

# Machine Learning Project 07 completed

November 21, 2019

## 1 Binary classification based on fully connected neural network

```
[1]: import torch
from torch.utils.data import Dataset, DataLoader
import torchvision.transforms as transforms
from torch.autograd import Variable
import torchvision
import os
import sys

#from scipy.special import xlogy

import matplotlib.pyplot as plt
import numpy as np
import time

transform = transforms.Compose([#transforms.Resize((256,256)),
                               transforms.Grayscale(), # the
                               ↪code transforms.Grayscale() is for changing the size [3,100,100] to [1, 100,
                               ↪100] (notice : [channel, height, width] )
                               transforms.ToTensor(),])

#train_data_path = 'relative path of training data set'
train_data_path = 'C:\\Users\\newmi\\OneDrive\\ 
↪ \\horse-or-human\\horse-or-human\\train'
trainset = torchvision.datasets.ImageFolder(root=train_data_path,
↪transform=transform)
# change the valuse of batch_size, num_workers for your program
# if shuffle=True, the data reshuffled at every epoch
loader_train = torch.utils.data.DataLoader(trainset, batch_size=30,
↪shuffle=False, num_workers=1)

validation_data_path = 'C:\\Users\\newmi\\OneDrive\\ 
↪ \\horse-or-human\\horse-or-human\\validation'
```

```

valset = torchvision.datasets.ImageFolder(root=validation_data_path,
    ↪transform=transform)
# change the valuse of batch_size, num_workers for your program
loader_test = torch.utils.data.DataLoader(valset, batch_size=30, shuffle=False,
    ↪num_workers=1)

```

- Loss function with a regularization term based on L2 norm
- Optimization Stochastic gradient descent with LeRU function

```

[357]: import torch
import torchvision.datasets as dsets
import torchvision.transforms as transforms
import random

# -----
# Set the flag for using cuda
# -----

device = 'cuda' if torch.cuda.is_available() else 'cpu'

if device == 'cuda':
    torch.cuda.manual_seed_all(777)

random.seed(111)
torch.manual_seed(777)
mnist_train=trainset
mnist_test =valset

# -----
# load neural network model
# -----

epochs = 153
batch_size = 8
learning_rate = 0.0001
weight_decay=1

loader_train = torch.utils.data.DataLoader(mnist_train, batch_size=batch_size,
    ↪shuffle=True, drop_last=True)
loader_test= torch.utils.data.DataLoader(mnist_test, batch_size=batch_size,
    ↪shuffle=True, drop_last=True)
linear1 = torch.nn.Linear(100*100, 50, bias=True)
linear2 = torch.nn.Linear(50,50, bias=True)
linear3 = torch.nn.Linear(50,11, bias=True)
relu = torch.nn.ReLU()

```

```

torch.nn.init.normal_(linear1.weight)
torch.nn.init.normal_(linear2.weight)
torch.nn.init.normal_(linear3.weight)

model = torch.nn.Sequential(linear1, relu, linear2, relu, linear3).to(device)

# -----
# optimization algorithm
# -----

loss_function = torch.nn.CrossEntropyLoss().to(device)
optimizer = torch.optim.SGD(model.parameters(),
    ↪lr=learning_rate,weight_decay=weight_decay)

# -----
# Initialization of loss array and accuracy array
# -----

loss_train_mean      = np.zeros(epochs)
loss_train_std       = np.zeros(epochs)
accuracy_train       = np.zeros(epochs)
accuracy_train_std   = np.zeros(epochs)

loss_test_mean       = np.zeros(epochs)
loss_test_std        = np.zeros(epochs)
accuracy_test        = np.zeros(epochs)

# -----
# Calculating for training the model
# -----

print('start')
for epoch in range(epochs):

    # ↪
    ↪-----
    ↪ function for training the model
    ↪-----

    ↪-----
    avg_loss_train = 0
    batch_count_train = len(loader_train)
    running_corrects_train = 0
    loss_accuracy      = []

```

```

loss_train          = []

for X, Y in loader_train:

    X = X.view(-1, 100*100).to(device)
    Y = Y.to(device)
    prediction = model(X)
    _, preds = torch.max(prediction, 1)
    loss = loss_function(prediction, Y)

    optimizer.zero_grad()
    loss.backward()
    optimizer.step()

    loss_train_batch = loss.item()/len(X)
    loss_train.append(loss_train_batch)
    running_corrects_train += torch.sum(preds == Y).item() /
    len(loader_train.dataset)
    loss_accuracy.append(running_corrects_train)

loss_train_mean[epoch] = np.mean(loss_train)
loss_train_std[epoch] = np.std(loss_train)
accuracy_train_std[epoch] = np.std(loss_accuracy)
accuracy_train[epoch]=running_corrects_train

#
→ -----
# function for testing the model
#
→ -----

avg_loss_test = 0
batch_count_test = len(loader_test)
running_corrects_test = 0
loss_test = []

for datas, labels in loader_test:

    datas = datas.view(-1, 100*100).to(device)
    labels = labels.to(device)
    prediction = model(datas)
    _, preds_test = torch.max(prediction, 1)
    loss = loss_function(prediction, labels)

```

```

    loss_test_batch      =loss.item()/len(datas)
    loss_test.append(loss_test_batch)
    running_corrects_test += torch.sum(preds_test == labels).item()

    loss_test_mean[epoch]      = np.mean(loss_test)
    running_corrects_test      = running_corrects_test / len(loader_test.
↪dataset)
    accuracy_test[epoch]      = running_corrects_test

    print("""[EPOCH %4d ] LOSS      : (TRAIN) %3.10f      (TEST): %3.10f
            ACCURACY : (TRAIN) %3.10f%%      (TEST): %3.
↪10f%%\n""")%(epoch,loss_train_batch,loss_test_batch,(running_corrects_train*100),(running_co

```

```

start
[EPOCH  0 ] LOSS      : (TRAIN) 1.7101622820      (TEST): 7.5678787231
            ACCURACY : (TRAIN) 49.8888888889%      (TEST): 62.8000000000%

[EPOCH  1 ] LOSS      : (TRAIN) 10.6595630646      (TEST): 11.3186035156
            ACCURACY : (TRAIN) 68.3333333333%      (TEST): 69.2000000000%

[EPOCH  2 ] LOSS      : (TRAIN) 2.8114550114      (TEST): 6.7598609924
            ACCURACY : (TRAIN) 71.2222222222%      (TEST): 69.6000000000%

[EPOCH  3 ] LOSS      : (TRAIN) 4.2264733315      (TEST): 3.7534179688
            ACCURACY : (TRAIN) 74.7777777778%      (TEST): 76.8000000000%

[EPOCH  4 ] LOSS      : (TRAIN) 1.4688758850      (TEST): 0.0000006631
            ACCURACY : (TRAIN) 77.4444444444%      (TEST): 78.0000000000%

[EPOCH  5 ] LOSS      : (TRAIN) 0.2081146538      (TEST): 8.4846649170
            ACCURACY : (TRAIN) 75.3333333333%      (TEST): 73.6000000000%

[EPOCH  6 ] LOSS      : (TRAIN) 1.2081222534      (TEST): 5.3128433228
            ACCURACY : (TRAIN) 76.7777777778%      (TEST): 76.4000000000%

[EPOCH  7 ] LOSS      : (TRAIN) 3.2477390766      (TEST): 2.7831389904
            ACCURACY : (TRAIN) 76.8888888889%      (TEST): 83.2000000000%

[EPOCH  8 ] LOSS      : (TRAIN) 2.1362209320      (TEST): 0.0000000000
            ACCURACY : (TRAIN) 78.4444444444%      (TEST): 79.2000000000%

[EPOCH  9 ] LOSS      : (TRAIN) 4.8848609924      (TEST): 3.4705963135
            ACCURACY : (TRAIN) 77.7777777778%      (TEST): 70.4000000000%

[EPOCH 10 ] LOSS      : (TRAIN) 1.5108711720      (TEST): 1.7557337284

```

		ACCURACY : (TRAIN) 78.0000000000%	(TEST): 85.6000000000%
[EPOCH	11 ]	LOSS : (TRAIN) 1.4595975876	(TEST): 1.4738998413
		ACCURACY : (TRAIN) 80.0000000000%	(TEST): 70.8000000000%
[EPOCH	12 ]	LOSS : (TRAIN) 2.2557175159	(TEST): 1.4123878479
		ACCURACY : (TRAIN) 80.8888888889%	(TEST): 86.0000000000%
[EPOCH	13 ]	LOSS : (TRAIN) 1.3138551712	(TEST): 1.5129566193
		ACCURACY : (TRAIN) 81.0000000000%	(TEST): 84.4000000000%
[EPOCH	14 ]	LOSS : (TRAIN) 1.2949386835	(TEST): 0.9536170959
		ACCURACY : (TRAIN) 82.2222222222%	(TEST): 77.6000000000%
[EPOCH	15 ]	LOSS : (TRAIN) 0.8715671897	(TEST): 0.0000000019
		ACCURACY : (TRAIN) 81.0000000000%	(TEST): 72.4000000000%
[EPOCH	16 ]	LOSS : (TRAIN) 0.3827417791	(TEST): 0.1433388740
		ACCURACY : (TRAIN) 83.4444444444%	(TEST): 84.0000000000%
[EPOCH	17 ]	LOSS : (TRAIN) 0.5972720385	(TEST): 0.0000000000
		ACCURACY : (TRAIN) 83.1111111111%	(TEST): 85.6000000000%
[EPOCH	18 ]	LOSS : (TRAIN) 1.0651102066	(TEST): 0.2868421674
		ACCURACY : (TRAIN) 82.0000000000%	(TEST): 85.2000000000%
[EPOCH	19 ]	LOSS : (TRAIN) 0.4663712382	(TEST): 2.0420131683
		ACCURACY : (TRAIN) 82.6666666667%	(TEST): 74.4000000000%
[EPOCH	20 ]	LOSS : (TRAIN) 0.7583829761	(TEST): 0.7109985352
		ACCURACY : (TRAIN) 84.8888888889%	(TEST): 82.4000000000%
[EPOCH	21 ]	LOSS : (TRAIN) 0.5283318758	(TEST): 0.0087740626
		ACCURACY : (TRAIN) 84.3333333333%	(TEST): 82.0000000000%
[EPOCH	22 ]	LOSS : (TRAIN) 0.1836417913	(TEST): 1.3823264837
		ACCURACY : (TRAIN) 83.7777777778%	(TEST): 81.6000000000%
[EPOCH	23 ]	LOSS : (TRAIN) 0.5266298056	(TEST): 1.2149221897
		ACCURACY : (TRAIN) 81.4444444444%	(TEST): 66.8000000000%
[EPOCH	24 ]	LOSS : (TRAIN) 0.3542418778	(TEST): 1.4568195343
		ACCURACY : (TRAIN) 82.2222222222%	(TEST): 84.0000000000%
[EPOCH	25 ]	LOSS : (TRAIN) 0.2227140516	(TEST): 0.5291764736
		ACCURACY : (TRAIN) 83.6666666667%	(TEST): 84.4000000000%
[EPOCH	26 ]	LOSS : (TRAIN) 0.1452812850	(TEST): 1.5303027630

		ACCURACY : (TRAIN) 83.3333333333%	(TEST): 66.0000000000%
[EPOCH	27 ]	LOSS : (TRAIN) 0.0026930172	(TEST): 0.3684248626
		ACCURACY : (TRAIN) 85.3333333333%	(TEST): 83.6000000000%
[EPOCH	28 ]	LOSS : (TRAIN) 0.3586052656	(TEST): 0.3123692870
		ACCURACY : (TRAIN) 86.0000000000%	(TEST): 85.6000000000%
[EPOCH	29 ]	LOSS : (TRAIN) 0.0441210940	(TEST): 0.4304371476
		ACCURACY : (TRAIN) 84.5555555556%	(TEST): 82.0000000000%
[EPOCH	30 ]	LOSS : (TRAIN) 0.1276044101	(TEST): 0.0000877297
		ACCURACY : (TRAIN) 85.0000000000%	(TEST): 86.0000000000%
[EPOCH	31 ]	LOSS : (TRAIN) 0.2566736042	(TEST): 0.3251261711
		ACCURACY : (TRAIN) 85.5555555556%	(TEST): 86.4000000000%
[EPOCH	32 ]	LOSS : (TRAIN) 0.4494494200	(TEST): 0.7458223104
		ACCURACY : (TRAIN) 83.2222222222%	(TEST): 80.8000000000%
[EPOCH	33 ]	LOSS : (TRAIN) 0.8801454902	(TEST): 2.3509998322
		ACCURACY : (TRAIN) 84.1111111111%	(TEST): 60.0000000000%
[EPOCH	34 ]	LOSS : (TRAIN) 0.1400978416	(TEST): 0.4563523233
		ACCURACY : (TRAIN) 85.8888888889%	(TEST): 84.0000000000%
[EPOCH	35 ]	LOSS : (TRAIN) 0.0667186454	(TEST): 0.0379672870
		ACCURACY : (TRAIN) 85.6666666667%	(TEST): 74.0000000000%
[EPOCH	36 ]	LOSS : (TRAIN) 0.4549833238	(TEST): 0.0001682621
		ACCURACY : (TRAIN) 85.4444444444%	(TEST): 73.2000000000%
[EPOCH	37 ]	LOSS : (TRAIN) 0.0821432620	(TEST): 0.5577158928
		ACCURACY : (TRAIN) 86.0000000000%	(TEST): 76.0000000000%
[EPOCH	38 ]	LOSS : (TRAIN) 0.2309833765	(TEST): 0.1449502856
		ACCURACY : (TRAIN) 86.8888888889%	(TEST): 76.4000000000%
[EPOCH	39 ]	LOSS : (TRAIN) 0.0718543455	(TEST): 0.1682496220
		ACCURACY : (TRAIN) 85.7777777778%	(TEST): 84.0000000000%
[EPOCH	40 ]	LOSS : (TRAIN) 0.0128387408	(TEST): 0.1427793354
		ACCURACY : (TRAIN) 87.3333333333%	(TEST): 83.2000000000%
[EPOCH	41 ]	LOSS : (TRAIN) 0.1994519085	(TEST): 0.2404633015
		ACCURACY : (TRAIN) 88.7777777778%	(TEST): 83.2000000000%
[EPOCH	42 ]	LOSS : (TRAIN) 0.1657607108	(TEST): 0.7279039621

		ACCURACY : (TRAIN) 87.1111111111%	(TEST): 69.2000000000%
[EPOCH	43 ]	LOSS : (TRAIN) 0.1356576532	(TEST): 0.2008712292
		ACCURACY : (TRAIN) 87.1111111111%	(TEST): 84.4000000000%
[EPOCH	44 ]	LOSS : (TRAIN) 0.1962220967	(TEST): 0.0021964649
		ACCURACY : (TRAIN) 86.4444444444%	(TEST): 85.6000000000%
[EPOCH	45 ]	LOSS : (TRAIN) 0.0609351024	(TEST): 0.0013723866
		ACCURACY : (TRAIN) 87.1111111111%	(TEST): 86.0000000000%
[EPOCH	46 ]	LOSS : (TRAIN) 0.2113592029	(TEST): 0.3927392364
		ACCURACY : (TRAIN) 86.0000000000%	(TEST): 76.4000000000%
[EPOCH	47 ]	LOSS : (TRAIN) 0.0242394879	(TEST): 0.1471698284
		ACCURACY : (TRAIN) 86.2222222222%	(TEST): 84.4000000000%
[EPOCH	48 ]	LOSS : (TRAIN) 0.0963171646	(TEST): 0.0034536682
		ACCURACY : (TRAIN) 87.6666666667%	(TEST): 86.4000000000%
[EPOCH	49 ]	LOSS : (TRAIN) 0.0148818213	(TEST): 0.0105860289
		ACCURACY : (TRAIN) 84.8888888889%	(TEST): 85.2000000000%
[EPOCH	50 ]	LOSS : (TRAIN) 0.0112080164	(TEST): 0.2010754049
		ACCURACY : (TRAIN) 87.2222222222%	(TEST): 85.2000000000%
[EPOCH	51 ]	LOSS : (TRAIN) 0.1293967366	(TEST): 0.2527229786
		ACCURACY : (TRAIN) 87.5555555556%	(TEST): 71.6000000000%
[EPOCH	52 ]	LOSS : (TRAIN) 0.0572891831	(TEST): 0.0008219515
		ACCURACY : (TRAIN) 87.1111111111%	(TEST): 86.4000000000%
[EPOCH	53 ]	LOSS : (TRAIN) 0.0367459133	(TEST): 0.0027312939
		ACCURACY : (TRAIN) 87.6666666667%	(TEST): 82.0000000000%
[EPOCH	54 ]	LOSS : (TRAIN) 0.0628332123	(TEST): 0.0494256914
		ACCURACY : (TRAIN) 88.4444444444%	(TEST): 86.8000000000%
[EPOCH	55 ]	LOSS : (TRAIN) 0.1075602844	(TEST): 0.0240656994
		ACCURACY : (TRAIN) 88.2222222222%	(TEST): 85.6000000000%
[EPOCH	56 ]	LOSS : (TRAIN) 0.0105240261	(TEST): 0.0001007967
		ACCURACY : (TRAIN) 87.1111111111%	(TEST): 84.8000000000%
[EPOCH	57 ]	LOSS : (TRAIN) 0.1518491805	(TEST): 0.1231509298
		ACCURACY : (TRAIN) 87.4444444444%	(TEST): 84.8000000000%
[EPOCH	58 ]	LOSS : (TRAIN) 0.1144929528	(TEST): 0.1934154630



		ACCURACY : (TRAIN) 89.3333333333%	(TEST): 87.2000000000%
[EPOCH	59 ]	LOSS : (TRAIN) 0.0084016919	(TEST): 0.0752337500
		ACCURACY : (TRAIN) 87.2222222222%	(TEST): 84.0000000000%
[EPOCH	60 ]	LOSS : (TRAIN) 0.0839517415	(TEST): 0.0617179014
		ACCURACY : (TRAIN) 90.7777777778%	(TEST): 86.4000000000%
[EPOCH	61 ]	LOSS : (TRAIN) 0.0247109495	(TEST): 0.0721326396
		ACCURACY : (TRAIN) 88.7777777778%	(TEST): 86.8000000000%
[EPOCH	62 ]	LOSS : (TRAIN) 0.0551798232	(TEST): 0.1147199199
		ACCURACY : (TRAIN) 89.4444444444%	(TEST): 82.8000000000%
[EPOCH	63 ]	LOSS : (TRAIN) 0.0550542511	(TEST): 0.0954831541
		ACCURACY : (TRAIN) 88.8888888889%	(TEST): 84.4000000000%
[EPOCH	64 ]	LOSS : (TRAIN) 0.0348847583	(TEST): 0.1705477834
		ACCURACY : (TRAIN) 89.0000000000%	(TEST): 80.0000000000%
[EPOCH	65 ]	LOSS : (TRAIN) 0.1160974205	(TEST): 0.0382045433
		ACCURACY : (TRAIN) 88.6666666667%	(TEST): 87.6000000000%
[EPOCH	66 ]	LOSS : (TRAIN) 0.1075214893	(TEST): 0.3259455562
		ACCURACY : (TRAIN) 89.7777777778%	(TEST): 76.0000000000%
[EPOCH	67 ]	LOSS : (TRAIN) 0.0672963113	(TEST): 0.1297023892
		ACCURACY : (TRAIN) 87.5555555556%	(TEST): 82.8000000000%
[EPOCH	68 ]	LOSS : (TRAIN) 0.0695476681	(TEST): 0.1164123341
		ACCURACY : (TRAIN) 88.4444444444%	(TEST): 86.4000000000%
[EPOCH	69 ]	LOSS : (TRAIN) 0.0917822719	(TEST): 0.1482853293
		ACCURACY : (TRAIN) 89.0000000000%	(TEST): 84.8000000000%
[EPOCH	70 ]	LOSS : (TRAIN) 0.0133291455	(TEST): 0.0718310401
		ACCURACY : (TRAIN) 90.2222222222%	(TEST): 83.6000000000%
[EPOCH	71 ]	LOSS : (TRAIN) 0.0995956957	(TEST): 0.1194402575
		ACCURACY : (TRAIN) 88.1111111111%	(TEST): 84.8000000000%
[EPOCH	72 ]	LOSS : (TRAIN) 0.0288109295	(TEST): 0.0302637126
		ACCURACY : (TRAIN) 90.5555555556%	(TEST): 85.6000000000%
[EPOCH	73 ]	LOSS : (TRAIN) 0.0527779311	(TEST): 0.1338444650
		ACCURACY : (TRAIN) 90.3333333333%	(TEST): 86.0000000000%
[EPOCH	74 ]	LOSS : (TRAIN) 0.0229614396	(TEST): 0.0180826597

		ACCURACY : (TRAIN) 90.555555556%	(TEST): 86.0000000000%
[EPOCH	75 ]	LOSS : (TRAIN) 0.0369064957	(TEST): 0.0287452228
		ACCURACY : (TRAIN) 90.666666667%	(TEST): 84.8000000000%
[EPOCH	76 ]	LOSS : (TRAIN) 0.0302421339	(TEST): 0.0308970548
		ACCURACY : (TRAIN) 90.333333333%	(TEST): 80.4000000000%
[EPOCH	77 ]	LOSS : (TRAIN) 0.0536190085	(TEST): 0.0378732942
		ACCURACY : (TRAIN) 89.666666667%	(TEST): 86.4000000000%
[EPOCH	78 ]	LOSS : (TRAIN) 0.0417512991	(TEST): 0.0408754349
		ACCURACY : (TRAIN) 90.666666667%	(TEST): 84.0000000000%
[EPOCH	79 ]	LOSS : (TRAIN) 0.0448198803	(TEST): 0.0310317148
		ACCURACY : (TRAIN) 90.333333333%	(TEST): 86.0000000000%
[EPOCH	80 ]	LOSS : (TRAIN) 0.0873731524	(TEST): 0.0619093478
		ACCURACY : (TRAIN) 91.222222222%	(TEST): 82.8000000000%
[EPOCH	81 ]	LOSS : (TRAIN) 0.0353192016	(TEST): 0.1316240281
		ACCURACY : (TRAIN) 91.0000000000%	(TEST): 86.0000000000%
[EPOCH	82 ]	LOSS : (TRAIN) 0.0369498655	(TEST): 0.0601974316
		ACCURACY : (TRAIN) 92.222222222%	(TEST): 86.0000000000%
[EPOCH	83 ]	LOSS : (TRAIN) 0.0710086226	(TEST): 0.1034784392
		ACCURACY : (TRAIN) 90.666666667%	(TEST): 86.4000000000%
[EPOCH	84 ]	LOSS : (TRAIN) 0.0724884570	(TEST): 0.0293114018
		ACCURACY : (TRAIN) 91.0000000000%	(TEST): 86.0000000000%
[EPOCH	85 ]	LOSS : (TRAIN) 0.0472045429	(TEST): 0.0215565134
		ACCURACY : (TRAIN) 91.777777778%	(TEST): 86.0000000000%
[EPOCH	86 ]	LOSS : (TRAIN) 0.0371654704	(TEST): 0.0429020561
		ACCURACY : (TRAIN) 91.222222222%	(TEST): 86.4000000000%
[EPOCH	87 ]	LOSS : (TRAIN) 0.0482804365	(TEST): 0.0716337562
		ACCURACY : (TRAIN) 91.0000000000%	(TEST): 86.8000000000%
[EPOCH	88 ]	LOSS : (TRAIN) 0.0612736642	(TEST): 0.0144867953
		ACCURACY : (TRAIN) 91.0000000000%	(TEST): 86.4000000000%
[EPOCH	89 ]	LOSS : (TRAIN) 0.0810004100	(TEST): 0.0416279621
		ACCURACY : (TRAIN) 91.0000000000%	(TEST): 86.0000000000%
[EPOCH	90 ]	LOSS : (TRAIN) 0.0704543069	(TEST): 0.0594339892

		ACCURACY : (TRAIN) 90.7777777778%	(TEST): 86.0000000000%
[EPOCH	91 ]	LOSS : (TRAIN) 0.0577649362	(TEST): 0.0337428562
		ACCURACY : (TRAIN) 92.0000000000%	(TEST): 85.2000000000%
[EPOCH	92 ]	LOSS : (TRAIN) 0.0663521439	(TEST): 0.0569543764
		ACCURACY : (TRAIN) 90.7777777778%	(TEST): 86.4000000000%
[EPOCH	93 ]	LOSS : (TRAIN) 0.0870496929	(TEST): 0.1062039062
		ACCURACY : (TRAIN) 90.1111111111%	(TEST): 81.2000000000%
[EPOCH	94 ]	LOSS : (TRAIN) 0.0474336371	(TEST): 0.0675864220
		ACCURACY : (TRAIN) 91.1111111111%	(TEST): 86.0000000000%
[EPOCH	95 ]	LOSS : (TRAIN) 0.0492111146	(TEST): 0.0489912480
		ACCURACY : (TRAIN) 91.2222222222%	(TEST): 85.6000000000%
[EPOCH	96 ]	LOSS : (TRAIN) 0.0523931384	(TEST): 0.1434294730
		ACCURACY : (TRAIN) 91.6666666667%	(TEST): 82.4000000000%
[EPOCH	97 ]	LOSS : (TRAIN) 0.0822136998	(TEST): 0.0775748491
		ACCURACY : (TRAIN) 90.8888888889%	(TEST): 87.2000000000%
[EPOCH	98 ]	LOSS : (TRAIN) 0.0353695638	(TEST): 0.0502970703
		ACCURACY : (TRAIN) 91.7777777778%	(TEST): 84.8000000000%
[EPOCH	99 ]	LOSS : (TRAIN) 0.0478905588	(TEST): 0.0423120297
		ACCURACY : (TRAIN) 91.5555555556%	(TEST): 86.8000000000%
[EPOCH	100 ]	LOSS : (TRAIN) 0.0525768138	(TEST): 0.0886075199
		ACCURACY : (TRAIN) 90.3333333333%	(TEST): 87.2000000000%
[EPOCH	101 ]	LOSS : (TRAIN) 0.0512108617	(TEST): 0.0998589620
		ACCURACY : (TRAIN) 90.2222222222%	(TEST): 86.0000000000%
[EPOCH	102 ]	LOSS : (TRAIN) 0.0699735284	(TEST): 0.0772321075
		ACCURACY : (TRAIN) 91.4444444444%	(TEST): 87.2000000000%
[EPOCH	103 ]	LOSS : (TRAIN) 0.0683528855	(TEST): 0.0675860792
		ACCURACY : (TRAIN) 90.8888888889%	(TEST): 86.4000000000%
[EPOCH	104 ]	LOSS : (TRAIN) 0.0287633762	(TEST): 0.0543810278
		ACCURACY : (TRAIN) 91.8888888889%	(TEST): 87.2000000000%
[EPOCH	105 ]	LOSS : (TRAIN) 0.0837516859	(TEST): 0.0408375114
		ACCURACY : (TRAIN) 90.3333333333%	(TEST): 87.6000000000%
[EPOCH	106 ]	LOSS : (TRAIN) 0.0790963024	(TEST): 0.0680795461

	ACCURACY : (TRAIN)	91.7777777778%	(TEST): 87.2000000000%
[EPOCH 107 ]	LOSS : (TRAIN)	0.0713751391	(TEST): 0.0419438407
	ACCURACY : (TRAIN)	91.1111111111%	(TEST): 87.2000000000%
[EPOCH 108 ]	LOSS : (TRAIN)	0.0514049567	(TEST): 0.0256325249
	ACCURACY : (TRAIN)	91.3333333333%	(TEST): 87.2000000000%
[EPOCH 109 ]	LOSS : (TRAIN)	0.0806162804	(TEST): 0.0650787503
	ACCURACY : (TRAIN)	91.1111111111%	(TEST): 86.8000000000%
[EPOCH 110 ]	LOSS : (TRAIN)	0.0530632548	(TEST): 0.0478601977
	ACCURACY : (TRAIN)	90.8888888889%	(TEST): 88.4000000000%
[EPOCH 111 ]	LOSS : (TRAIN)	0.0772599727	(TEST): 0.0342754796
	ACCURACY : (TRAIN)	91.2222222222%	(TEST): 87.2000000000%
[EPOCH 112 ]	LOSS : (TRAIN)	0.0673160926	(TEST): 0.0590111613
	ACCURACY : (TRAIN)	90.6666666667%	(TEST): 88.0000000000%
[EPOCH 113 ]	LOSS : (TRAIN)	0.0529230051	(TEST): 0.0232043155
	ACCURACY : (TRAIN)	90.7777777778%	(TEST): 88.0000000000%
[EPOCH 114 ]	LOSS : (TRAIN)	0.0569535457	(TEST): 0.0534576476
	ACCURACY : (TRAIN)	91.1111111111%	(TEST): 87.2000000000%
[EPOCH 115 ]	LOSS : (TRAIN)	0.0561788306	(TEST): 0.0206116345
	ACCURACY : (TRAIN)	91.0000000000%	(TEST): 88.0000000000%
[EPOCH 116 ]	LOSS : (TRAIN)	0.0535793230	(TEST): 0.0849098042
	ACCURACY : (TRAIN)	91.3333333333%	(TEST): 85.2000000000%
[EPOCH 117 ]	LOSS : (TRAIN)	0.0522100180	(TEST): 0.0675779283
	ACCURACY : (TRAIN)	90.2222222222%	(TEST): 86.4000000000%
[EPOCH 118 ]	LOSS : (TRAIN)	0.0703808591	(TEST): 0.0606353357
	ACCURACY : (TRAIN)	91.4444444444%	(TEST): 87.6000000000%
[EPOCH 119 ]	LOSS : (TRAIN)	0.0679025352	(TEST): 0.0468985923
	ACCURACY : (TRAIN)	92.0000000000%	(TEST): 88.4000000000%
[EPOCH 120 ]	LOSS : (TRAIN)	0.0556007624	(TEST): 0.0293764975
	ACCURACY : (TRAIN)	90.3333333333%	(TEST): 88.4000000000%
[EPOCH 121 ]	LOSS : (TRAIN)	0.0551166721	(TEST): 0.0565171167
	ACCURACY : (TRAIN)	90.2222222222%	(TEST): 85.6000000000%
[EPOCH 122 ]	LOSS : (TRAIN)	0.0524467267	(TEST): 0.0683104247

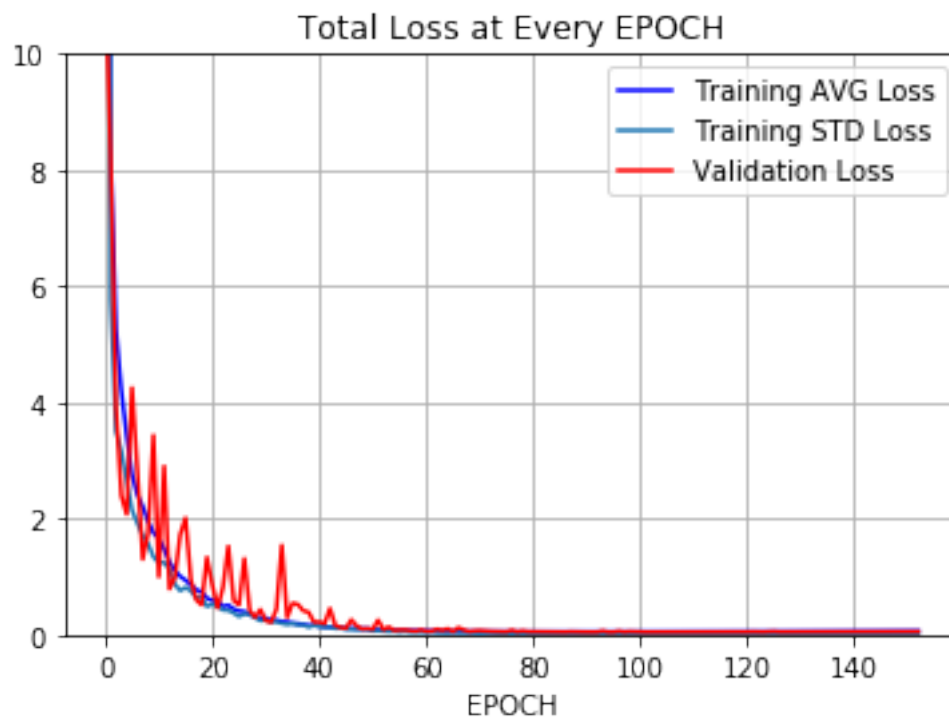
	ACCURACY : (TRAIN)	90.3333333333%	(TEST): 88.4000000000%
[EPOCH 123 ]	LOSS : (TRAIN)	0.0731893182	(TEST): 0.0513792261
	ACCURACY : (TRAIN)	90.2222222222%	(TEST): 88.4000000000%
[EPOCH 124 ]	LOSS : (TRAIN)	0.0693496093	(TEST): 0.0238928609
	ACCURACY : (TRAIN)	90.7777777778%	(TEST): 86.0000000000%
[EPOCH 125 ]	LOSS : (TRAIN)	0.0759429559	(TEST): 0.0245910380
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 82.8000000000%
[EPOCH 126 ]	LOSS : (TRAIN)	0.0474294089	(TEST): 0.0348118916
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 88.4000000000%
[EPOCH 127 ]	LOSS : (TRAIN)	0.0809649602	(TEST): 0.0517672747
	ACCURACY : (TRAIN)	90.3333333333%	(TEST): 88.4000000000%
[EPOCH 128 ]	LOSS : (TRAIN)	0.0469507128	(TEST): 0.0762151331
	ACCURACY : (TRAIN)	90.7777777778%	(TEST): 87.6000000000%
[EPOCH 129 ]	LOSS : (TRAIN)	0.1045569777	(TEST): 0.0836037397
	ACCURACY : (TRAIN)	90.7777777778%	(TEST): 87.6000000000%
[EPOCH 130 ]	LOSS : (TRAIN)	0.0512563735	(TEST): 0.0387681983
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 88.4000000000%
[EPOCH 131 ]	LOSS : (TRAIN)	0.0550930314	(TEST): 0.0327292122
	ACCURACY : (TRAIN)	89.6666666667%	(TEST): 86.0000000000%
[EPOCH 132 ]	LOSS : (TRAIN)	0.0881444886	(TEST): 0.0514562465
	ACCURACY : (TRAIN)	89.7777777778%	(TEST): 88.4000000000%
[EPOCH 133 ]	LOSS : (TRAIN)	0.0706293508	(TEST): 0.0803722143
	ACCURACY : (TRAIN)	90.2222222222%	(TEST): 86.4000000000%
[EPOCH 134 ]	LOSS : (TRAIN)	0.0648734644	(TEST): 0.0869651586
	ACCURACY : (TRAIN)	90.0000000000%	(TEST): 88.8000000000%
[EPOCH 135 ]	LOSS : (TRAIN)	0.0572009310	(TEST): 0.0617421195
	ACCURACY : (TRAIN)	89.3333333333%	(TEST): 87.6000000000%
[EPOCH 136 ]	LOSS : (TRAIN)	0.0831570774	(TEST): 0.1037914082
	ACCURACY : (TRAIN)	89.7777777778%	(TEST): 84.8000000000%
[EPOCH 137 ]	LOSS : (TRAIN)	0.0729062930	(TEST): 0.0657114387
	ACCURACY : (TRAIN)	89.5555555556%	(TEST): 89.6000000000%
[EPOCH 138 ]	LOSS : (TRAIN)	0.0573560633	(TEST): 0.0374204591

	ACCURACY : (TRAIN)	89.7777777778%	(TEST): 88.4000000000%
[EPOCH 139 ]	LOSS : (TRAIN)	0.0557487570	(TEST): 0.1205469370
	ACCURACY : (TRAIN)	89.5555555556%	(TEST): 88.4000000000%
[EPOCH 140 ]	LOSS : (TRAIN)	0.0651108697	(TEST): 0.0366722085
	ACCURACY : (TRAIN)	89.7777777778%	(TEST): 86.4000000000%
[EPOCH 141 ]	LOSS : (TRAIN)	0.0801285878	(TEST): 0.0460619181
	ACCURACY : (TRAIN)	89.4444444444%	(TEST): 89.6000000000%
[EPOCH 142 ]	LOSS : (TRAIN)	0.0755276754	(TEST): 0.0410710946
	ACCURACY : (TRAIN)	89.2222222222%	(TEST): 87.6000000000%
[EPOCH 143 ]	LOSS : (TRAIN)	0.0729495138	(TEST): 0.0571087040
	ACCURACY : (TRAIN)	89.1111111111%	(TEST): 88.4000000000%
[EPOCH 144 ]	LOSS : (TRAIN)	0.0621790513	(TEST): 0.0615073480
	ACCURACY : (TRAIN)	89.1111111111%	(TEST): 88.8000000000%
[EPOCH 145 ]	LOSS : (TRAIN)	0.0759018660	(TEST): 0.0736735463
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 87.2000000000%
[EPOCH 146 ]	LOSS : (TRAIN)	0.0521004274	(TEST): 0.0522844791
	ACCURACY : (TRAIN)	89.1111111111%	(TEST): 85.2000000000%
[EPOCH 147 ]	LOSS : (TRAIN)	0.0572534986	(TEST): 0.0634582192
	ACCURACY : (TRAIN)	89.6666666667%	(TEST): 88.8000000000%
[EPOCH 148 ]	LOSS : (TRAIN)	0.0663139075	(TEST): 0.0413329266
	ACCURACY : (TRAIN)	90.0000000000%	(TEST): 89.2000000000%
[EPOCH 149 ]	LOSS : (TRAIN)	0.0503552705	(TEST): 0.0347821116
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 85.6000000000%
[EPOCH 150 ]	LOSS : (TRAIN)	0.0842194408	(TEST): 0.0548142642
	ACCURACY : (TRAIN)	88.7777777778%	(TEST): 88.8000000000%
[EPOCH 151 ]	LOSS : (TRAIN)	0.0597380549	(TEST): 0.0412292555
	ACCURACY : (TRAIN)	88.3333333333%	(TEST): 90.0000000000%
[EPOCH 152 ]	LOSS : (TRAIN)	0.0706750378	(TEST): 0.0390633158
	ACCURACY : (TRAIN)	90.1111111111%	(TEST): 90.0000000000%

## 1.1 Plot the Total loss train and test

```
[358]: plt.plot(loss_train_mean,color="b",label='Training AVG Loss')
plt.plot(loss_train_std,label='Training STD Loss')
plt.plot(loss_test_mean,color="r",label='Validation Loss')
plt.legend(loc='upper right')
plt.ylim([0,10])
plt.grid()
plt.title("Total Loss at Every EPOCH")
plt.xlabel("EPOCH")
```

```
[358]: Text(0.5, 0, 'EPOCH')
```



## 1.2 [Learning Curve] : Training Loss with Standard Deviation.

```
[359]: from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import learning_curve
from sklearn.model_selection import ShuffleSplit
from sklearn.svm import SVC
from sklearn.datasets import load_digits

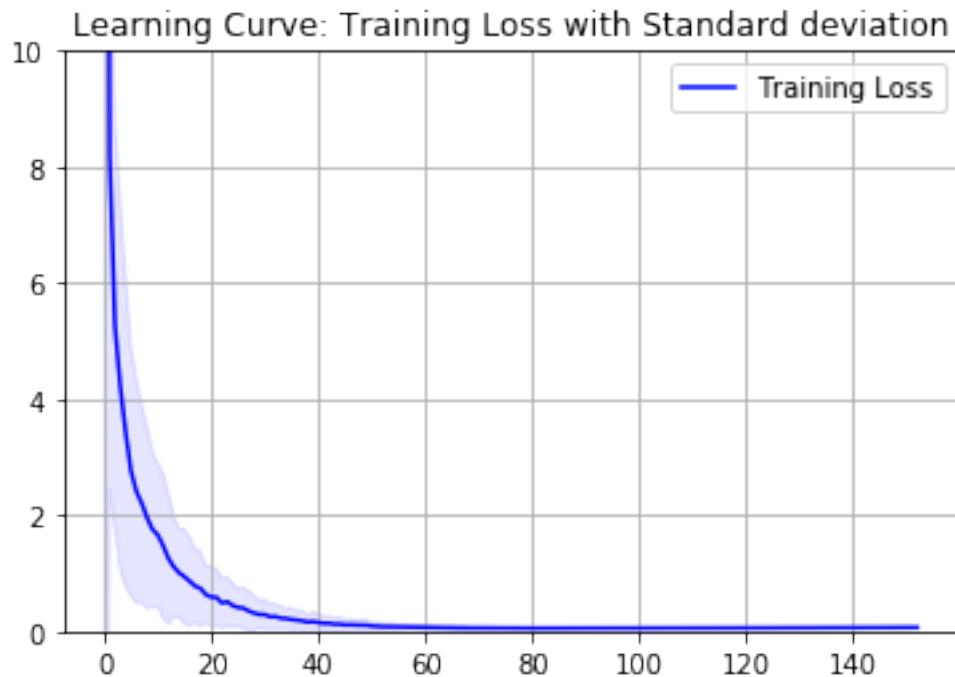
train_sizes=np.array(range(epochs))
```

```

plt.title("Learning Curve: Training Loss with Standard deviation")
plt.grid()
plt.ylim([0,10])
plt.fill_between(train_sizes,loss_train_mean - loss_train_std, loss_train_mean_
↳ + loss_train_std, alpha=0.1,color="b")
plt.plot(train_sizes, loss_train_mean, color="b",label="Training Loss")
plt.legend(loc='upper right')

plt.show()

```



### 1.3 [Learning Curve] : Train and test Accuracy with Standard Deviation.

```

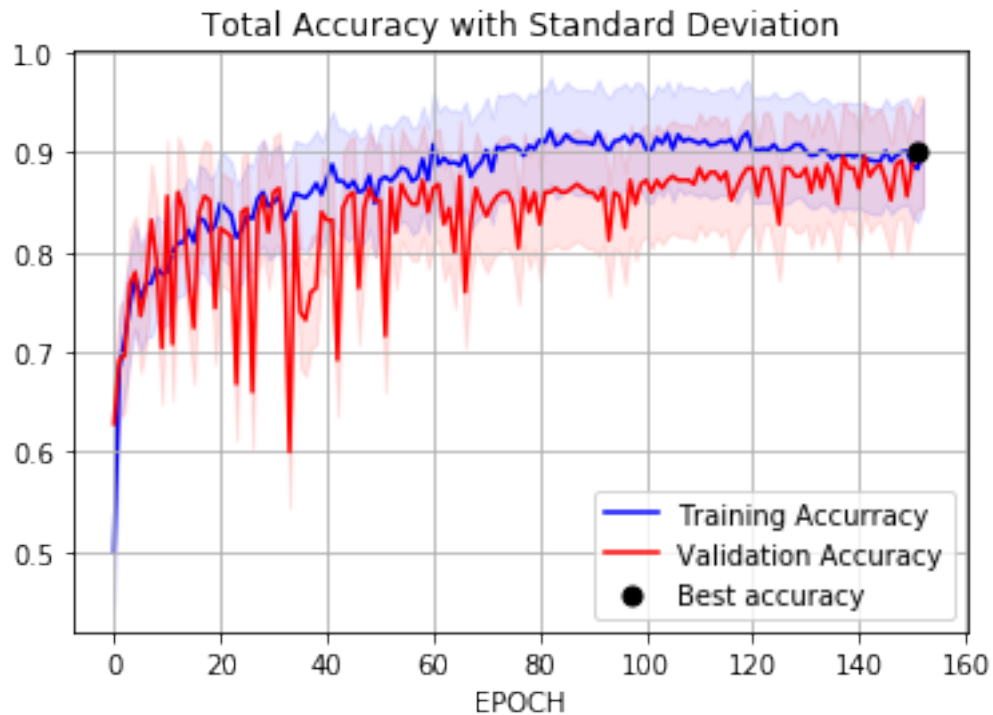
[360]: max_index=np.argmax(accuracy_test)
accuracy_train_std=np.std(accuracy_train)
accuracy_test_std=np.std(accuracy_test)
plt.fill_between(train_sizes,accuracy_train -
↳ accuracy_train_std,accuracy_train + accuracy_train_std, alpha=0.
↳ 1,color="b")
plt.fill_between(train_sizes,accuracy_test - accuracy_test_std,accuracy_test
↳ + accuracy_test_std, alpha=0.1,color="r")
plt.plot(accuracy_train, color="b",zorder=1,label="Training Accuracy")
plt.plot(accuracy_test,color="r",zorder=2,label='Validation Accuracy')

```



```
plt.
    ↳scatter(max_index,accuracy_test[max_index],c='black',s=50,zorder=10,label='Best_
    ↳accuracy')
plt.legend(loc='lower right')
plt.grid()
plt.title("Total Accuracy with Standard Deviation")
plt.xlabel("EPOCH")
```

[360]: Text(0.5, 0, 'EPOCH')



#### 1.4 Visualize Table : Present the final loss and accuracy at convergence

```
[361]: max_index=np.argmax(accuracy_test)
print("""
```

```

                <At convergence>
+-----+-----+-----+
|      At convergence      |      Loss      |      Accuracy      |
+-----+-----+-----+
|      Training            |      %6.3f      |      %6.2f %%      |
+-----+-----+-----+
|      Validation          |      %6.3f      |      %6.2f %%      |
+-----+-----+-----+
```

```
"""
```

```
%(loss_train_mean[epochs-1],accuracy_train[epochs-1]*100,loss_test_mean[epochs-1],accuracy_test[epochs-1])
```

<At convergence>

At convergence	Loss	Accuracy
Training	0.067	90.11 %
Validation	0.056	90.00 %

```
[ ]:
```