In [ ]:
```python
import numpy as np
import torch as torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime
import random
from scipy.stats import norm

import os
import pathlib

from Model import *
from utils import *

torch.autograd.set_detect_anomaly(True)
start_time=datetime.now().strftime('%B %d - %H:%M:%S')
```

In [ ]:
```python
#Global parameters
GlobalParams1=Params(param_type='k1',target_type='indicator',trick='clamp',loss_type='BCELoss',delta=0.01,w=1,lr=0.0001)
dB1 = SampleBMIncr(GlobalParams=GlobalParams1)
init_x1 =  Sample_Init(GlobalParams=GlobalParams1)
init_c1= torch.zeros_like(init_x1)

GlobalParams2=Params(param_type='k2',target_type='indicator',trick='clamp',loss_type='BCELoss',delta=0.01,w=1,lr=0.0001)
dB2 = SampleBMIncr(GlobalParams=GlobalParams2)   ## TODO: same dB?????
init_x2 =  Sample_Init(GlobalParams=GlobalParams2)
init_c2= torch.zeros_like(init_x2)

NT1=GlobalParams1.NT1
NT2=GlobalParams1.NT2
dt=GlobalParams1.dt
device=GlobalParams1.device
learning_rate = GlobalParams1.lr

#Forward Loss
forward_losses = []

#How many batches
MaxBatch= 750

#How many optimization steps per batch
OptimSteps= 25

#Train on a single batch?
single_batch = True
```

```
#Set up main models for y0 and z (z will be list of models)
v0_model_main1 = Network(scaler_type='sigmoid')
u0_model_main1 = Network(scaler_type='sigmoid')
y0_model_main1 = Network(scaler_type='sigmoid')

zv_models_main1 = [Network() for i in range(NT1)]
zu_models_main1 = [Network() for i in range(NT1)]
zy_models_main1 = [Network() for i in range(NT2)]
main_models1=Main_Models(GlobalParams=GlobalParams1)
main_models1.create(v0_model=v0_model_main1,
                    u0_model=u0_model_main1,
                    y0_model=y0_model_main1,
                    zv_models=zv_models_main1,
                    zu_models=zu_models_main1,
                    zy_models=zy_models_main1,
                    forward_loss=forward_losses,
                    dB=dB1,
                    init_x=init_x1,
                    init_c=init_c1)

v0_model_main2 = Network(scaler_type='sigmoid')
u0_model_main2 = Network(scaler_type='sigmoid')
y0_model_main2 = Network(scaler_type='sigmoid')

zv_models_main2 = [Network() for i in range(NT1)]
zu_models_main2 = [Network() for i in range(NT1)]
zy_models_main2 = [Network() for i in range(NT2)]
main_models2=Main_Models(GlobalParams=GlobalParams2)
main_models2.create(v0_model=v0_model_main2,
                    u0_model=u0_model_main2,
                    y0_model=y0_model_main2,
                    zv_models=zv_models_main2,
                    zu_models=zu_models_main2,
                    zy_models=zy_models_main2,
                    forward_loss=forward_losses,
                    dB=dB2,
                    init_x=init_x2,
                    init_c=init_c2)

pop1_dict={'dB':dB1,
           'init_x':init_x1 ,
           'init_c':init_c1 ,
           'GlobalParams':GlobalParams1,
           'main_models':main_models1}

pop2_dict={'dB':dB2,
           'init_x':init_x2 ,
           'init_c':init_c2 ,
           'GlobalParams':GlobalParams2,
           'main_models':main_models2}
```

In [ ]:

```
#Define optimization parameters
params=[]
params = list(main_models1.v0_model.parameters())+\
         list(main_models1.u0_model.parameters())+\
         list(main_models1.y0_model.parameters())+\
         list(main_models2.v0_model.parameters())+\
         list(main_models2.u0_model.parameters())+\
         list(main_models2.y0_model.parameters())
for i in range(NT1):
    params += list(main_models1.zv_models[i].parameters())
    params += list(main_models1.zu_models[i].parameters())
    params += list(main_models2.zv_models[i].parameters())
    params += list(main_models2.zu_models[i].parameters())

for i in range(NT2):
    params += list(main_models1.zy_models[i].parameters())
    params += list(main_models2.zy_models[i].parameters())

#Set up optimizer and scheduler
optimizer = optim.Adamax(params, lr=learning_rate)
scheduler = torch.optim.lr_scheduler.StepLR(optimizer, step_size=100, gamma=0.95)

for k in range(0,MaxBatch):

    print("Batch Number: ", k+1)
    sloss=0
    #optimize main network wrt the foward loss
    for l in range(0,OptimSteps):
        optimizer.zero_grad()
        loss = get_foward_loss(pop1_dict=pop1_dict, pop2_dict=pop2_dict)
        loss.backward()
        # torch.nn.utils.clip_grad_norm_(parameters=params,max_norm=0.7)
        optimizer.step()
        scheduler.step()
        nloss = loss.detach().numpy()
        sloss += nloss
        # print('OptimStep: '+ str(l+1))
        # print('forward_loss: ' + str(nloss))
    avgloss = sloss/OptimSteps
    print("Average Error Est: ", avgloss)
    forward_losses.append(avgloss)

    #Generate a new batch if using multiple batches
    if(not single_batch):

        dB1 = SampleBMIncr(GlobalParams=GlobalParams1)
        init_x1 =  Sample_Init(GlobalParams=GlobalParams1)
        init_c1= torch.zeros_like(init_x1)
        pop1_dict={'dB':dB1,
                'init_x':init_x1 ,
                'init_c':init_c1 ,
```

```
            'GlobalParams':GlobalParams1,
            'main_models':main_models1}

    dB2 = SampleBMIncr(GlobalParams=GlobalParams2)   ## TODO: same dB?????
    init_x2 =  Sample_Init(GlobalParams=GlobalParams2)
    init_c2= torch.zeros_like(init_x2)
    pop2_dict={'dB':dB2,
            'init_x':init_x2 ,
            'init_c':init_c2 ,
            'GlobalParams':GlobalParams2,
            'main_models':main_models2}
```

```
Batch Number:  511
Average Error Est:  1.6824539709091186
Batch Number:  512
Average Error Est:  1.6865029382705687
Batch Number:  513
Average Error Est:  1.684301724433899
Batch Number:  514
Average Error Est:  1.6838127660751343
Batch Number:  515
Average Error Est:  1.6852887630462647
Batch Number:  516
Average Error Est:  1.6869824409484864
Batch Number:  517
Average Error Est:  1.685720772743225
Batch Number:  518
Average Error Est:  1.683901653289795
Batch Number:  519
Average Error Est:  1.6836091184616089
Batch Number:  520
Average Error Est:  1.6861657094955445
Batch Number:  521
Average Error Est:  1.6839993810653686
Batch Number:  522
Average Error Est:  1.6859577989578247
Batch Number:  523
Average Error Est:  1.6876707649230958
Batch Number:  524
Average Error Est:  1.68301260471344
Batch Number:  525
Average Error Est:  1.6881186294555663
Batch Number:  526
Average Error Est:  1.6858909606933594
Batch Number:  527
Average Error Est:  1.6855929899215698
Batch Number:  528
Average Error Est:  1.6834970760345458
Batch Number:  529
Average Error Est:  1.6862510299682618
Batch Number:  530
Average Error Est:  1.6863108444213868
Batch Number:  531
Average Error Est:  1.6869402360916137
Batch Number:  532
Average Error Est:  1.6851856184005738
Batch Number:  533
Average Error Est:  1.6861200046539306
Batch Number:  534
Average Error Est:  1.6854363012313842
Batch Number:  535
Average Error Est:  1.6847105503082276
Batch Number:  536
```

```
Average Error Est:  1.688040804862976
Batch Number:  537
Average Error Est:  1.6848394203186035
Batch Number:  538
Average Error Est:  1.6905893564224244
Batch Number:  539
Average Error Est:  1.6880903100967408
Batch Number:  540
Average Error Est:  1.687672996520996
Batch Number:  541
Average Error Est:  1.6845133590698242
Batch Number:  542
Average Error Est:  1.6838258123397827
Batch Number:  543
Average Error Est:  1.6850829601287842
Batch Number:  544
Average Error Est:  1.6830358219146728
Batch Number:  545
Average Error Est:  1.6877766895294188
Batch Number:  546
Average Error Est:  1.6819846439361572
Batch Number:  547
Average Error Est:  1.6824767303466797
Batch Number:  548
Average Error Est:  1.6843154907226563
Batch Number:  549
Average Error Est:  1.686055154800415
Batch Number:  550
Average Error Est:  1.6807348537445068
Batch Number:  551
Average Error Est:  1.6849817371368407
Batch Number:  552
Average Error Est:  1.6858178520202636
Batch Number:  553
Average Error Est:  1.6840467929840088
Batch Number:  554
Average Error Est:  1.6850014543533325
Batch Number:  555
Average Error Est:  1.6862866020202636
Batch Number:  556
Average Error Est:  1.6829587316513062
Batch Number:  557
Average Error Est:  1.685247507095337
Batch Number:  558
Average Error Est:  1.685205111503601
Batch Number:  559
Average Error Est:  1.6863167142868043
Batch Number:  560
Average Error Est:  1.6792222452163696
Batch Number:  561
Average Error Est:  1.6849729442596435
```

```
Batch Number:  562
Average Error Est:  1.6844683408737182
Batch Number:  563
Average Error Est:  1.6848991107940674
Batch Number:  564
Average Error Est:  1.6826937770843506
Batch Number:  565
Average Error Est:  1.6852263975143433
Batch Number:  566
Average Error Est:  1.6875631046295165
Batch Number:  567
Average Error Est:  1.6825312852859498
Batch Number:  568
Average Error Est:  1.6866102266311644
Batch Number:  569
Average Error Est:  1.6836942911148072
Batch Number:  570
Average Error Est:  1.6856303882598878
Batch Number:  571
Average Error Est:  1.6851705598831177
Batch Number:  572
Average Error Est:  1.683547887802124
Batch Number:  573
Average Error Est:  1.6831574487686156
Batch Number:  574
Average Error Est:  1.6853477668762207
Batch Number:  575
Average Error Est:  1.6859348106384278
Batch Number:  576
Average Error Est:  1.6865145921707154
Batch Number:  577
Average Error Est:  1.687988429069519
Batch Number:  578
Average Error Est:  1.6861838483810425
Batch Number:  579
Average Error Est:  1.6853610801696777
Batch Number:  580
Average Error Est:  1.6885153579711913
Batch Number:  581
Average Error Est:  1.686638717651367
Batch Number:  582
Average Error Est:  1.6861371278762818
Batch Number:  583
Average Error Est:  1.6896436023712158
Batch Number:  584
Average Error Est:  1.6871112537384034
Batch Number:  585
Average Error Est:  1.6829937982559204
Batch Number:  586
Average Error Est:  1.6810900783538818
Batch Number:  587
```

```
Average Error Est:  1.6829934453964233
Batch Number:  588
Average Error Est:  1.6852147817611693
Batch Number:  589
Average Error Est:  1.6844466543197631
Batch Number:  590
Average Error Est:  1.683593430519104
Batch Number:  591
Average Error Est:  1.6839492893218995
Batch Number:  592
Average Error Est:  1.6814739608764648
Batch Number:  593
Average Error Est:  1.6864438915252686
Batch Number:  594
Average Error Est:  1.6801946020126344
Batch Number:  595
Average Error Est:  1.6818813848495484
Batch Number:  596
Average Error Est:  1.6812762880325318
Batch Number:  597
Average Error Est:  1.6839154386520385
Batch Number:  598
Average Error Est:  1.6851806926727295
Batch Number:  599
Average Error Est:  1.682796883583069
Batch Number:  600
Average Error Est:  1.6848869514465332
Batch Number:  601
Average Error Est:  1.685491828918457
Batch Number:  602
Average Error Est:  1.6862303256988525
Batch Number:  603
Average Error Est:  1.6767702913284301
Batch Number:  604
Average Error Est:  1.6828896951675416
Batch Number:  605
Average Error Est:  1.6865469121932983
Batch Number:  606
Average Error Est:  1.6831800842285156
Batch Number:  607
Average Error Est:  1.6846273422241211
Batch Number:  608
Average Error Est:  1.685716052055359
Batch Number:  609
Average Error Est:  1.6838742208480835
Batch Number:  610
Average Error Est:  1.6888931846618653
Batch Number:  611
Average Error Est:  1.6852046060562134
Batch Number:  612
Average Error Est:  1.688194055557251
```

```
Batch Number:  613
Average Error Est:  1.690322813987732
Batch Number:  614
Average Error Est:  1.686228675842285
Batch Number:  615
Average Error Est:  1.6872141456604004
Batch Number:  616
Average Error Est:  1.6870223522186278
Batch Number:  617
Average Error Est:  1.6827640438079834
Batch Number:  618
Average Error Est:  1.6850918436050415
Batch Number:  619
Average Error Est:  1.6828772306442261
Batch Number:  620
Average Error Est:  1.688934144973755
Batch Number:  621
Average Error Est:  1.6815928649902343
Batch Number:  622
Average Error Est:  1.682413902282715
Batch Number:  623
Average Error Est:  1.6860722732543945
Batch Number:  624
Average Error Est:  1.6873084163665772
Batch Number:  625
Average Error Est:  1.684215989112854
Batch Number:  626
Average Error Est:  1.6833538866043092
Batch Number:  627
Average Error Est:  1.6812486410140992
Batch Number:  628
Average Error Est:  1.6839825057983397
Batch Number:  629
Average Error Est:  1.681229591369629
Batch Number:  630
Average Error Est:  1.6797702169418336
Batch Number:  631
Average Error Est:  1.6853371858596802
Batch Number:  632
Average Error Est:  1.679506883621216
Batch Number:  633
Average Error Est:  1.6794397926330566
Batch Number:  634
Average Error Est:  1.6761370277404786
Batch Number:  635
Average Error Est:  1.674396014213562
Batch Number:  636
Average Error Est:  1.6776970100402833
Batch Number:  637
Average Error Est:  1.676101622581482
Batch Number:  638
```

```
Average Error Est:  1.681574397087097
Batch Number:  639
Average Error Est:  1.6819225454330444
Batch Number:  640
Average Error Est:  1.6818053436279297
Batch Number:  641
Average Error Est:  1.6825605821609497
Batch Number:  642
Average Error Est:  1.6782325506210327
Batch Number:  643
Average Error Est:  1.6864886951446534
Batch Number:  644
Average Error Est:  1.6818683195114135
Batch Number:  645
Average Error Est:  1.6851281547546386
Batch Number:  646
Average Error Est:  1.6851215553283692
Batch Number:  647
Average Error Est:  1.687384376525879
Batch Number:  648
Average Error Est:  1.6805355310440064
Batch Number:  649
Average Error Est:  1.6880376958847045
Batch Number:  650
Average Error Est:  1.6786086940765381
Batch Number:  651
Average Error Est:  1.6902610778808593
Batch Number:  652
Average Error Est:  1.6886256551742553
Batch Number:  653
Average Error Est:  1.6848976039886474
Batch Number:  654
Average Error Est:  1.6857871532440185
Batch Number:  655
Average Error Est:  1.6815054273605348
Batch Number:  656
Average Error Est:  1.6894613885879517
Batch Number:  657
Average Error Est:  1.6843632030487061
Batch Number:  658
Average Error Est:  1.6845126485824584
Batch Number:  659
Average Error Est:  1.685157494544983
Batch Number:  660
Average Error Est:  1.6840274810791016
Batch Number:  661
Average Error Est:  1.6913042497634887
Batch Number:  662
Average Error Est:  1.684259819984436
Batch Number:  663
Average Error Est:  1.6896032285690308
```

```
Batch Number:  664
Average Error Est:  1.6816156911849975
Batch Number:  665
Average Error Est:  1.6872336673736572
Batch Number:  666
Average Error Est:  1.682069149017334
Batch Number:  667
Average Error Est:  1.6856876087188721
Batch Number:  668
Average Error Est:  1.6882993698120117
Batch Number:  669
Average Error Est:  1.6858474493026734
Batch Number:  670
Average Error Est:  1.6833963775634766
Batch Number:  671
Average Error Est:  1.6908513498306275
Batch Number:  672
Average Error Est:  1.6809662008285522
Batch Number:  673
Average Error Est:  1.6843592166900634
Batch Number:  674
Average Error Est:  1.6780561447143554
Batch Number:  675
Average Error Est:  1.6812781953811646
Batch Number:  676
Average Error Est:  1.681375036239624
Batch Number:  677
Average Error Est:  1.6808476114273072
Batch Number:  678
Average Error Est:  1.6797665166854858
Batch Number:  679
Average Error Est:  1.6836106491088867
Batch Number:  680
Average Error Est:  1.6845550298690797
Batch Number:  681
Average Error Est:  1.6731566429138183
Batch Number:  682
Average Error Est:  1.679128737449646
Batch Number:  683
Average Error Est:  1.675900831222534
Batch Number:  684
Average Error Est:  1.6745230770111084
Batch Number:  685
Average Error Est:  1.6719684648513793
Batch Number:  686
Average Error Est:  1.6771343660354614
Batch Number:  687
Average Error Est:  1.6738365507125854
Batch Number:  688
Average Error Est:  1.6759105110168457
Batch Number:  689
```

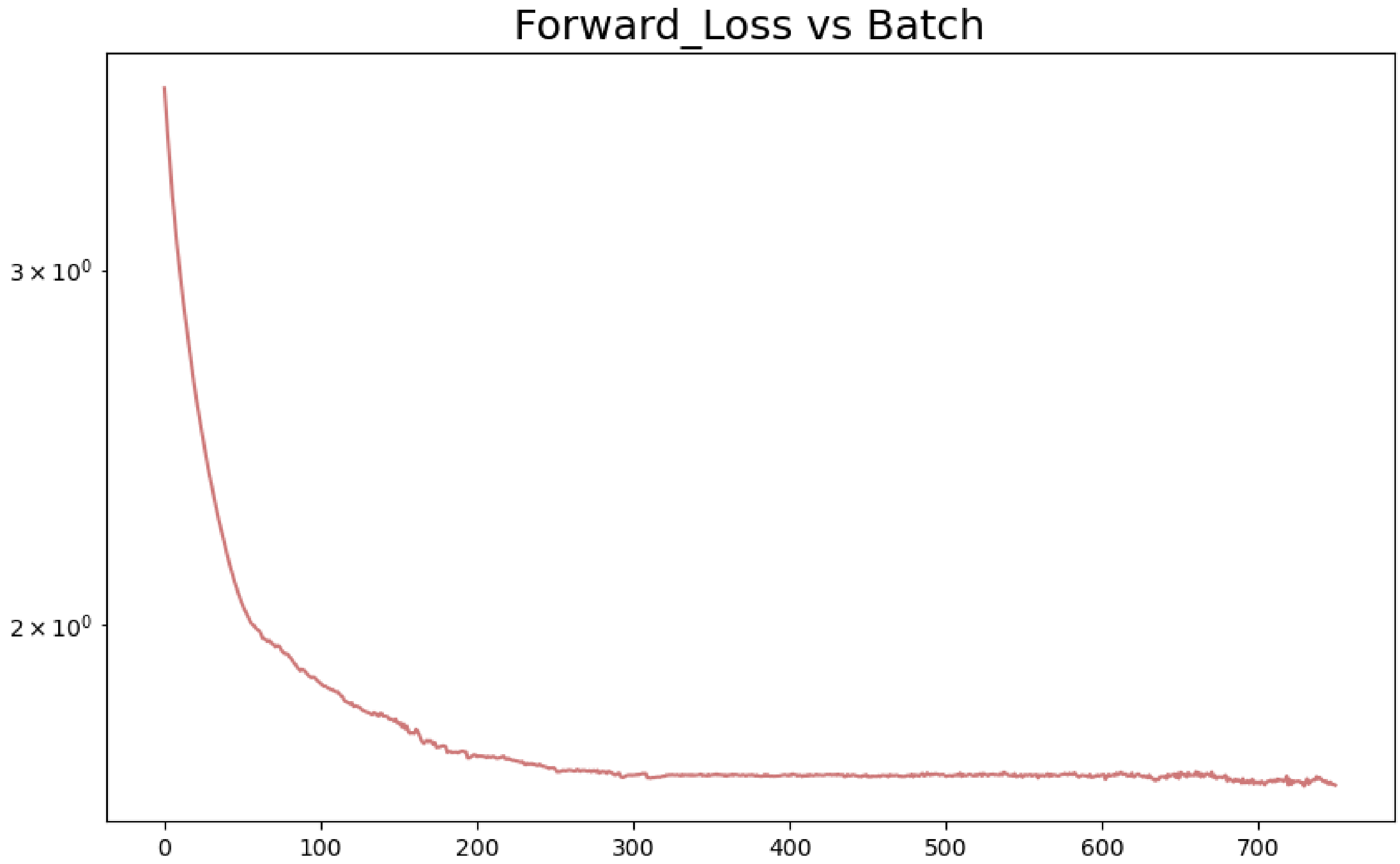
```
Average Error Est:  1.6743533945083617
Batch Number:  690
Average Error Est:  1.6694537258148194
Batch Number:  691
Average Error Est:  1.6782470369338989
Batch Number:  692
Average Error Est:  1.6697843503952026
Batch Number:  693
Average Error Est:  1.6711703205108643
Batch Number:  694
Average Error Est:  1.6745183372497559
Batch Number:  695
Average Error Est:  1.6691656684875489
Batch Number:  696
Average Error Est:  1.6727126741409302
Batch Number:  697
Average Error Est:  1.6771802616119384
Batch Number:  698
Average Error Est:  1.6680183982849122
Batch Number:  699
Average Error Est:  1.672044849395752
Batch Number:  700
Average Error Est:  1.6694517087936402
Batch Number:  701
Average Error Est:  1.6715086364746095
Batch Number:  702
Average Error Est:  1.6681405019760132
Batch Number:  703
Average Error Est:  1.673098177909851
Batch Number:  704
Average Error Est:  1.6696885585784913
Batch Number:  705
Average Error Est:  1.6662631511688233
Batch Number:  706
Average Error Est:  1.6711353397369384
Batch Number:  707
Average Error Est:  1.6733931827545165
Batch Number:  708
Average Error Est:  1.673092474937439
Batch Number:  709
Average Error Est:  1.6752186584472657
Batch Number:  710
Average Error Est:  1.6720041513442994
Batch Number:  711
Average Error Est:  1.6752968072891234
Batch Number:  712
Average Error Est:  1.672436056137085
Batch Number:  713
Average Error Est:  1.6775158643722534
Batch Number:  714
Average Error Est:  1.6743745613098144
```

```
Batch Number:  715
Average Error Est:  1.6760258150100709
Batch Number:  716
Average Error Est:  1.6756342649459839
Batch Number:  717
Average Error Est:  1.672943639755249
Batch Number:  718
Average Error Est:  1.676301393508911
Batch Number:  719
Average Error Est:  1.683203854560852
Batch Number:  720
Average Error Est:  1.6661761045455932
Batch Number:  721
Average Error Est:  1.6763416051864624
Batch Number:  722
Average Error Est:  1.6670189094543457
Batch Number:  723
Average Error Est:  1.6738581132888795
Batch Number:  724
Average Error Est:  1.6723020458221436
Batch Number:  725
Average Error Est:  1.6725559520721436
Batch Number:  726
Average Error Est:  1.6734485340118408
Batch Number:  727
Average Error Est:  1.673474612236023
Batch Number:  728
Average Error Est:  1.670504002571106
Batch Number:  729
Average Error Est:  1.670697979927063
Batch Number:  730
Average Error Est:  1.6634385442733766
Batch Number:  731
Average Error Est:  1.6711410522460937
Batch Number:  732
Average Error Est:  1.6782033967971801
Batch Number:  733
Average Error Est:  1.6678334045410157
Batch Number:  734
Average Error Est:  1.6719546031951904
Batch Number:  735
Average Error Est:  1.6749273538589478
Batch Number:  736
Average Error Est:  1.6755289697647096
Batch Number:  737
Average Error Est:  1.6755341100692749
Batch Number:  738
Average Error Est:  1.6819331073760986
Batch Number:  739
Average Error Est:  1.6775695896148681
Batch Number:  740
```

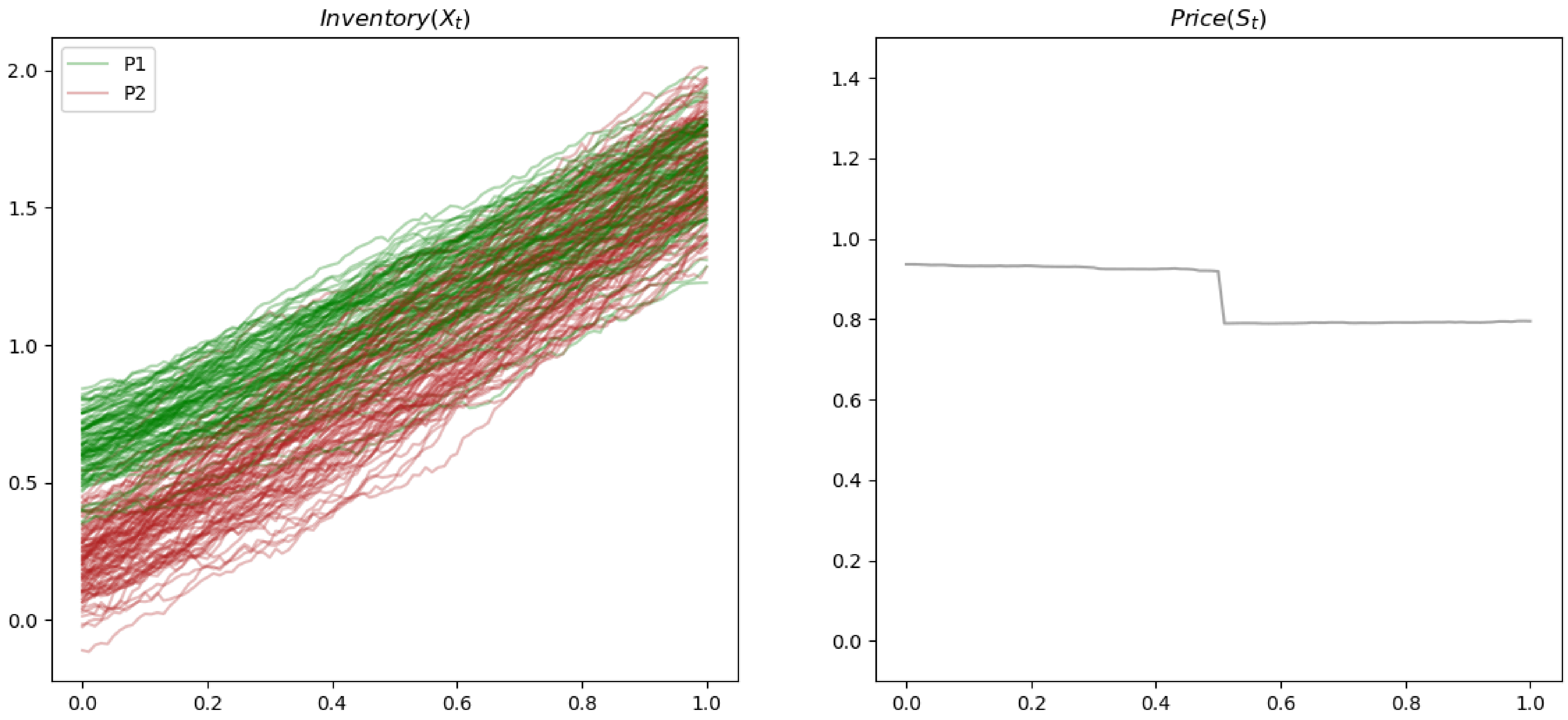
```
Average Error Est:  1.6803805255889892
Batch Number:  741
Average Error Est:  1.6775617504119873
Batch Number:  742
Average Error Est:  1.673789668083191
Batch Number:  743
Average Error Est:  1.6724073266983033
Batch Number:  744
Average Error Est:  1.6738758420944213
Batch Number:  745
Average Error Est:  1.6689344453811645
Batch Number:  746
Average Error Est:  1.667996597290039
Batch Number:  747
Average Error Est:  1.671448721885681
Batch Number:  748
Average Error Est:  1.6672642755508422
Batch Number:  749
Average Error Est:  1.667117977142334
Batch Number:  750
Average Error Est:  1.665627236366272
```

In [ ]:

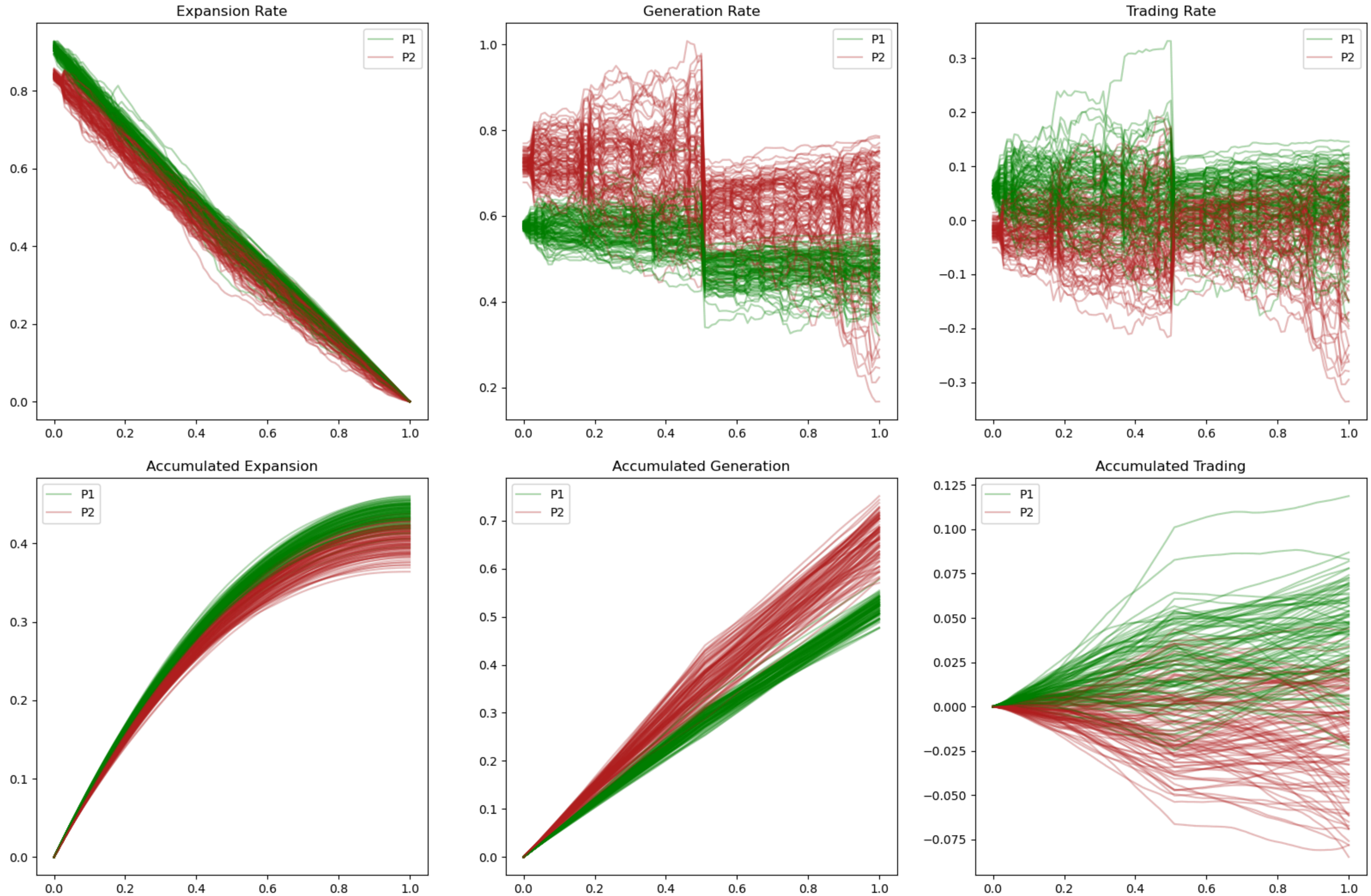
```
plot=plot_results(pop1_dict=pop1_dict, pop2_dict=pop2_dict, loss=forward_losses)
plot.FwdLoss(log=True)
plot.Inventory_And_Price()
plot.Decomposition_Inventory()
plot.Key_Processes()
plot.Terminal_Convergence()
```

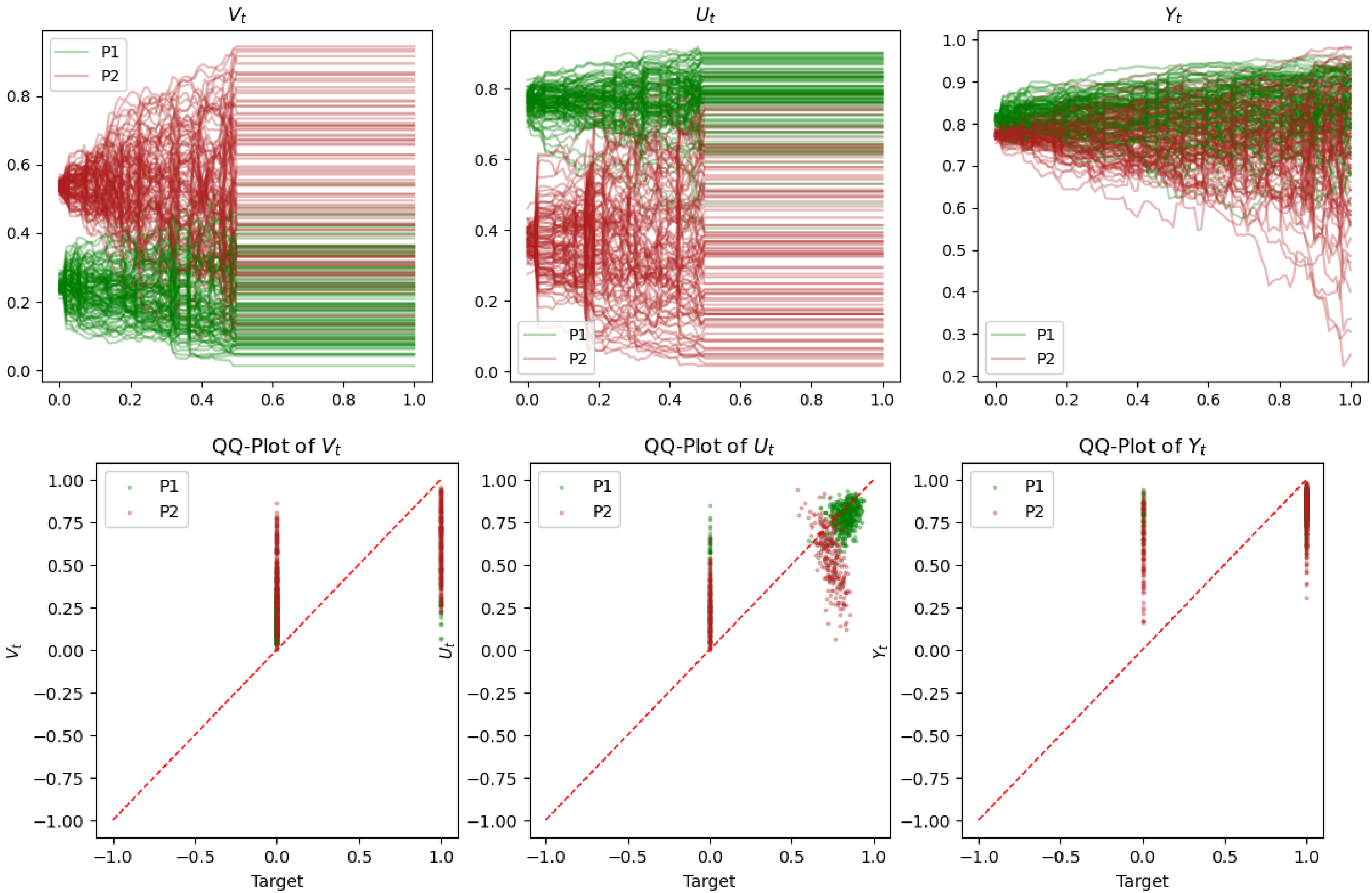


Forward_Loss vs Batch
3 × 10^0
2 × 10^0
0
100
200
300
400
500
600
700

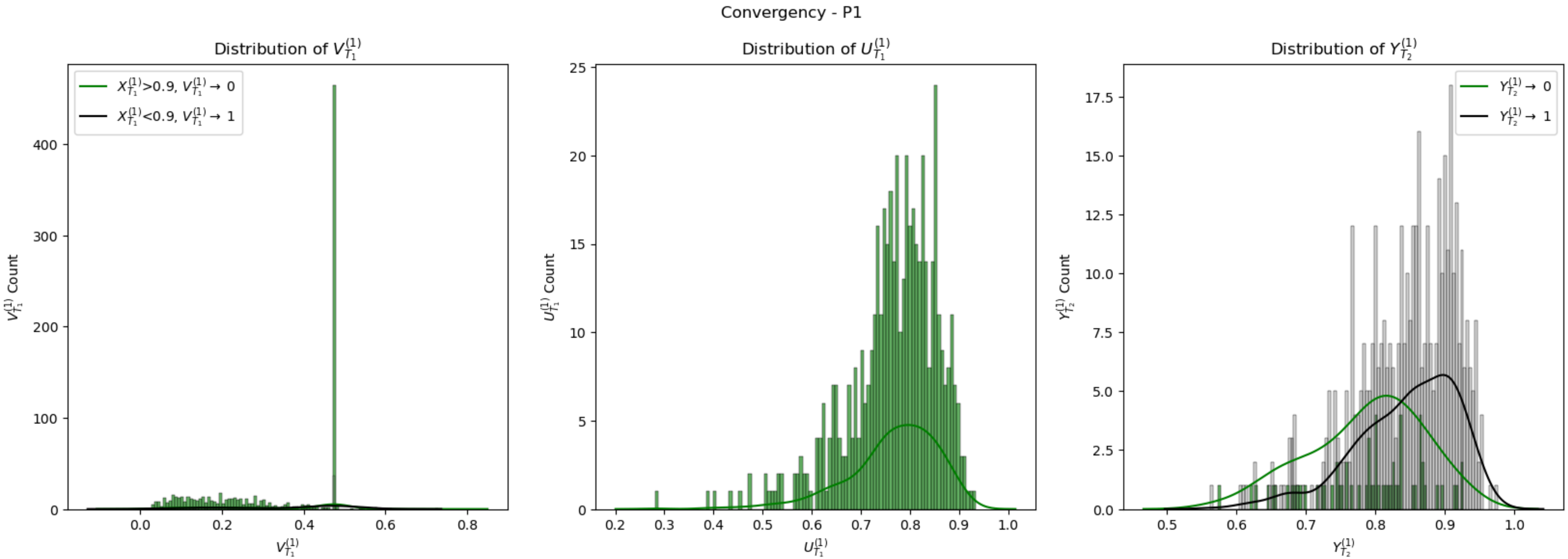


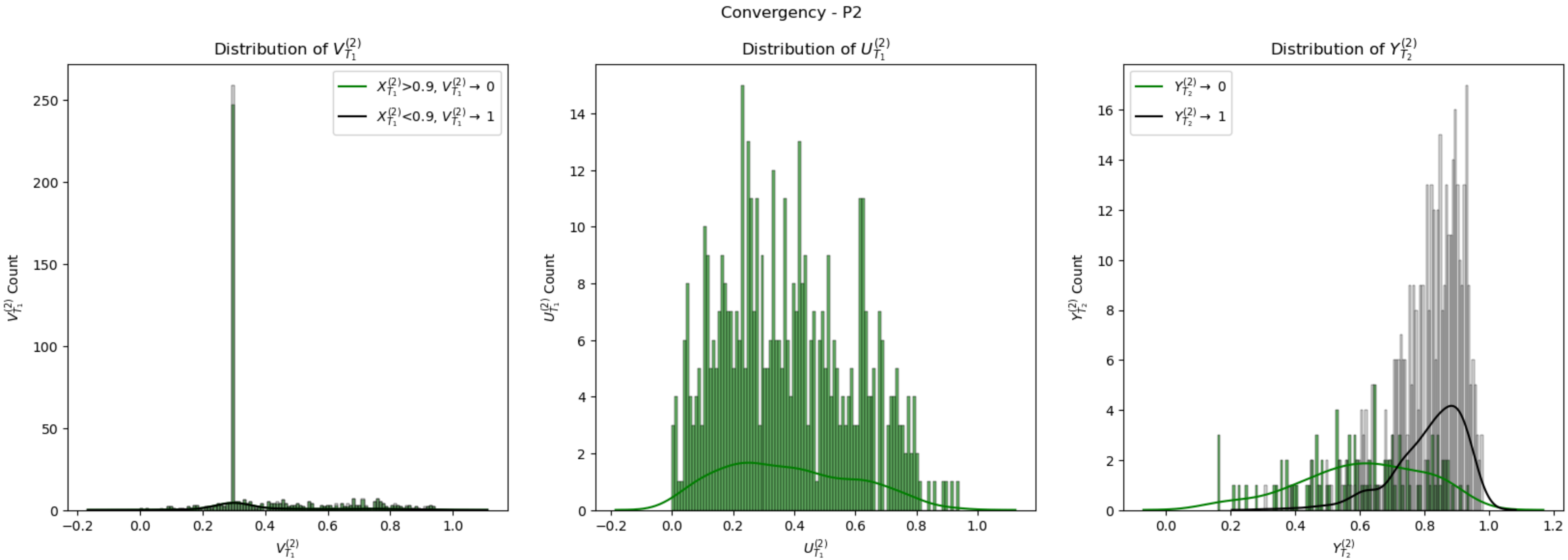
$Inventory(X_t)$
P1
P2
2.0
1.5
1.0
0.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0
$Price(S_t)$
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


Expansion Rate
Generation Rate
Trading Rate
Accumulated Expansion
Accumulated Generation
Accumulated Trading
P1
P2



V_t
U_t
Y_t
P1
P2
QQ-Plot of V_t
QQ-Plot of U_t
QQ-Plot of Y_t
Target


Convergency - P1
Distribution of $V_{T_1}^{(1)}$
$X_{T_1}^{(1)}$>0.9, $V_{T_1}^{(1)} \to 0$
$X_{T_1}^{(1)}$<0.9, $V_{T_1}^{(1)} \to 1$
$V_{T_1}^{(1)}$ Count
$V_{T_1}^{(1)}$
Distribution of $U_{T_1}^{(1)}$
$U_{T_1}^{(1)}$ Count
$U_{T_1}^{(1)}$
Distribution of $Y_{T_2}^{(1)}$
$Y_{T_2}^{(1)} \to 0$
$Y_{T_2}^{(1)} \to 1$
$Y_{T_2}^{(1)}$ Count
$Y_{T_2}^{(1)}$


Convergency - P2
Distribution of $V_{T_1}^{(2)}$
$X_{T_1}^{(2)}$>0.9, $V_{T_1}^{(2)} \to 0$
$X_{T_1}^{(2)}$<0.9, $V_{T_1}^{(2)} \to 1$
$V_{T_1}^{(2)}$ Count
$V_{T_1}^{(2)}$
Distribution of $U_{T_1}^{(2)}$
$U_{T_1}^{(2)}$ Count
$U_{T_1}^{(2)}$
Distribution of $Y_{T_2}^{(2)}$
$Y_{T_2}^{(2)} \to 0$
$Y_{T_2}^{(2)} \to 1$
$Y_{T_2}^{(2)}$ Count
$Y_{T_2}^{(2)}$

Save The Models

In [ ]:
```
print(f"{len(main_models1.loss)} steps\nStarted @ {start_time}\nSaved @ {datetime.now().strftime('%B %d - %H:%M:%S')}")   ## to examine whether
dir_path=pathlib.Path(os.getcwd(),
                      'Results',
                      'Best Models Saved',
                      'sigmoid_ind_0.0001lr_750steps_BCE_1w')
dir_path.mkdir()
path1=pathlib.Path(dir_path,'pop1.pt')
path2=pathlib.Path(dir_path,'pop2.pt')
main_models1.save_entire_models(path=path1)
main_models2.save_entire_models(path=path2)
```

```
750 steps
Started @ August 07 - 14:20:05
Saved @ August 07 - 20:59:53
```

Load The Models

In [ ]:
```
# # GlobalParams1=Params(param_type='k1',target_type='indicator',trick='clamp',loss_type='MSELoss',delta=0.03,K=0.9,lr=0.01)
# # GlobalParams2=Params(param_type='k2',target_type='indicator',trick='clamp',loss_type='MSELoss',delta=0.03,K=0.9,lr=0.01)
# models1=Main_Models(GlobalParams=GlobalParams1)
# models2=Main_Models(GlobalParams=GlobalParams2)
# path_dir=pathlib.Path(os.getcwd(),
#                       "Results",
#                       "Best Models Saved",
#                       'minmax_logit_0.01lr_300steps_BCELogits_1_dt')
# path1=pathlib.Path(path_dir,'pop1.pt')
# path2=pathlib.Path(path_dir,'pop2.pt')
# model_dict1=models1.load_entire_models(path=path1,overwrite=True)
# model_dict2=models2.load_entire_models(path=path2,overwrite=True)

# dB1=model_dict1['dB']
# init_x1=model_dict1['init_x']
# init_c1=model_dict1['init_c']
# pop1_dict= {'dB':dB1,
#             'init_x':init_x1,
#             'init_c':init_c1,
#             'GlobalParams':GlobalParams1,
#             'main_models':models1}

# dB2=model_dict2['dB']
# init_x2=model_dict2['init_x']
# init_c2=model_dict2['init_c']

# pop2_dict= {'dB':dB2,
#             'init_x':init_x2 ,
#             'init_c':init_c2 ,
```

```
#             'GlobalParams':GlobalParams2,
#             'main_models':models2}
# dt=GlobalParams1.dt
# NT1=GlobalParams1.NT1
# NT2=GlobalParams1.NT2
# NumTrain=GlobalParams1.NumTrain
# K=GlobalParams1.K
# loss=models1.loss

# plot=plot_results(pop1_dict=pop1_dict,pop2_dict=pop2_dict,loss=loss)
# plot.FwdLoss(log=True)
# plot.Inventory_And_Price()
# plot.Decomposition_Inventory()
# plot.Key_Processes()
# # plot.Terminal_Convergence()
```