4.

Dataset of Hand-Written images of digits

60,000 examples.

each example → 28X28 matrix.

* Convert Integer datatype - float32

(i) Caluculated the mean for each cell inthe 28X28 matrix i.e 784 cells from the 60,000 samples for each digit from 0-9. So, there will be a 2D-array of 784X10

(ii) Similar to the part (i);
Caluculated the variance for each cell inthe 28X28 matrix i.e 784 cells from the 60,000 samples for each digit from 0-9. So, there will be a 2D-array of 784X10.

(iii) For every digit 0-9 form the 784X10 set calculate the max. eigen value for each digit and corresponding vector and store them in the vector.

We will store these values and vectors for each digit 0 – 9.

- Also in Q4_a.py we are printing all these values to console.

(a) Q4_b.py ⇒ code Figure1 to Figure10 images.

Here 28^2 eigenvalues and eigenvectors are calculated for each digit and they are sorted and stored.

Here from the graph we can see the eigenvalues becomes very less or negligible after a certain extent. These disturbances are most likely caused due to external noise and disturbance. So we can neglect all the eigenvalues after a certain extent of N.

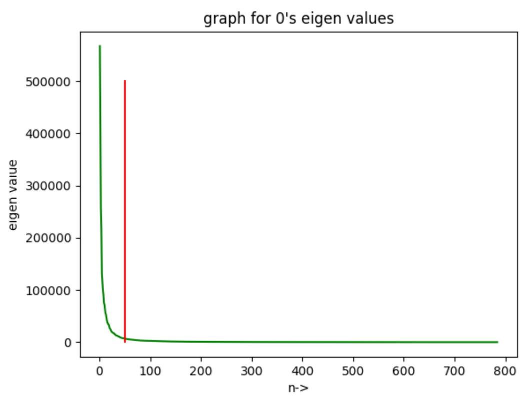
From the graph we have got some values for each digit 0 – 9 and also have plotted the graphs from 0 – 9.

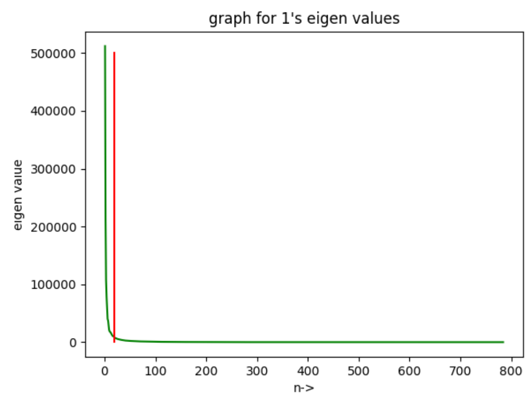
$n_0 = 50$ $n_1 = 20$ $n_2 = 60$ $n_3 = 50$ $n_4 = 50$

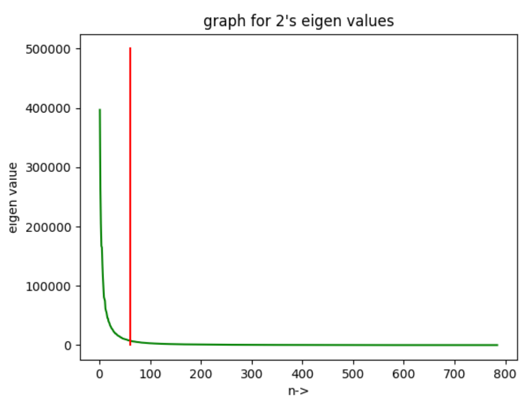
$n_5 = 50$ $n_6 = 50$ $n_7 = 50$ $n_8 = 50$ $n_9 = 50$

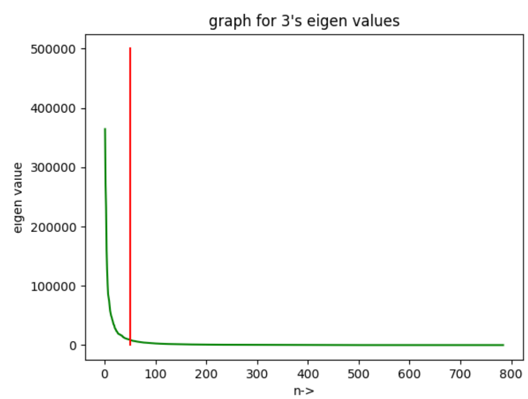
APPROXIMATE VALUES.

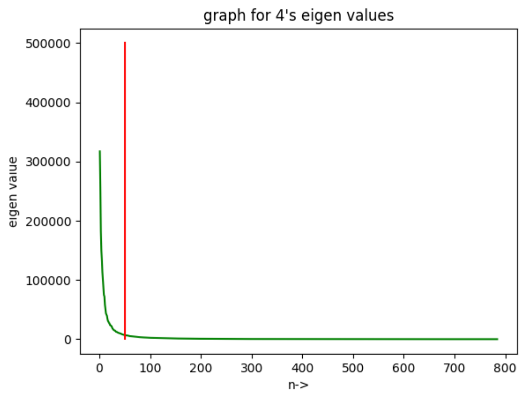
Graphs of Eigen values for digits 0-9

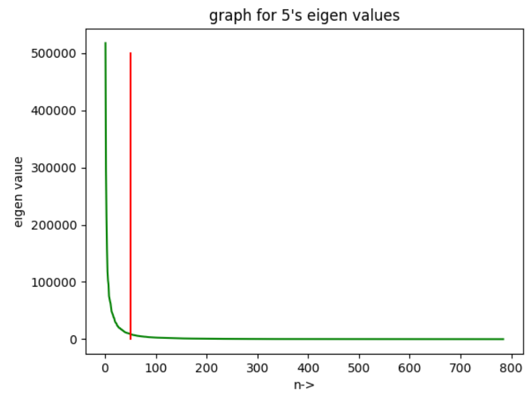


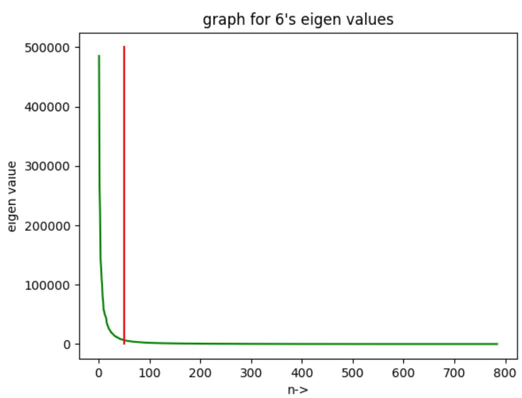


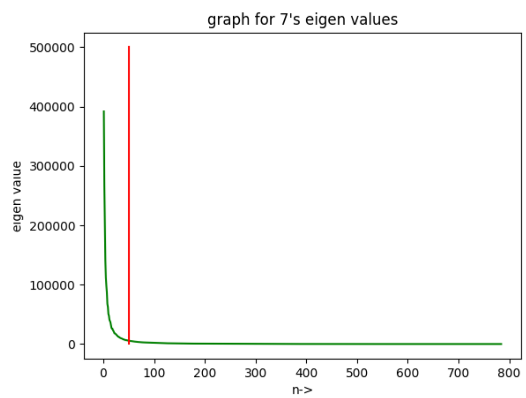


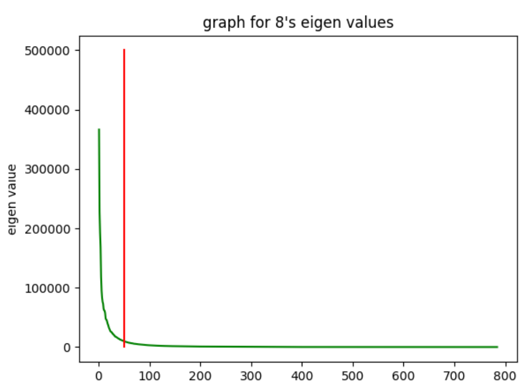


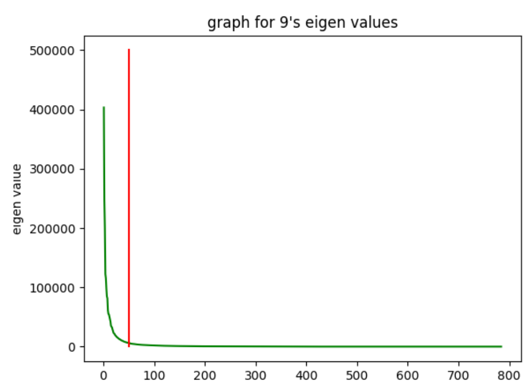

graph for 4's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->


graph for 5's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->


graph for 6's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->


graph for 7's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->


graph for 8's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->


graph for 9's eigen values
eigen value
0
100000
200000
300000
400000
500000
0 100 200 300 400 500 600 700 800
n->

b.

For each digit we will plot 3 images.

$$\left[\mu - \sqrt{\lambda_1} v_1 \quad \mu \quad \mu + \sqrt{\lambda_1} v_1 \right]$$

* 10 images are are plotted with all the 3 images plots in the same image side by side for

$$\left[\mu - \sqrt{\lambda_1} v_1 \quad \mu \quad \mu + \sqrt{\lambda_1} v_1 \right].$$

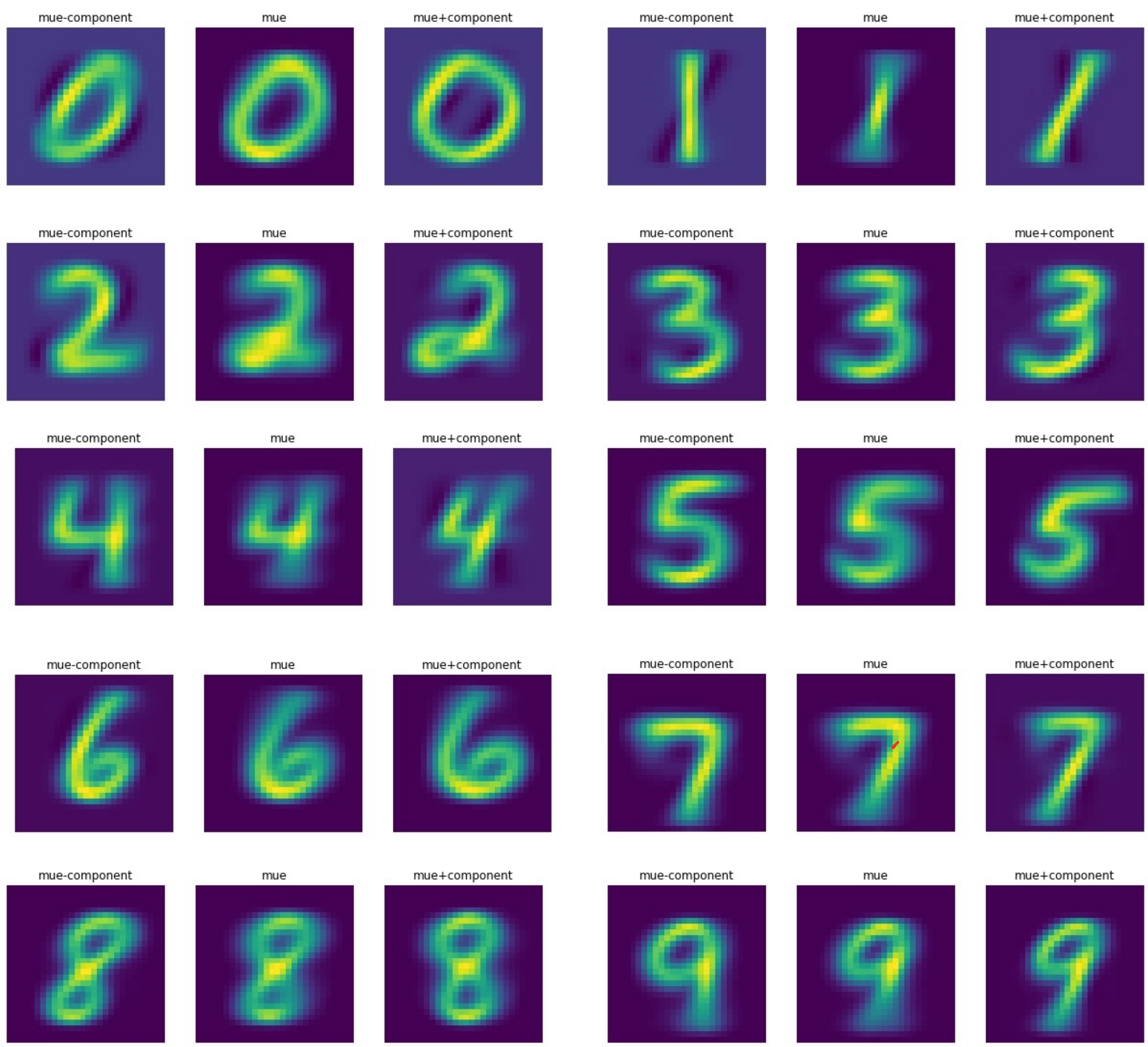
OBSERVATIONS:

- $\mu - \sqrt{\lambda_1} v$ to $\mu + \sqrt{\lambda_1} v$ highlights the space where most of the digit examples lie in.
- All the variations in the digit are almost covered in the region of $\mu - \sqrt{\lambda_1} v$ to $\mu_1 + \sqrt{\lambda_1} v$.
- This gives a trend as how most of the users write the numbers.
- Just like how most the matter of gaussian distribution lie $\mu - \sigma$ to $\mu + \sigma$ similar to that

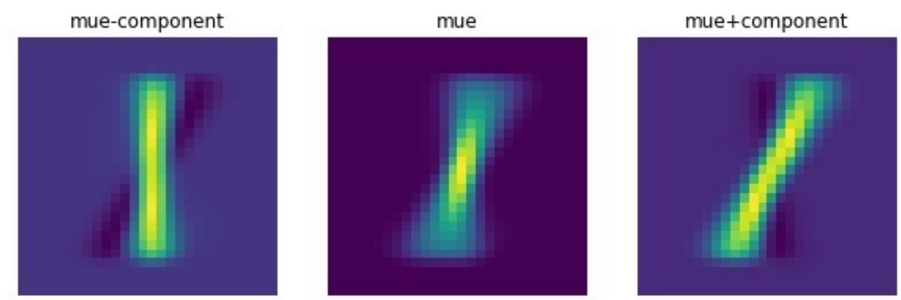
here also most of variations lie $\mu - \sqrt{\lambda_1} V$ to $\mu + \sqrt{\lambda_1} V$.

where $\sqrt{\lambda_1}$ is <u>like</u> variance.

- Because there always is a style to the writers numbers, it is not a random process. Though some amount of variations are handled.

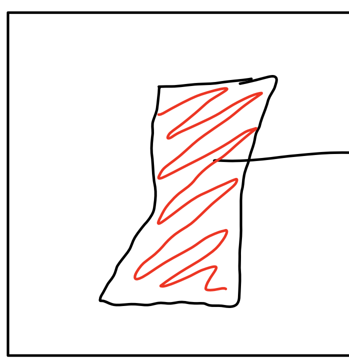


Special Observation in the case of Number 1:



From the image we can say that

from μ image that



This is the region where most of styles where 1 is written.

And $\mu + \sqrt{\lambda_1}\, v$ has the variation of slant 1 here

and $\mu - \sqrt{\lambda_1}\, v$ has the variation of 1 written straight.

→ So in the region $\mu - \sqrt{\lambda_1}\, v$ to $\mu + \sqrt{\lambda_1}\, v$ has all the variation from straight to slant.