# Assignment03

March 27, 2019

Assignment03: Visualize average images based on L2-norm
Software Engineering
20154652 Lee Dong Jae

In [5]:
```
import matplotlib.pyplot as plt
import numpy as np

file_data                = "mnist_train.csv"
handle_file        = open(file_data, "r")
data                        = handle_file.readlines()
handle_file.close()

size_row         = 28    # height of the image
size_col           = 28    # width of the image

num_image         = len(data)
count                = 0     # count for the number of images|
```

normalize the values of the input data to be [0, 1]

In [6]:
```
def normalize(data):

    data_normalized = (data - min(data)) / (max(data) - min(data))

    return(data_normalized)
```

example of distance function between two vectors x and y

In [7]:
```
def distance(x, y):

    d = (x - y) ** 2
    #s = np.sum(d)
    #r = np.sqrt(s)

    return(d)
```

make a matrix each column of which represents an images in a vector form

```
In [8]: list_image  = np.empty((size_row * size_col, num_image), dtype=float)
        list_label  = np.empty(num_image, dtype=int)

In [9]: for line in data:

            line_data   = line.split(',')
            label       = line_data[0]
            im_vector   = np.asfarray(line_data[1:])
            im_vector   = normalize(im_vector)

            list_label[count]       = label
            list_image[:, count]    = im_vector

            count += 1
```

plot first 100 images out of 10,000 with their labels

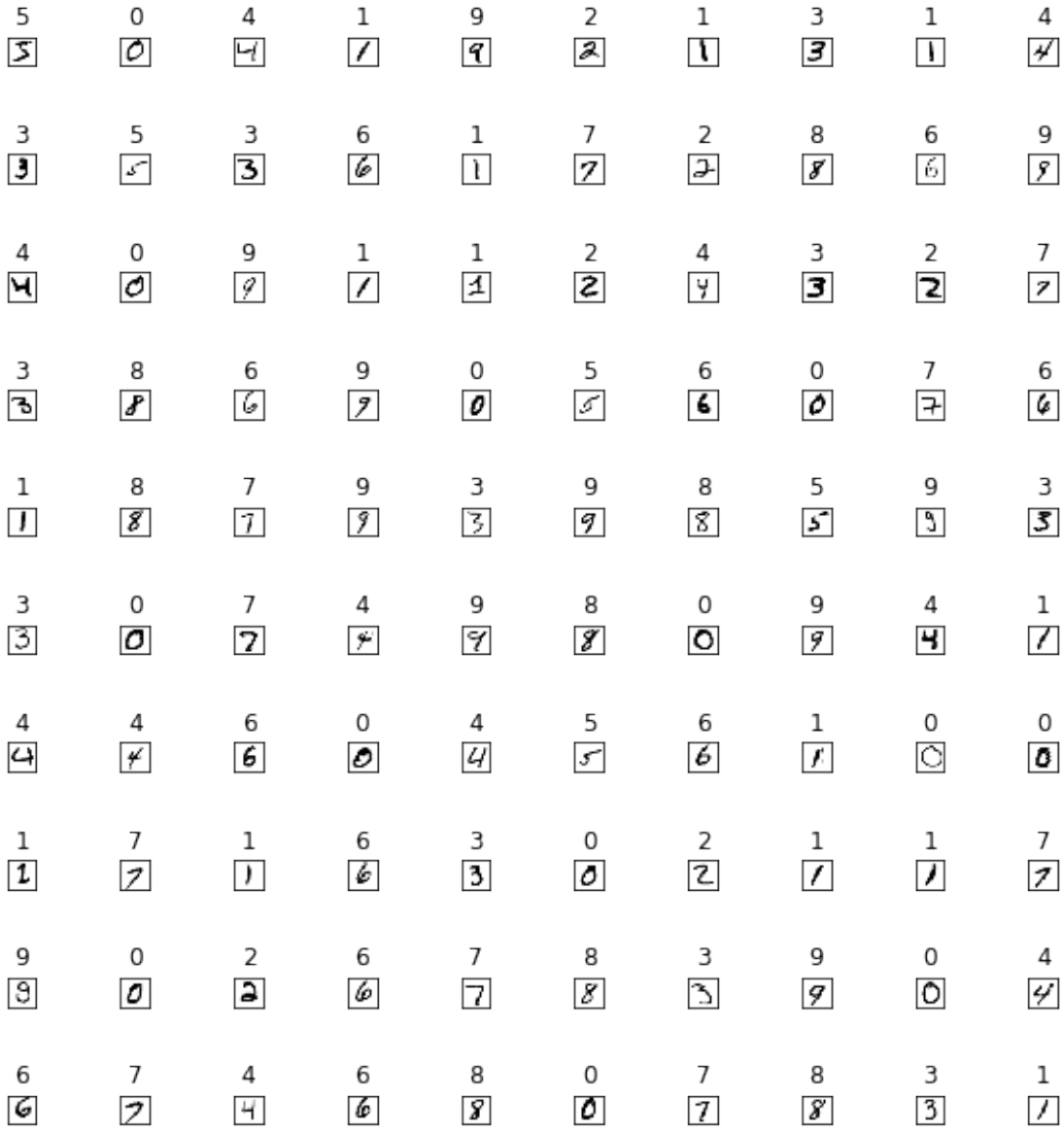
```
In [38]: f1 = plt.figure(figsize=(10,10))
         for i in range(100):

             label       = list_label[i]
             im_vector   = list_image[:, i]
             im_matrix   = im_vector.reshape((size_row, size_col))

             plt.subplot(10, 10, i+1)
             plt.subplots_adjust(hspace = 3)
             plt.title(label)
             plt.imshow(im_matrix, cmap='Greys', interpolation='None')

             frame   = plt.gca()
             frame.axes.get_xaxis().set_visible(False)
             frame.axes.get_yaxis().set_visible(False)

         plt.show()
```




Create a list of values and labels for the images

```
In [11]: ans_list  = np.zeros((size_col*size_row, 10), dtype=float)
```

Store the distance of the image in a list of corresponding labels for each image

```
In [12]: for i in range(60000):
             ans_list[:, list_label[i]] += distance(list_image[:,i], 0.0)
```

Distance obtained by label using L2-norm

```
In [13]: for i in range(10):
             ans_list[:, i] = np.sqrt(ans_list[:, i])
```

Visualize average images for each digits

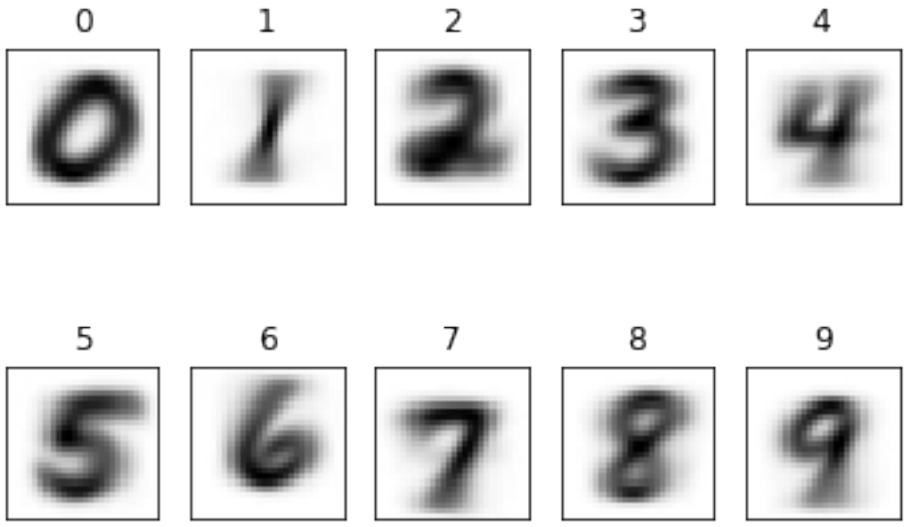
In [14]:
```
for i in range(10):

    label       = i
    im_vector   = ans_list[:, i]
    im_matrix   = im_vector.reshape((size_row, size_col))

    plt.subplot(2, 5, i+1)
    plt.title(label)
    plt.imshow(im_matrix, cmap='Greys', interpolation='None')

    frame   = plt.gca()
    frame.axes.get_xaxis().set_visible(False)
    frame.axes.get_yaxis().set_visible(False)

plt.show()
```




In [ ]: