assignment03

March 27, 2019

In [15]:
```
#
# 20141775 ParkJinHyuk
#
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]
#
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
#
def distance(x, y):

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

    return(s)
```

```
#
# calcuate the values of the input data in l2-norm
#
def norm(x):
    r = np.sqrt(x.T * x)

    return(r)

#
# make a matrix each column of which represents an images in a vector form
#
list_image  = np.empty((size_row * size_col, num_image), dtype=float)
list_label  = np.empty(num_image, dtype=int)
list_norm   = [0 for i in range(10)]
list_count  = [0 for i in range(10)]

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
    list_norm[int(label)]   += norm(im_vector)
    list_count[int(label)]  += 1

    count += 1


#
# each index get mean-value
#
for i in range(10):
    list_norm[i] /= list_count[i]

#
# plot 0 ~ 9 mean-images with their labels
#
f1 = plt.figure(1)

for i in range(10):

    label       = i
    im_vector   = list_norm[i]
    im_matrix   = im_vector.reshape((size_row, size_col))
```

```
    plt.subplot(1, 10, 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()
```