

RESULTS

01. plot examples of the input training images for '0'

In []:

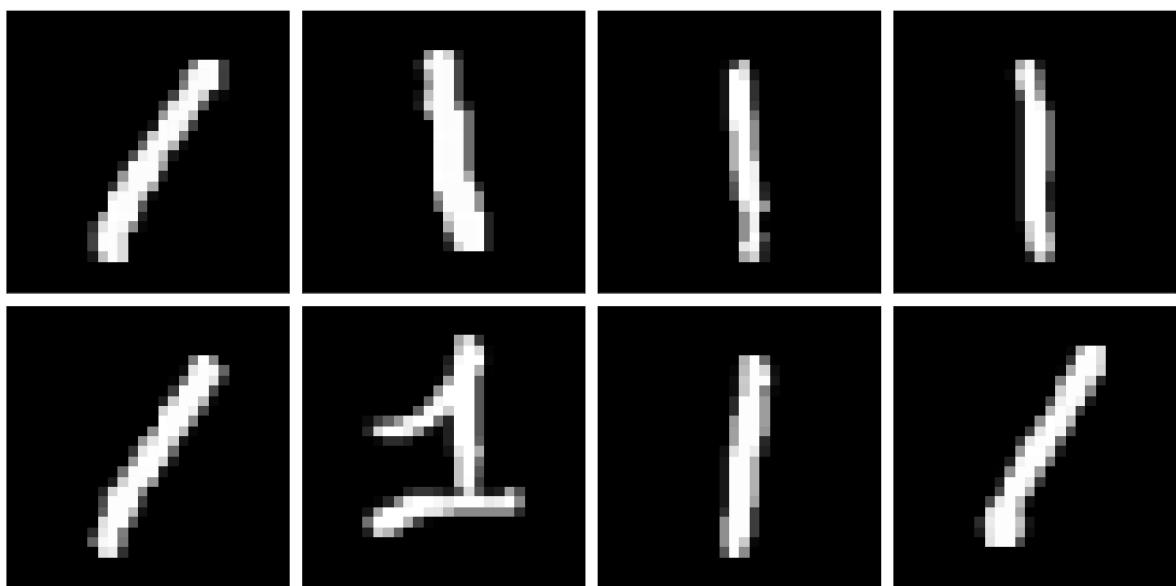
```
function_results_01(x_train, index_data_0, nRow, nCol)
```



02. plot examples of the input training images for '1'

In []:

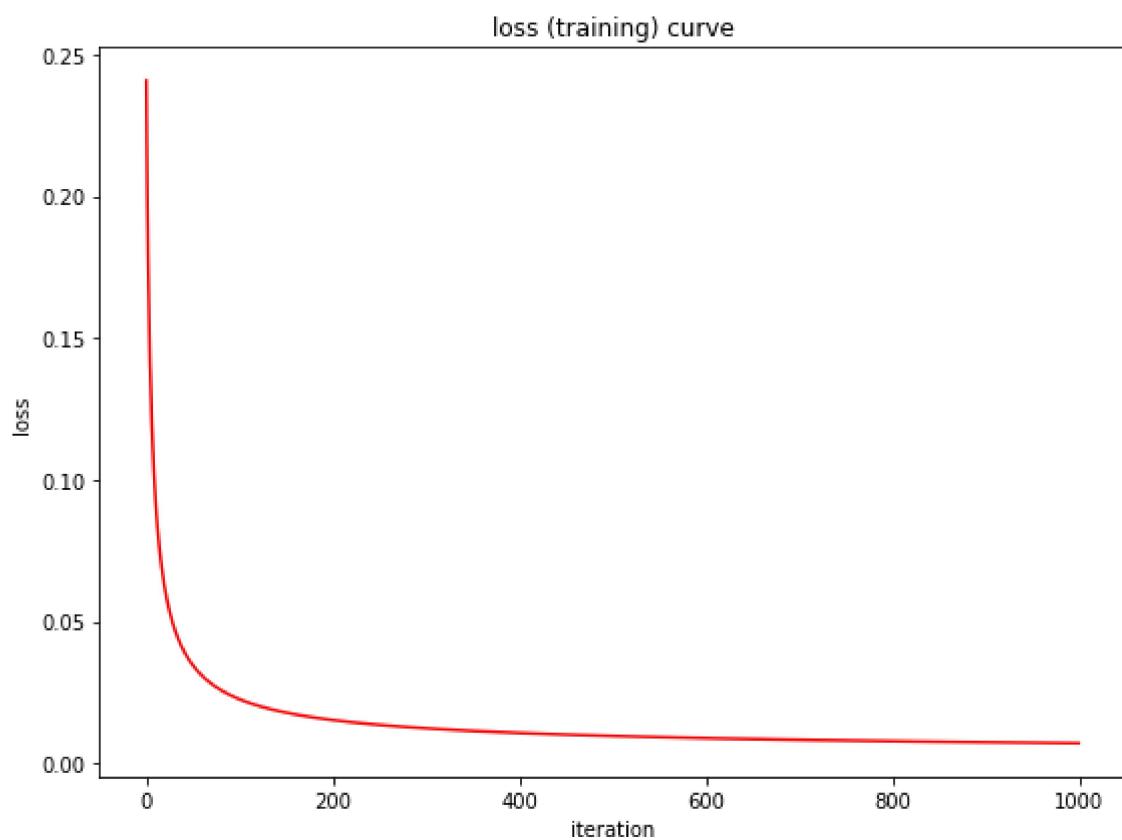
```
function_results_02(x_train, index_data_1, nRow, nCol)
```



03. plot the training loss curve (x-axis: iteration, y-

axis: loss)

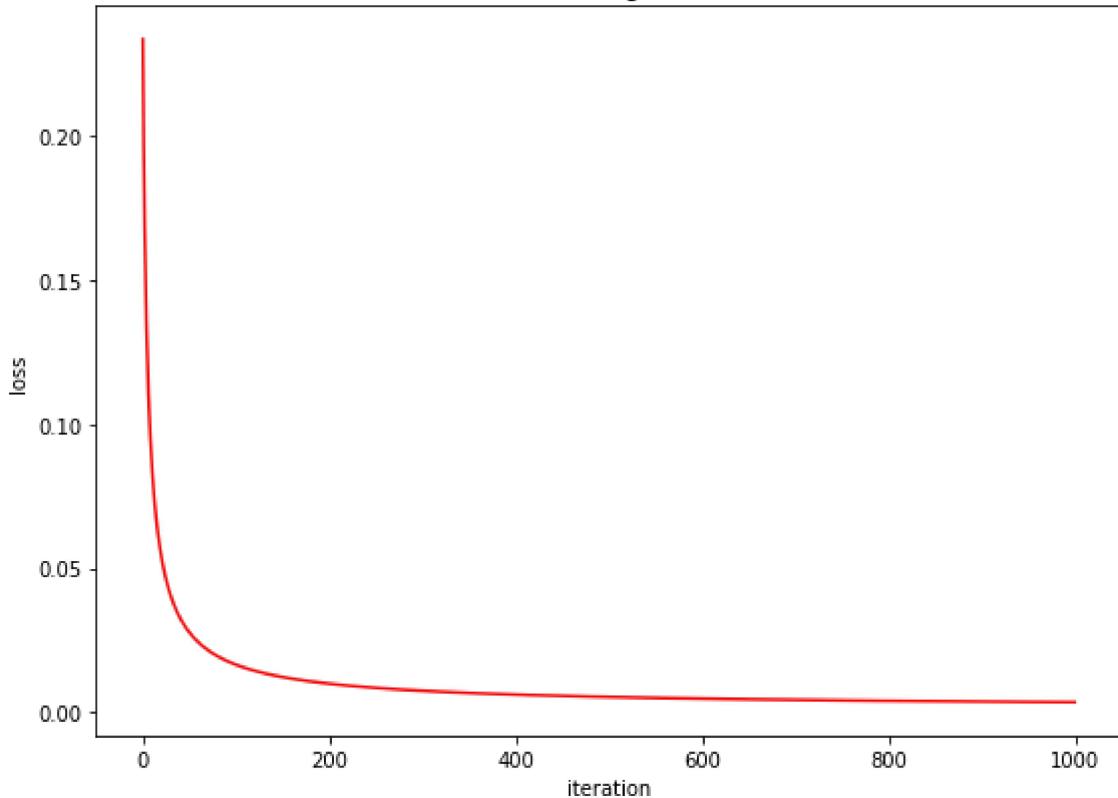
```
In [ ]: function_results_03(loss_train_iteration)
```



04. plot the testing loss curve (x-axis: iteration, y-axis: loss)

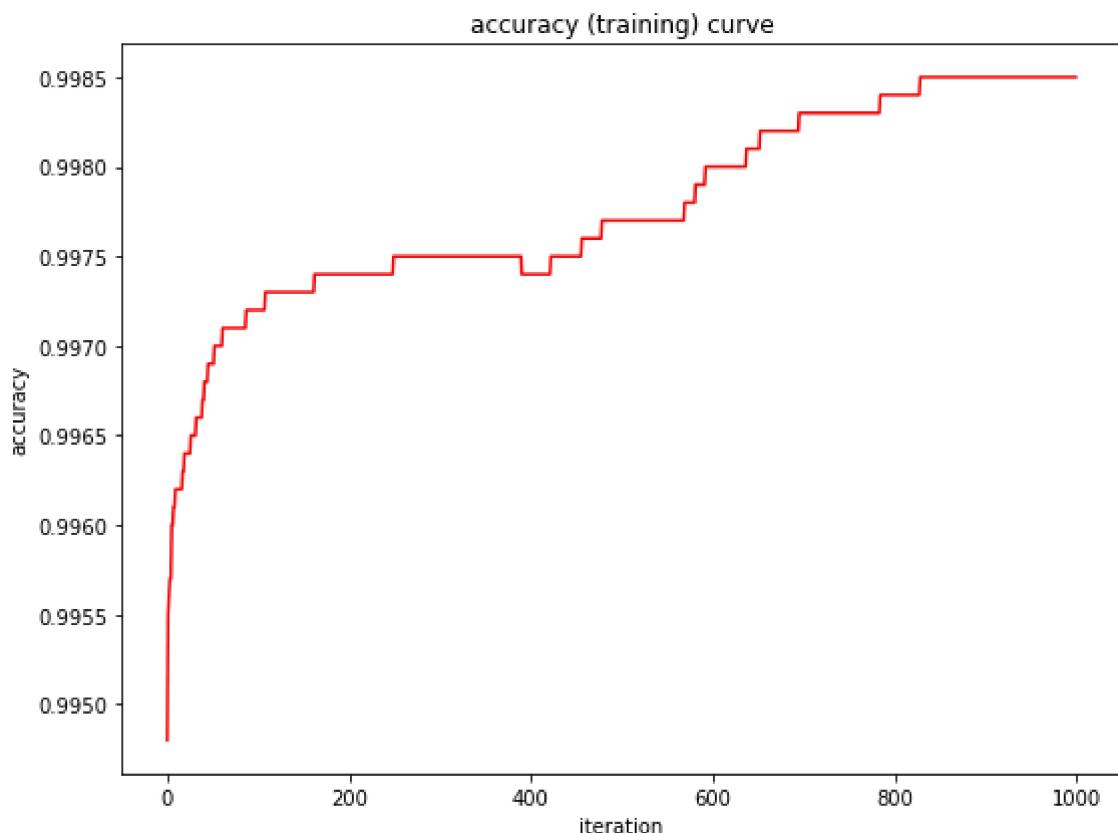
```
In [ ]: function_results_04(loss_test_iteration)
```

loss (testing) curve



05. plot the training accuracy curve (x-axis: iteration, y-axis: accuracy)

```
In [ ]: function_results_05(accuracy_train_iteration)
```

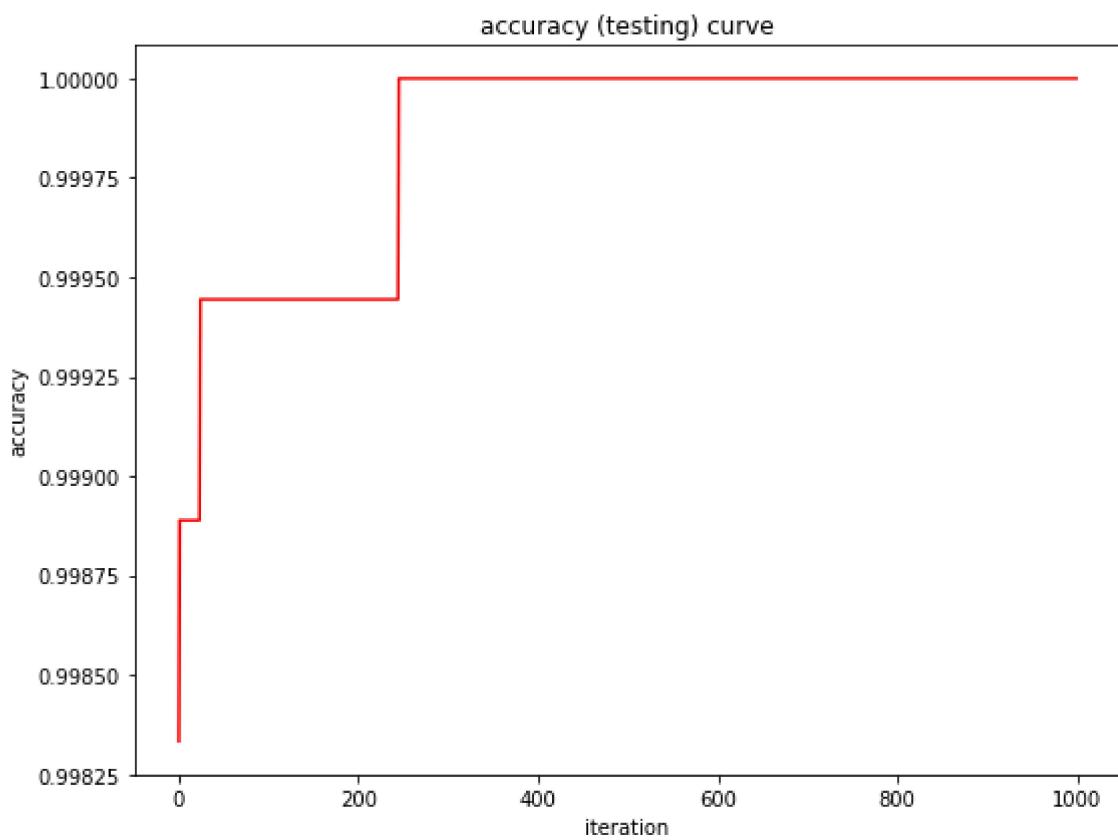


06. plot the testing accuracy curve (x-axis:

iteration, y-axis: accuracy)

In []:

```
function_results_06(accuracy_test_iteration)
```

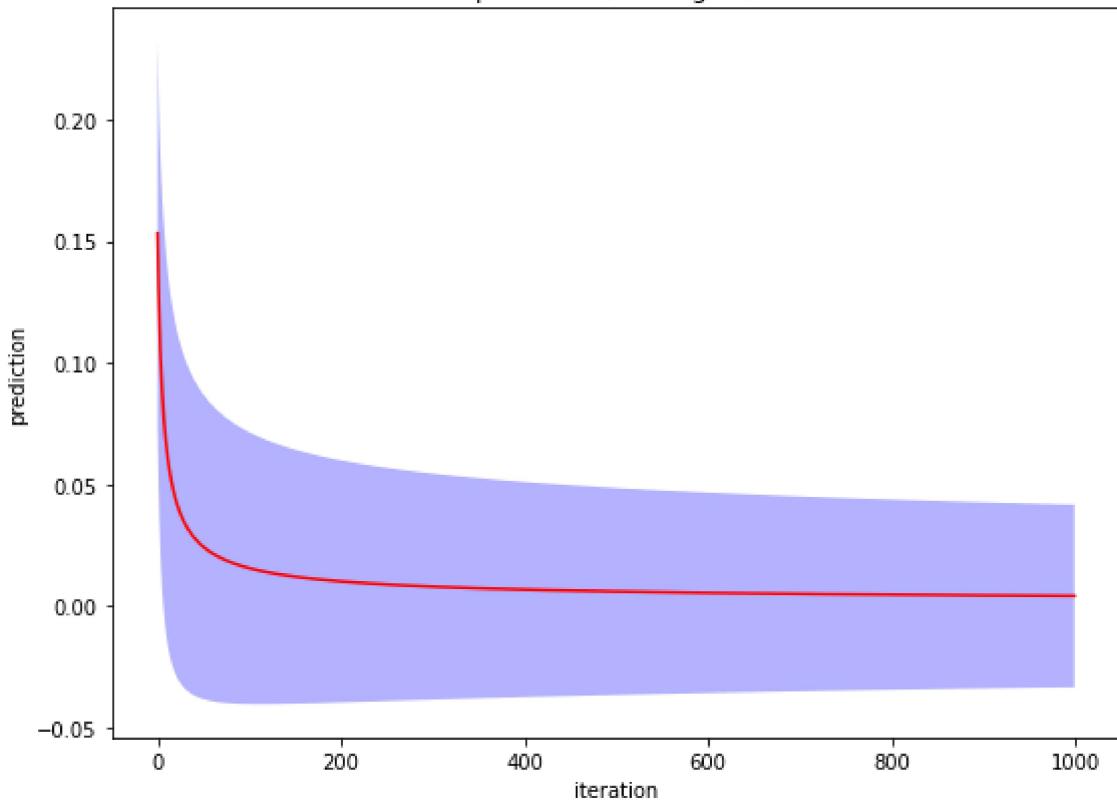


07. plot the training prediction curve (mean and std) for image 0 (x-axis: iteration, y-axis: prediction)

In []:

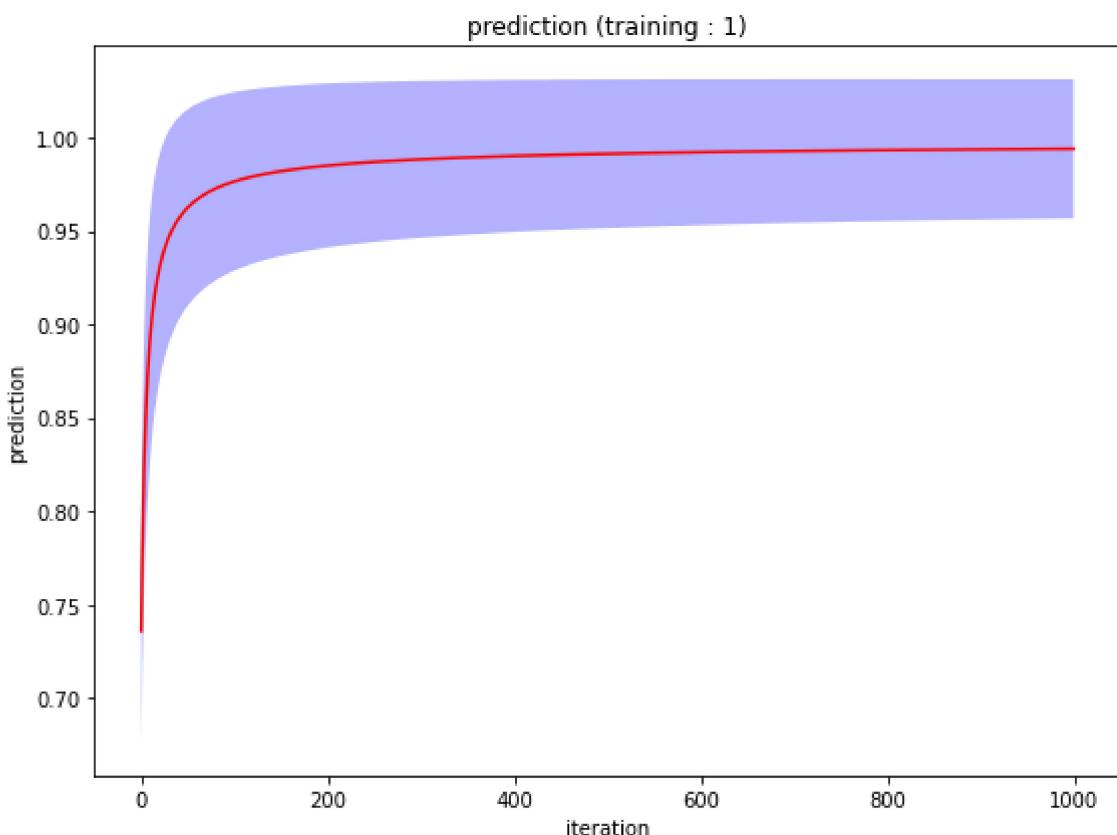
```
function_results_07(pred_0_train_mean_iteration, pred_0_train_std_iteration)
```

prediction (training : 0)



08. plot the training prediction curve (mean and std) for image 1 (x-axis: iteration, y-axis: prediction)

In []: `function_results_08(pred_1_train_mean_iteration, pred_1_train_std_iteration)`

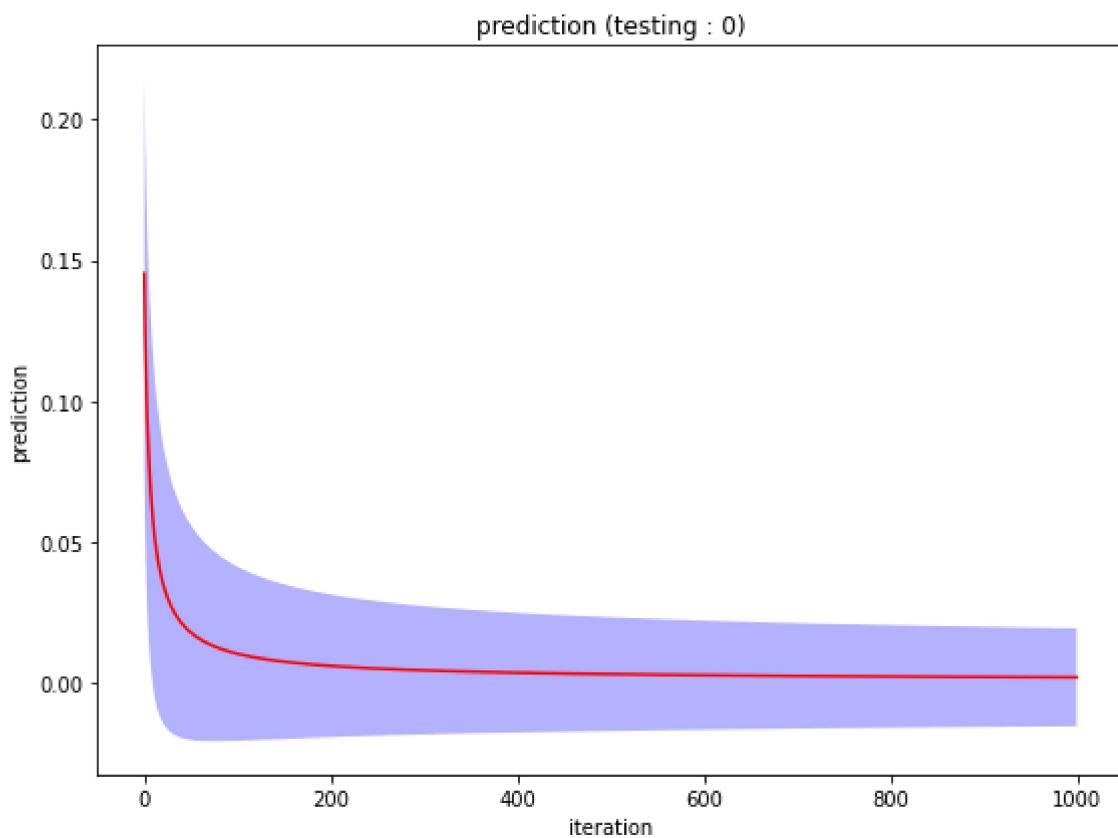


09. plot the testing prediction curve (mean and

std) for image 0 (x-axis: iteration, y-axis: prediction)

In []:

```
function_results_09(pred_0_test_mean_iteration, pred_0_test_std_iteration)
```

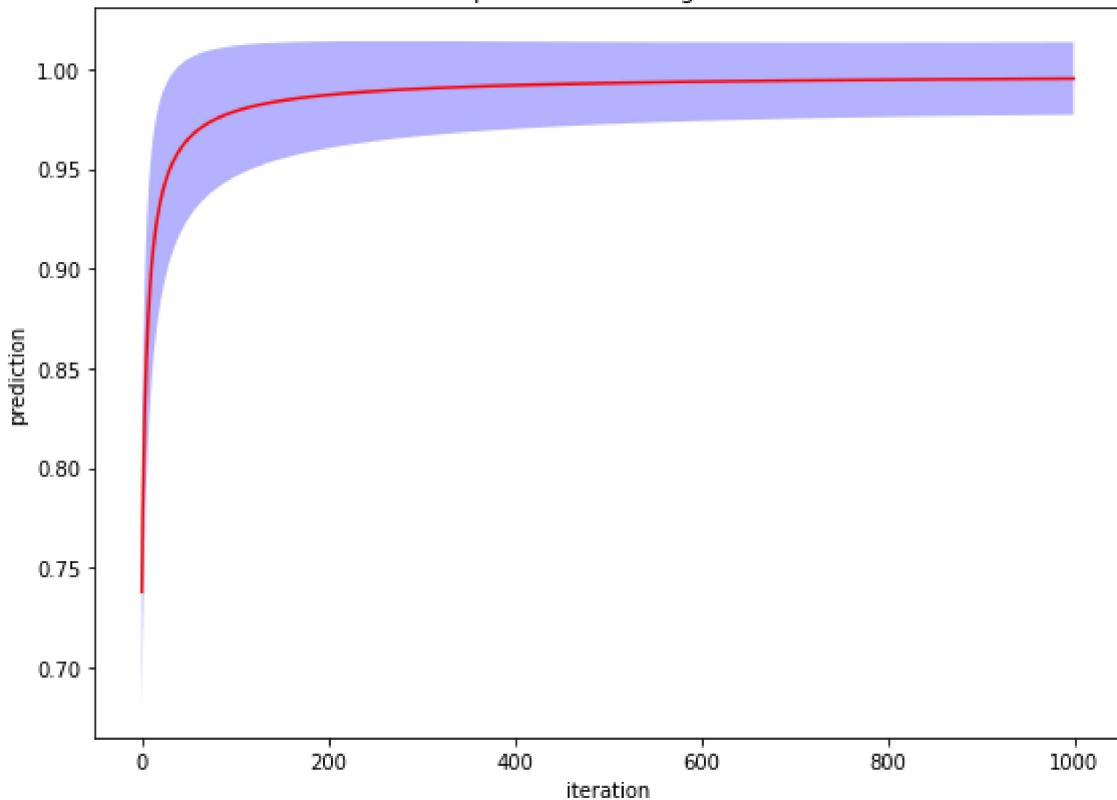


10. plot the testing prediction curve (mean and std) for image 1 (x-axis: iteration, y-axis: prediction)

In []:

```
function_results_10(pred_1_test_mean_iteration, pred_1_test_std_iteration)
```

prediction (testing : 1)



11. print the training loss at iterations 0, 100, 200, 300, 400, 500, 600, 700, 800, 900

```
In [ ]: function_results_11(loss_train_iteration, index)
```

```
index = 0, value = 0.2408591875
index = 100, value = 0.0225553021
index = 200, value = 0.0151963959
index = 300, value = 0.0122820566
index = 400, value = 0.0106472273
index = 500, value = 0.0095737094
index = 600, value = 0.0088019146
index = 700, value = 0.0082133315
index = 800, value = 0.0077454629
index = 900, value = 0.0073619313
```

12. print the testing loss at iterations 0, 100, 200, 300, 400, 500, 600, 700, 800, 900

```
In [ ]: function_results_12(loss_test_iteration, index)
```

```
index = 0, value = 0.2338833327
index = 100, value = 0.0164594767
index = 200, value = 0.0098669205
index = 300, value = 0.0073959195
index = 400, value = 0.0060733233
index = 500, value = 0.0052411869
index = 600, value = 0.0046664217
index = 700, value = 0.0042445050
index = 800, value = 0.0039212256
index = 900, value = 0.0036655191
```

13. print the training accuracy at iterations 0, 100, 200, 300, 400, 500, 600, 700, 800, 900

In []:

```
function_results_13(accuracy_train_iteration, index)
```

```
index = 0, value = 0.9948000000
index = 100, value = 0.9972000000
index = 200, value = 0.9974000000
index = 300, value = 0.9975000000
index = 400, value = 0.9974000000
index = 500, value = 0.9977000000
index = 600, value = 0.9980000000
index = 700, value = 0.9983000000
index = 800, value = 0.9984000000
index = 900, value = 0.9985000000
```

14. print the testing accuracy at iterations 0, 100, 200, 300, 400, 500, 600, 700, 800, 900

In []:

```
function_results_14(accuracy_test_iteration, index)
```

```
index = 0, value = 0.9983333333
index = 100, value = 0.9994444444
index = 200, value = 0.9994444444
index = 300, value = 1.0000000000
index = 400, value = 1.0000000000
index = 500, value = 1.0000000000
index = 600, value = 1.0000000000
index = 700, value = 1.0000000000
index = 800, value = 1.0000000000
index = 900, value = 1.0000000000
```