Introduction

For assignment 1 we chose to train an AI on doom using vizdoom which develops the AI using only visual information. The two scenarios we chose were “defend the line” and “deadly corridor”. The two personalities being used to “beat” the levels are “survival” and “kill_fast”.

Personalities

Survival

This personality is focused on surviving. The characteristics we wanted to emulate with it include moving around and to not get hit. We did this by rewarding it for being alive, moving around, and regaining health, and penalizing it for being idle, taking damage, and dying.

Kill_fast

This personality is more so focused on killing. It’s designed to be aggressive and focused on quick kills, consistently engaging the enemy and overall efficient combat. This was done by rewarding it for dealing damage, killing and having enemies visible on screen while it was only penalized for time to discourage idling.

Scenarios

Defend the line

This scenario’s main goal is to survive the onslaught of monsters as long as possible.

Kill_fast

The kill_fast personality performs exceptionally well in this scenario, due to its parameters it takes care of the enemies as soon as possible allowing it to survive longer. The parameter that makes the bot prioritize having an enemy in its visuals allows the bot to dodge incoming attacks and return fire appropriately.

Survival

Does not perform well in this situation. Due to its parameters it ends up wall hugging which ends up hindering its ability to stay alive since in the scenario for it to effectively “defend the line” it can only move left and right.

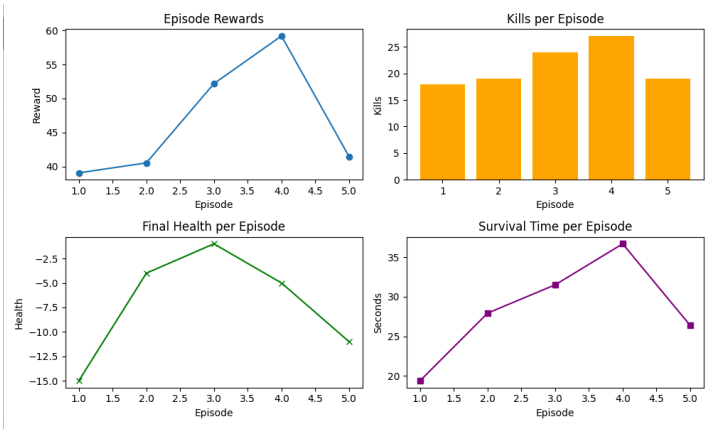
Experiments

1 algo, 2 personalities

The main focus is to see how each personality completes the level with the same hparams.

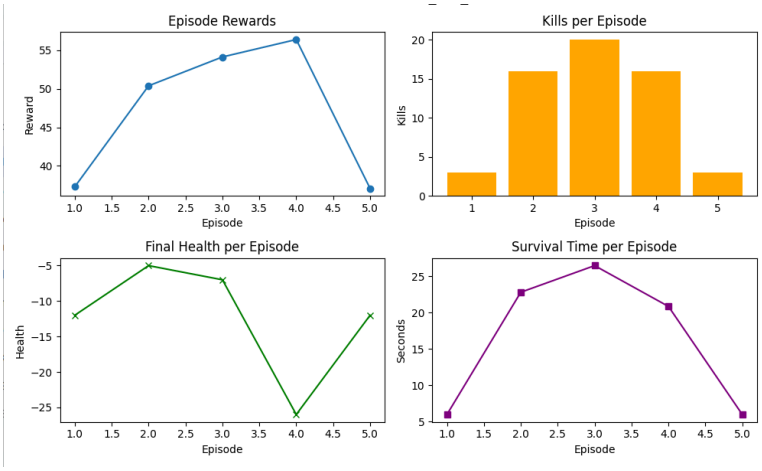
```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=3.5e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.025,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

kill_fast



Across the episodes for kill_fast the AI was able to survive for 20+ seconds, its kills ranged from 15 - 28 and overall was able to keep itself alive due to the reward pushing it to focus on kills and the AI was able to last longer in situations it got more kills.

Survival



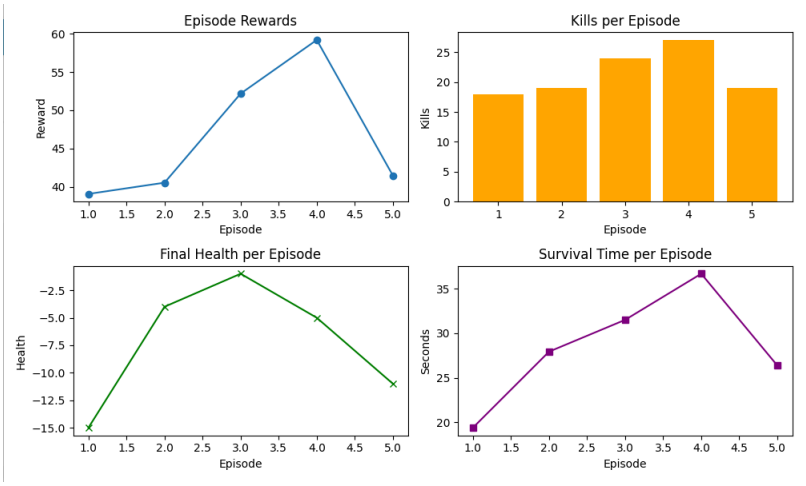
Survival had a harder time due to it getting overwhelmed with the amount of enemies and not being able to retaliate well enough leading it to meet its end early on. The Ai didn't exhibit any noteworthy strategies either.

1 persona, vast hparam changes

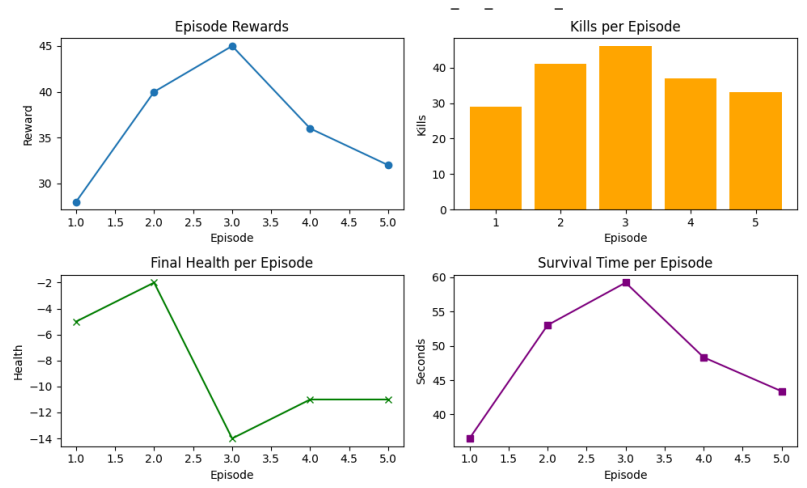
This experiment focuses on how hparam changes affect a personality, the kill_fast personality was chosen due to its compatibility with the scenario and overall success.

kill_fast

```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=3.5e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.025,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```




```python
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=1e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.01,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```



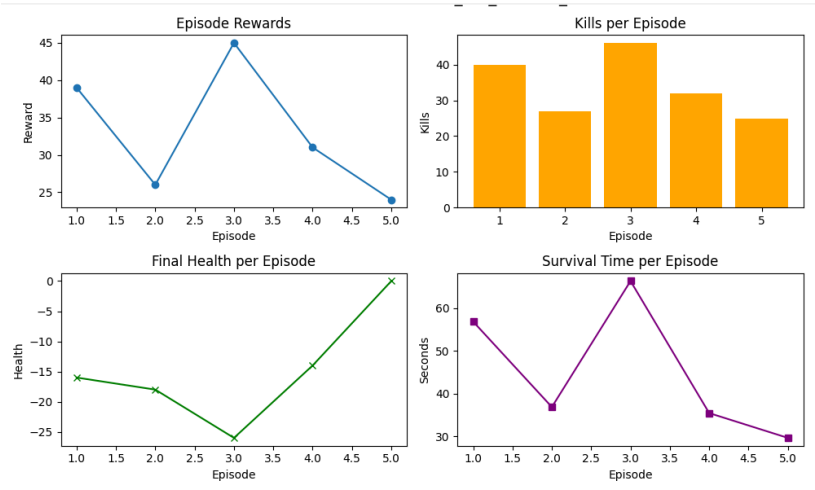
Overall the second set of hparams which had a lower learning rate and a lower ent_coef had a much better performance garnering more kills, and surviving much longer than the first set, this is due to the fact that the smaller hparams allow the AI to not be so random and not “overreact to every little reward signal” allowing it to perform much better than its higher counterpart.

Personalities on recommended hparams for kill_fast

This experiment was to see how both personalities on the recommended hparams of kill_fast would fare. This main value that was increased was the learning rate which helps them adapt to new kill strategies and find the fastest way to get kills.

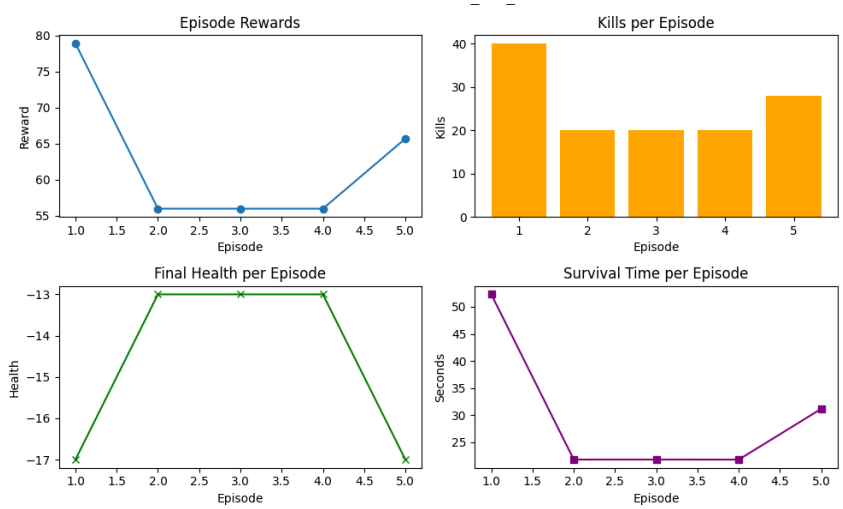
```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=6e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.02,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

Kill_fast



For the Kill_fast personality it adopted a strategy of staying at the enemies spawn point to get fast kills but this leads to it getting attacked by the enemies from the other spawn points which end up damaging the Ai until it dies.

Survival



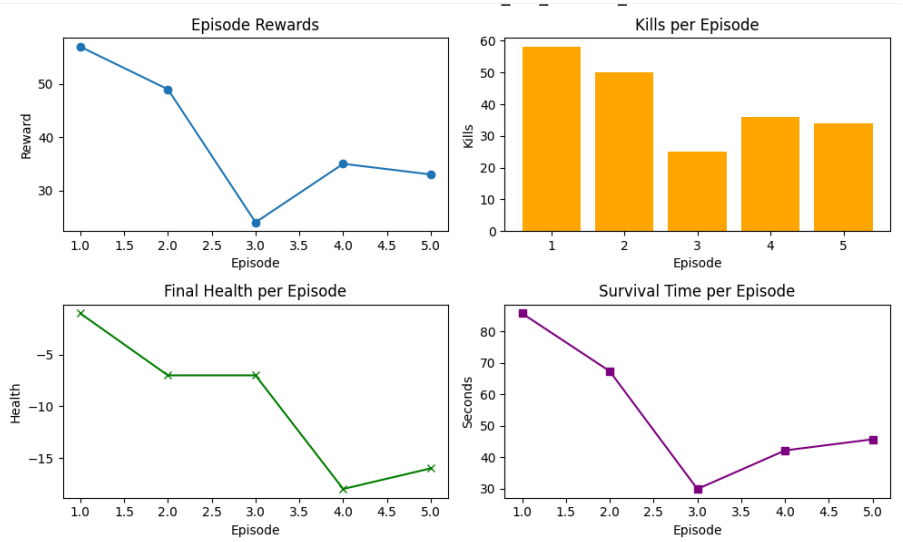
Survival had a strong start with the first episode, being able to survive for 50+ seconds but in later episodes started to hug the walls leading to the enemies spawning in to overwhelm it and eventually kill it.

Personalities on recommended hparams for survival

This experiment was to see how both personalities on the recommended hparams of survival would fare. These hparams were set to maximize longevity and minimize reckless aggression, which is done with low exploration and a lower learning making it create strategies that would promote slow gameplay.

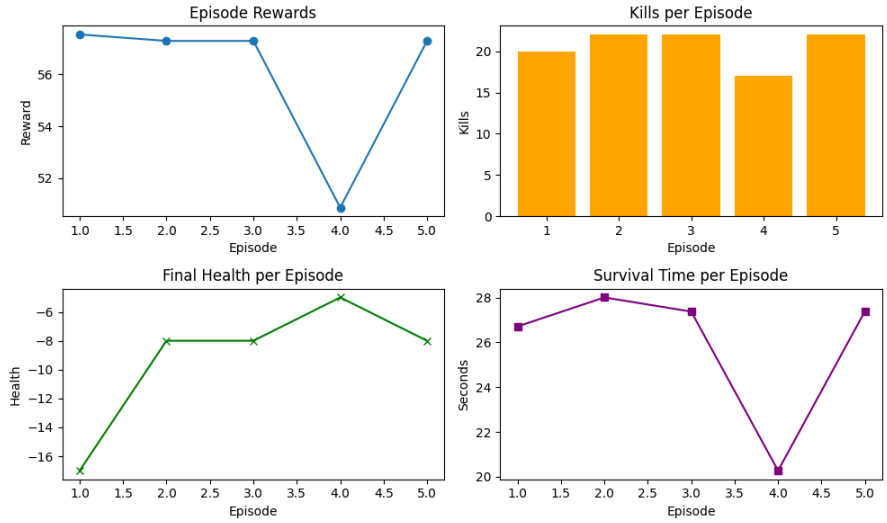
```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=2.5e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.005,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

Kill_fast



Due to the low learning rate the Ai was able to survive much longer and get much more kills, the low learning rate allowing it to experiment more and not get stuck in strategies that will end up letting it get cornered and killed.

Survival



The survival Ai had consistent results, due to the scenario and the controls allowed it wasn't able to shine as well as it could.

Deadly Corridor

This scenario's main goal is to get to the end of the corridor. The corridor is filled with sections that contain two enemies each that attack simultaneously. The AI does struggle to "beat" this level due to its demanding playstyle and the technical ability needed overall to beat it.

Kill_fast

The kill_fast personality does not perform well in this scenario. Since the scenario also requires the AI to move forward, it ends up making the AI focus on one side letting the enemies on the other side damage it and eventually kill it.

Survival

Performs better in this scenario since it's more focused on getting to the end of the hallway rather than killing the enemies.

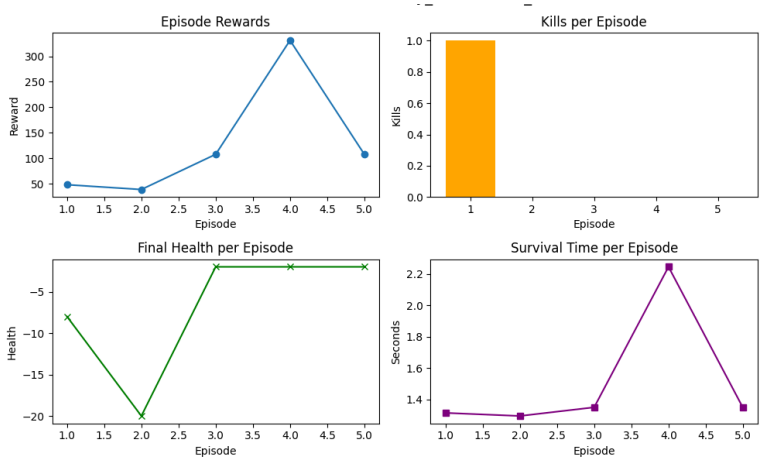
Experiments

The same experiments that were done for Defend the line.

1 algo, 2 personalities

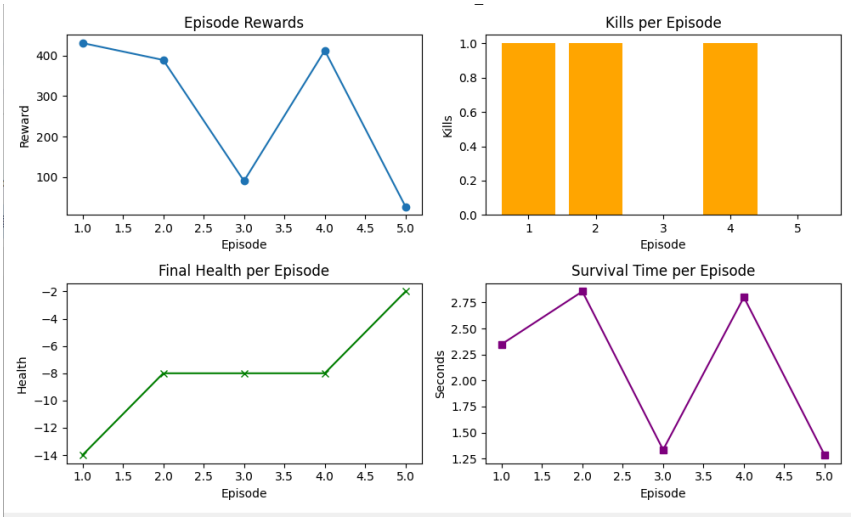
```
# Create PPO model
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=3.5e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.025,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

Kill_fast



Since kill_fast focused more on killing then it did on getting to the end it did not make it through once and wasn't able to react to 2 enemies in close proximity in time. It focused more on one side and was then either killed before it could shoot or get one kill before getting killed.

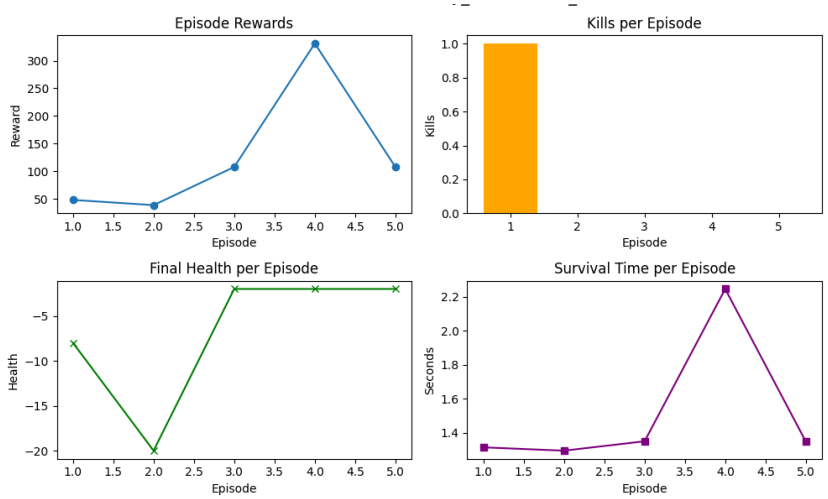
Survival



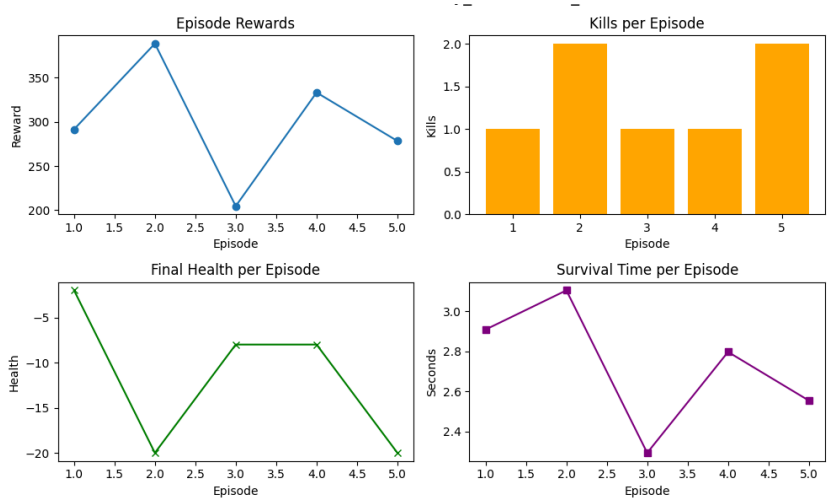
Survival was able to last longer since it was able to get past a few of the enemies but its strategy is reliant on luck which ends up failing most of the time. It was able to turn towards the enemy but since it was only focused on one side the other side was able to damage it.

1 persona, vast hparam changes

```python
# Create PPO model
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=3.5e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.025,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```




```python
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=1e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.005,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

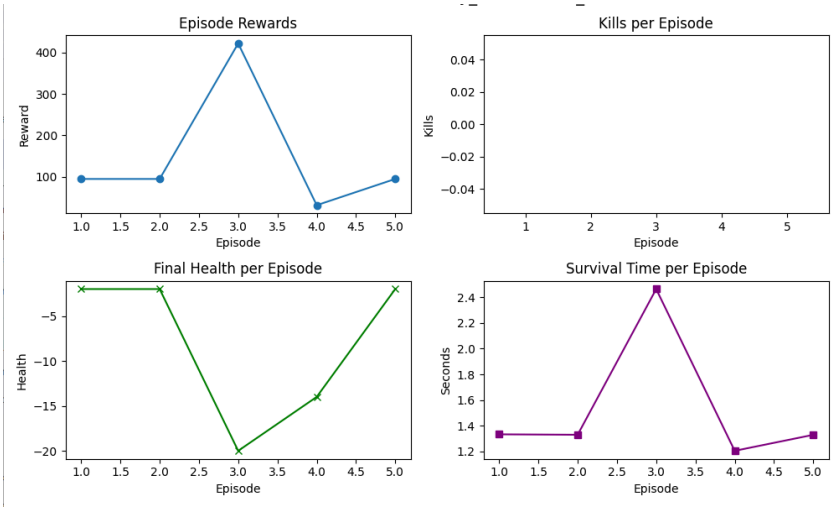


The personality chosen was kill_fast, with the lower hparams it performed better due to it not being so aggressive with the objectives. Since the Ai is also being rewarded for its distance it tends to rush to the end which ends up getting it killed.

Personalities on recommended hparams for kill_fast

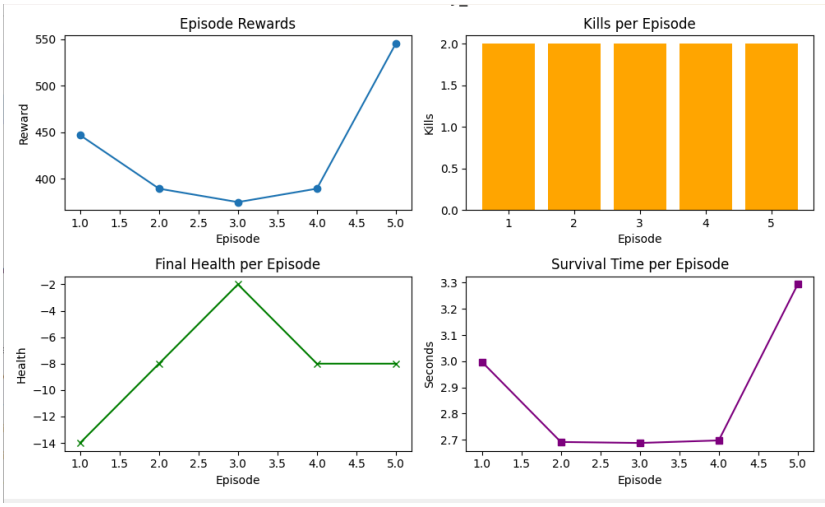
```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=6e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.02,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

Kill_fast



The kill_fast personality was only focused on moving forward. This may be because of the high learning rate and it being rewarded for distance. Looking at the graph for episode rewards it reaches over 400 in scenario three without any kills which means it was only being rewarded for distance and the distance it covered gave it the boost.

Survival

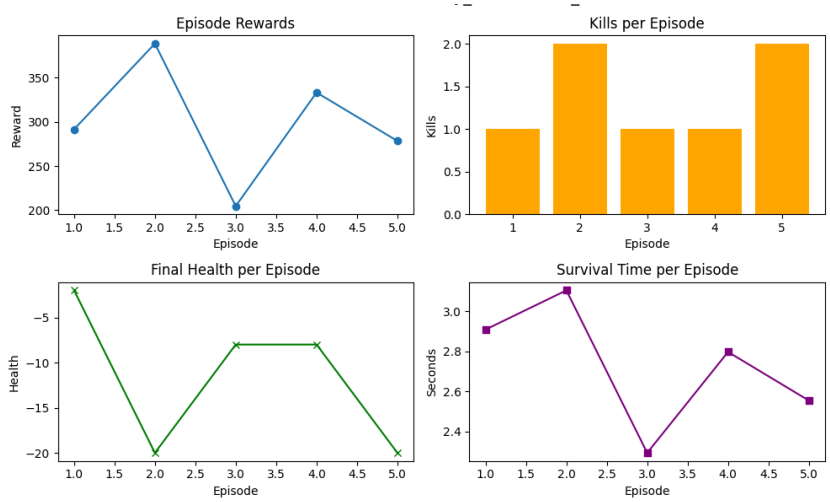


The survival personality surprisingly got more kills than kill_fast personality and this is because of the reward parameters. Survival is being rewarded not only for surviving but also moving around and in this scenario it can move forward and turn which lets it face the enemy and when it sees the enemy it attacks it.

Personalities on recommended hparams for survival

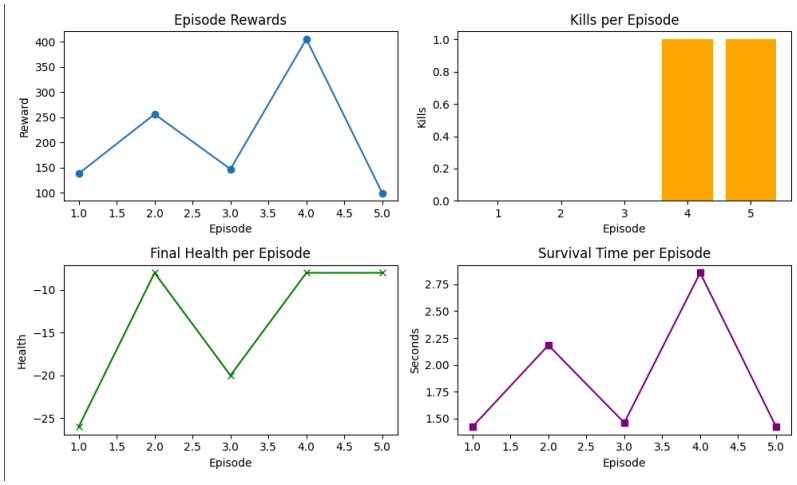
```
model = PPO(
    policy="CnnPolicy",
    env=env,
    verbose=1,
    tensorboard_log=f"./logs_vizdoom/{args.scenario}",
    n_steps=1024,
    batch_size=64,
    learning_rate=1e-4,
    gamma=0.99,
    clip_range=0.2,
    ent_coef=0.005,  # encourages exploration
    seed=args.seed,  # ensures deterministic initialization
)
```

Kill_fast



Kill_fast was able to perform well but wasn't able to make it to the end, due to it being overwhelmed with the 2 enemies in each chamber. It was more so focused on going forward and keeping an eye on the left side which is where the kills come from as well.

survival



In overall distance the Ai performs better but due to it being less aggressive in general compared to kill_fast, it doesn't go for the enemies and ends up taking damage over time and dying. Getting to the end is possible but very rare.