Open-ended Deep Reinforcement Learning for a Bitcoin Trading Bot

Adrian Altermatt

IEEE
Advancing Technology for Humanity

Bitcoin trading

- **24/7 Market**: Unlike traditional stock markets, Bitcoin trading operates 24/7, allowing continuous trading opportunities without any closing hours.
- **Volatility**: Bitcoin is known for its high volatility, with significant price swings that can occur within short periods, offering both opportunities and risks for traders.
- **Technical Analysis**: Traders often use technical analysis tools and indicators, such as moving averages, relative strength index (RSI), and Fibonacci retracement levels, to make informed trading decisions.
- **Market Sentiment**: Market sentiment, driven by news, social media, and public perceptions, plays a significant role in Bitcoin's price movements and trading activity.

https://www.pexels.com/photo/close-up-shot-of-bitcoins-5980585/

Rainbow Agent [1]

- **DoubleQ-learning:** Adding a Target Network that is used in the loss function and upgrade once every tau steps. [2]
- **Distributional RL** : Approximating the probability distributions of the Q-values instead of the Q-values themself. [3]
- **Dueling Networks**: Divide neural net stream into two branches, an action stream and a value stream. Both of them combined formed the Q-action values. [4]
- **Multi-step learning** : Making Temporal Difference bigger than classic DQN (where TD = 1). [5]

```
ainbow:
__init__(self,
    nb_states,
    nb_actions,
    gamma,

    replay_capacity,
    learning_rate,
    batch_size,
    epsilon_function = lambda episode, step : max(0.001, (1 - 5E-5)** step),
    # Model buildes
    window = 1, # 1 = Classic , 1> = RNN
    units = [32, 32],
    dropout = 0,
    adversarial = False,
    noisy = False,
    # Double DQN
    tau = 500,
    # Multi Steps replay
    multi_steps = 1,
    # Distributional
    distributional = False, nb_atoms = 51, v_min= -200, v_max= 200,
    # Prioritized replay
    prioritized_replay = False, prioritized_replay_alpha =0.65, prioritized_re
    # Vectorized envs
    simultaneous_training_env = 1,
    train_every = 1,
    name = "Rainbow",
):
self.name = name
self.nb_states = nb_states
```


[1] Hessel, Matteo, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, und David Silver. „Rainbow: Combining Improvements in Deep Reinforcement Learning“. arXiv, 6. Oktober 2017. https://doi.org/10.48550/arXiv.1710.02298.

[2] Hasselt, Hado van, Arthur Guez, und David Silver. „Deep Reinforcement Learning with Double Q-learning“. arXiv, 8. Dezember 2015. https://doi.org/10.48550/arXiv.1509.06461.

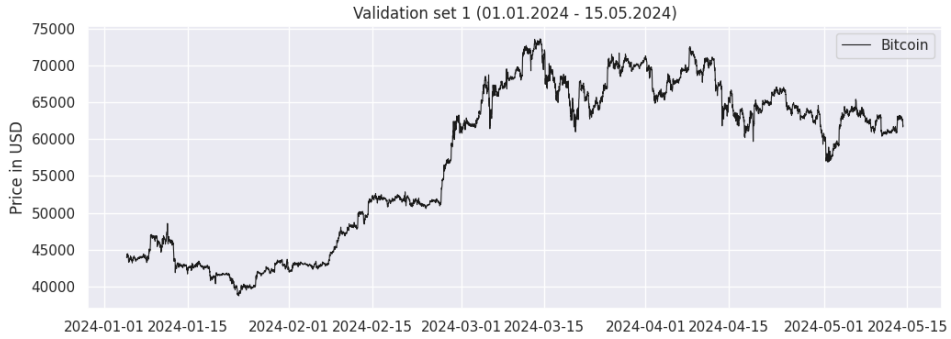
[3] Bellemare, Marc G., Will Dabney, und Rémi Munos. „A Distributional Perspective on Reinforcement Learning“. arXiv, 21. Juli 2017. https://doi.org/10.48550/arXiv.1707.06887.

[4] Freitas, Nando de. „Dueling Network Architectures for Deep Reinforcement Learning“, Nr. arXiv:1511.06581 (2015). https://www.cs.ox.ac.uk/publications/publication10201-abstract.html.

[5] De Asis, Kristopher, J. Fernando Hernandez-Garcia, G. Zacharias Holland, und Richard S. Sutton. „Multi-step Reinforcement Learning: A Unifying Algorithm“. arXiv, 11. Juni 2018. https://doi.org/10.48550/arXiv.1703.01327.

Validation sets

- Training data
  - 01.01.2019 – 31.12.2023
- Validation set 1
  - 01.01.2024 – 15.05.2024
  - Bullish market
- Validation set 2
  - 01.12.2017 – 01.12.2018
  - Bearish market



Base Bot

- Trained for 20 hours.
- Input Window of 15.
- Two actions.
  - 0: Buy or Hold USD
  - 1: Buy or Hold Bitcoin

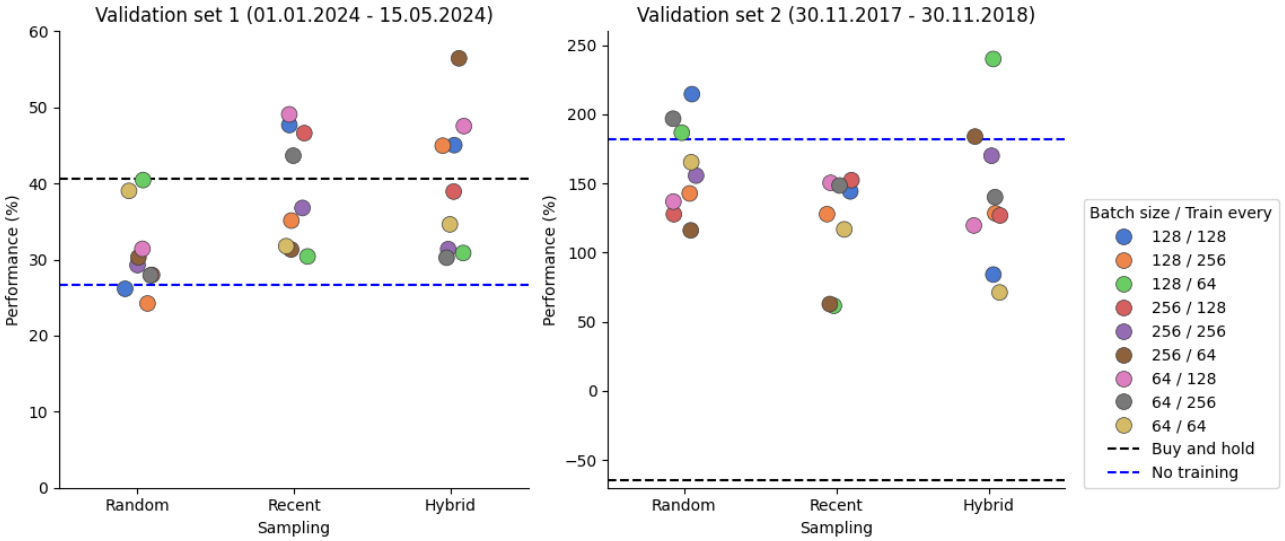
- https://github.com/ClementPerroud/Rainbow-Agent

Open-ended Learning

- Also known as:
  - Continuous Learning
  - Lifelong Learning
  - Incremental Learning
- Self improving
- Adaptive

Comparison of Sampling methods

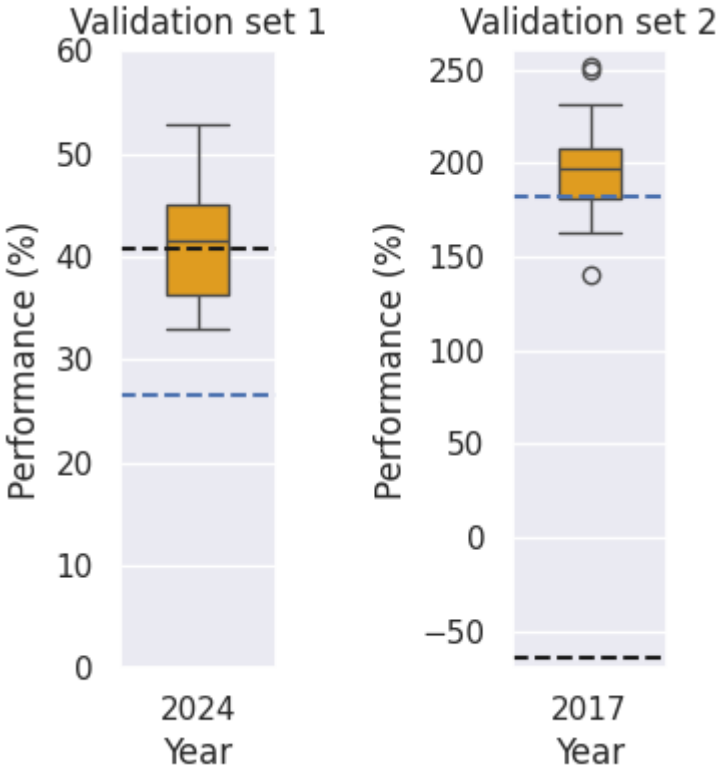
- **Hyperparameter optimization**
  - Batch size
  - Training interval
- In **Random sampling** the samples are chosen randomly from the memory replay.
- In **Recent sampling** the most recent samples are chosen from the memory replay.
- In **Hybrid sampling** half the batch size is chosen randomly, and the other half are the most recent ones.



Running the best configuration multiple times

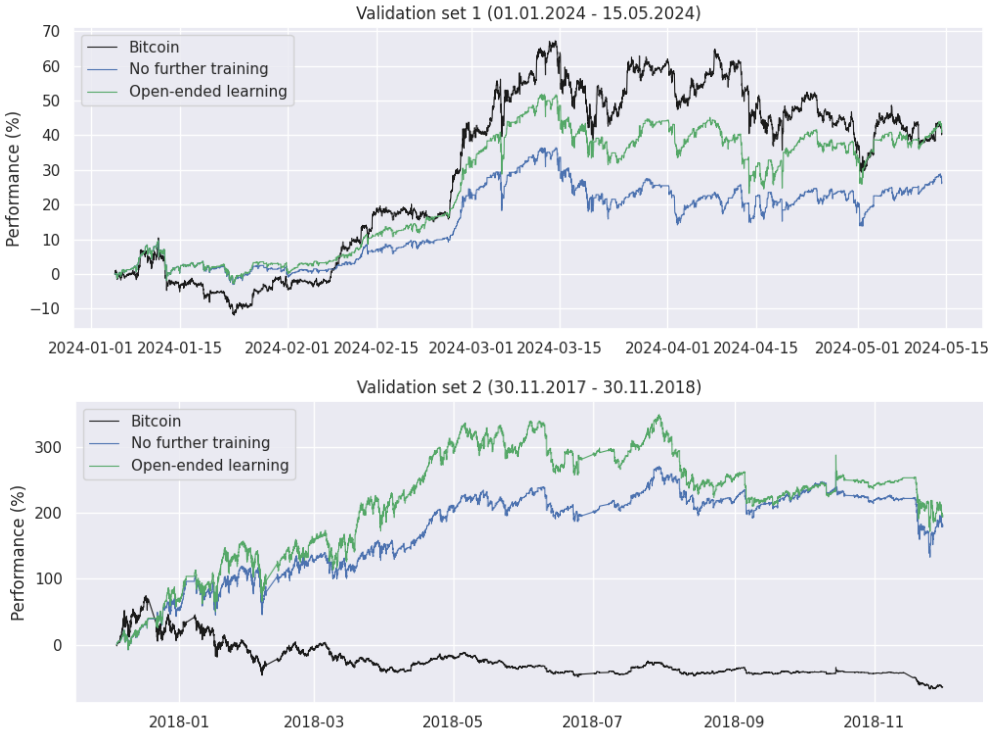
Batch size: 256, Train every: 64, Mode: Hybrid

- Boxplot over of the same configuration **21 runs**.
- Validation set one
  - Base Bot beaten every time.
  - Was able to beat market unlike the base bot.
- Validation set two
  - Base bot beaten more times than not.



Analyzing the median run

- Validation set 1
  - Able to pull away from base bot.
  - Able to beat the market just in the end.
- Validation set 2
  - Able to pull away from base bot.
  - The gap closes after 7 months.



Conclusion

- Generally, seems to work well in declining market.
- Open-ended learning is very interesting for the bitcoin market.
- Hybrid mode seems promising.
  - Different percentages of random and recent samples might be interesting.
- Gap closes after some time.
  - Maybe a full retraining could solve this issue.