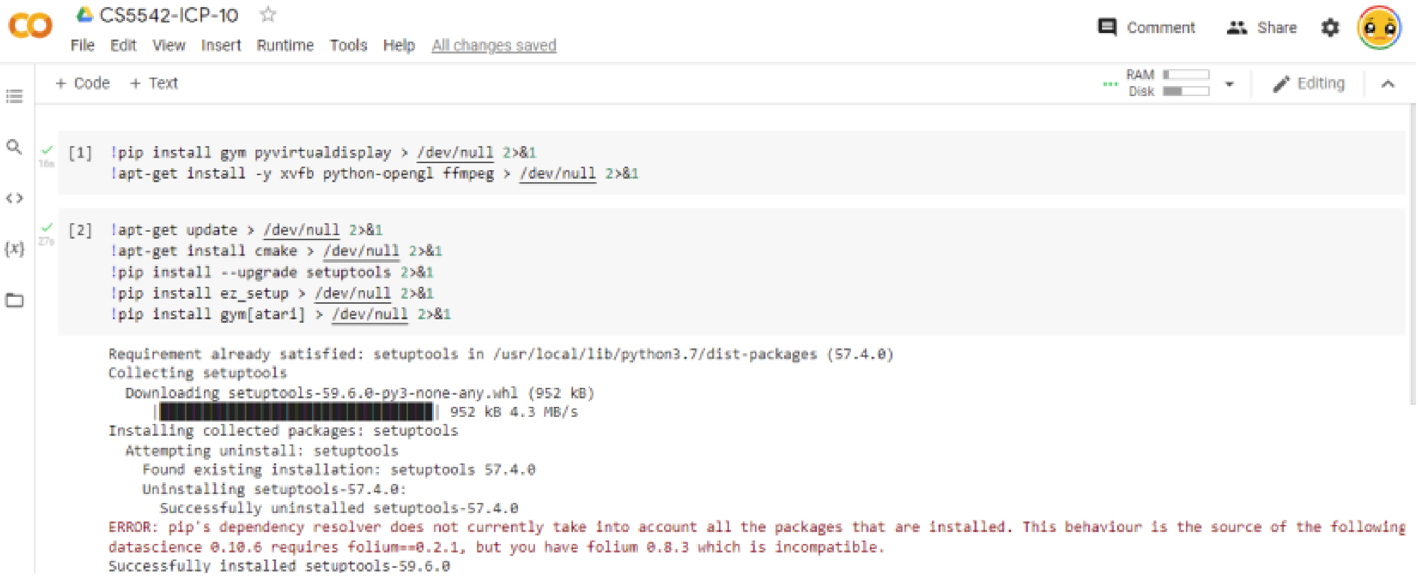
ICP 10

Edit New Page Jump to bottom

JaydenT4864 edited this page 12 minutes ago · 2 revisions

Code

Setup all libraries and access the game simulator

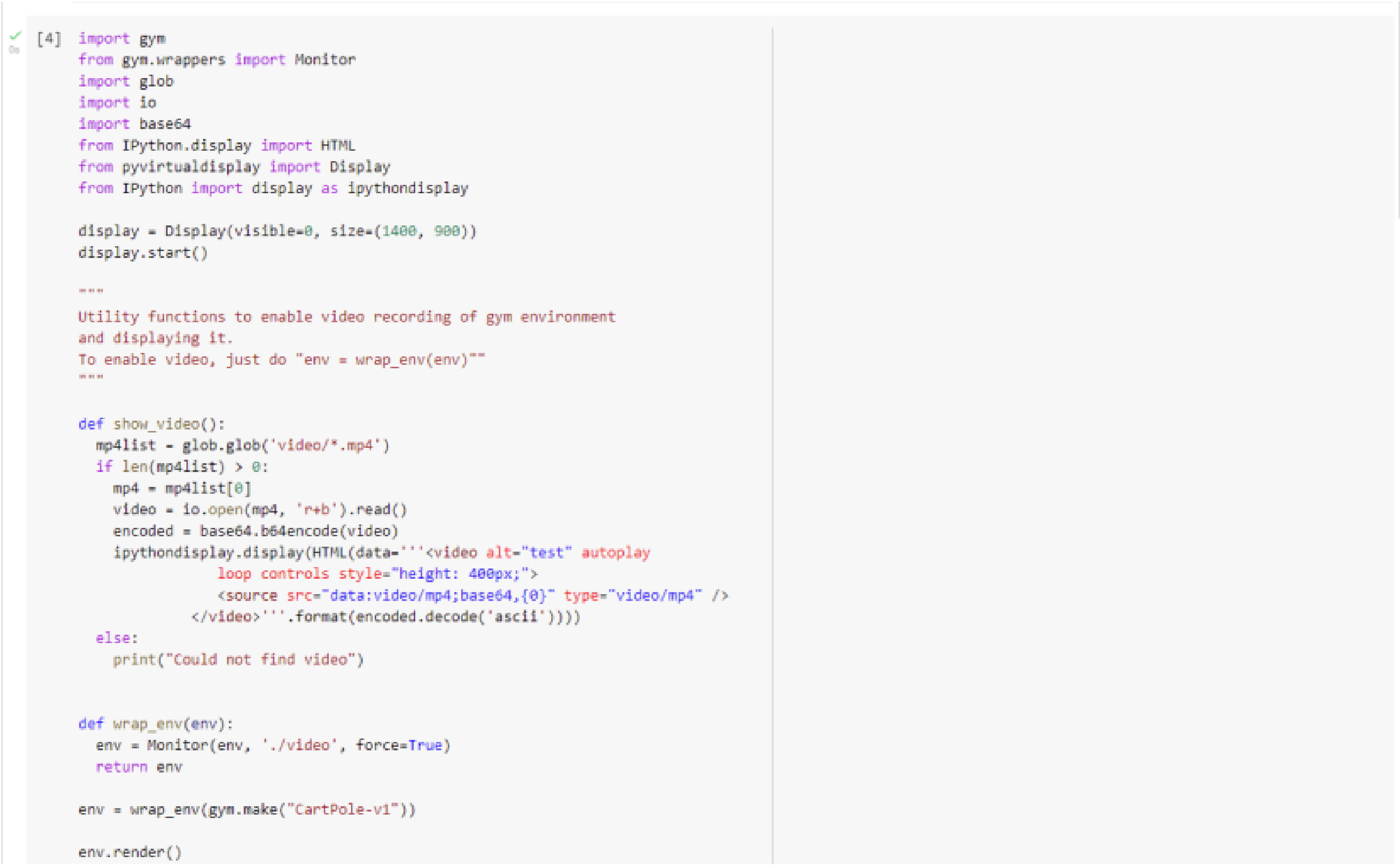


```
[1]  !pip install gym pyvirtualdisplay > /dev/null 2>&1
     !apt-get install -y xvfb python-opengl ffmpeg > /dev/null 2>&1
```

```
[2]  !apt-get update > /dev/null 2>&1
     !apt-get install cmake > /dev/null 2>&1
     !pip install --upgrade setuptools 2>&1
     !pip install ez_setup > /dev/null 2>&1
     !pip install gym[atari] > /dev/null 2>&1
```

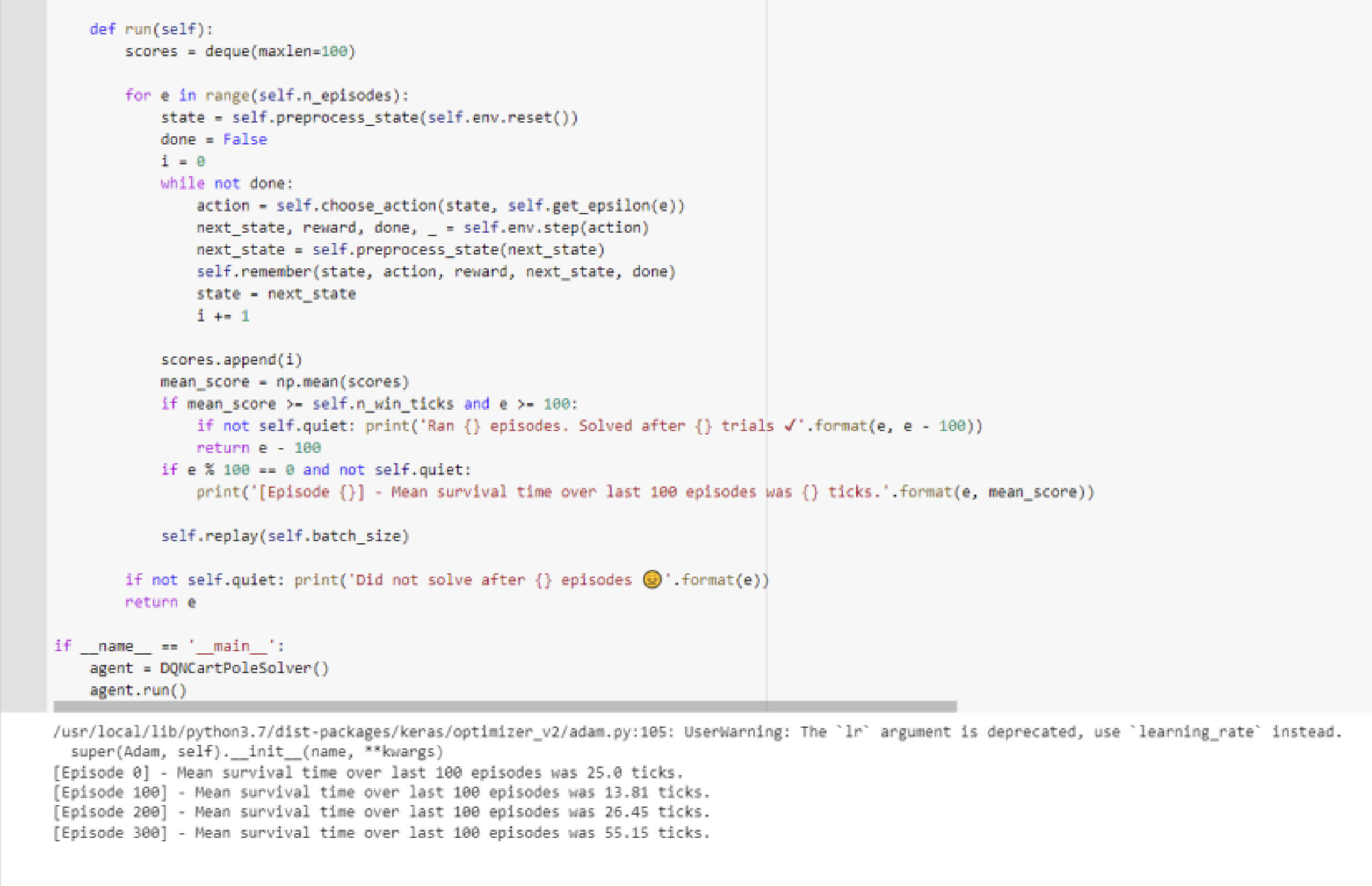
```
Requirement already satisfied: setuptools in /usr/local/lib/python3.7/dist-packages (57.4.0)
Collecting setuptools
  Downloading setuptools-59.6.0-py3-none-any.whl (952 kB)
     |████████████████████████████████| 952 kB 4.3 MB/s
Installing collected packages: setuptools
  Attempting uninstall: setuptools
    Found existing installation: setuptools 57.4.0
    Uninstalling setuptools-57.4.0:
      Successfully uninstalled setuptools-57.4.0
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following
datascience 0.10.6 requires folium==0.2.1, but you have folium 0.8.3 which is incompatible.
Successfully installed setuptools-59.6.0
```

Define the DQN model.



```
import gym
from gym.wrappers import Monitor
import glob
import io
import base64
from IPython.display import HTML
from pyvirtualdisplay import Display
from IPython import display as ipythondisplay

display = Display(visible=0, size=(1400, 900))
display.start()

"""
Utility functions to enable video recording of gym environment
and displaying it.
To enable video, just do "env = wrap_env(env)""
"""

def show_video():
  mp4list = glob.glob('video/*.mp4')
  if len(mp4list) > 0:
    mp4 = mp4list[0]
    video = io.open(mp4, 'r+b').read()
    encoded = base64.b64encode(video)
    ipythondisplay.display(HTML(data='''<video alt="test" autoplay
                loop controls style="height: 400px;">
                <source src="data:video/mp4;base64,{0}" type="video/mp4" />
             </video>'''.format(encoded.decode('ascii'))))
  else:
    print("Could not find video")


def wrap_env(env):
  env = Monitor(env, './video', force=True)
  return env

env = wrap_env(gym.make("CartPole-v1"))

env.render()
```

This result took me more than 30 minutes and it still processing. I just stop it and create this report. A tick show for the time of the game runs before solved.



```
    def run(self):
        scores = deque(maxlen=100)

        for e in range(self.n_episodes):
            state = self.preprocess_state(self.env.reset())
            done = False
            i = 0
            while not done:
                action = self.choose_action(state, self.get_epsilon(e))
                next_state, reward, done, _ = self.env.step(action)
                next_state = self.preprocess_state(next_state)
                self.remember(state, action, reward, next_state, done)
                state = next_state
                i += 1

            scores.append(i)
            mean_score = np.mean(scores)
            if mean_score >= self.n_win_ticks and e >= 100:
                if not self.quiet: print('Ran {} episodes. Solved after {} trials ✓'.format(e, e - 100))
                return e - 100
            if e % 100 == 0 and not self.quiet:
                print('[Episode {}] - Mean survival time over last 100 episodes was {} ticks.'.format(e, mean_score))

            self.replay(self.batch_size)

        if not self.quiet: print('Did not solve after {} episodes 😞'.format(e))
        return e

if __name__ == '__main__':
    agent = DQNCartPoleSolver()
    agent.run()
```

```
/usr/local/lib/python3.7/dist-packages/keras/optimizer_v2/adam.py:105: UserWarning: The `lr` argument is deprecated, use `learning_rate` instead.
  super(Adam, self).__init__(name, **kwargs)
[Episode 0] - Mean survival time over last 100 episodes was 25.0 ticks.
[Episode 100] - Mean survival time over last 100 episodes was 13.81 ticks.
[Episode 200] - Mean survival time over last 100 episodes was 26.45 ticks.
[Episode 300] - Mean survival time over last 100 episodes was 55.15 ticks.
```

Conclusion

This is the most difficult in all ICPs. I am happy that the Google Colab did not mess with the OpenAi environment. However, I dont think this result correct or not. Although, it is very interesting to work with something that I am not similar. I did not have any OpenAI environment in the past, or the game simulation. So this is my first experience with this OpenAI environment.

+ Add a custom footer

Pages 11

Find a Page...

Home

ICP 1

ICP 10

Code

Conclusion

- ICP 2
- ICP 3
- ICP 4
- ICP 5
- ICP 6
- ICP 7
- ICP 8
- ICP 9

+ Add a custom sidebar

Clone this wiki locally

```
https://github.com/JaydenT4864/CS5542---Big-Data.wiki.git
```