howest
/ we develop people

Research Project & Bachelor Proef

Zelfrijdende AI met routebeschrijving

Vergelijking van modellen

Team Carlier Alex

Module Research Project

Richting Multimedia and Creative Technology (MCT)

Schooljaar 2021-2022

1 – Vergelijking

1.1 – Vergelijking – PPO

1.1.1 – Vergelijking – PPO – beta

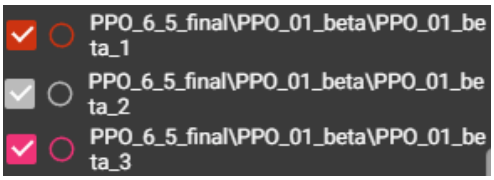
De waarde van beta controleert de ratio van exploratie (hoog) VS exploitatie (laag).

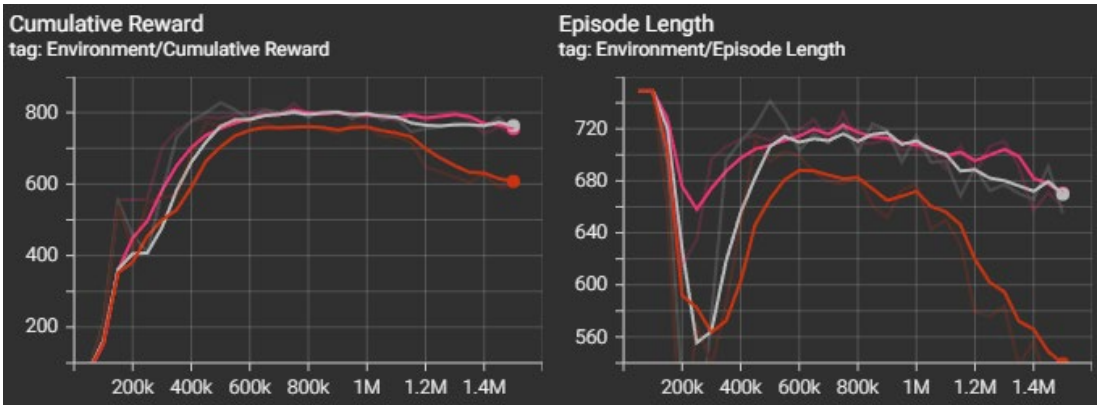
beta: 3.0e-4 beta: 5.0e-4 beta: 7.0e-4

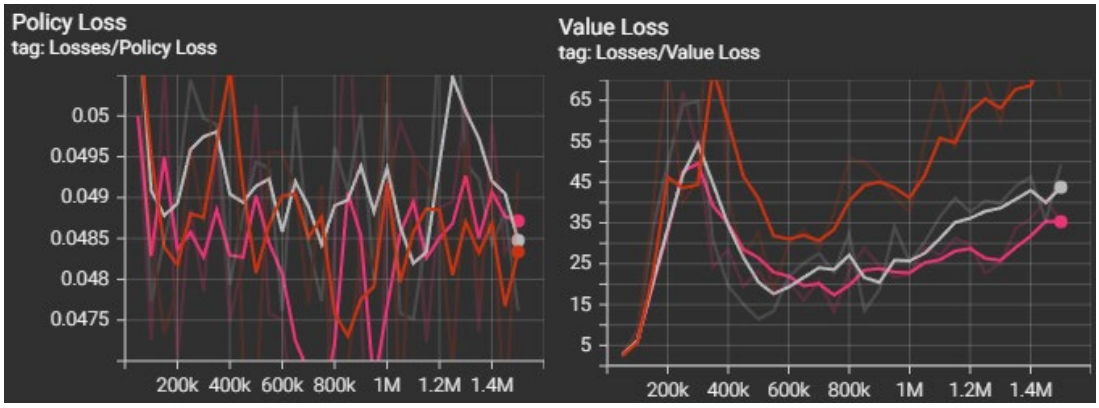
Bij hogere exploratie komt de agent meer diverse scenario's tegen, maar de behavior lijkt ook minder sterk getraind te zijn (meer onzekerheid in het nemen van acties) en vertoond soms ongewenst/random gedrag. Kleinere beta waarde heeft meer consistent gedrag, maar leid tot minder variërend gedrag die niet met meerdere scenario's om kan.

Merkwaardig is dat de uitwijking van het pad groter wordt bij een grotere beta waardes. Daarnaast kan in de grafieken waargenomen worden dat deze ook tot hogere rewards leiden. Dit wijst er op een fout mogelijke in de reward signalen die dit gedrag aanmoedigt.

Minder exploratie (3.0e-4) was meer voordelig voor de behavior, alhoewel rewards lager vallen.







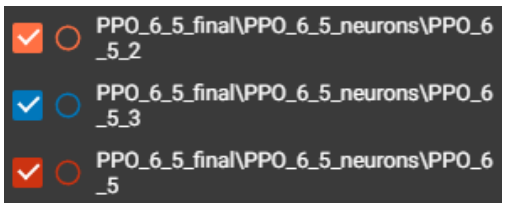
1.1.2 – Vergelijking – PPO – hidden units

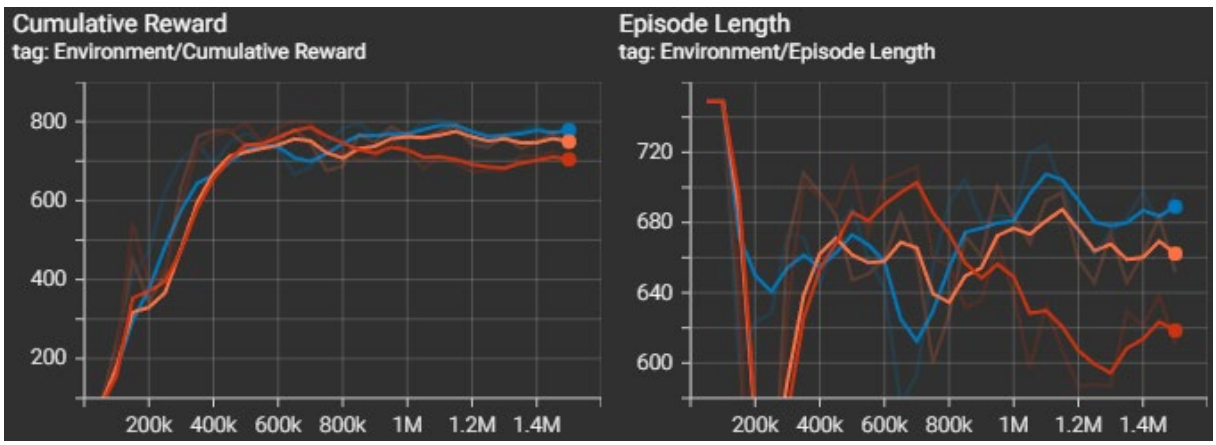
Het aantal neurons/hidden units per laag heeft een invloed hoe complex de behavior van een getraind model kan zijn.

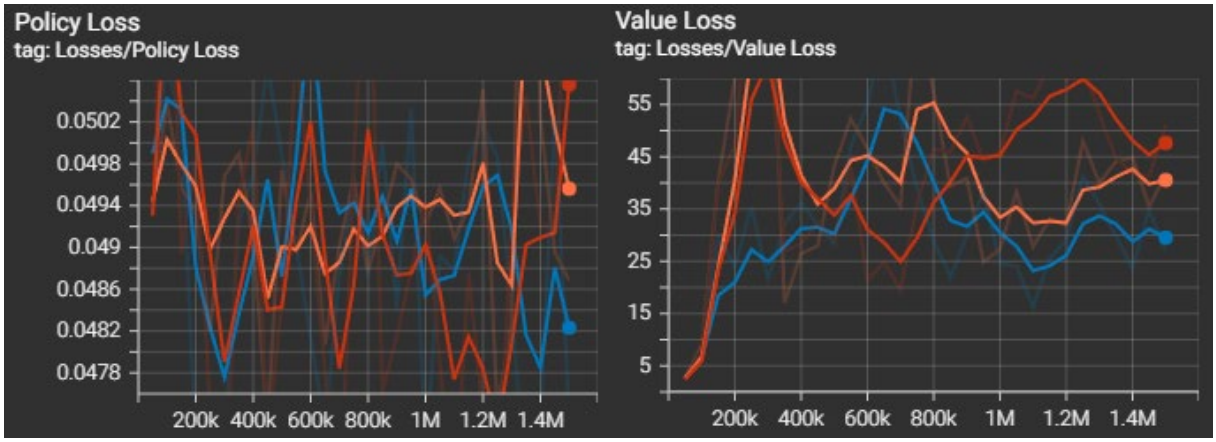
hidden_units: 64 hidden_units: 128 hidden_units: 256

In dit scenario zorgde een hoger aantal neurons per laag voor complexer gedrag en een beter begrip van de omgeving, maar leidde ook tot bepaalde ongewenste karakteristieken in het model zoals te sterk uitwijken van het pad, waarna de agent doorheeft dat hij niet meer op tijd kan draaien voor de checkpoint en komt stil te staan om negatieve rewards te voorkomen. Grotere netwerken vergen ook meer training.

Kleinere waardes zorgen voor meer simplistisch gedrag wat meer gewenst is gezien de specifieke omgeving. Er zit niet veel verschil in de rewards en het simpelere model kan sneller het einde bereiken.





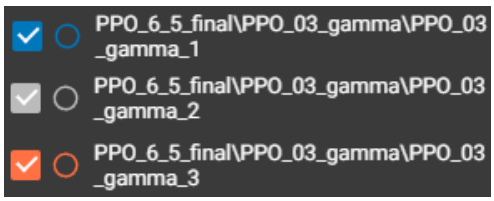


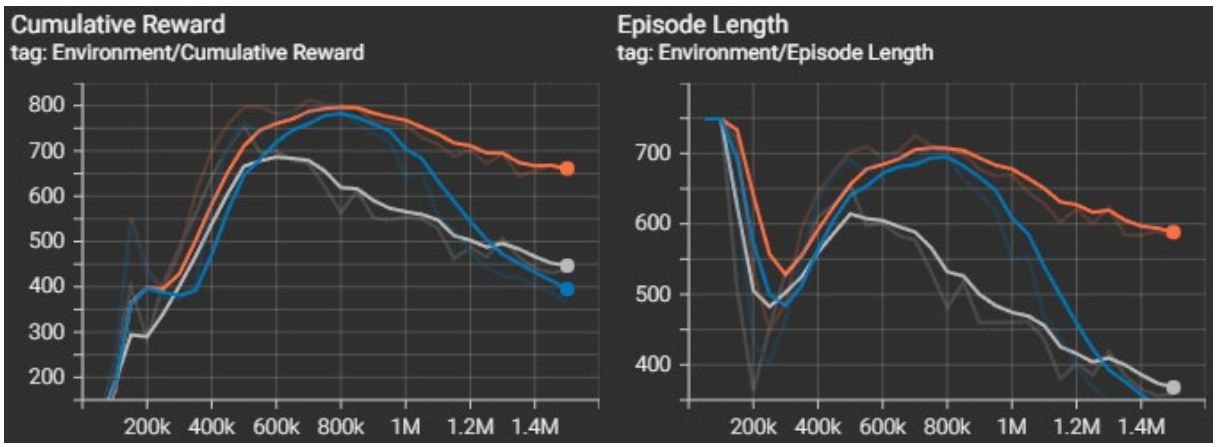
1.1.3 – Vergelijking – PPO – gamma

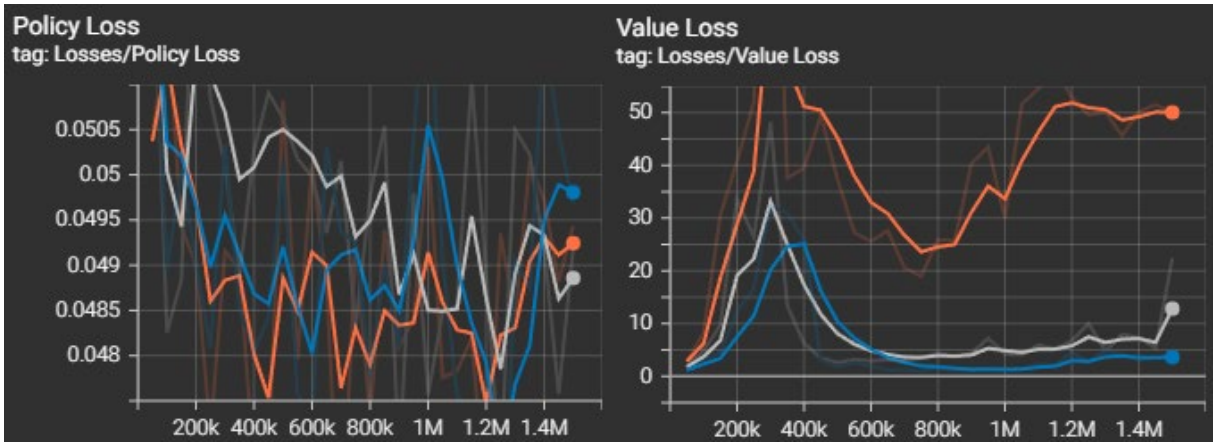
Gamma is een expressie van de zekerheid van de rewards in te toekomst. Des te groter, des te meer zekerheid.

gamma: 0.9 gamma: 0.95 gamma: 0.99

Bij een hogere gamma reageert de agent soms onvoorspelbaar. De ene keer rijd het traag, de andere keer gaat het zo snel als mogelijk vooruit en mist de bocht bijna. Een kleinere gamma heeft meer moeite met toekomstige situaties. De gemiddelde gamma heeft de meest stabiele en graduele training. Het heeft ook de meest voordelige behavior en kan hoeken correct inschatten





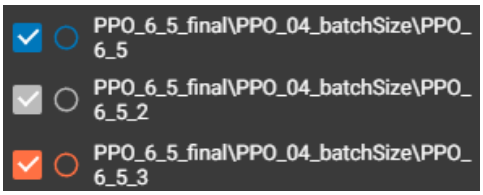


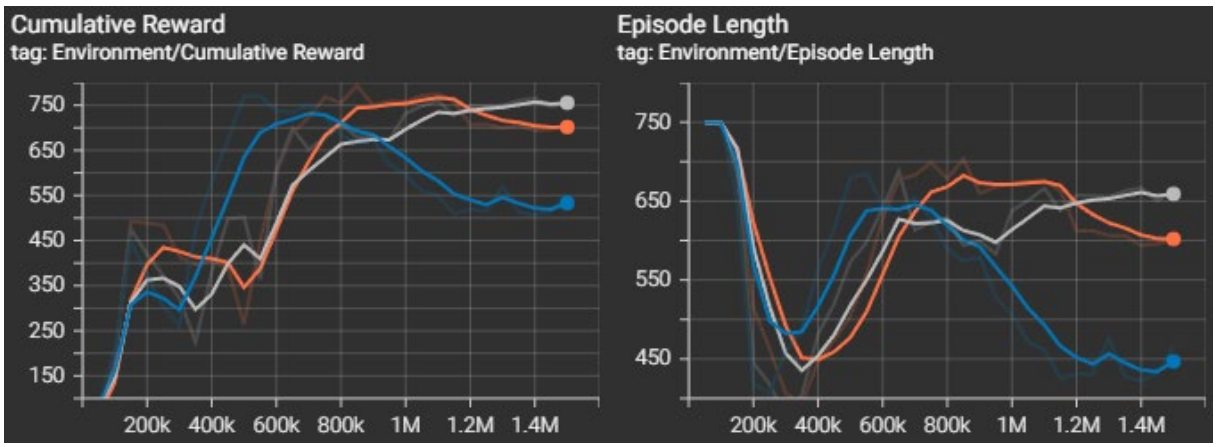
1.1.4 – Vergelijking – PPO – batch size

De batch size geeft aan hoeveel samples per update worden gebruikt vanuit de buffer. Grotere batch sizes leiden tot stabielere training.

batch_size: 256 batch_size: 512 batch_size: 1024

Grotere batches leidde tot meer variërend gedrag, met soms ook negatieve gevolgen (te veel uitgeweken, traag rijden, grote bochten, checkpoints missen, ..), maar de agent kan wel betere rewards halen. De batch size van 256 heeft een lagere cumulatieve reward en een hogere loss op de policy, maar kan in sommige situaties beter presteren.





1.2 – Vergelijking – SAC

1.2.1 – Vergelijking – SAC – batch size

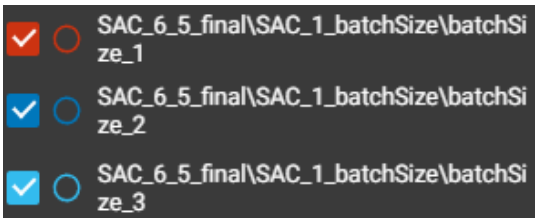
De batch size geeft aan hoeveel samples per update worden gebruikt vanuit de buffer. Grotere batch sizes leiden tot stabielere training.

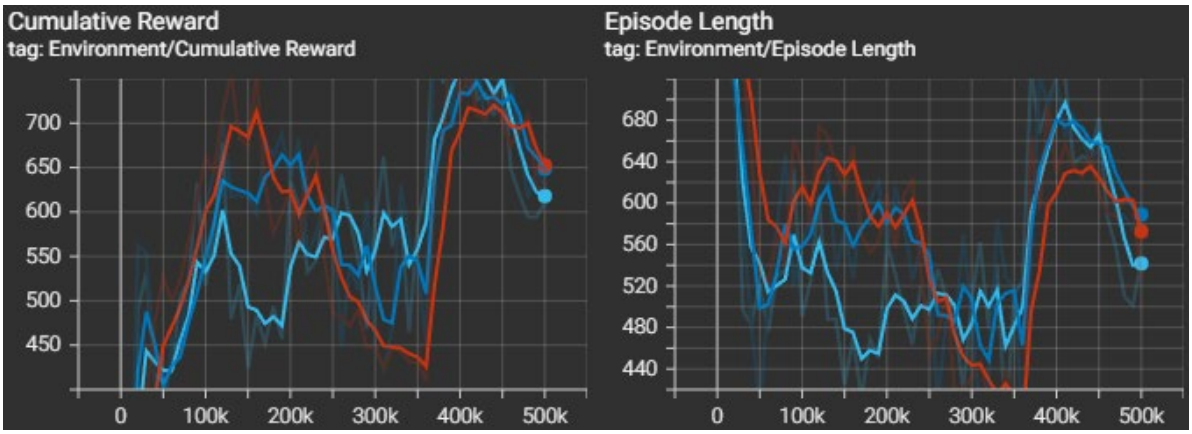
batch_size: 512 batch_size: 1024 batch_size: 2048

In deze environment is het niet wenselijk om een te grote of te kleine batch size te hebben.

- te laag ➔ niet genoeg samples gezien waardoor het model simplistisch gedrag vertoond, zwakker is getraind en niet beseft wanneer het moet draaien
- te hoog aantal ➔ meer variërende acties, maar kan in slechte situaties terechtkomen.

Alhoewel een grotere batch size de environment beter interpreteert, leert het ook negatieve acties. Dit lijkt op overfitting op bepaalde reward signalen. In een betere omgevingsopstelling zou een grotere batch voordeliger kunnen zijn, maar momenteel werkt 1024 het beste.





1.2.2 – Vergelijking – SAC – hidden units

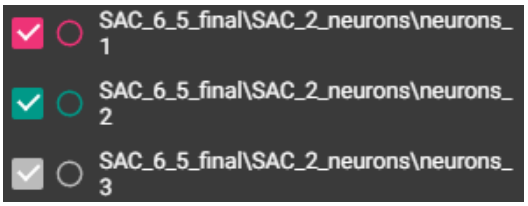
Het aantal hidden units/neurons per laag bepaalt hoe complex het model kan zijn. Des te meer neurons, des te meer training is vereist en kan leiden tot onverwacht gedrag. Maar te weinig neurons kan leiden tot simpele acties met negatieve gevolgen op lange termijn.

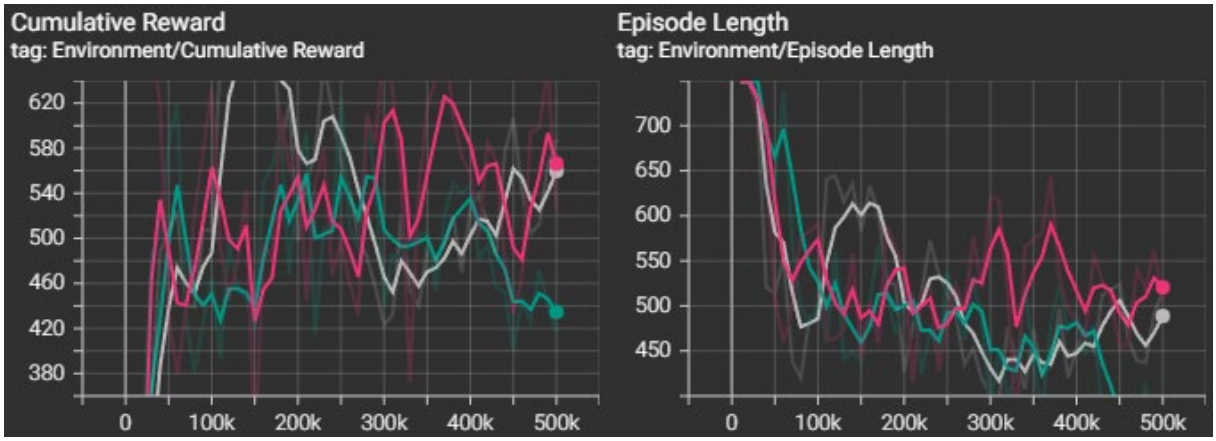
hidden_units: 32 hidden_units: 64 hidden_units: 128

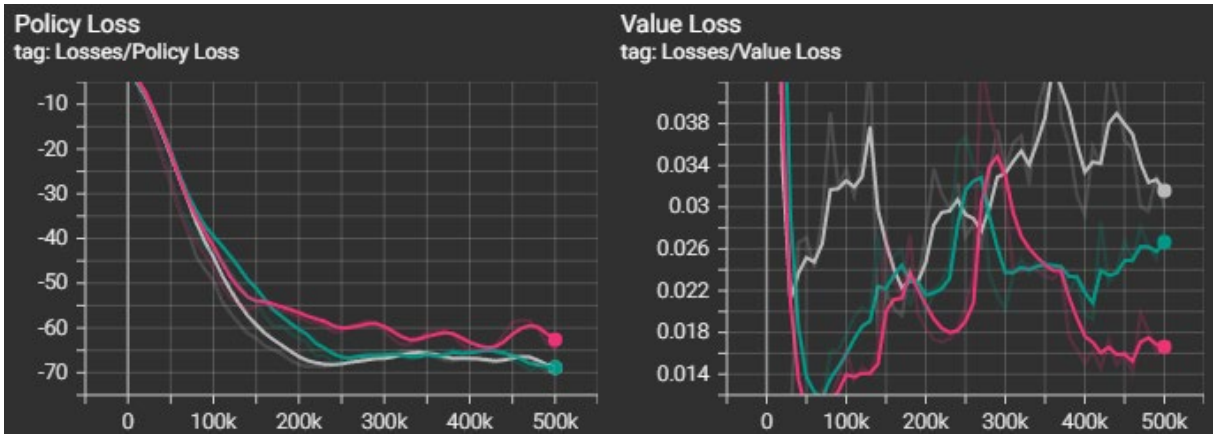
Bij 32 neurons zien we simplistisch gedrag waarbij de environment niet voldoende kan worden geïnterpreteerd om complexere bewegingen te maken.

128 neurons lijkt ook wel voordelig gedrag te vertonen, maar faalt vaker door meer complexe acties te nemen (zoals sterk uitwijken voor checkpoints wat leid tot een terminal step bij de volgende).

64 hidden units per laag vertoond het meest gebalanceerde gedrag. Het kan vlot navigeren en is minder geneigd om random acties te nemen, alhoewel hij soms nog faalt.







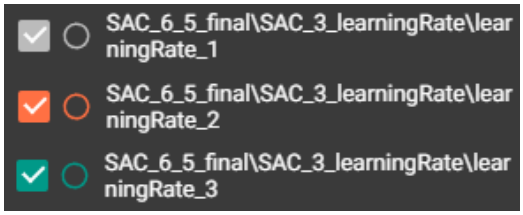
1.2.3 – Vergelijking – SAC – learning rate

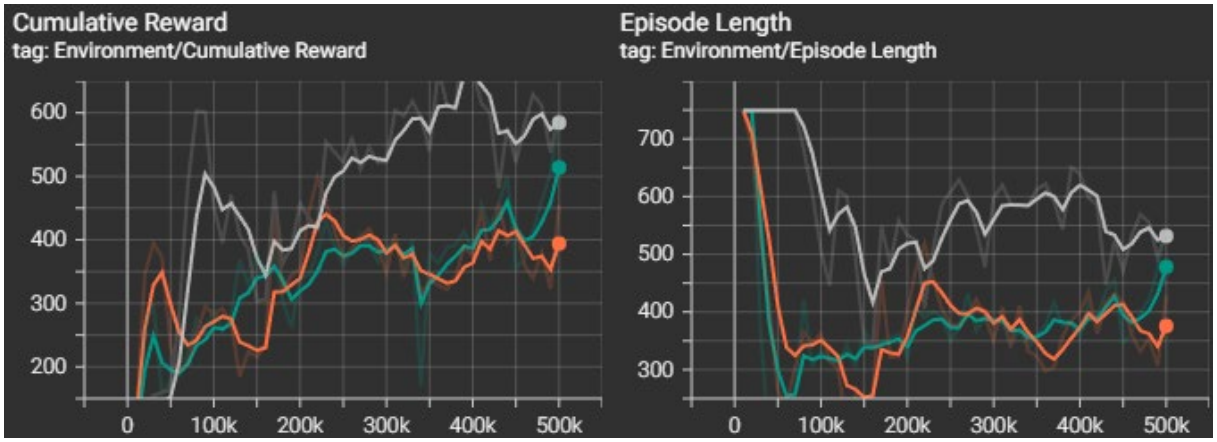
De learning rate geeft aan hoe sterk de gradient descent is tijdens het updaten van het model (aka hoe sterk het model wordt geüpdatet). De learning rate waarde verlagen helpt bij onstabiele training, maar training zal trager verlopen.

learning_rate: 1.0e-4 learning_rate: 3.0e-4 learning_rate: 5.0e-4

De hogere waarde heeft wel sterkere training ondergaan en zal sterker reageren op de observaties. Maar vanaf een learning rate van 3.0e-4 kan de agent slechter presteren doordat het meer gefocust is op onmiddellijke observaties en rewards dan het verloop van de episode over een langere termijn. Dit zorgt voornamelijk dat de agent soms te sterk naar links/rechts stuurt en bochten niet volledig meer kan halen.

De learning rate van 1.0e-4 heeft de meest stabiele behavior en heeft betere scores.





1.2.4 – Vergelijking – SAC – init entcoef

De initiële entropy coëfficiënt determineert hoeveel exploratie gebeurt bij het begin van de training. Een hogere waarde resulteert in meer exploratie.

init_entcoef: 0.1 init_entcoef: 0.3 init_entcoef: 0.5

Meer exploratie leidde tot meer variatie in de behavior, maar kan soms te snel rijden en zijn hoeken missen, terwijl een kleinere waarde resulteert in minder afwijkingen van het gedrag, maar ook zorgt voor minder flexibiliteit.