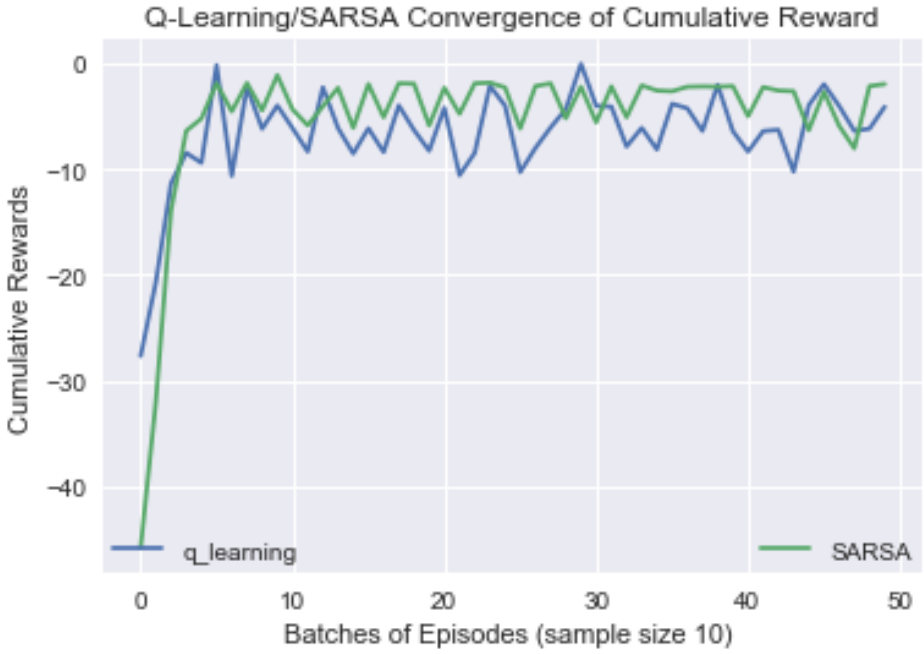
# Q-Learning/SARSA Reinforcement Learning

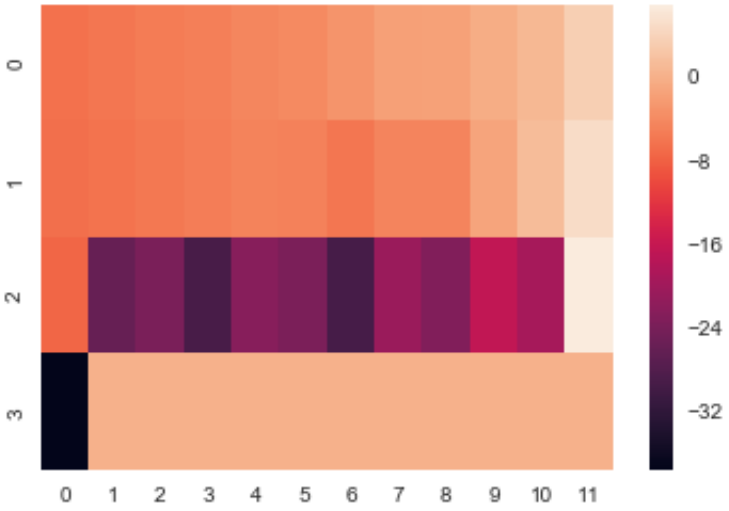
## Reward Convergence



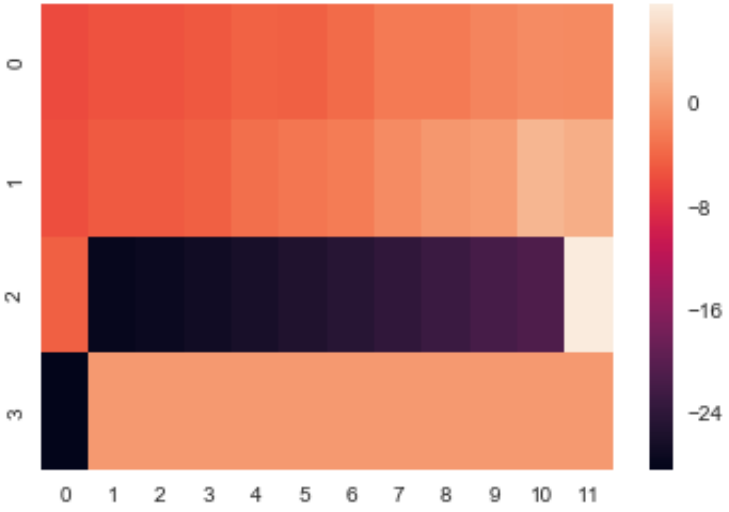

- Results obtained training the policy network with 500 episodes of consecutive games.
- Learning rate chosen as 0.5 in order to prevent fast convergence.
- Discount factor chosen as 0.9 indicates amount of stress our agent last upon importance of future rewards
- Epsilon 0.1 indicates explore/exploit ratio agent takes %10 random actions and fallows the policy %90 when it needs to take an action.
- Visualization obtained by sampling the consecutive episodes into groups of batches which have 10 episodes.
- Summed up cumulative rewards of episodes within the batch and applied normalization to the batch.
- SARSA seems to slightly outperform q-learning in terms of reward convergence. Further investigation of number of policy updates in order to reach the goal should be evaluated.

**HeatMaps Generated:**

**SARSA:**




**Q-learning:**




- **COMMENTS**
- Even though, state to HeatMap

Mapping might not be accurate it illustrates properties of SARSA and Q-learning algorithms on-policy/off-policy properties. SARSA is an on-policy learning algorithm where q-value depends on action performed by current policy and not of the greedy policy.

- SARSA evaluates state-action values together to determine value of the state. On the other hand q-learning decides based on looking at the state value alone to determine which action to take.
- The resulting affect is that q-learning learns the optimal policy which moves along the cliff but random exploration causes it to fall off. Thus, incurring higher penalties. On the other hand SARSA finds a safer but not the optimal path which is further away from the cliff.
- Decreasing epsilon value shows faster convergence and can be implemented. It's intuitively makes sense since once policy learned (environment dynamics) random exploration can be decreased.

**Visualizing paths taken with Q-learning and SARSA**

**Notation**

**1's indicate visited cells**
**0's indicate not visited cells**

Agent trained with SARSA after 500 episodes

```
[[1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]]
```

Agent trained with Q-learning after 500 episodes

```
[[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]]
```

**Graphs visualizing number of iterations taken to complete the episode.**

Expected:
SARSA should complete the episode by taking larger number of steps because it follows a safer path away from cliff.

NOTE: Graph obtained by applying standard normal normalization and grouping episodes by batches of 10.