REINFORCE

```
!apt-get install -y xvfb

!pip install \
  pygame \
  gym==0.23.1 \
  pytorch-lightning==1.6 \
  pyvirtualdisplay
```

```
!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/libtcm_debug.so.1 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/libtbbmalloc.so.2 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/libtbbbind_2_5.so.3 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/libur_loader.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_adapter_opencl.so.0 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/libhwloc.so.15 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/libtcm.so.1 is not a symbolic link
```

```
!pip install pygame
```

```
Requirement already satisfied: pygame in /usr/local/lib/python3.10/dist-packages (2.6.1)
```

```
!pip install gym==0.23.1
```

```
Collecting gym==0.23.1
  Downloading gym-0.23.1.tar.gz (626 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 626.2/626.2 kB 9.6 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) (1.26.4)
Requirement already satisfied: cloudpickle>=1.2.0 in /usr/local/lib/python3.10/dist-packages (from gym==0.23.1) (3.1.0)
Requirement already satisfied: gym_notices>=0.0.4 in /usr/local/lib/python3.10/dist-packages (from gym==0.23.1) (0.0.8)
Building wheels for collected packages: gym
  Building wheel for gym (pyproject.toml) ... done
  Created wheel for gym: filename=gym-0.23.1-py3-none-any.whl size=701372 sha256=bd3c0c6fa0c84780874f387b88779c426185c249032df9bd9e8
  Stored in directory: /root/.cache/pip/wheels/1a/00/fb/fe5cf2860fb9b7bc860e28f00095a1f42c7b726dd6f42d1acc
Successfully built gym
Installing collected packages: gym
  Attempting uninstall: gym
    Found existing installation: gym 0.25.2
    Uninstalling gym-0.25.2:
      Successfully uninstalled gym-0.25.2
Successfully installed 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 14.8 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 37.1 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 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 0x7ce7aa90da80>
```

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 RecordVideo, RecordEpisodeStatistics, \
  NormalizeObservation, NormalizeReward


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

```
def plot_policy(policy):
  pos = np.linspace(-4.8, 4.8, 100)
  vel = np.random.random(size=(10000, 1)) * 0.1
  ang = np.linspace(-0.418, 0.418, 100)
  ang_vel = np.random.random(size=(10000, 1)) * 0.1

  g1, g2 = np.meshgrid(pos, ang)
  grid = np.stack((g1,g2), axis=-1)
  grid = grid.reshape(-1, 2)
  grid = np.hstack((grid, vel, ang_vel))

  probs = policy(grid).detach().numpy()
  probs_left = probs[:, 0]

  probs_left = probs_left.reshape(100, 100)
  probs_left = np.flip(probs_left, axis=1)

  plt.figure(figsize=(8, 8))
  plt.imshow(probs_left, cmap='coolwarm')
  plt.colorbar()
  plt.clim(0, 1)
  plt.title("P(left | s)", size=20)
  plt.xlabel("Cart Position", size=14)
  plt.ylabel("Pole angle", size=14)
  plt.xticks(ticks=[0, 50, 100], labels=['-4.8', '0', '4.8'])
  plt.yticks(ticks=[100, 50, 0], labels=['-0.418', '0', '0.418'])


def test_env(env_name, policy, obs_rms):
  env = gym.make(env_name)
  env = RecordVideo(env, 'videos', episode_trigger=lambda e: True)
  env = NormalizeObservation(env)
  env.obs_rms = obs_rms

  for episode in range(10):
    done = False
    obs = env.reset()
    while not done:
      action = policy(obs).multinomial(1).cpu().item()
      obs, _, done, _ = env.step(action)
  del env


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>")
```

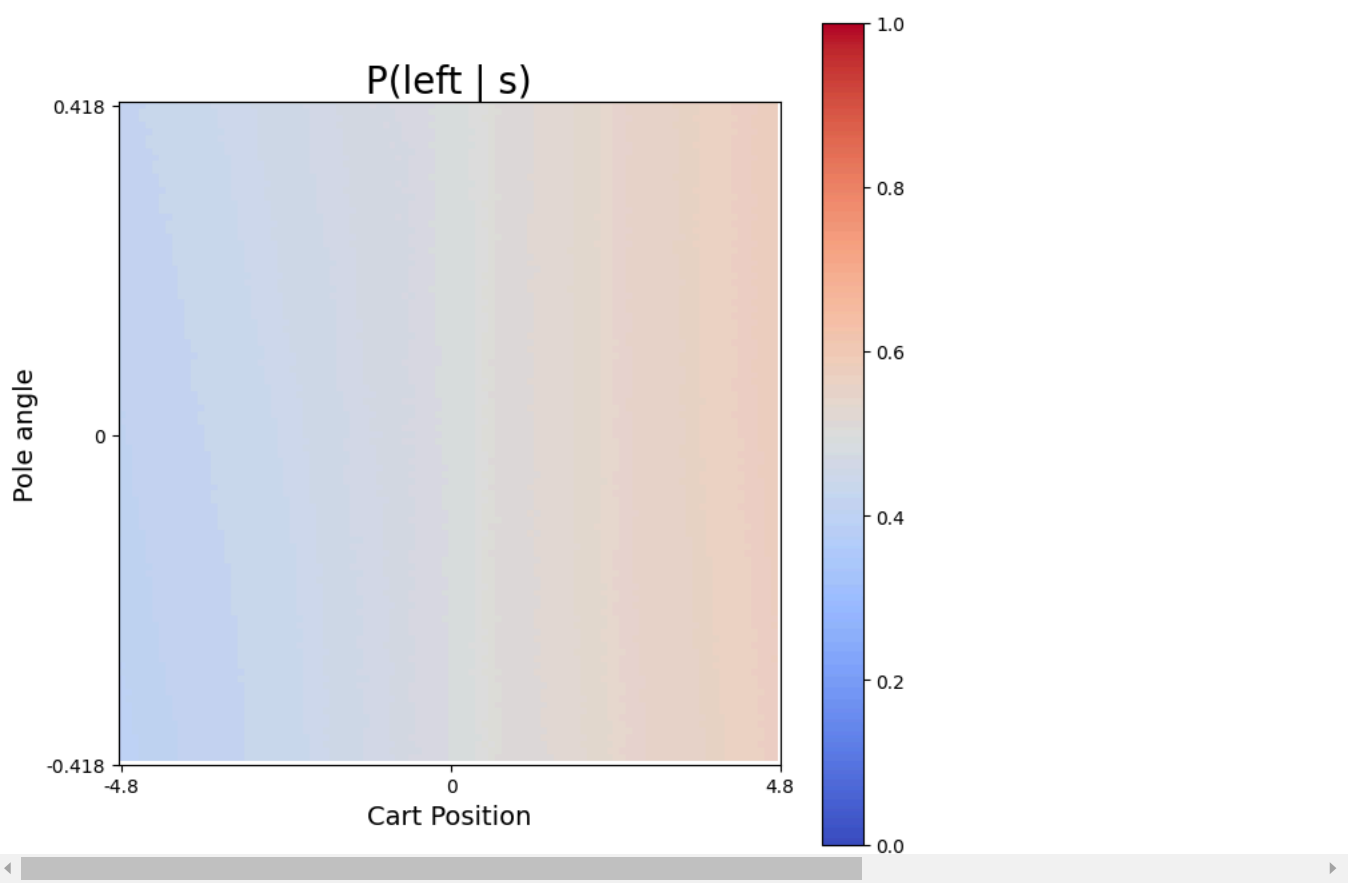
Create the policy

```
class GradientPolicy(nn.Module):

  def __init__(self, in_features, n_actions, hidden_size=128):
    super().__init__()
    self.fc1 = nn.Linear(in_features, hidden_size)
    self.fc2 = nn.Linear(hidden_size, hidden_size)
    self.fc3 = nn.Linear(hidden_size, n_actions)
```

```
  def forward(self, x):
    x = torch.tensor(x).float().to(device)
    x = F.relu(self.fc1(x))
    x = F.relu(self.fc2(x))
    x = F.softmax(self.fc3(x), dim=-1)
    return x
```

Plot the untrained policy

```
policy = GradientPolicy(4, 2)
grid = plot_policy(policy)
```




```
grid
```

Create the environment

```
env = gym.vector.make("CartPole-v1", num_envs=2)
env.reset()
env.observation_space, env.action_space
```

```
(Box([[-4.8000002e+00 -3.4028235e+38 -4.1887903e-01 -3.4028235e+38]
  [-4.8000002e+00 -3.4028235e+38 -4.1887903e-01 -3.4028235e+38]], [[4.8000002e+00 3.4028235e+38 4.1887903e-01 3.4028235e+38]
  [4.8000002e+00 3.4028235e+38 4.1887903e-01 3.4028235e+38]], (2, 4), float32),
 MultiDiscrete([2 2]))
```

```
actions = np.array([0, 0])
next_obs, rewards, dones, infos = env.step(actions)
```

```
next_obs
```

```
array([[ 0.03400568, -0.21859774,  0.01964536,  0.283147  ],
       [-0.00514356, -0.23008856,  0.03757748,  0.30924633]],
      dtype=float32)
```

```
rewards
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
  and should_run_async(code)
array([1., 1.])
```

```
dones
```

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

```
infos
```

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

```
def create_env(env_name, num_envs):
  env = gym.vector.make(env_name, num_envs=num_envs)
  env = RecordEpisodeStatistics(env)
  env = NormalizeObservation(env)
  env = NormalizeReward(env)
  return env
```

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```

Create the dataset

```
class RLDataset(IterableDataset):

  def __init__(self, env, policy, steps_per_epoch, gamma):
    self.env = env
    self.policy = policy
    self.steps_per_epoch = steps_per_epoch
    self.gamma = gamma
    self.obs = env.reset()

  @torch.no_grad()
  def __iter__(self):
    transitions = []

    for step in range(self.steps_per_epoch):
      action = self.policy(self.obs)
      action = action.multinomial(1).cpu().numpy()
      next_obs, reward, done, info = self.env.step(action.flatten())
      transitions.append((self.obs, action, reward, done))
      self.obs = next_obs

    obs_b, action_b, reward_b, done_b = map(np.stack, zip(*transitions))

    running_return = np.zeros(self.env.num_envs, dtype=np.float32)
    return_b = np.zeros_like(reward_b)

    for row in range(self.steps_per_epoch - 1, -1, -1):
      running_return = reward_b[row] + (1 - done_b[row]) * self.gamma * running_return
      return_b[row] = running_return

    num_samples = self.env.num_envs * self.steps_per_epoch
    obs_b = obs_b.reshape(num_samples, -1)
    action_b = action_b.reshape(num_samples, -1)
    return_b = return_b.reshape(num_samples, -1)

    idx = list(range(num_samples))
    random.shuffle(idx)

    for i in idx:
      yield obs_b[i], action_b[i], return_b[i]
```

Create the REINFORCE algorithm

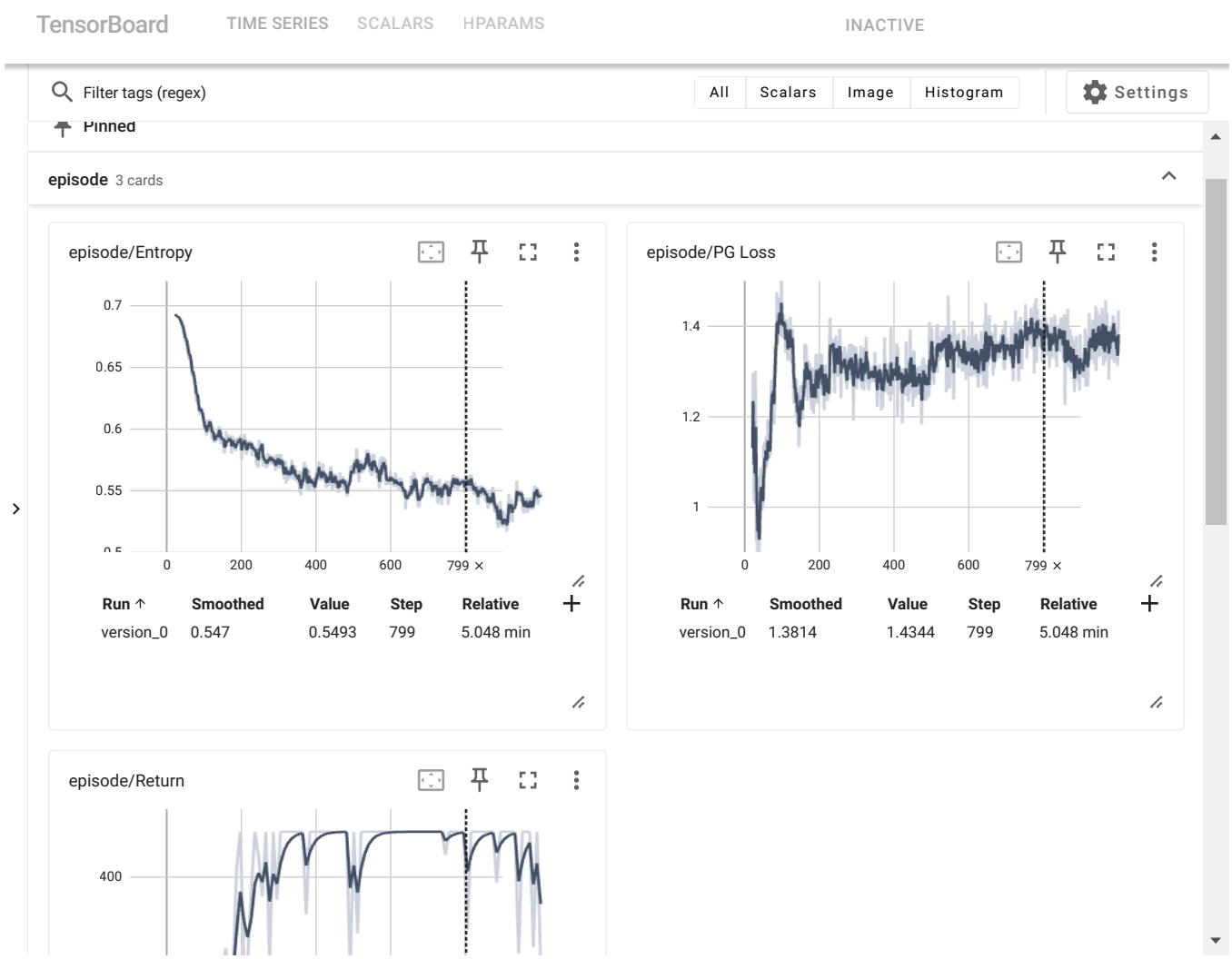
```
class Reinforce(LightningModule):

  def __init__(self, env_name, num_envs=8, samples_per_epoch=1000, batch_size=1024,
               hidden_size=64, policy_lr=0.001, gamma=0.99, entropy_coef=0.001, optim=AdamW):

    super().__init__()
```

```
    self.env = create_env(env_name, num_envs=num_envs)

    obs_size = self.env.single_observation_space.shape[0]
    n_actions = self.env.single_action_space.n

    self.policy = GradientPolicy(obs_size, n_actions, hidden_size)
    self.dataset = RLDataset(self.env, self.policy, samples_per_epoch, gamma)

    self.save_hyperparameters()

  # Configure optimizers.
  def configure_optimizers(self):
    return self.hparams.optim(self.policy.parameters(), lr=self.hparams.policy_lr)

  def train_dataloader(self):
    return DataLoader(dataset=self.dataset, batch_size=self.hparams.batch_size)

  # Training step.
  def training_step(self, batch, batch_idx):
    obs, actions, returns = batch

    probs = self.policy(obs)
    log_probs = torch.log(probs + 1e-6)
    action_log_prob = log_probs.gather(1, actions)

    entropy = - torch.sum(probs * log_probs, dim=-1, keepdim=True)

    pg_loss = - action_log_prob * returns
    loss = (pg_loss - self.hparams.entropy_coef * entropy).mean()

    self.log("episode/PG Loss", pg_loss.mean())
    self.log("episode/Entropy", entropy.mean())

    return loss

  def on_train_epoch_end(self):
    self.log("episode/Return", self.env.return_queue[-1])
```

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




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

Train the policy

```
algo = Reinforce('CartPole-v1')

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=100,
    log_every_n_steps=1
)

trainer.fit(algo)
```

```
INFO:pytorch_lightning.utilities.rank_zero:GPU available: False, used: False
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.callbacks.model_summary:
  | Name   | Type           | Params | Mode
-------------------------------------------------
0 | policy | GradientPolicy | 4.6 K  | train
-------------------------------------------------
4.6 K     Trainable params
0         Non-trainable params
4.6 K     Total params
0.018     Total estimated model params size (MB)
4         Modules in train mode
0         Modules in eval mode
Epoch 98:                                                                                                  8/? [00:03<00:00,  2.25it/s, v_num=0]
INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=100` reached.
```

Check the resulting policy

```
test_env('CartPole-v1', algo.policy, algo.env.obs_rms)
```

```
display_video(episode=1)
```

0:02 / 0:08

Plot the trained policy

```
plot_policy(algo.policy)
```