MLT-07 Summary Document

Reinforcement Learning largely deals with training an agent in a given environment. It has three primary elements: states of the environment, actions of the agent and the reward given by the environment to the agent based on the action taken. It rewards the desired action and punishes the undesired behavior of the model. Reinforcement learning follows the Markov Decision Process as the core logic.

State: The state is the current situation of the environment (stock market in our case). We can define various features such as the percentage change, RSI, volatility and so on to represent states. It tries to capture all the information that is required to take action.

Actions: Actions are taken by the agent in the given state. It can be "Buy", "Hold" and "Sell." in case of trading.

Reward: The reward function can use the following parameters to calculate the reward or punishment:

- Percentage change of price in the given interval
- The direction of the percentage change in the given interval
- Final Returns of a trade
- Returns in the given time-frequency
- etc.

Epsilon: The RL model uses this value to tradeoff between exploration or exploitation policies.

Policy: The policy helps the RL model to decide, which action to take depending upon the current state. There are two policies used in RL, exploration and exploitation.
At the start of the game, the model does not know which action would lead to the best reward. Therefore, it starts with exploration policy. As the RL model learns with the help of Epsilon value, the exploration rate reduces, and it moves towards exploitation policy.

Q Table: Q-Table is a table with columns as actions, row labels as states, and cross-dimensional values as the rewards. In other words, it is a list of rewards for each state and action pair. An example of a Q table for a trading model could be:

States	Buy	Sell
S1	1	0
S2	0	1
S3	0	0

- S1, S2, S3 are states with a combination of values with RSI, pct change, volatility etc.
- "Buy" and "Sell" are actions.

Bellman Equation: Bellman equation helps in updating the Q table, which is the function of reward.

$$Q(s,a) = r(s,a) + \gamma \max Q(s',a)$$

- r is the “immediate” reward of action “a”
- Q is the “cumulative” reward of action “a” (Q-table)
- s’ is the state we end up with after performing the action “a”.
- Gamma symbol(γ) is referred to as the learning rate.

Input Feature Selection: Following features can be considered that can represent states of an environment:

- OHLCV:
  - Percentage change in the Open, High, Low, Close prices and Volume.
  - The above can be chosen for the last *n* periods.
- Technical indicators: Some of the examples are:
  - RSI
  - MACD
  - Moving Average crossover
- The market behaves differently in different time periods. Therefore, following features can be considered:
  - Hour of the day
  - Day of the week
  - The month of the year
  - Week of the year.
- Different time granularity
  - OHLCV data and Technical Indicator Data in various time frequencies:
    - 5 mins
    - 15 mins
    - 1 Hour
    - 1 Day
- Correlation with other assets
  - Negative correlation with other assets or asset classes in the different time intervals.
  - Positive correlation with other assets or asset classes in the different time intervals.
- Alternative data: Some of the examples of Alternative data that can be considered are:
  - Sentiment analysis data
  - Website traffic
  - Mobile app downloads, rating, usage per instance
  - Geolocation statistics.

Type of Algorithm: Following types of Neural network algorithms can be considered:

- Recurrent neural network: Inputs are processed only in the forward direction.
  - Ideal for sequential or time-series data.
- Convolutional NN:
  - Ideal for images. it can be used if you are using chart patterns for predictions
- LSTM:
  - It is derived from RNN, capable of learning long term dependencies.