

QF632-2025-W8

Number of participants: 22



1. What does the reward signal tell an RL agent?

11 correct answers
out of 12 respondents

How unpredictable the environment is



1 vote

Whether its last action was good or bad for its long-term goal



11 votes

The exact next action to take



0 votes

How many states exist in the environment



0 votes



2. In a standard RL problem, exploration is needed because:

13 correct answers
out of 14 respondents

The transition dynamics are always stochastic.



1 vote

The agent's initial policy is assumed optimal.



0 votes

The agent does not initially know which actions yield high long-term rewards.



13 votes

The discount factor, γ , is less than 1.



0 votes



3. Why is the discount factor (γ) usually set to a value less than 1?

14 correct answers
out of 16 respondents

To force the agent to act randomly early on



2 votes

Because future rewards are considered slightly less important than immediate ones



14 votes

To make the environment deterministic



0 votes

So the agent never finishes an episode



0 votes

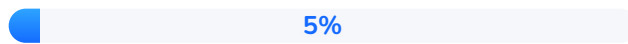


4.

If an agent keeps receiving sparse rewards (only at episode end), which technique can make learning easier?

9 correct answers
out of 19 respondents

Add more hidden layers to the network by default



1 vote

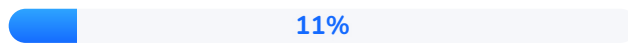


Introduce shaping rewards that give small hints along the way



9 votes

Lower the discount factor to zero



2 votes

Turn the episodic task into a continuing task without resetting



7 votes



5. In exploration–exploitation for portfolio RL, too much exploration leads to:

10 correct answers
out of 16 respondents

Overfitting to the training period



6 votes

Excessive random trades that hurt performance in live deployment



10 votes

Conservatively holding cash only



0 votes

Guaranteed higher Sharpe ratio



0 votes



Why might an RL agent overfit 6. when trained on historical price series?

12 correct answers
out of 12 respondents

Because it uses too
large a discount
factor ($\gamma \approx 1$)

0%

0 votes

Because it
memorizes past
market patterns
that don't repeat

100%

12 votes

Because
transaction costs
are ignored

0%

0 votes

Because states are
Markovian

0%

0 votes



7.

What's the main advantage of using a distributional RL approach (predicting full return distribution) over standard RL in portfolio tasks?

10 correct answers
out of 10 respondents

It guarantees convergence faster

0%

0 votes

It captures the shape and tail risks of the return distribution, improving risk management

100%

10 votes

It requires no function approximation

0%

0 votes

It avoids the need for bootstrapping

0%

0 votes




In a partially observable market
8. model, integrating an RNN (e.g. LSTM) enables the agent to

9 correct answers
out of 10 respondents

Reduce the action
space to discrete
trades

0%

0 votes

 Infer hidden
volatility regimes
from past prices
and actions

90%

9 votes

Eliminate the need
to model
transaction costs

10%

1 vote

Ensure the hedge
is always perfect

0%

0 votes



**A key downside of discrete-time
9. rebalancing (e.g. daily) versus
continuous hedging is that:**

9 correct answers
out of 11 respondents

The Markov
property breaks
down



2 votes

Transaction costs
disappear



0 votes



Tracking error
builds up between
re-hedging steps



9 votes

Option Greeks
become constant



0 votes



10. In multi-objective RL for hedging (balancing return, variance, and cost), a common approach to scalarise the vector reward $(r^{\text{return}}, r^{\text{var}}, r^{\text{cost}})$ is to

10 correct answers
out of 10 respondents



Use a weighted sum $w_1 r^{\text{return}} - w_2 r^{\text{var}} - w_3 r^{\text{cost}}$, tuning w_i to reflect risk-return-cost trade-offs

100%

10 votes

Ignore variance entirely

0%

0 votes

Optimise only r^{return} then adjust for cost later

0%

0 votes

Train three separate agents without sharing experience

0%

0 votes