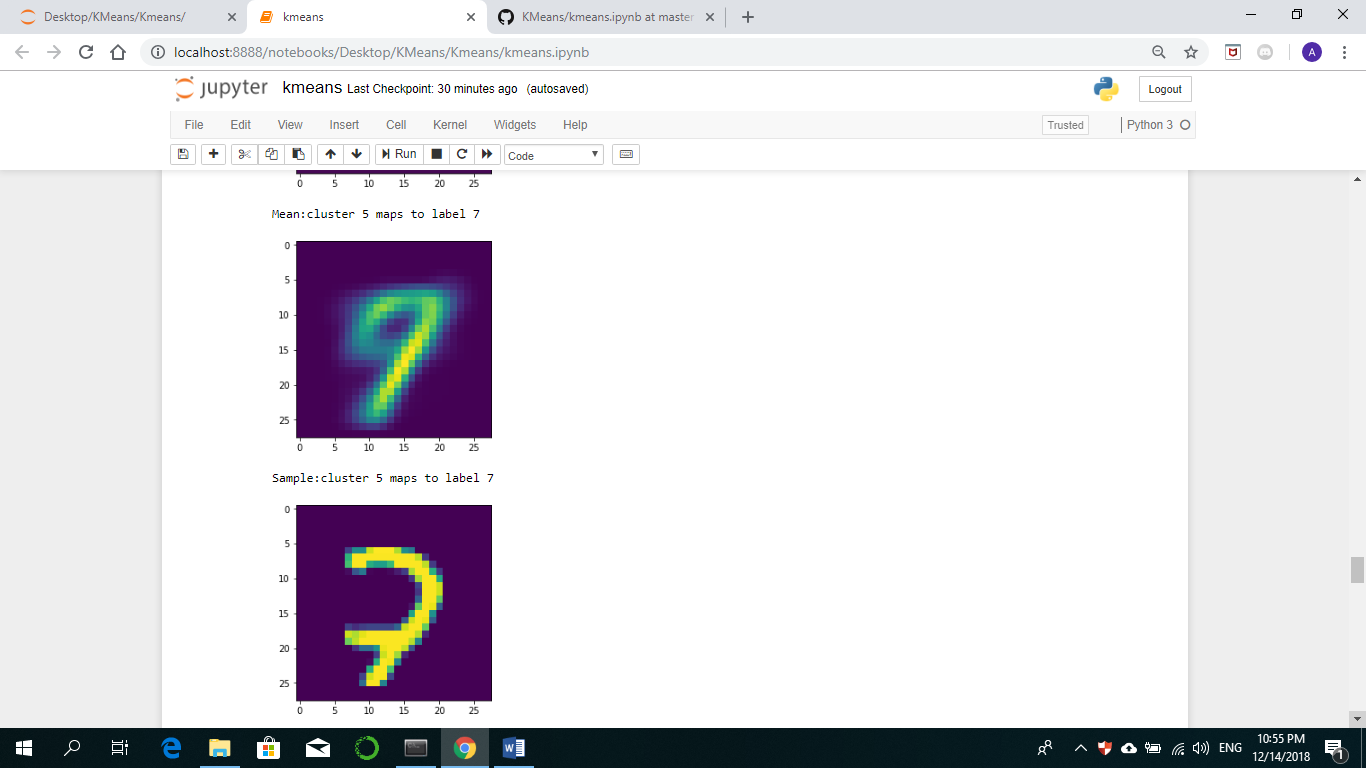
Run3: Max= 1850150.4099327181, Min= 1645926.690151455

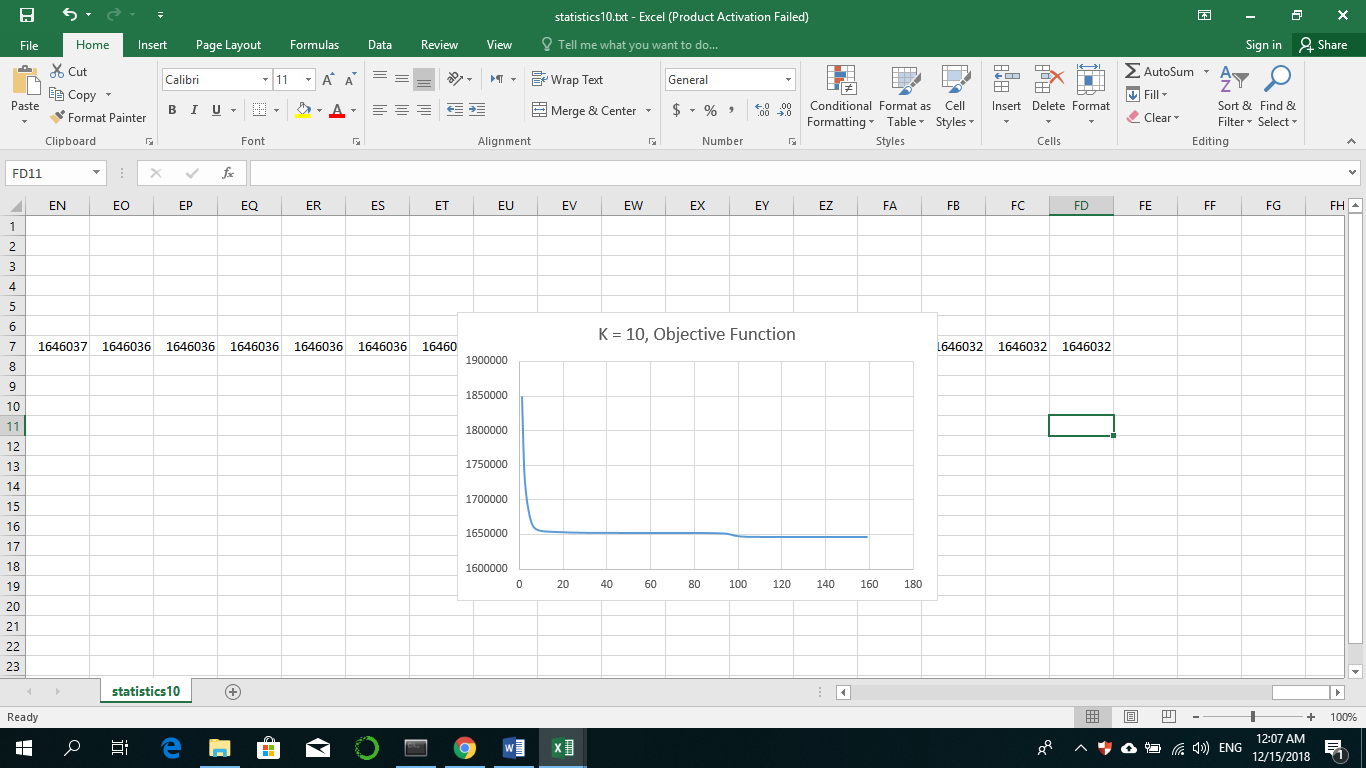
Run4: Max= 1848641.2613696465, Min= 1646031.9300724927

Run5: Max= 1824945.3455359973, Min= 1665512.9291154451

Train Accuracy = 0.5920, Test Accuracy = 0.5883







For k = 12:

Run1: Max= 1810431.295040314, Min= 1598411.390134471

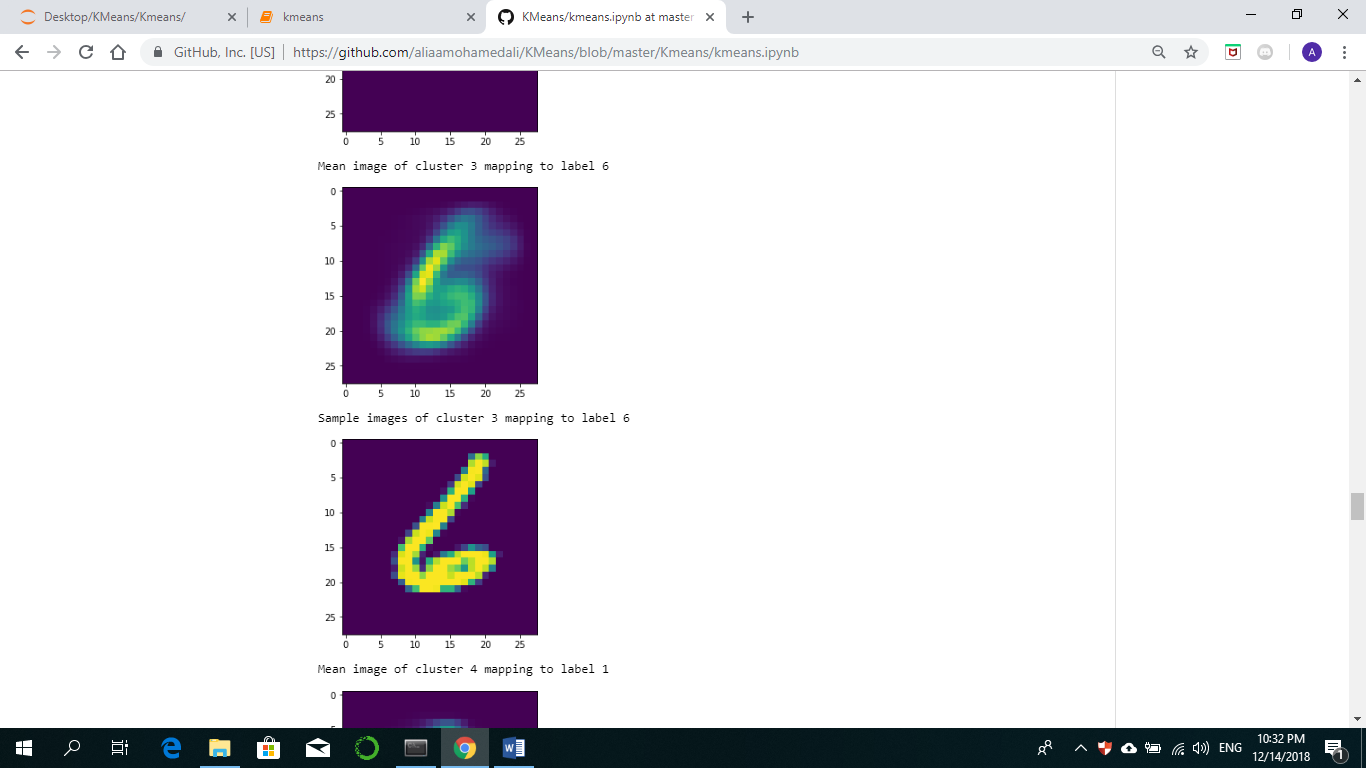
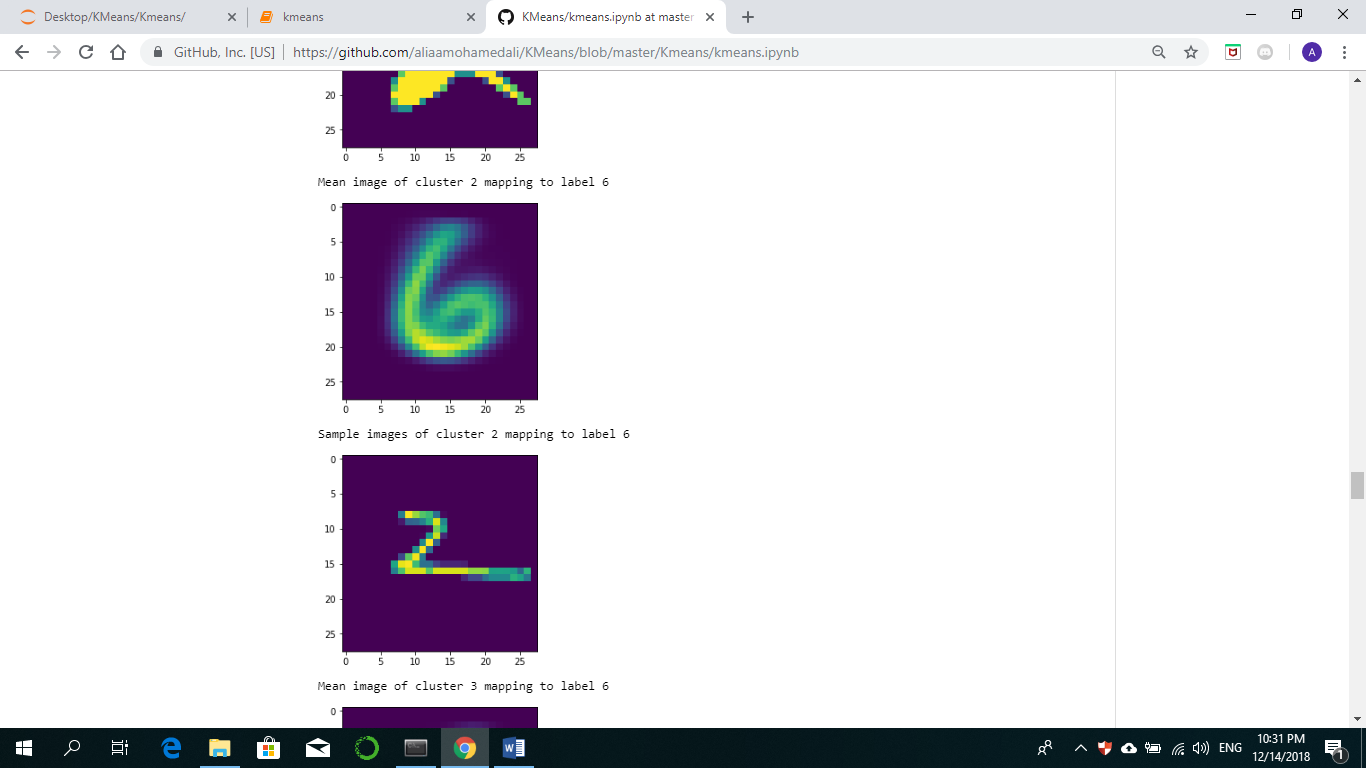
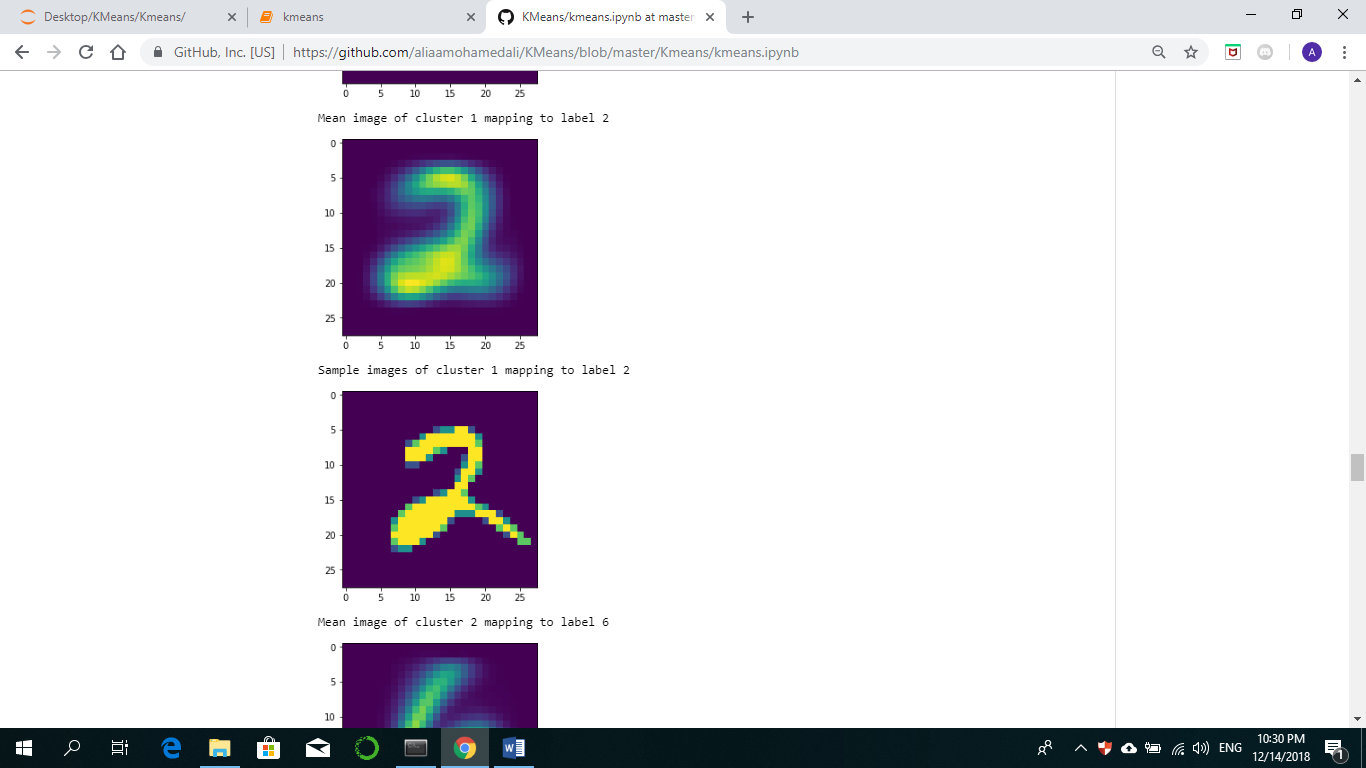
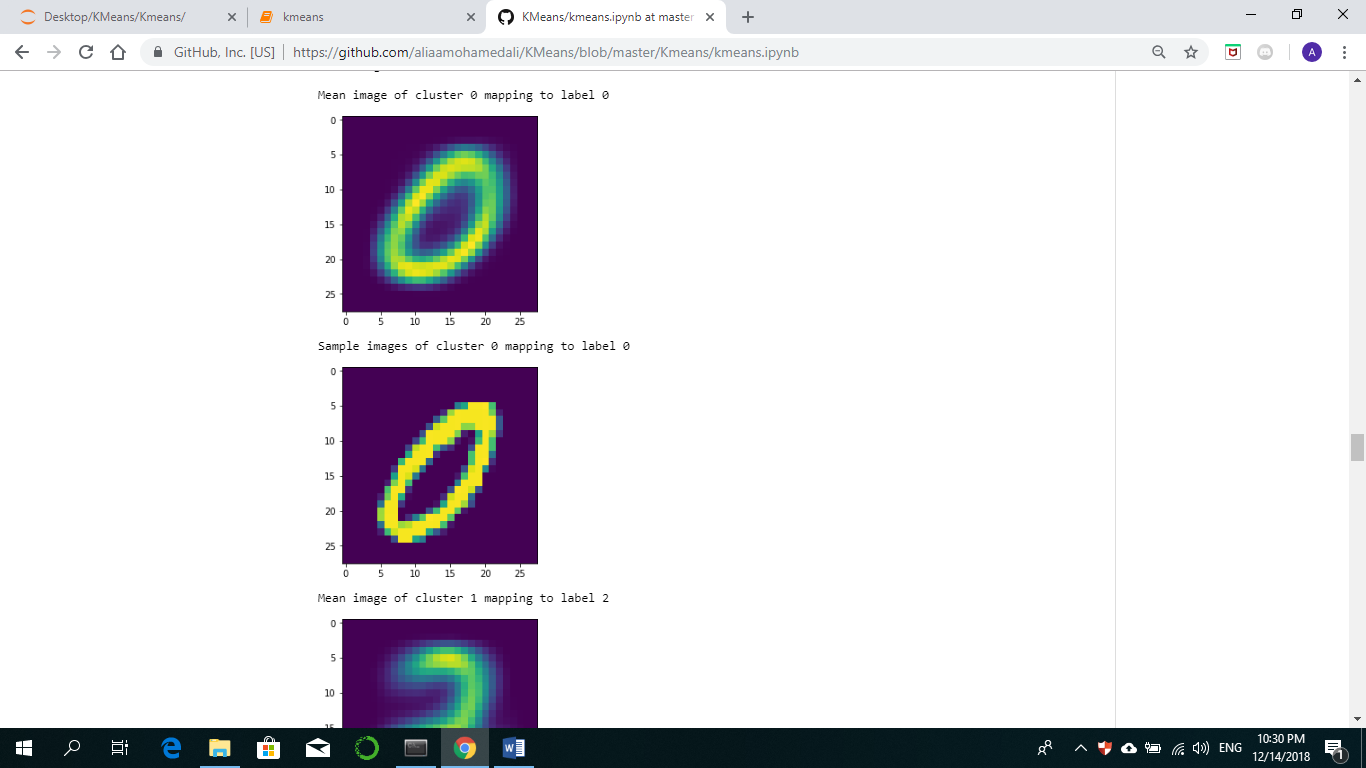
Run2: Max= 1806245.0786733285, Min= 1597737.9590808046

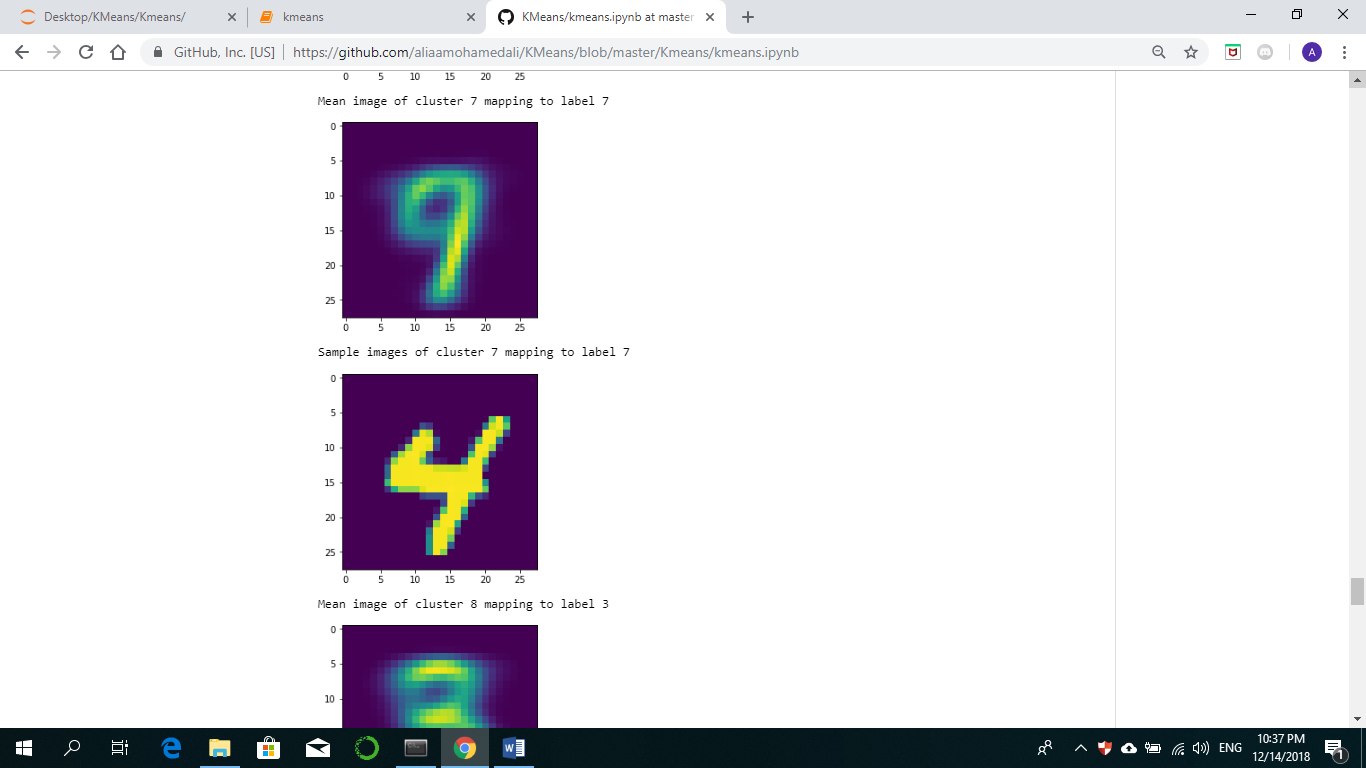
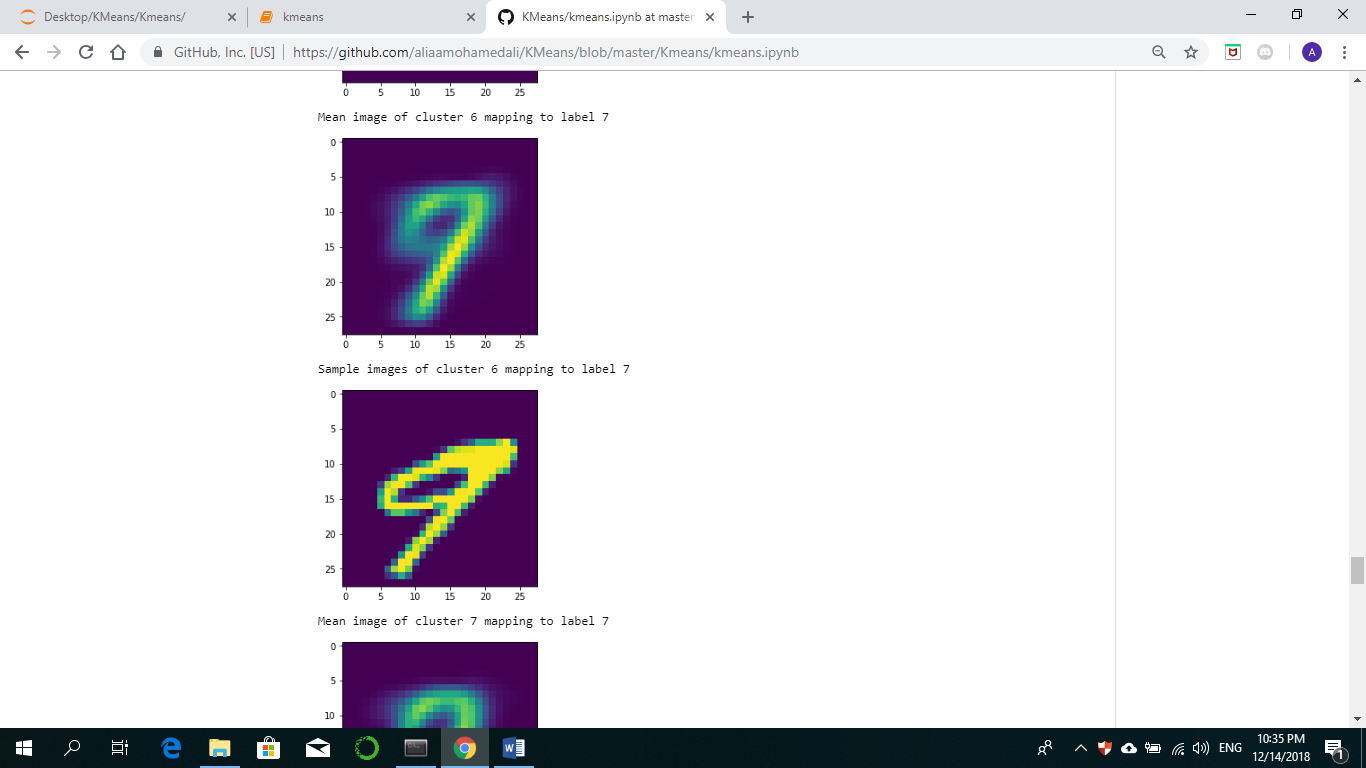
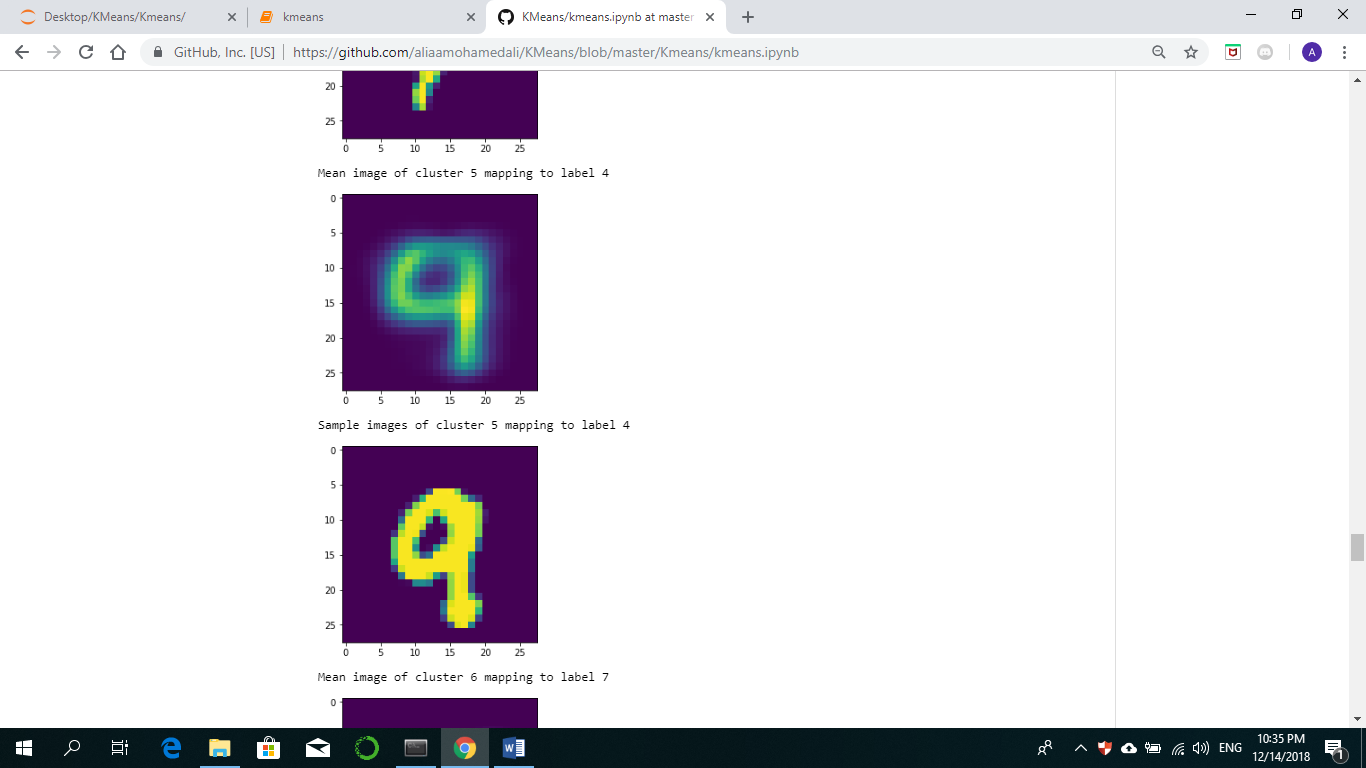
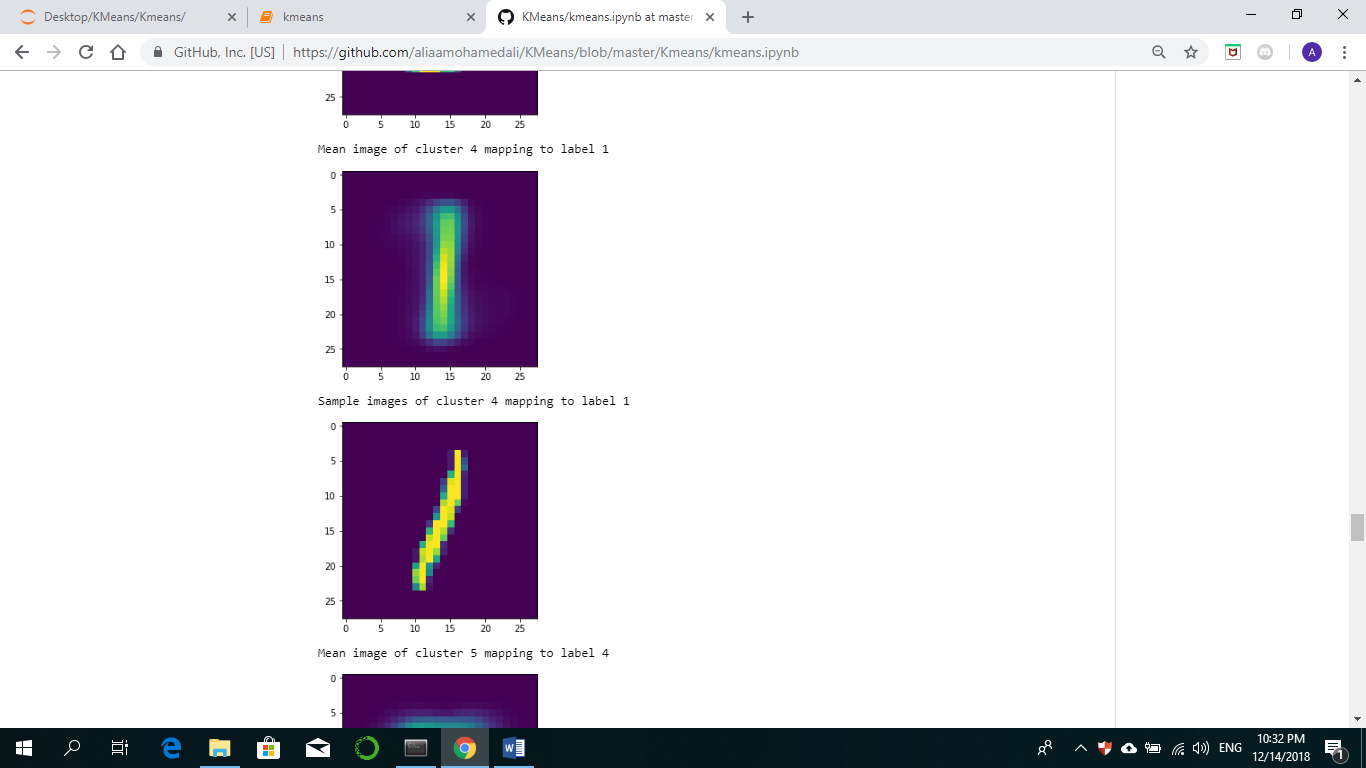
Run3: Max= 1774087.2499047131, Min= 1597737.9511413954

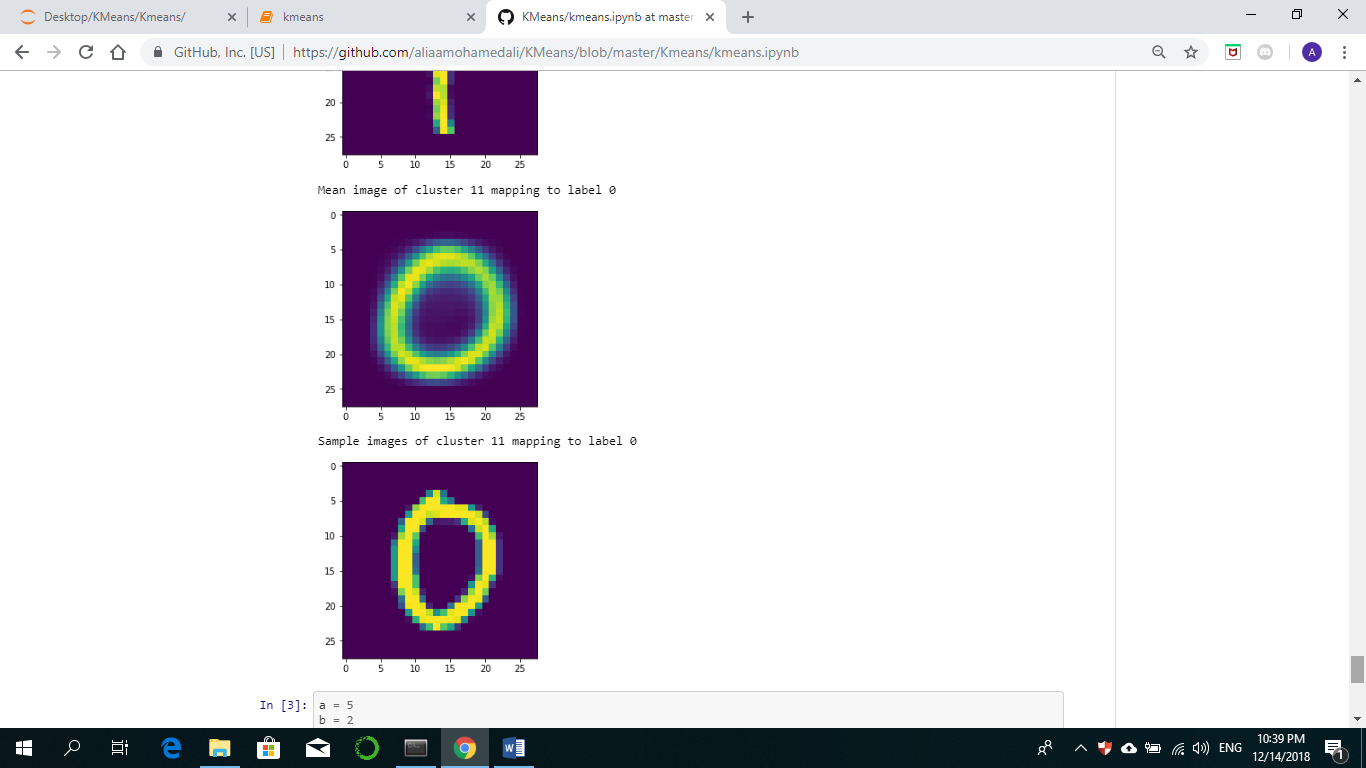
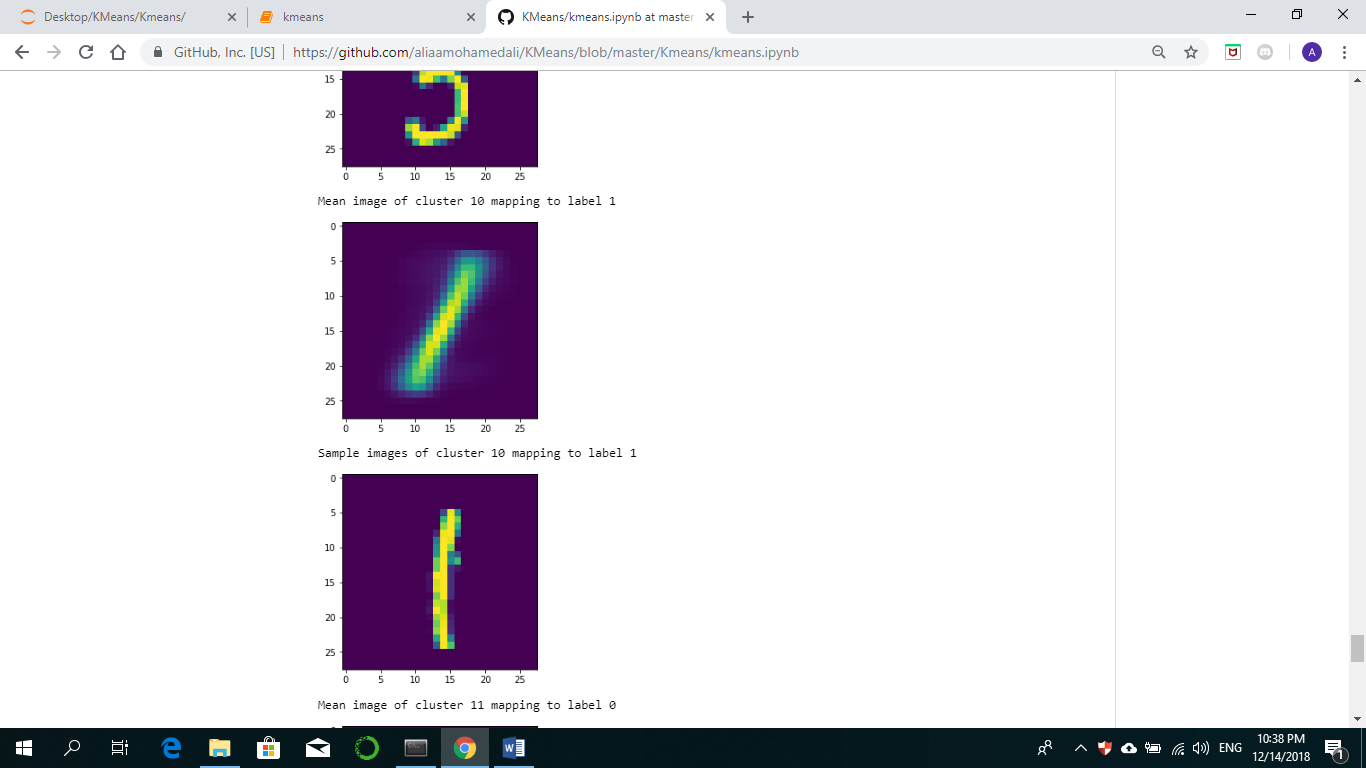
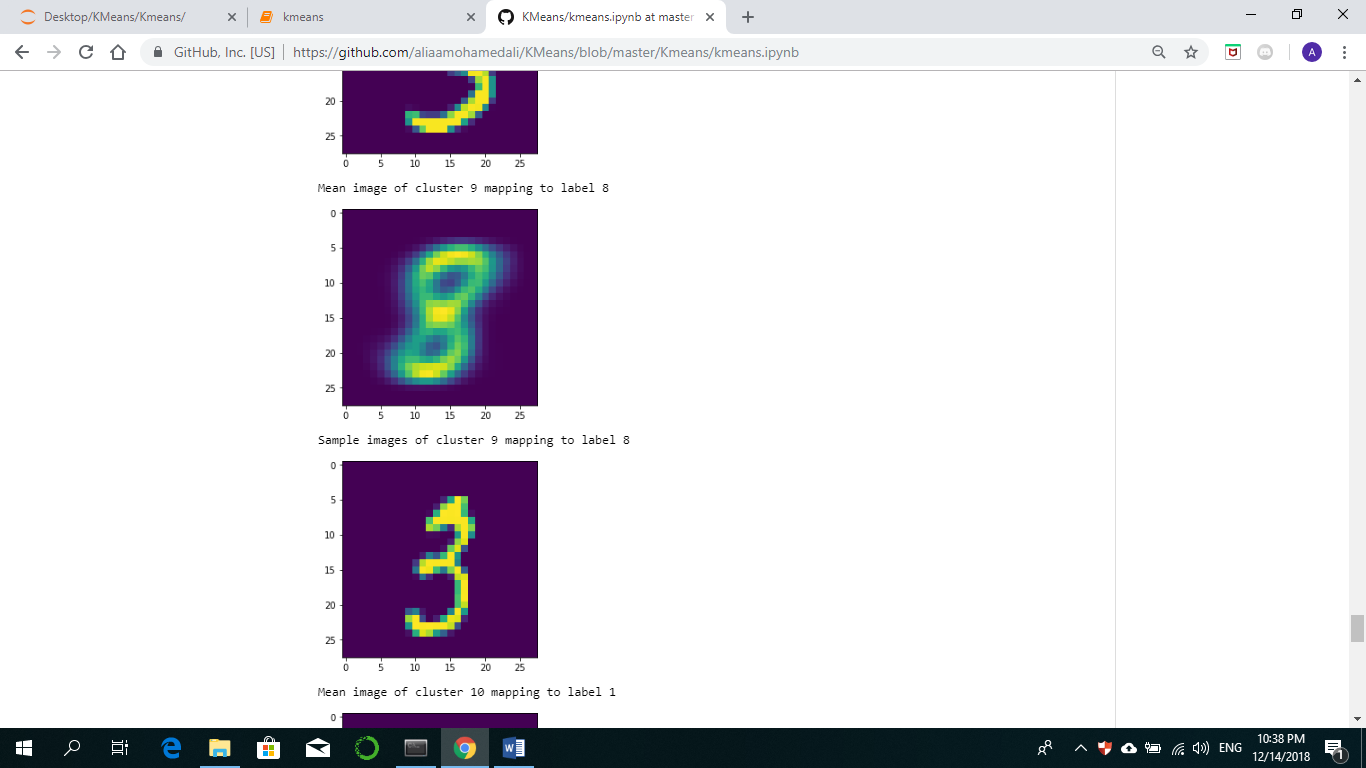
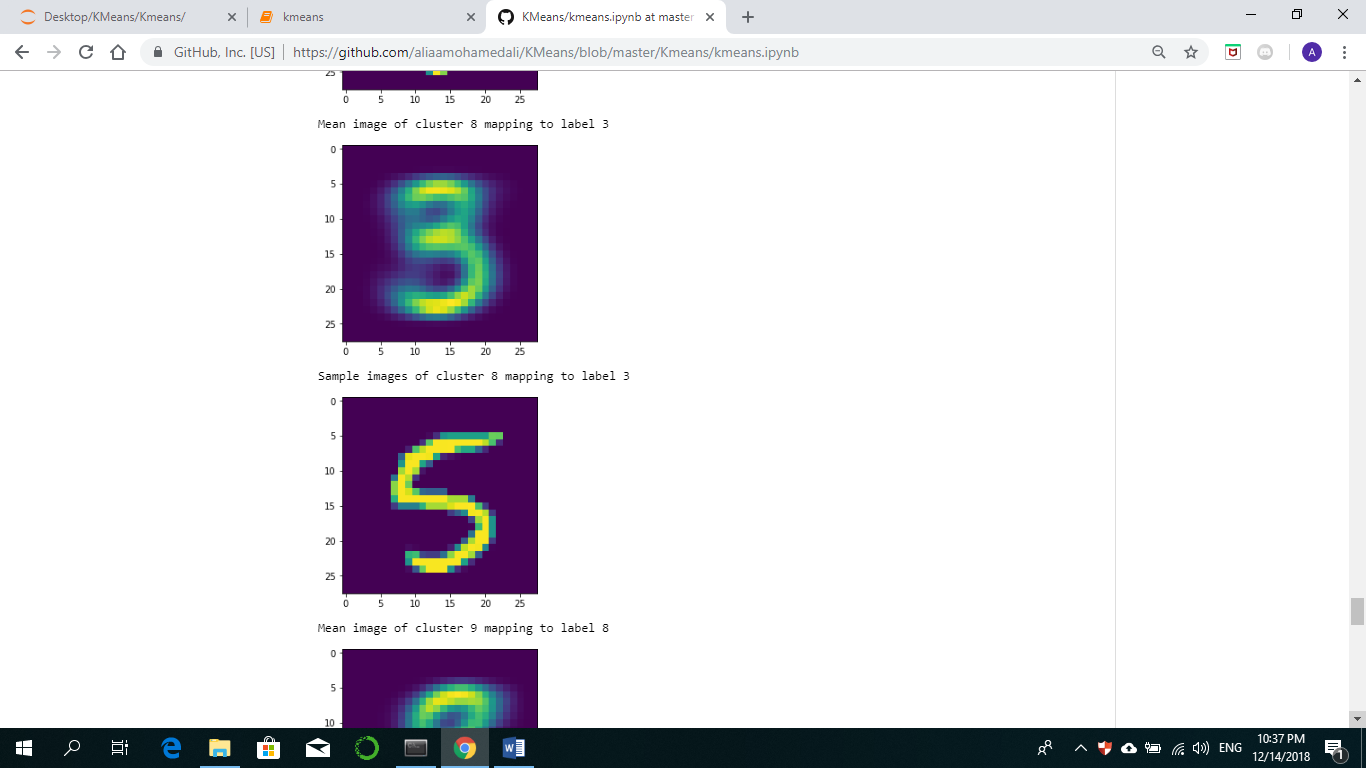
Run4: Max= 1777668.7371507126, Min= 1597737.4860982485

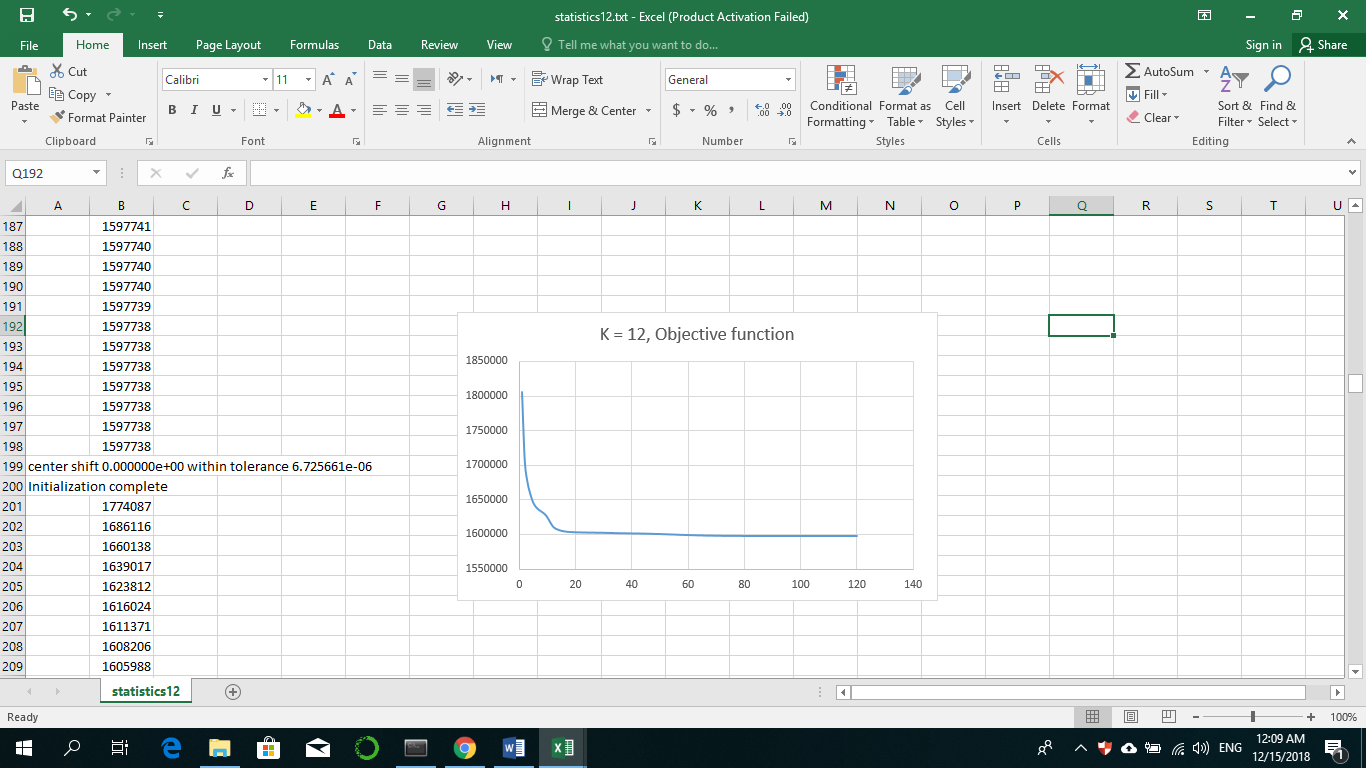
Run5: Max= 1779308.8004898103, Min= 1598417.5902334058

Train Accuracy = 0.6065, Test Accuracy = 0.6035









**Conclusion:**

After 300 iterations the model seems to have found mean images corresponding to the digits of the dataset, we ‘inspect’ if the images in the cluster correspond to said digit using the labels and it shows that most of the image in the cluster really correspond to the digit the center mean image shows.