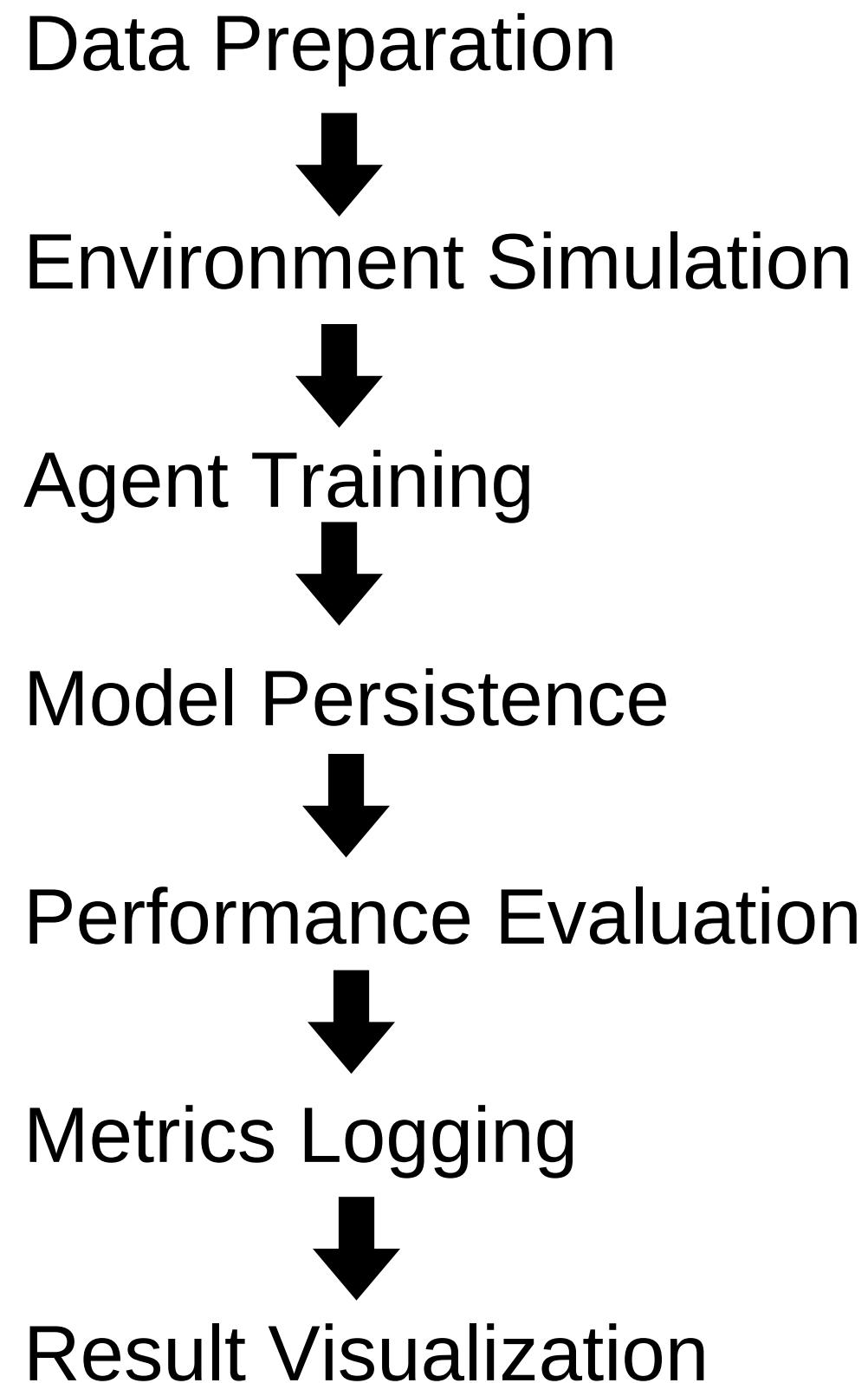


Short Summary about Inventory Management

**By
Vaseegaran B**

Workflow



1. Dataset Handling

- Path check: Located M5 Forecasting dataset in
D:\Trail\Project_root\data\raw\m5-forecasting-accuracy.
- Files loaded: sales_train_validation.csv, calendar.csv, sell_prices.csv.
- Preprocessing steps:
 - Merge datasets.
 - Clean missing/irrelevant values.
 - Perform feature engineering (time, demand, prices, promotions).
- Data prepared for environment simulation

2. Environment Simulation

- Custom Inventory Management Environment created using Gymnasium API.
- Observation includes: inventory levels, demand signals, day features, etc.
- Action space: reorder decision (how much to order).
- Reward: profit (sales revenue – holding cost – stockout penalty).

3. Agent Training

- Algorithm: DQN (Deep Q-Network).
- Timesteps: 5,000 training steps.
- Device: CPU.
- Model trained and saved at:
➤ `models/dqn_inventory_model.zip`.

4. Evaluation Phase

- Episodes: 3 simulation runs.
- For each episode:
 - Agent interacts with environment using learned policy.
 - Records reward, service level (demand served ÷ total demand), and profit.
- Results aggregated into metrics.

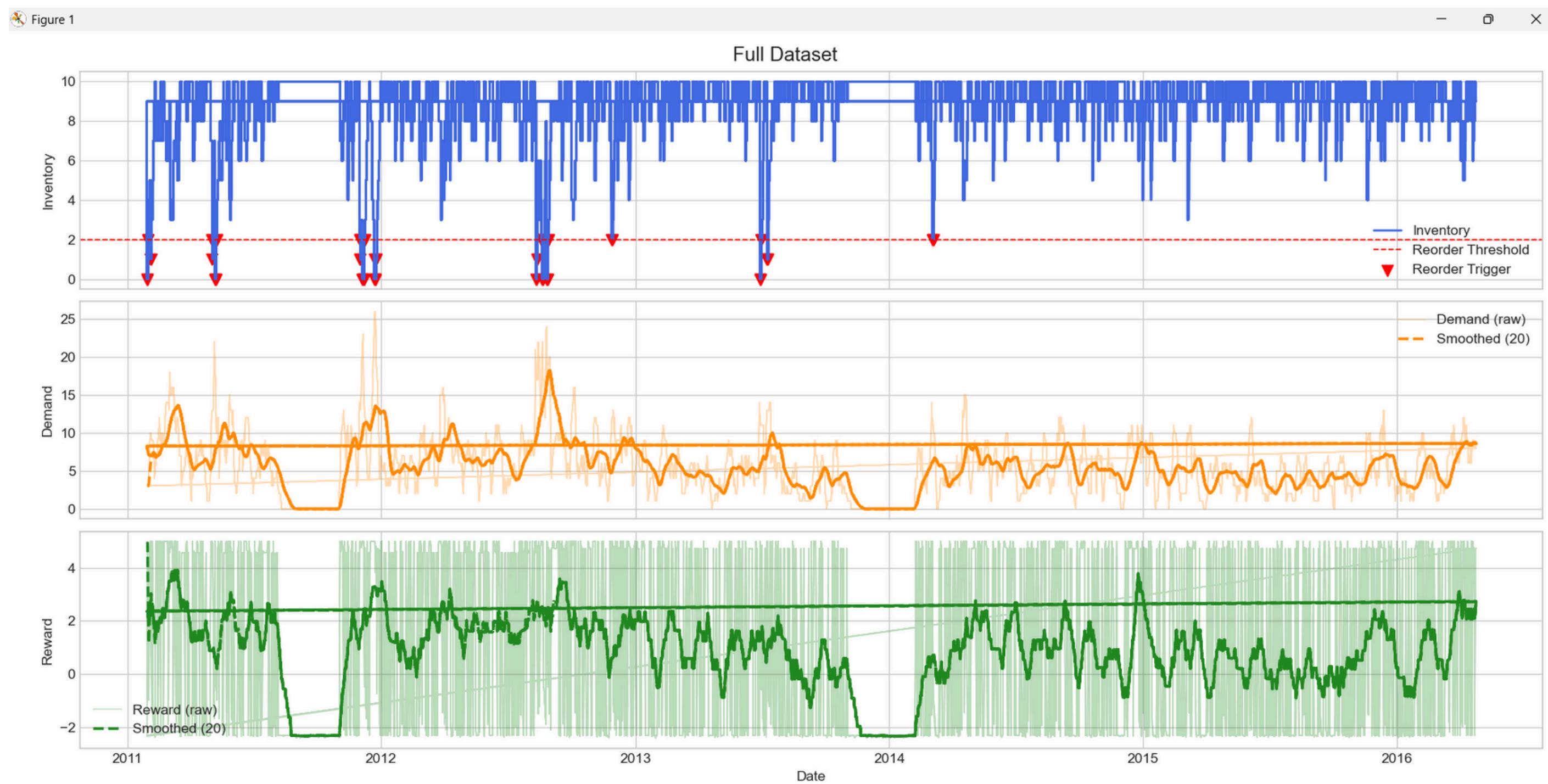
5. Results

- Agent interacts with environment using learned policy.
- Average Reward: 1554.03
- Average Service Level: 0.92 ($\approx 92\%$)
- Average Profit: 1554.03
- Metrics saved to: ./reports/metrics.json.

./reports/metrics.json

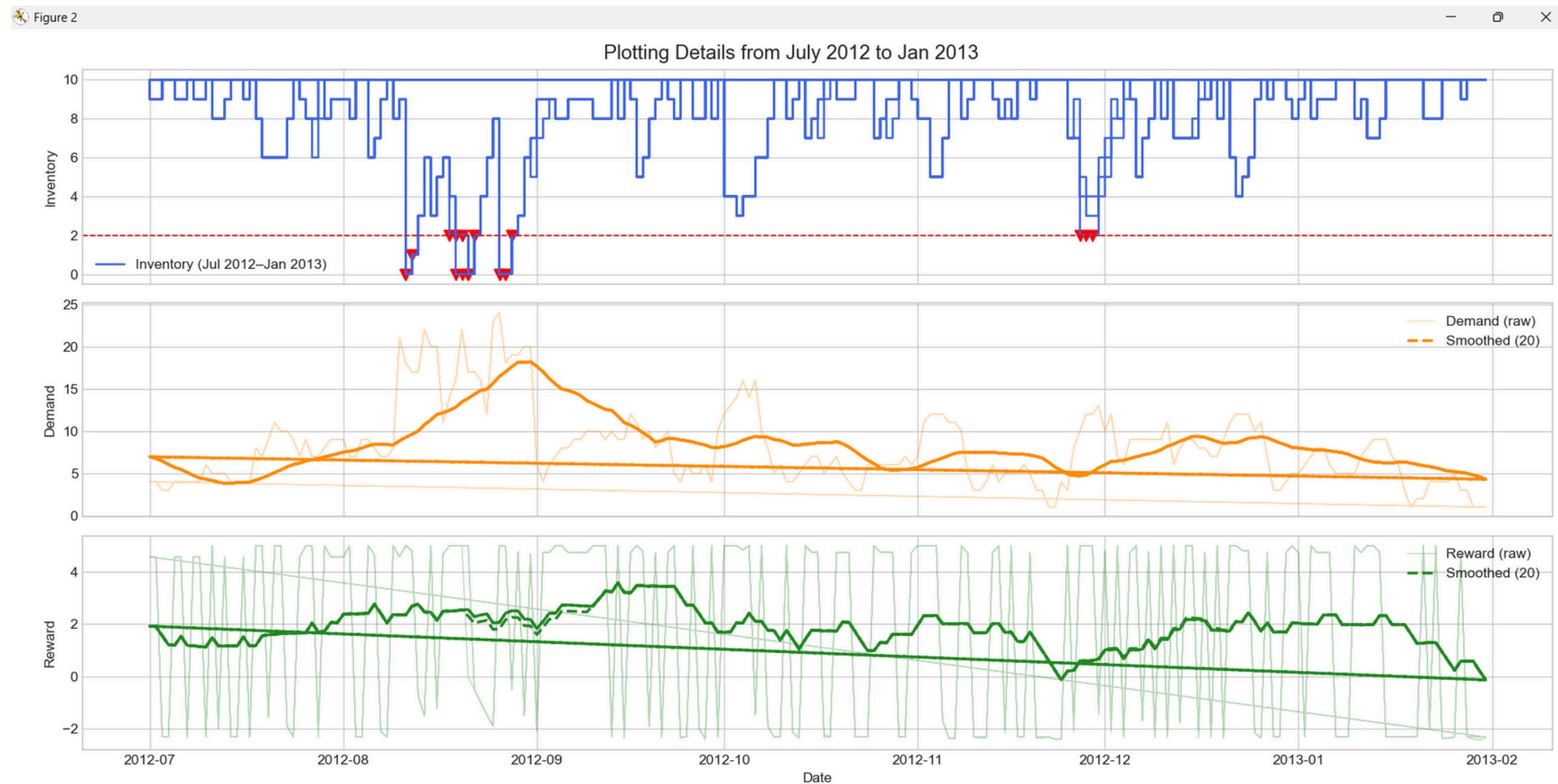
```
reports > {} metrics.json > [ ] profits > # 2
  1  {
  2    "agent": "dqn",
  3    "timesteps": 5000,
  4    "episodes": 3,
  5    "item_id": "FOODS_1_001",
  6    "store_id": "CA_1",
  7    "avg_reward": 1554.0286865234375,
  8    "avg_service_level": 0.9175310051701716,
  9    "avg_profit": 1554.0286865234375,
 10    "rewards": [
 11      1552.949951171875,
 12      1555.9190673828125,
 13      1553.217041015625
 14    ],
 15    "profits": [
 16      1552.949951171875,
 17      1555.9190673828125,
 18      1553.217041015625
 19    ],
 20    "service_levels": [
 21      0.9187363212484537,
 22      0.915786468741079,
 23      0.918070225520982
 24    ]
 25 }
```

Result Visualization



Plotting Inventory Dynamics, Demand, Reward(Full Dataset)

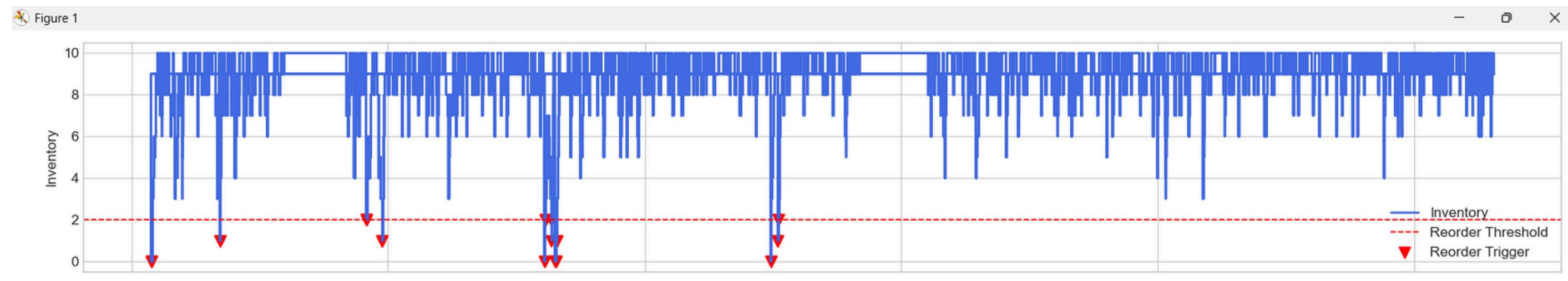
Result Visualization



Plotting Inventory Dynamics,Demand,Reward(July-2012 to Jan 2013)

1. Inventory Dynamics (Top Plot)

- Line plot (blue step) shows the inventory level over time.
- Red dashed line marks the reorder threshold (inventory ≤ 2).
- Red downward triangles (scatter) indicate reorder triggers, i.e., points when inventory fell below the threshold.
- Interpretation:
 - The agent places orders to maintain inventory above the threshold.
 - Sudden drops show demand exceeding inventory.
 - Reorder markers highlight how often the policy triggers restocking.



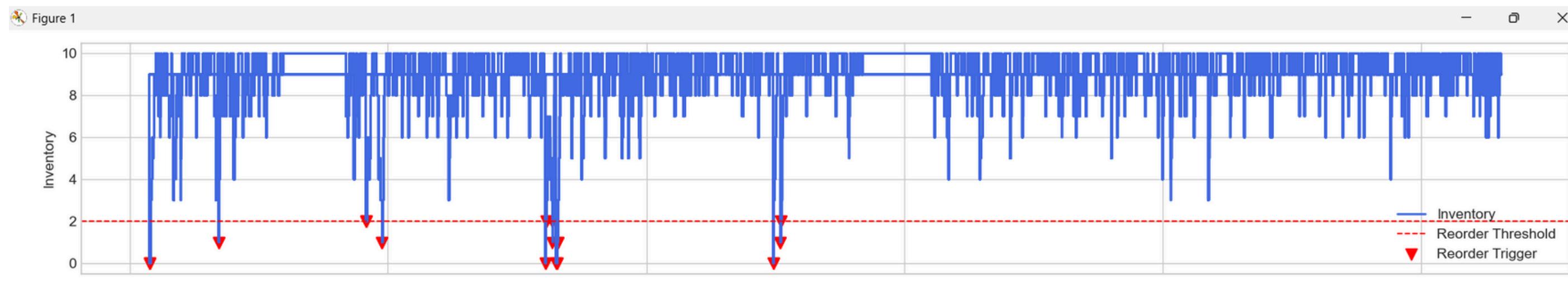
Inventory Plot (Top)

X-axis:

- Date (calendar time, starting from 2011-01-29, increasing daily).
- Represents time progression (days).

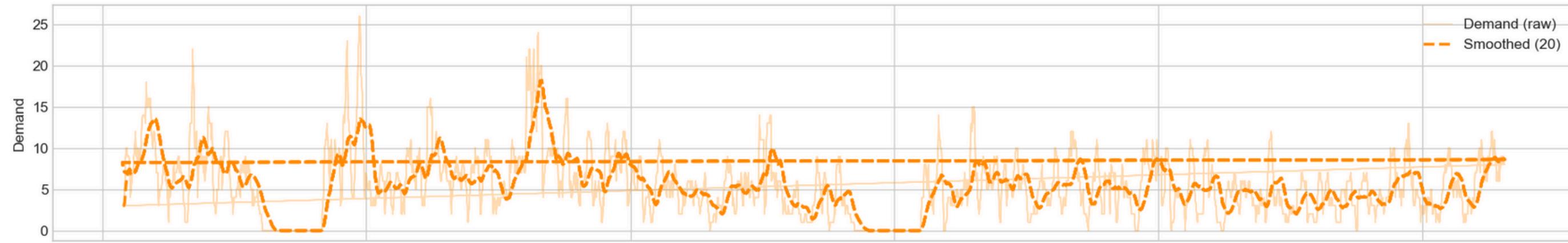
Y-axis:

- Inventory (integer values).
- Represents the stock level available at each step.
- Additional markers:
 - Red dashed line: reorder threshold at 2 units.
 - Red triangles: points where $\text{inventory} \leq \text{threshold}$.



2. Demand (Middle Plot)

- Orange thin line (semi-transparent) shows raw daily demand.
- Orange dashed line is a smoothed 20-day rolling average.
- Interpretation:
 - Helps visualize trends in demand over time.
 - Peaks in demand may correspond to inventory dips and reorder events.
 - Smoothing removes daily fluctuations, showing the overall demand trend.



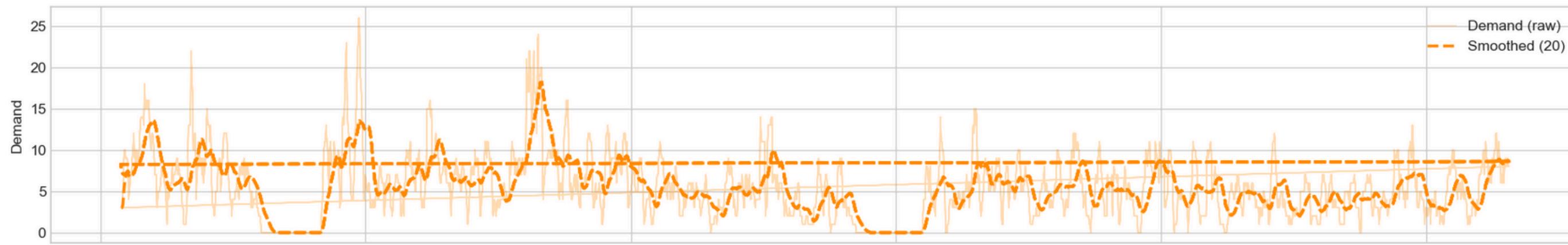
Demand Plot (Middle)

X-axis:

Same as above → date (time in days).

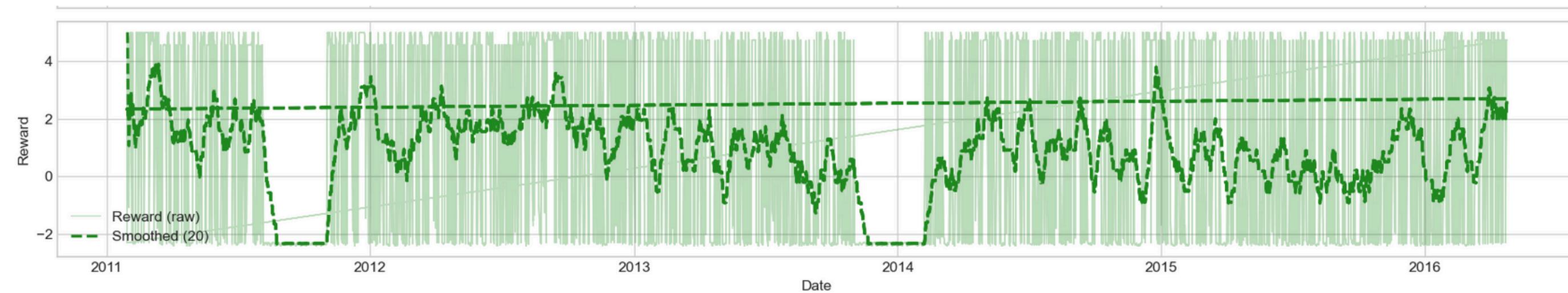
Y-axis:

- demand (number of units requested by customers).
- Two versions shown:
 - Raw demand: daily fluctuations (thin orange line).
 - Smoothed demand: rolling average (dashed orange line).



3. Reward (Bottom Plot)

- Green thin line (semi-transparent) shows raw reward per step.
- Green dashed line is a smoothed rolling average.
- Interpretation:
 - Reward reflects profitability, service level, and penalties (holding and stockout).
 - Sharp drops indicate steps with high unmet demand or inventory costs.
 - Smooth trend shows overall learning performance of the RL agent over time.



Reward Plot (Bottom)

X-axis:

Same as above → date (time in days).

Y-axis:

- reward (numeric value of agent's immediate profit/penalty per step).
- Two versions shown:
 - Raw reward: per-step values (thin green line).
 - Smoothed reward: rolling average (dashed green line).

