In [ ]:
```
import time
import gym
import random
import torch
import numpy as np
from collections import deque
import matplotlib.pyplot as plt
import math
import sys
import os
from Manifold_Analysis import Manifold_analysis
sys.path.append('../../../')
from algos.agents.dqn_agent import DQNAgent
from algos.models.dqn_cnn import DQNCnn
from algos.preprocessing.stack_frame import preprocess_frame, stack_frame


# importing required libraries
from mpl_toolkits.mplot3d import Axes3D



from sklearn.datasets import load_iris
from sklearn.decomposition import PCA, IncrementalPCA
from sklearn.manifold import MDS
from sklearn.manifold import Isomap
from sklearn.metrics import pairwise_distances

from matplotlib.ticker import NullFormatter
from collections import OrderedDict
from functools import partial
from sklearn import manifold, datasets
```

```
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/ale_py/roms/utils.
py:90: DeprecationWarning: SelectableGroups dict interface is deprecated. U
se select.
  for external in metadata.entry_points().get(self.group, []):
```

In [ ]:
```
import warnings
from scipy.sparse import (spdiags, SparseEfficiencyWarning, csc_matrix,
    csr_matrix, isspmatrix, dok_matrix, lil_matrix, bsr_matrix)
warnings.simplefilter('ignore',SparseEfficiencyWarning)
```

Init GPU or CPU

In [ ]:
```
# if gpu is to be used
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print("Device: ", device)
```

```
Device:  cpu
```

Create Environment

In [ ]:
```
env = gym.make('SpaceInvaders-v0')
env.seed(0)

def stack_frames(frames, state, is_new=False):
    frame = preprocess_frame(state, (8, -12, -12, 4), 84)
    frames = stack_frame(frames, frame, is_new)

    return frames
```

```
A.L.E: Arcade Learning Environment (version +978d2ce)
[Powered by Stella]
```

Create Agent

In [ ]:
```
INPUT_SHAPE = (4, 84, 84)
ACTION_SIZE = env.action_space.n
SEED = 0
GAMMA = 0.99            # discount factor
BUFFER_SIZE = 100000    # replay buffer size
BATCH_SIZE = 64         # Update batch size
LR = 0.0001             # learning rate
TAU = 1e-3              # for soft update of target parameters
UPDATE_EVERY = 1        # how often to update the network
UPDATE_TARGET = 10000   # After which thershold replay to be started
EPS_START = 0.99        # starting value of epsilon
EPS_END = 0.01          # Ending value of epsilon
EPS_DECAY = 100          # Rate by which epsilon to be decayed

epsilon_by_epsiode = lambda frame_idx: EPS_END + (EPS_START - EPS_END) * ma

start_epoch = 0
scores = []
scores_window = deque(maxlen=20)
length_trial = 700
```

In [ ]:
```
agent_init = DQNAgent(INPUT_SHAPE, ACTION_SIZE, SEED, device, BUFFER_SIZE,
agent_1000 = DQNAgent(INPUT_SHAPE, ACTION_SIZE, SEED, device, BUFFER_SIZE,
agent_2000 = DQNAgent(INPUT_SHAPE, ACTION_SIZE, SEED, device, BUFFER_SIZE,
agent_3000 = DQNAgent(INPUT_SHAPE, ACTION_SIZE, SEED, device, BUFFER_SIZE,

Liste_agents = {
    "agent_init" : agent_init,
    "agent_1000" : agent_1000,
    "agent_2000" : agent_2000,
    "agent_3000" : agent_3000
}
```

Load Agents corresponding to different part of the training

In [ ]:

```
for a in Liste_agents.keys():
    if a != "agent_init":

        PATH = os.path.join(os.getcwd(),"Models_training_dqn_space_invader'
        print(PATH)
        checkpoint = torch.load(PATH , map_location=torch.device('cpu'))
        Liste_agents[a].policy_net.load_state_dict(checkpoint['modelA_state
        Liste_agents[a].target_net.load_state_dict(checkpoint['modelB_state
        Liste_agents[a].optimizer.load_state_dict(checkpoint['optimizer_sta
```

```
/Users/charlottebeylier/Documents/PhD/Atari1.0/Reinforcement-Learning_modif
/cgames/02_space_invader/Analysis/Models_training_dqn_space_invader/agent_1
000.pt
/Users/charlottebeylier/Documents/PhD/Atari1.0/Reinforcement-Learning_modif
/cgames/02_space_invader/Analysis/Models_training_dqn_space_invader/agent_2
000.pt
/Users/charlottebeylier/Documents/PhD/Atari1.0/Reinforcement-Learning_modif
/cgames/02_space_invader/Analysis/Models_training_dqn_space_invader/agent_3
000.pt
```

In [ ]:

```
Names_hook = ["fc1","Conv_1","Conv_2","Conv_3"]
Liste_activation = [[[] for i in range(len(Names_hook))] for j in range(ler
i = 0
agent_number = 0
for agent in Liste_agents.values():

    while i < length_trial:
        agent.registration()
        score = 0
        #print(Liste_activation[agent_number])
        Liste_activation[agent_number] = [[] for i in range(len(Names_hook)
        print(Liste_activation[agent_number])
        state = stack_frames(None, env.reset(), True)

        while True:
            #env.render()
            action = agent.act(state)

            for h in range(len(Names_hook)):
                #print(torch.flatten(agent.activation[Names_hook[h]]).unsqu
                Liste_activation[agent_number][h].append(torch.flatten(ager

            next_state, reward, done, _ = env.step(action)
            score += reward
            state = stack_frames(state, next_state, False)
            i +=1
            if done:
                print('\nAgent number :{} \tFinal score: {:.2f} \tNumber of

                break
        env.close()


        agent.detach()

    agent_number += 1
    i = 0
```

```
[[], [], [], []]

Agent number :0         Final score: 0.00       Number of steps: 930[[], []
, [], []]

Agent number :1         Final score: 315.00     Number of steps: 1353[[], [
], [], []]

Agent number :2         Final score: 425.00     Number of steps: 866[[], []
, [], []]

Agent number :3         Final score: 15.00      Number of steps: 920
```

In [ ]:

```
print(len(Liste_activation[3][0]))
```

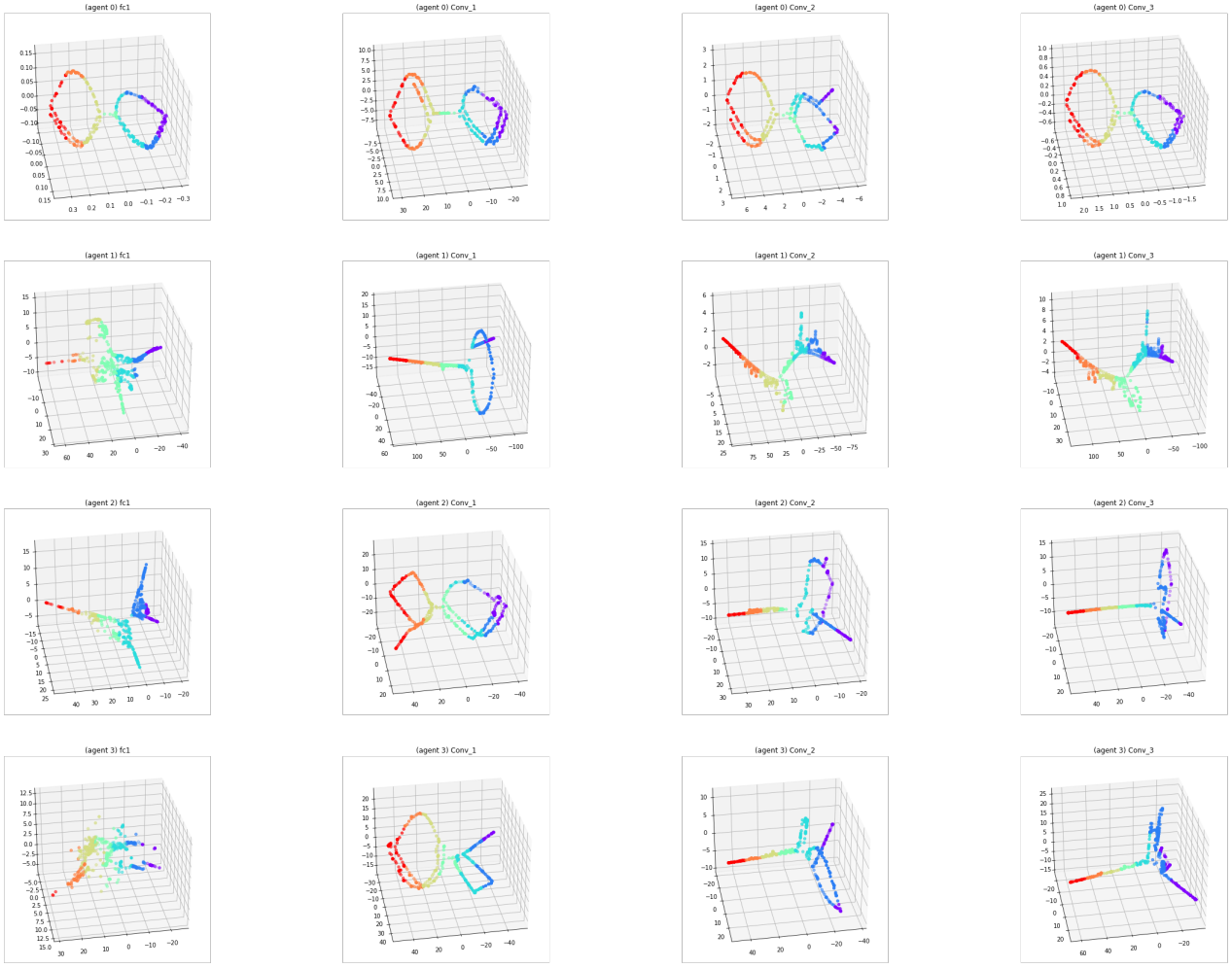
```
920
```

In [ ]:
```
analysis = Manifold_analysis(length_trial)
```

Compute the Isomap

In [ ]:
```
embedding = Isomap(n_neighbors=15,n_components=3)
i = 1
fig = plt.figure(figsize=(40, 30))
for agent_number in range(len(Liste_agents)):
    print("agent_number",agent_number)
    for layer in range(len(Names_hook)):
        activity_layer = analysis.prepro(Liste_activation[agent_number][lay
        X = embedding.fit_transform(activity_layer)
        colorize = dict(c=X[:, 0], cmap=plt.cm.get_cmap('rainbow', 7))
        ax = fig.add_subplot(len(Liste_agents), len(Names_hook),i,projectio
        sc = ax.scatter3D(X[:, 0], X[:, 1], X[:, 2],**colorize)
        ax.view_init(azim=80, elev=30)
        ax.axis("tight")
        ax.set_title("(agent %s) %s" % (agent_number, Names_hook[layer]))
        i+=1
plt.show()
```

```
agent_number 0
agent_number 1
agent_number 2
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
isomap.py:324: UserWarning: The number of connected components of the neigh
bors graph is 2 > 1. Completing the graph to fit Isomap might be slow. Incr
ease the number of neighbors to avoid this issue.
  self._fit_transform(X)
agent_number 3
```



MDS plot

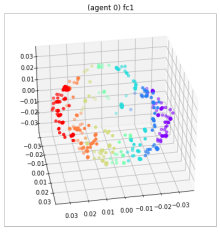
In [ ]:

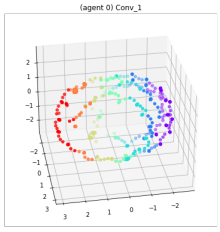
```
D = pairwise_distances(activation)
model = MDS(n_components=dimension, dissimilarity='precomputed', random_sta
out = model.fit_transform(D)
```

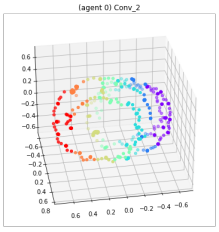
In [ ]:

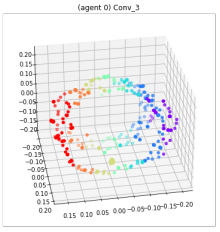
```
model = MDS(n_components=3, dissimilarity='precomputed', random_state=1)
i = 1
fig = plt.figure(figsize=(40, 30))

fig2 = plt.figure(figsize=(40, 30))
for agent_number in range(len(Liste_agents)):
    print("agent_number",agent_number)
    for layer in range(len(Names_hook)):
        activity_layer = analysis.prepro(Liste_activation[agent_number][lay

        D = pairwise_distances(activity_layer)
        X  = model.fit_transform(D)
        colorize = dict(c=X[:, 0], cmap=plt.cm.get_cmap('rainbow', 7))
        ax = fig.add_subplot(len(Liste_agents), len(Names_hook),i,projectio
        sc = ax.scatter3D(X[:, 0], X[:, 1], X[:, 2],**colorize)
        ax.view_init(azim=80, elev=30)
        ax.axis("tight")
        ax.set_title("(agent %s) %s" % (agent_number, Names_hook[layer]))

        ax2 = fig2.add_subplot(len(Liste_agents), len(Names_hook),i)
        ax2.imshow(D, zorder=2, cmap='Blues', interpolation='nearest')
        ax2.axis("tight")
        ax2.set_title("(agent %s) %s" % (agent_number, Names_hook[layer]))




        i+=1
plt.show()
```

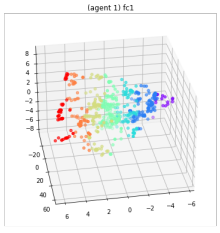
```
agent_number 0
agent_number 1
agent_number 2
agent_number 3
```

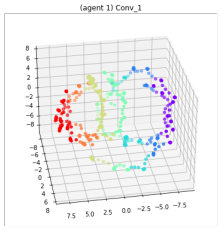

(agent 0) fc1

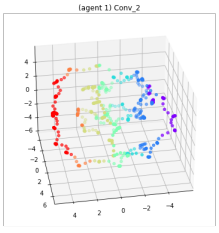

(agent 0) Conv_1

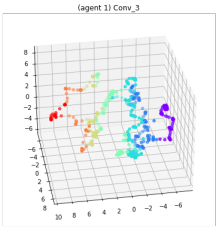

(agent 0) Conv_2

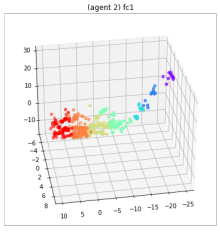

(agent 0) Conv_3

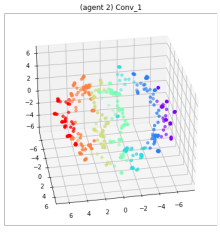

(agent 1) fc1

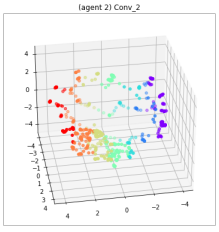

(agent 1) Conv_1

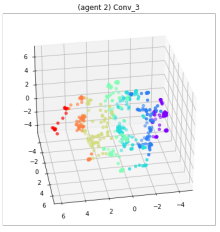

(agent 1) Conv_2

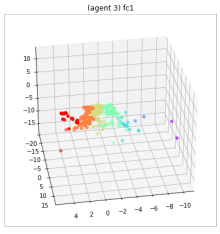

(agent 1) Conv_3

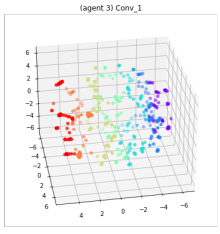

(agent 2) fc1

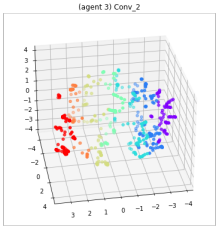

(agent 2) Conv_1

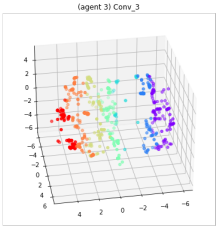

(agent 2) Conv_2

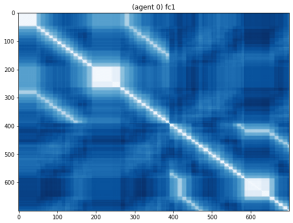

(agent 2) Conv_3

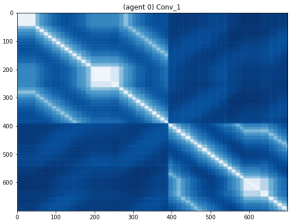

(agent 3) fc1

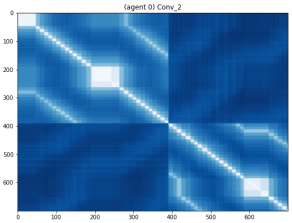

(agent 3) Conv_1

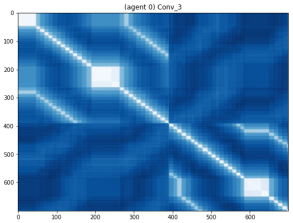

(agent 3) Conv_2

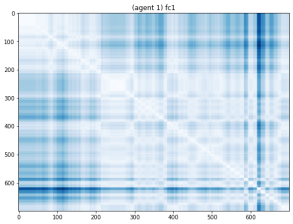

(agent 3) Conv_3

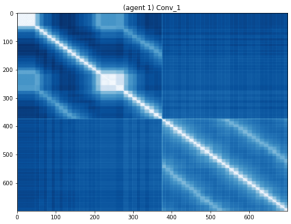

(agent 0) fc1

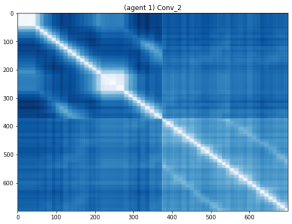

(agent 0) Conv_1

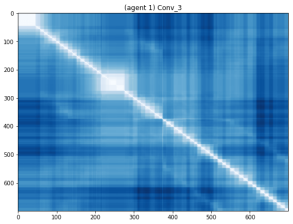

(agent 0) Conv_2

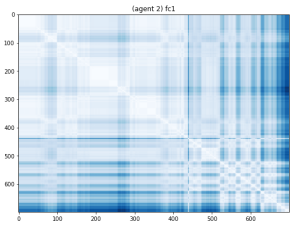

(agent 0) Conv_3

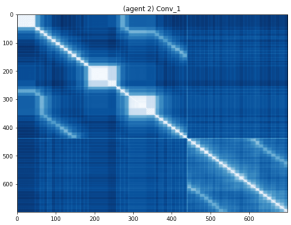

(agent 1) fc1

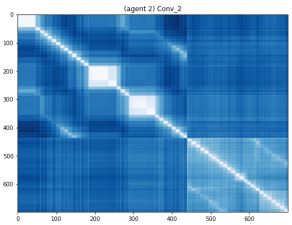

(agent 1) Conv_1

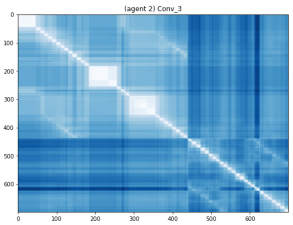

(agent 1) Conv_2

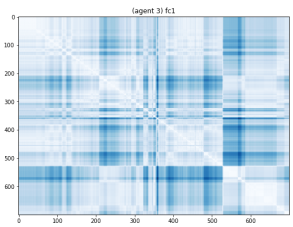

(agent 1) Conv_3

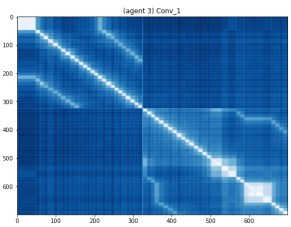

(agent 2) fc1

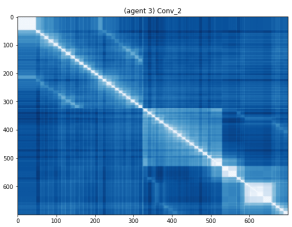

(agent 2) Conv_1

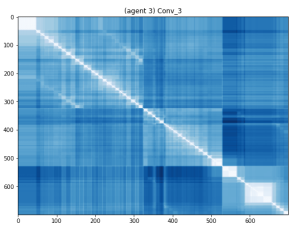

(agent 2) Conv_2


(agent 2) Conv_3


(agent 3) fc1


(agent 3) Conv_1


(agent 3) Conv_2


(agent 3) Conv_3

t-SNE

In [ ]:

```
model = manifold.TSNE(n_components=3, init="pca", random_state=0)

i = 1
fig = plt.figure(figsize=(40, 30))
for agent_number in range(len(Liste_agents)):
    print("agent_number",agent_number)
    for layer in range(len(Names_hook)):
        activity_layer = analysis.prepro(Liste_activation[agent_number][lay
        X  = model.fit_transform(activity_layer)
        colorize = dict(c=X[:, 0], cmap=plt.cm.get_cmap('rainbow', 7))
        ax = fig.add_subplot(len(Liste_agents), len(Names_hook),i,projectio
        sc = ax.scatter3D(X[:, 0], X[:, 1], X[:, 2],**colorize)
        ax.view_init(azim=80, elev=30)
        ax.axis("tight")
        ax.set_title("(agent %s) %s" % (agent_number, Names_hook[layer]))
        i+=1
plt.show()
```

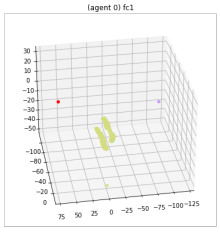
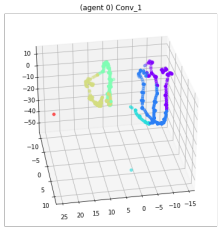
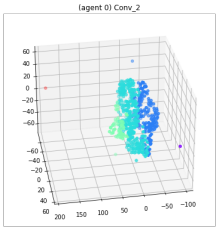
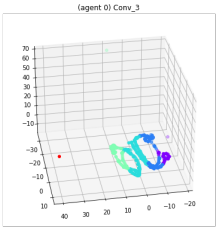
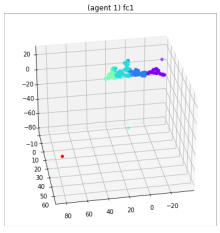
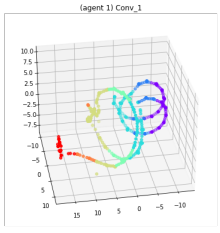
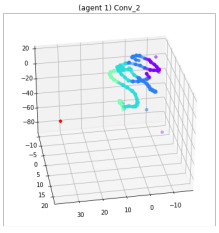
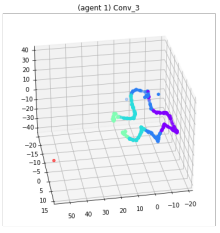
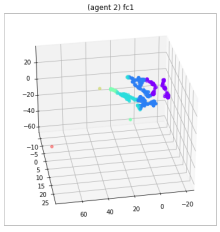
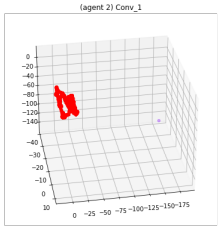
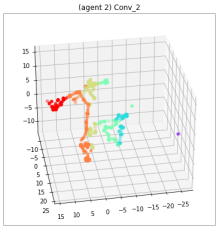
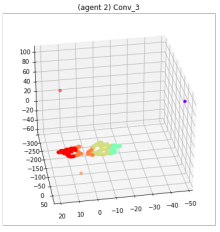
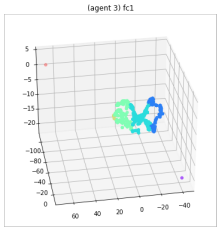
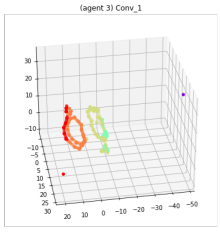
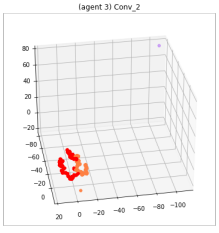
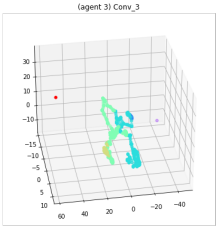
```
agent_number 0
```

```
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
agent_number 1
```

```
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
agent_number 2
```

```
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
agent_number 3
```

```
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:793: FutureWarning: The default learning rate in TSNE will change
from 200.0 to 'auto' in 1.2.
  FutureWarning,
/opt/anaconda3/envs/atari1.0/lib/python3.7/site-packages/sklearn/manifold/_
t_sne.py:986: FutureWarning: The PCA initialization in TSNE will change to
have the standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure b
etter convergence.
  FutureWarning,
```


(agent 0) fc1

(agent 0) Conv_1

(agent 0) Conv_2

(agent 0) Conv_3

(agent 1) fc1

(agent 1) Conv_1

(agent 1) Conv_2

(agent 1) Conv_3

(agent 2) fc1

(agent 2) Conv_1

(agent 2) Conv_2

(agent 2) Conv_3

(agent 3) fc1

(agent 3) Conv_1

(agent 3) Conv_2

(agent 3) Conv_3

In [ ]: