

Förstärkningsinlärning med Unity ML Agents: Python-kommandon och parametrar för inträning och statistik

KONFIGURATIONSFIL:

Inlärningsalgoritmen (PPO) som sköts av Python API:t behöver en konfigurationsfil. Olika exempel kan laddas ner härifrån: <https://github.com/Unity-Technologies/ml-agents/tree/main/config/ppo>. De olika konfigurationsfilerna hänger ihop med olika exempel som kan laddas ner. För enkla exempel fungerar *Basic.yaml* riktigt bra till en början!

I konfigurationsfilen kan olika parametrar för nervernätet och inlärningen definieras. Här de viktigaste parametrarna och några riktgivande råd:

summary_freq = Efter hur många steg (1 steg = 1 handling) som statistik visas på konsolen

max_steps = Maximalt antal steg vid inträningen (default är 500 000 vilket inte räcker till i alla situationer)

batch_size = Om Action type (i behaviour parameter skriptet) är diskret bör värdet vara mellan 32 och 512. Om Action type (i behaviour parameter skriptet) är continuous bör värdet vara mellan 512 och 5120

buffer_size = Bör vara flera ggr större än batch_size (2048 – 409600)

hidden_units = Mellan 32 och 512. För enkla inträninga bör värdet vara lågt. För mer avancerade inträningsproblem bör värdet vara högt

num_layers = Mellan 1 och 3. Default är 2. För enkla problem lågt värde, för komplexa problem högt värde

PYTHON-KOMMANDON FÖR INTRÄNING:

Träna genom att köra programmet/spelet i Unity-editorn:

```
mlagents-learn pathToYamlfile --run-id="MyAi"
```

Samma som ovan men skriver över MyAi-foldern om den redan finns

```
mlagents-learn pathToYamlFile --run-id="MyAi" --force
```

Träna genom att köra programmet/spelet utanför Unity, dvs som ett standalone program. Spelet startas i ett 512x512 pixlars fönster och med en "speed up" på 10x. Kräver att man skapat en .exe fil av spelet i Unity editorn via File->Build Settings -> Build. *pathToExeFile* nedan är sökstigen till .exe filen efter "build"

```
mlagents-learn pathToYamlFile --run-id="MyAi" --env=pathToExeFile --width=512 --height=512 --time-scale= 10
```

Samma som föregående men skapar 10 instanser av spelet/programmet för att försnabba och stabilisera inläringen

```
mlagents-learn pathToYamlFile --run-id="MyAi" --env=pathToExeFile --width=512 --height=512 --time-scale= 10 --num-envs=10
```

Samma som föregående men kör inläringen utan att rendera någon grafik ("går snabbare och rekommenderas om man är säker på att spelet/inläringen är buggfri")

```
mlagents-learn pathToYamlFile --run-id="MyAi" --env=pathToExeFile --no-graphics --time-scale= 10 --num-envs=10
```

VISUALISERING AV TRÄNINGSTATISTIK

Kör följande kommando i Anaconda Powershell Prompt där *result* är sökstigen till katalogen där träningsdata sparats med *mlagents-learn*

```
tensorboard --logdir results --port 6006
```

Öppna sedan en webbläsare och surfa på localhost:6006

