Distributional Deep Q-Networks

```
!apt-get update && apt-get install -y xvfb
```

```
Unpacking libxkbfile1:amd64 (1:1.1.0-1build3) ...
Selecting previously unselected package x11-xkb-utils.
Preparing to unpack .../3-x11-xkb-utils_7.7+5build4_amd64.deb ...
Unpacking x11-xkb-utils (7.7+5build4) ...
Selecting previously unselected package xfonts-encodings.
Preparing to unpack .../4-xfonts-encodings_1%3a1.0.5-0ubuntu2_all.deb ...
Unpacking xfonts-encodings (1:1.0.5-0ubuntu2) ...
Selecting previously unselected package xfonts-utils.
Preparing to unpack .../5-xfonts-utils_1%3a7.7+6build2_amd64.deb ...
Unpacking xfonts-utils (1:7.7+6build2) ...
Selecting previously unselected package xfonts-base.
Preparing to unpack .../6-xfonts-base_1%3a1.0.5_all.deb ...
Unpacking xfonts-base (1:1.0.5) ...
Selecting previously unselected package xserver-common.
Preparing to unpack .../7-xserver-common_2%3a21.1.4-2ubuntu1.7~22.04.12_all.deb ...
Unpacking xserver-common (2:21.1.4-2ubuntu1.7~22.04.12) ...
Selecting previously unselected package xvfb.
Preparing to unpack .../8-xvfb_2%3a21.1.4-2ubuntu1.7~22.04.12_amd64.deb ...
Unpacking xvfb (2:21.1.4-2ubuntu1.7~22.04.12) ...
Setting up libfontenc1:amd64 (1:1.1.4-1build3) ...
Setting up xfonts-encodings (1:1.0.5-0ubuntu2) ...
Setting up libxkbfile1:amd64 (1:1.1.0-1build3) ...
Setting up libxfont2:amd64 (1:2.0.5-1build1) ...
Setting up x11-xkb-utils (7.7+5build4) ...
Setting up xfonts-utils (1:7.7+6build2) ...
Setting up xfonts-base (1:1.0.5) ...
Setting up xserver-common (2:21.1.4-2ubuntu1.7~22.04.12) ...
Setting up xvfb (2:21.1.4-2ubuntu1.7~22.04.12) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for fontconfig (2.13.1-4.2ubuntu5) ...
Processing triggers for libc-bin (2.35-0ubuntu3.4) ...
/sbin/ldconfig.real: /usr/local/lib/libur_adapter_opencl.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_loader.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbb.so.12 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_adapter_level_zero.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtcm.so.1 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libhwloc.so.15 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_0.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtcm_debug.so.1 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libumf.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc_proxy.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_5.so.3 is not a symbolic link
```

```
!pip install gym[atari,accept-rom-license]==0.23.1
```

```
Collecting gym==0.23.1 (from gym[accept-rom-license,atari]==0.23.1)
  Downloading gym-0.23.1.tar.gz (626 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 626.2/626.2 kB 33.9 MB/s eta 0:00:00
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing metadata (pyproject.toml) ... done
Requirement already satisfied: numpy>=1.18.0 in /usr/local/lib/python3.10/dist-packages (from gym==0.23.1->gym[accept-rom-license,at
Requirement already satisfied: cloudpickle>=1.2.0 in /usr/local/lib/python3.10/dist-packages (from gym==0.23.1->gym[accept-rom-licer
Requirement already satisfied: gym_notices>=0.0.4 in /usr/local/lib/python3.10/dist-packages (from gym==0.23.1->gym[accept-rom-licer
Collecting ale-py~=0.7.4 (from gym[accept-rom-license,atari]==0.23.1)
  Downloading ale_py-0.7.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (8.1 kB)
Collecting autorom~=0.4.2 (from autorom[accept-rom-license]~=0.4.2; extra == "accept-rom-license"->gym[accept-rom-license,atari]==0
  Downloading AutoROM-0.4.2-py3-none-any.whl.metadata (2.8 kB)
Requirement already satisfied: importlib-resources in /usr/local/lib/python3.10/dist-packages (from ale-py~=0.7.4->gym[accept-rom-li
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from autorom~=0.4.2->autorom[accept-rom-license]~=0
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from autorom~=0.4.2->autorom[accept-rom-license
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from autorom~=0.4.2->autorom[accept-rom-license]~=0
Collecting AutoROM.accept-rom-license (from autorom[accept-rom-license]~=0.4.2; extra == "accept-rom-license"->gym[accept-rom-licens
  Downloading AutoROM.accept-rom-license-0.6.1.tar.gz (434 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 434.7/434.7 kB 36.8 MB/s eta 0:00:00
```

```
    Installing build dependencies ... done
    Getting requirements to build wheel ... done
    Preparing metadata (pyproject.toml) ... done
  Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->autorom~=0.4.2->a
  Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->autorom~=0.4.2->autorom[accep
  Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->autorom~=0.4.2->autoror
  Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->autorom~=0.4.2->autoror
  Downloading ale_py-0.7.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.6 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.6/1.6 MB 55.3 MB/s eta 0:00:00
  Downloading AutoROM-0.4.2-py3-none-any.whl (16 kB)
  Building wheels for collected packages: gym, AutoROM.accept-rom-license
    Building wheel for gym (pyproject.toml) ... done
    Created wheel for gym: filename=gym-0.23.1-py3-none-any.whl size=701367 sha256=acba4f02837f0a683313928a9424a8a09880abd27e50888a424
    Stored in directory: /root/.cache/pip/wheels/1a/00/fb/fe5cf2860fb9b7bc860e28f00095a1f42c7b726dd6f42d1acc
    Building wheel for AutoROM.accept-rom-license (pyproject.toml) ... done
    Created wheel for AutoROM.accept-rom-license: filename=AutoROM.accept_rom_license-0.6.1-py3-none-any.whl size=446667 sha256=5f5ce2
    Stored in directory: /root/.cache/pip/wheels/6b/1b/ef/a43ff1a2f1736d5711faa1ba4c1f61be1131b8899e6a057811
  Successfully built gym AutoROM.accept-rom-license
  Installing collected packages: gym, ale-py, AutoROM.accept-rom-license, autorom
    Attempting uninstall: gym
      Found existing installation: gym 0.25.2
      Uninstalling gym-0.25.2:
        Successfully uninstalled gym-0.25.2
  Successfully installed AutoROM.accept-rom-license-0.6.1 ale-py-0.7.5 autorom-0.4.2 gym-0.23.1
```

```
!pip install pytorch-lightning
```

```
  Collecting pytorch-lightning
    Downloading pytorch_lightning-2.4.0-py3-none-any.whl.metadata (21 kB)
  Requirement already satisfied: torch>=2.1.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning) (2.5.1+cu121)
  Requirement already satisfied: tqdm>=4.57.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning) (4.66.6)
  Requirement already satisfied: PyYAML>=5.4 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning) (6.0.2)
  Requirement already satisfied: fsspec>=2022.5.0 in /usr/local/lib/python3.10/dist-packages (from fsspec[http]>=2022.5.0->pytorch-lig
  Collecting torchmetrics>=0.7.0 (from pytorch-lightning)
    Downloading torchmetrics-1.6.0-py3-none-any.whl.metadata (20 kB)
  Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning) (24.2)
  Requirement already satisfied: typing-extensions>=4.4.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning) (4.12.2
  Collecting lightning-utilities>=0.10.0 (from pytorch-lightning)
    Downloading lightning_utilities-0.11.9-py3-none-any.whl.metadata (5.2 kB)
  Requirement already satisfied: aiohttp!=4.0.0a0,!=4.0.0a1 in /usr/local/lib/python3.10/dist-packages (from fsspec[http]>=2022.5.0->p
  Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from lightning-utilities>=0.10.0->pytorch-ligh
  Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from torch>=2.1.0->pytorch-lightning) (3.16.1)
  Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=2.1.0->pytorch-lightning) (3.4.2)
  Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=2.1.0->pytorch-lightning) (3.1.4)
  Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from torch>=2.1.0->pytorch-lightning) (1.13
  Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy==1.13.1->torch>=2.1.0->pyto
  Requirement already satisfied: numpy>1.20.0 in /usr/local/lib/python3.10/dist-packages (from torchmetrics>=0.7.0->pytorch-lightning
  Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->
  Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fsspec
  Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fsspec[htt
  Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fsspec
  Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fssp
  Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fsspec
  Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->fsspec
  Requirement already satisfied: async-timeout<6.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp!=4.0.0a0,!=4.0.0a1->
  Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch>=2.1.0->pytorch-lightn
  Requirement already satisfied: idna>=2.0 in /usr/local/lib/python3.10/dist-packages (from yarl<2.0,>=1.17.0->aiohttp!=4.0.0a0,!=4.0
  Downloading pytorch_lightning-2.4.0-py3-none-any.whl (815 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 815.2/815.2 kB 37.0 MB/s eta 0:00:00
  Downloading lightning_utilities-0.11.9-py3-none-any.whl (28 kB)
  Downloading torchmetrics-1.6.0-py3-none-any.whl (926 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 926.4/926.4 kB 48.3 MB/s eta 0:00:00
  Installing collected packages: lightning-utilities, torchmetrics, pytorch-lightning
  Successfully installed lightning-utilities-0.11.9 pytorch-lightning-2.4.0 torchmetrics-1.6.0
```

```
!pip install stable-baselines3==1.4.0
```

```
  Collecting stable-baselines3==1.4.0
    Downloading stable_baselines3-1.4.0-py3-none-any.whl.metadata (3.9 kB)
  Collecting gym<0.20,>=0.17 (from stable-baselines3==1.4.0)
    Downloading gym-0.19.0.tar.gz (1.6 MB)
       ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.6/1.6 MB 32.3 MB/s eta 0:00:00
    error: subprocess-exited-with-error

    × python setup.py egg_info did not run successfully.
    │ exit code: 1
    ╰─> See above for output.

    note: This error originates from a subprocess, and is likely not a problem with pip.
    Preparing metadata (setup.py) ... error
  error: metadata-generation-failed

  × Encountered error while generating package metadata.
  ╰─> See above for output.

  note: This is an issue with the package mentioned above, not pip.
```

```
    hint: See above for details.
```

```
!pip install pyvirtualdisplay
```

```
Collecting pyvirtualdisplay
  Downloading PyVirtualDisplay-3.0-py3-none-any.whl.metadata (943 bytes)
Downloading PyVirtualDisplay-3.0-py3-none-any.whl (15 kB)
Installing collected packages: pyvirtualdisplay
Successfully installed pyvirtualdisplay-3.0
```

Setup virtual display

```
from pyvirtualdisplay import Display
Display(visible=False, size=(1400, 900)).start()
```

```
<pyvirtualdisplay.display.Display at 0x7f069610f550>
```

Import the necessary code libraries

```
import copy
import torch
import random
import gym
import matplotlib

import numpy as np
import matplotlib.pyplot as plt

import torch.nn.functional as F

from collections import deque, namedtuple
from IPython.display import HTML
from base64 import b64encode

from torch import nn
from torch.utils.data import DataLoader
from torch.utils.data.dataset import IterableDataset
from torch.optim import AdamW

from pytorch_lightning import LightningModule, Trainer

from gym.wrappers import TransformObservation, NormalizeObservation, \
  NormalizeReward, RecordVideo, RecordEpisodeStatistics, AtariPreprocessing


device = 'cuda:0' if torch.cuda.is_available() else 'cpu'
num_gpus = torch.cuda.device_count()
```

```
def display_video(episode=0):
  video_file = open(f'/content/videos/rl-video-episode-{episode}.mp4', "r+b").read()
  video_url = f"data:video/mp4;base64,{b64encode(video_file).decode()}"
  return HTML(f"<video width=600 controls><source src='{video_url}'></video>")
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipykernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
  and should_run_async(code)
```

Create the Deep Q-Network

```
import math
from torch.nn.init import kaiming_uniform_, zeros_

class NoisyLinear(nn.Module):

  def __init__(self, in_features, out_features, sigma):
    super(NoisyLinear, self).__init__()
    self.w_mu = nn.Parameter(torch.empty((out_features, in_features)))
    self.w_sigma = nn.Parameter(torch.empty((out_features, in_features)))
    self.b_mu = nn.Parameter(torch.empty((out_features)))
    self.b_sigma = nn.Parameter(torch.empty((out_features)))

    kaiming_uniform_(self.w_mu, a=math.sqrt(5))
    kaiming_uniform_(self.w_sigma, a=math.sqrt(5))
    zeros_(self.b_mu)
    zeros_(self.b_sigma)
  def forward(self, x, sigma=0.5):
    if self.training:
      w_noise = torch.normal(0, sigma, size=self.w_mu.size()).to(device)
      b_noise = torch.normal(0, sigma, size=self.b_mu.size()).to(device)
      return F.linear(x, self.w_mu + self.w_sigma * w_noise, self.b_mu + self.b_sigma * b_noise)
    else:
      return F.linear(x, self.W_mu, self.b_mu)
```

```
class DQN(nn.Module):

  def __init__(self, hidden_size, obs_shape, n_actions, atoms=51, sigma=0.5):
    super().__init__()
    self.atoms = atoms
    self.n_actions = n_actions

    self.conv = nn.Sequential(
      nn.Conv2d(obs_shape[0], 64, kernel_size=3),
      nn.MaxPool2d(kernel_size=4),
      nn.ReLU(),
      nn.Conv2d(64, 64, kernel_size=3),
      nn.MaxPool2d(kernel_size=4),
      nn.ReLU()
    )
    conv_out_size = self._get_conv_out(obs_shape)
    self.head = nn.Sequential(
      NoisyLinear(conv_out_size, hidden_size, sigma=sigma),
      nn.ReLU(),
    )

    self.fc_adv = NoisyLinear(hidden_size, self.n_actions * self.atoms, sigma=sigma)
    self.fc_value = NoisyLinear(hidden_size, self.atoms, sigma=sigma)

  def _get_conv_out(self, shape):
    conv_out = self.conv(torch.zeros(1, *shape))
    return int(np.prod(conv_out.size()))

  def forward(self, x):
    x = self.conv(x.float()).view(x.size()[0], -1)
    x = self.head(x)
    adv = self.fc_adv(x).view(-1, self.n_actions, self.atoms)  # (B, A, N)
    value = self.fc_value(x).view(-1, 1, self.atoms)  # (B, 1, N)
    q_logits = value + adv - adv.mean(dim=1, keepdim=True)  # (B, A, N)
    q_probs = F.softmax(q_logits, dim=-1)  # (B, A, N)
    return q_probs
```

Create the policy

```
def greedy(state, net, support):
  state = torch.tensor([state]).to(device)
  q_value_probs = net(state)  # (1, A, N)
  q_values = (support * q_value_probs).sum(dim=-1)  # (1, A)
  action = torch.argmax(q_values, dim=-1)  # (1, 1)
  action = int(action.item())  # ()
  return action
```

Create the replay buffer

```
class ReplayBuffer:

  def __init__(self, capacity):
    self.buffer = deque(maxlen=capacity)
    self.priorities = deque(maxlen=capacity)
    self.capacity = capacity
    self.alpha = 0.0  # anneal.
    self.beta = 1.0  # anneal.
    self.max_priority = 0.0

  def __len__(self):
    return len(self.buffer)

  def append(self, experience):
    self.buffer.append(experience)
    self.priorities.append(self.max_priority)

  def update(self, index, priority):
    if priority > self.max_priority:
      self.max_priority = priority
    self.priorities[index] = priority

  def sample(self, batch_size):
    prios = np.array(self.priorities, dtype=np.float64) + 1e-4 # Stability constant.
    prios = prios ** self.alpha
    probs = prios / prios.sum()

    weights = (self.__len__() * probs) ** -self.beta
    weights = weights / weights.max()

    idx = random.choices(range(self.__len__()), weights=probs, k=batch_size)
    sample = [(i, weights[i], *self.buffer[i]) for i in idx]
    return sample
```

```
class RLDataset(IterableDataset):

  def __init__(self, buffer, sample_size=400):
    self.buffer = buffer
    self.sample_size = sample_size

  def __iter__(self):
    for experience in self.buffer.sample(self.sample_size):
      yield experience
```

Create the environment

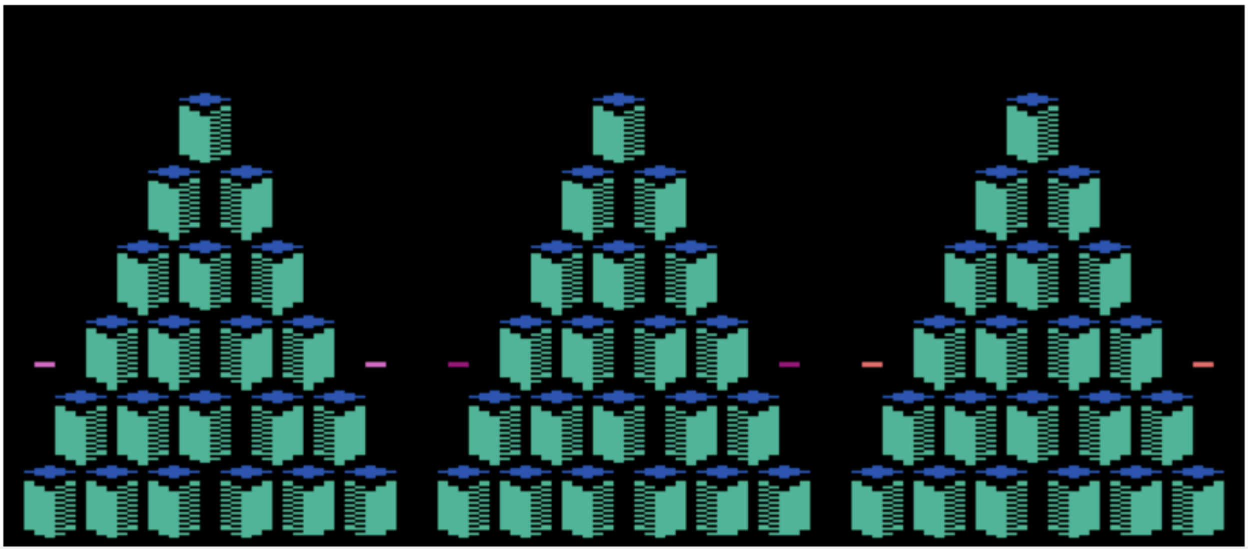
```
env = gym.make('QbertNoFrameskip-v4')
```

```
env.observation_space, env.action_space
```

```
(Box(0, 255, (210, 160, 3), uint8), Discrete(6))
```

```
frames = []
i = 60
skip = 8
obs = env.reset()
done = False

while not done:
  frames.append(obs)
  obs, _, done, _ = env.step(env.action_space.sample())

frames = np.hstack([frames[i], frames[i+skip], frames[i+2*skip]])
plt.figure(figsize=(12, 8))
plt.axis('off')
plt.imshow(frames)
```

```
<matplotlib.image.AxesImage at 0x7f05c228f130>
```

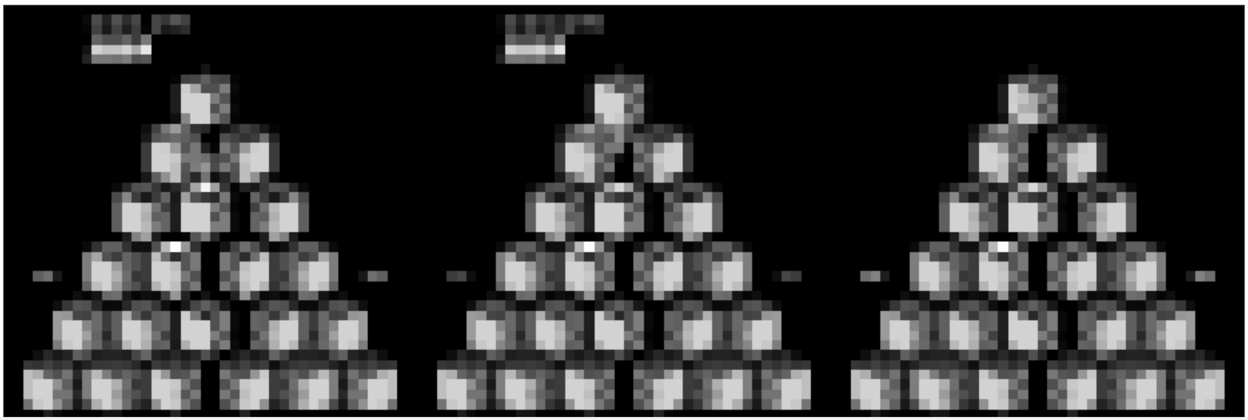


```
env = AtariPreprocessing(env, frame_skip=8, screen_size=42)
```

```
frames = []
i = 40
skip = 1
```

```
obs = env.reset()
done = False

while not done:
  frames.append(obs)
  obs, _, done, _ = env.step(env.action_space.sample())

img = np.hstack([frames[i], frames[i+skip], frames[i+2*skip]])
plt.figure(figsize=(12, 8))
plt.axis('off')
plt.imshow(img, cmap='gray')
```

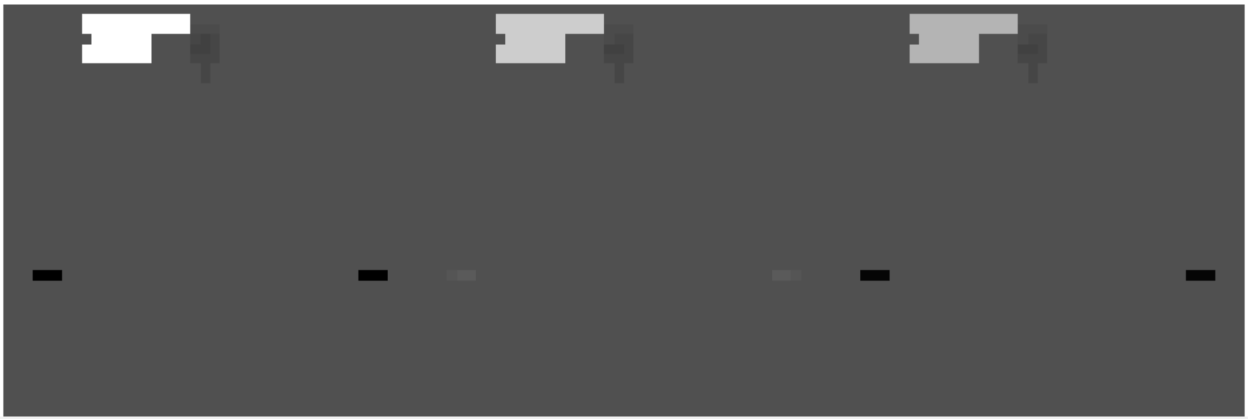
```
<matplotlib.image.AxesImage at 0x7f05bec80550>
```



```
env = NormalizeObservation(env)
```

```
frames = []
i = 100
skip = 1

for i in range(20):
  obs = env.reset()
  done = False
  while not done:
    frames.append(obs)
    obs, _, done, _ = env.step(env.action_space.sample())

img = np.hstack([frames[i], frames[i+skip], frames[i+2*skip]])
plt.figure(figsize=(12, 8))
plt.axis('off')
plt.imshow(img.squeeze(), cmap='gray')
```

```
<matplotlib.image.AxesImage at 0x7f05bec7d3c0>
```



```
def create_environment(name):
  env = gym.make(name)
  env = RecordVideo(env, 'videos', episode_trigger=lambda e: e % 100 == 0)
  env = AtariPreprocessing(env, frame_skip=8, screen_size=42)
  env = RecordEpisodeStatistics(env)
  env = TransformObservation(env, lambda x: x[np.newaxis,:,:])
  env.observation_space = gym.spaces.Box(low=0, high=1, shape=(1, 42, 42), dtype=np.float32)
  env = NormalizeObservation(env)
  env = NormalizeReward(env)
  return env
```

```
env = create_environment('QbertNoFrameskip-v4')
frames = []
for episode in range(10):
  done = False
  obs = env.reset()
  while not done:
    frames.append(obs)
    action = env.action_space.sample()
    obs, _, done, _ = env.step(action)
```

```
/usr/local/lib/python3.10/dist-packages/gym/wrappers/monitoring/video_recorder.py:341: DeprecationWarning: Use shutil.which instead
  if distutils.spawn.find_executable("avconv") is not None:
/usr/local/lib/python3.10/dist-packages/gym/wrappers/monitoring/video_recorder.py:421: DeprecationWarning: distutils Version classe
  if distutils.version.LooseVersion(
```

```
display_video(episode=0)
```

0:40 / 0:40

Create the Deep Q-Learning algorithm

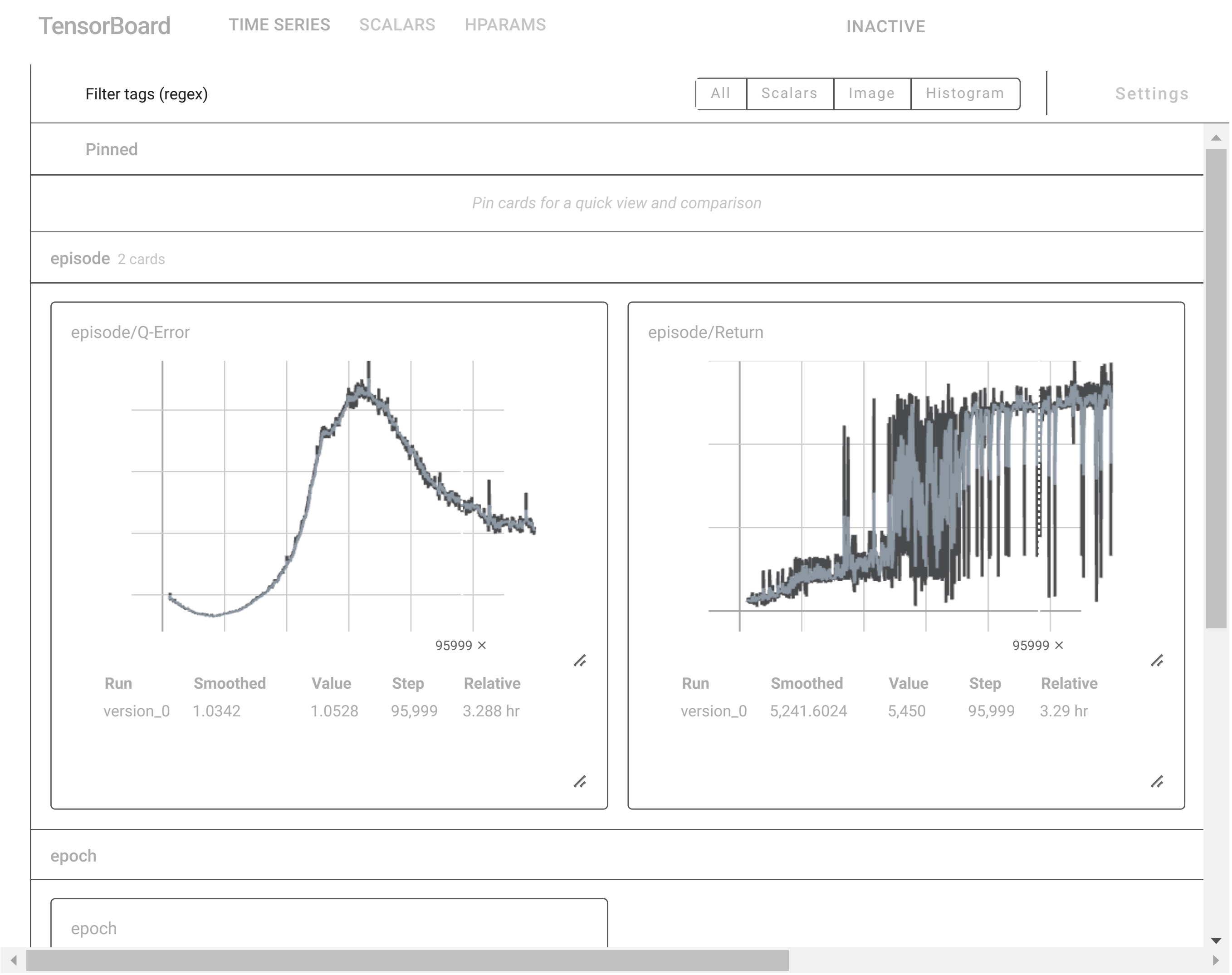
```
class DeepQLearning(LightningModule):

  # Initialize.
  def __init__(self, env_name, policy=greedy, capacity=100_000,
               batch_size=256, lr=1e-3, hidden_size=128, gamma=0.99,
               loss_fn=F.smooth_l1_loss, optim=AdamW, samples_per_epoch=10_000,
               sync_rate=10, sigma=0.5, a_start=0.5, a_end=0.0, a_last_episode=100,
               b_start=0.4, b_end=1.0, b_last_episode=100, n_steps=3,
               v_min=-10.0, v_max=10.0, atoms=51):

    super().__init__()

    self.support = torch.linspace(v_min, v_max, atoms, device=device)  # (N)
    self.delta = (v_max - v_min) / (atoms - 1)

    self.env = create_environment(env_name)
    obs_size = self.env.observation_space.shape
    n_actions = self.env.action_space.n

    self.q_net = DQN(hidden_size, obs_size, n_actions, atoms=atoms, sigma=sigma)

    self.target_q_net = copy.deepcopy(self.q_net)

    self.policy = policy
    self.buffer = ReplayBuffer(capacity=capacity)

    self.save_hyperparameters()

    while len(self.buffer) < self.hparams.samples_per_epoch:
      print(f"{len(self.buffer)} samples in experience buffer. Filling...")
      self.play_episode()

  @torch.no_grad()
  def play_episode(self, policy=None):
    state = self.env.reset()
    done = False
    transitions = []

    while not done:
      if policy:
        action = policy(state, self.q_net, self.support)
      else:
        action = self.env.action_space.sample()

      next_state, reward, done, info = self.env.step(action)
      exp = (state, action, reward, done, next_state)
      transitions.append(exp)
      state = next_state

    for i, (s, a, r, d, ns) in enumerate(transitions):
      batch = transitions[i:i+self.hparams.n_steps]
      ret = sum([t[2] * self.hparams.gamma**j for j, t in enumerate(batch)])
      _, _, _, ld, ls = batch[-1]
      self.buffer.append((s, a, ret, ld, ls))

  # Forward.
  def forward(self, x):
    return self.q_net(x)

  # Configure optimizers.
  def configure_optimizers(self):
    q_net_optimizer = self.hparams.optim(self.q_net.parameters(), lr=self.hparams.lr)
    return [q_net_optimizer]

  # Create dataloader.
  def train_dataloader(self):
    dataset = RLDataset(self.buffer, self.hparams.samples_per_epoch)
    dataloader = DataLoader(
        dataset=dataset,
        batch_size=self.hparams.batch_size
    )
    return dataloader

  # Training step.
  def training_step(self, batch, batch_idx):
    indices, weights, states, actions, returns, dones, next_states = batch
    returns = returns.unsqueeze(1)
    dones = dones.unsqueeze(1)
    batch_size = len(indices)

    q_value_probs = self.q_net(states)  # (B, A, N)

    action_value_probs = q_value_probs[range(batch_size), actions, :]  # (B, N)
    log_action_value_probs = torch.log(action_value_probs + 1e-6)  # (B, N)

    with torch.no_grad():
      next_q_value_probs = self.q_net(next_states)  # (B, A, N)
      next_q_values = (next_q_value_probs * self.support).sum(dim=-1)  # (B, A)
      next_actions = next_q_values.argmax(dim=-1)  # (B,)

      next_q_value_probs = self.target_q_net(next_states)  # (B, A, N)
      next_action_value_probs = next_q_value_probs[range(batch_size), next_actions, :]  # (B, N)

    m = torch.zeros(batch_size * self.hparams.atoms, device=device, dtype=torch.float64)  # (B * N)

    Tz = returns + ~dones * self.hparams.gamma**self.hparams.n_steps * self.support.unsqueeze(0)  # (B, N)

    Tz.clamp_(min=self.hparams.v_min, max=self.hparams.v_max)  # (B, N)
    b = (Tz - self.hparams.v_min) / self.delta  # (B, N)
    l, u = b.floor().long(), b.ceil().long()  # (B, N)

    offset = torch.arange(batch_size, device=device).view(-1, 1) * self.hparams.atoms  # (B, 1)

    l_idx = (l + offset).flatten()  # (B * N)
    u_idx = (u + offset).flatten()  # (B * N)

    upper_probs = (next_action_value_probs * (u - b)).flatten()  # (B * N)
    lower_probs = (next_action_value_probs * (b - l)).flatten()  # (B * N)

    m.index_add_(dim=0, index=l_idx, source=upper_probs)
    m.index_add_(dim=0, index=u_idx, source=lower_probs)

    m = m.reshape(batch_size, self.hparams.atoms)  # (B, N)

    cross_entropies = - (m * log_action_value_probs).sum(dim=-1)  # (B,)

    for idx, e in zip(indices, cross_entropies):
      self.buffer.update(idx, e.detach().item())

    loss = (weights * cross_entropies).mean()

    self.log('episode/Q-Error', loss)
    return loss

  # Training epoch end.
  def on_train_epoch_end(self):
    alpha = max(
        self.hparams.a_end,
        self.hparams.a_start - self.current_epoch / self.hparams.a_last_episode
    )
    beta = min(
        self.hparams.b_end,
        self.hparams.b_start + self.current_epoch / self.hparams.b_last_episode
    )
    self.buffer.alpha = alpha
    self.buffer.beta = beta

    self.play_episode(policy=self.policy)
    self.log('episode/Return', self.env.return_queue[-1])

    if self.current_epoch % self.hparams.sync_rate == 0:
      self.target_q_net.load_state_dict(self.q_net.state_dict())
```

```
import pytorch_lightning as pl
import warnings
warnings.filterwarnings('ignore')
```

Purge logs and run the visualization tool (Tensorboard)

```
!rm -r /content/lightning_logs/
!rm -r /content/videos/
%load_ext tensorboard
%tensorboard --logdir /content/lightning_logs/
```

```
rm: cannot remove '/content/lightning_logs/': No such file or directory
```



episode/Q-Error

Run	Smoothed	Value	Step	Relative
version_0	1.0342	1.0528	95,999	3.288 hr

episode/Return

Run	Smoothed	Value	Step	Relative
version_0	5,241.6024	5,450	95,999	3.29 hr

Train the policy

```
algo = DeepQLearning(
  'QbertNoFrameskip-v4',
  lr=0.0001,
  sigma=0.5,
  hidden_size=512,
  a_last_episode=2_000,
  b_last_episode=2_000,
  n_steps=8,
)

trainer = pl.Trainer(
    accelerator="gpu" if num_gpus else "cpu", # Use 'gpu' if num_gpus is greater than 0, otherwise use 'cpu'
    devices=1, # Specify the number of GPUs or 'auto' for automatic detection
    max_epochs=2400,
    log_every_n_steps=1
)

trainer.fit(algo)
```

```
180 samples in experience buffer. Filling...
346 samples in experience buffer. Filling...
528 samples in experience buffer. Filling...
688 samples in experience buffer. Filling...
865 samples in experience buffer. Filling...
1022 samples in experience buffer. Filling...
1237 samples in experience buffer. Filling...
1388 samples in experience buffer. Filling...
1566 samples in experience buffer. Filling...
1738 samples in experience buffer. Filling...
1945 samples in experience buffer. Filling...
2114 samples in experience buffer. Filling...
2274 samples in experience buffer. Filling...
2479 samples in experience buffer. Filling...
2639 samples in experience buffer. Filling...
2797 samples in experience buffer. Filling...
2954 samples in experience buffer. Filling...
3137 samples in experience buffer. Filling...
3286 samples in experience buffer. Filling...
3501 samples in experience buffer. Filling...
3707 samples in experience buffer. Filling...
3856 samples in experience buffer. Filling...
4004 samples in experience buffer. Filling...
4173 samples in experience buffer. Filling...
4322 samples in experience buffer. Filling...
4509 samples in experience buffer. Filling...
4650 samples in experience buffer. Filling...
4791 samples in experience buffer. Filling...
4954 samples in experience buffer. Filling...
5138 samples in experience buffer. Filling...
5285 samples in experience buffer. Filling...
5430 samples in experience buffer. Filling...
5576 samples in experience buffer. Filling...
5698 samples in experience buffer. Filling...
5876 samples in experience buffer. Filling...
6021 samples in experience buffer. Filling...
6169 samples in experience buffer. Filling...
6335 samples in experience buffer. Filling...
6517 samples in experience buffer. Filling...
6669 samples in experience buffer. Filling...
6810 samples in experience buffer. Filling...
6963 samples in experience buffer. Filling...
7116 samples in experience buffer. Filling...
7304 samples in experience buffer. Filling...
7525 samples in experience buffer. Filling...
7697 samples in experience buffer. Filling...
7857 samples in experience buffer. Filling...
8020 samples in experience buffer. Filling...
8203 samples in experience buffer. Filling...
8344 samples in experience buffer. Filling...
8506 samples in experience buffer. Filling...
8670 samples in experience buffer. Filling...
8819 samples in experience buffer. Filling...
9035 samples in experience buffer. Filling...
9181 samples in experience buffer. Filling...
9407 samples in experience buffer. Filling...
9635 samples in experience buffer. Filling...
9778 samples in experience buffer. Filling...
9967 samples in experience buffer. Filling...
INFO:pytorch_lightning.utilities.rank_zero:GPU available: True (cuda), used: True
INFO:pytorch_lightning.utilities.rank_zero:TPU available: False, using: 0 TPU cores
INFO:pytorch_lightning.utilities.rank_zero:HPU available: False, using: 0 HPUs
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]
INFO:pytorch_lightning.callbacks.model_summary:
  | Name         | Type | Params | Mode 
----------------------------------------------
0 | q_net        | DQN  | 667 K  | train
1 | target_q_net | DQN  | 667 K  | train
----------------------------------------------
1.3 M     Trainable params
0         Non-trainable params
1.3 M     Total params
5.336     Total estimated model params size (MB)
26        Modules in train mode
0         Modules in eval mode
Epoch 2399:                                                    40/? [00:06<00:00, 6.03it/s, v_num=0]
INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=2400` reached.
```

Check the resulting policy

```
display_video(episode=2000)
```

1:58 / 1:58

```
!zip -r /content/lightning_logs.zip /content/lightning_logs
```

```
adding: content/lightning_logs/ (stored 0%)
adding: content/lightning_logs/version_0/ (stored 0%)
adding: content/lightning_logs/version_0/hparams.yaml (deflated 39%)
adding: content/lightning_logs/version_0/events.out.tfevents.1733248631.c8211f42db4f.493.0 (deflated 72%)
adding: content/lightning_logs/version_0/checkpoints/ (stored 0%)
adding: content/lightning_logs/version_0/checkpoints/epoch=2399-step=96000.ckpt (deflated 7%)
```

```
!zip -r /content/videos.zip /content/videos
```

```
adding: content/videos/ (stored 0%)
adding: content/videos/rl-video-episode-2300.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-1200.mp4 (deflated 18%)
adding: content/videos/rl-video-episode-2200.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-500.mp4 (deflated 21%)
adding: content/videos/rl-video-episode-1800.mp4 (deflated 16%)
adding: content/videos/rl-video-episode-1200.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-0.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-2000.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-900.mp4 (deflated 17%)
adding: content/videos/rl-video-episode-600.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-1100.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-100.mp4 (deflated 20%)
adding: content/videos/rl-video-episode-1900.mp4 (deflated 15%)
adding: content/videos/rl-video-episode-1800.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-1400.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-2100.mp4 (deflated 16%)
adding: content/videos/rl-video-episode-2100.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-300.mp4 (deflated 19%)
adding: content/videos/rl-video-episode-100.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-0.mp4 (deflated 22%)
adding: content/videos/rl-video-episode-1500.mp4 (deflated 16%)
adding: content/videos/rl-video-episode-500.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-700.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-1600.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-800.meta.json (deflated 61%)
adding: content/videos/rl-video-episode-1400.mp4 (deflated 15%)
adding: content/videos/rl-video-episode-700.mp4 (deflated 21%)
```