Detecting Rewards Deterioration in Episodic Reinforcement Learning

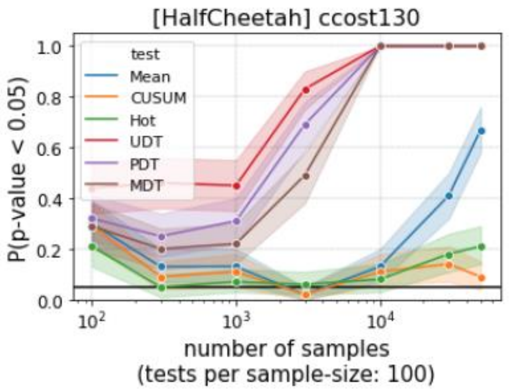
Ido Greenberg & Shie Mannor, ICML 2021

Problem

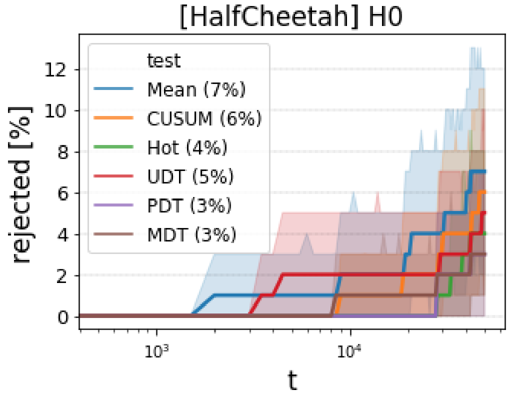
- Reinforcement Learning: learn a policy to make decisions in various states
- Reasonable real-world pipeline: train → freeze → test → production
- What if the world changes / the agent falters for any reason?
- **Re-exploring policies online is often forbidden / unsuccessful**
  - E.g. in autonomous driving: pass control to human driver
  - E.g. in insulin injector: send patient to the doctor
- **Key is noticing performance degradation ASAP**
- Common assumptions (Markov, known model, etc.) do not hold

Framework and solution approach

- Focus on rewards (no need to learn a states-dependent model)
- Episodic framework: i.i.d episodes of length T
  - Rewards within episodes: **NOT [independent, identically-distributed, or Markov]**
  - Reference dataset: “valid” recorded episodes (e.g. the test period before marketing)
- **Pseudo-algorithm: for every test-point (e.g. several times per episode):**
  - **If the rewards of the last few episodes are small wrt the reference data – raise a flag**
- How to summarize the rewards (i.e. what is the **test-statistic**)?
- What is “small” (i.e. how to choose the **test-threshold**)?



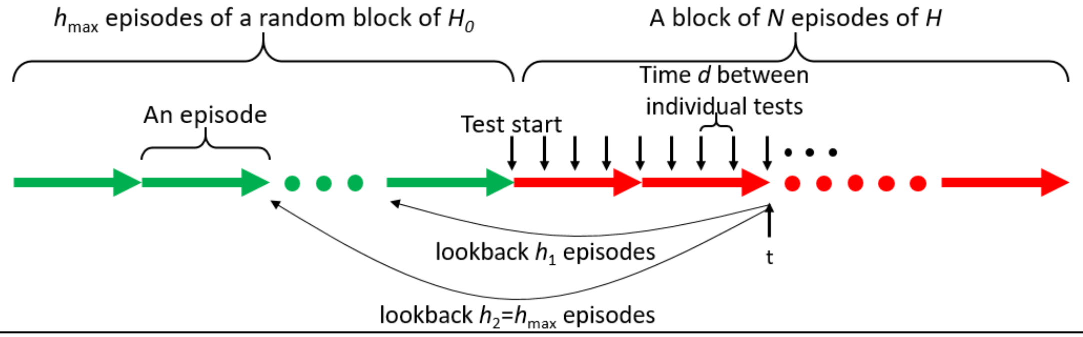
Due to sub-optimal handling of noise, standard tests sometimes become *worse* when getting more data



<u>Threshold tuning</u>: all 6 tests were successfully tuned to yield ~5% false-alarms within 50 episodes

<u>Sequential test for a scenario *H*</u>

<u>Repeat for *M=100* different data-blocks of *H*:</u>

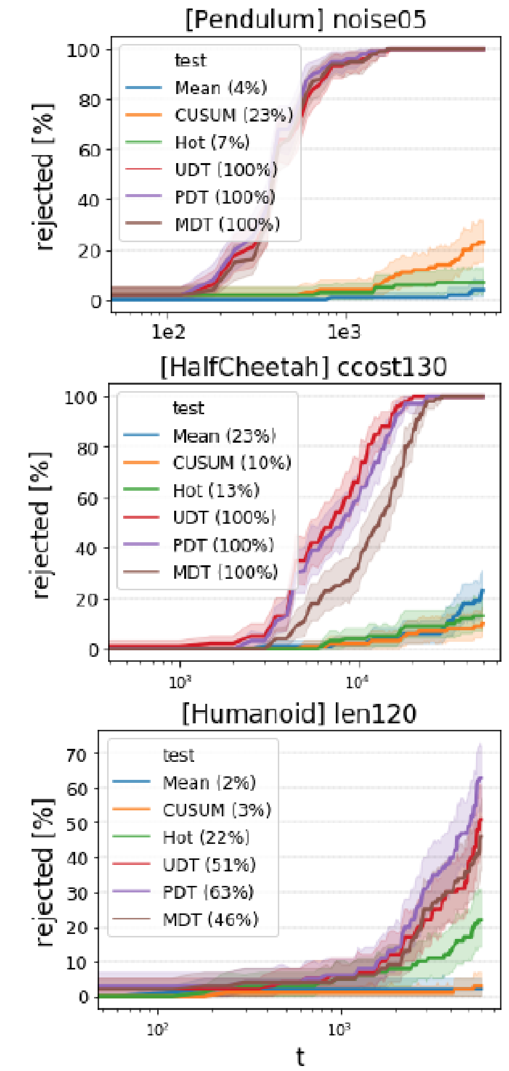

h_{max} episodes of a random block of H_0
A block of N episodes of H
Time d between individual tests
An episode
Test start
lookback h_1 episodes
t
lookback h_2=h_{max} episodes

Test statistic

- Naïve: mean reward (e.g. over last few episodes)
- Our suggestion: **weighted mean** $\boldsymbol{W \cdot r}$ $\boldsymbol{(W \coloneqq \mathbf{1}^\top \Sigma^{-1})}$
  - $\Sigma \in R^{T \times T}$ is the covariance matrix, estimated from the reference data
  - Intuition for independent rewards: $w_t = 1/\sigma_t^2$ ⇒ look where it’s less noisy
- Theorems:
  - Uniform degradation over time-steps + normal rewards ⇒ **optimal test** (max power)
  - Without normality ⇒ still better than simple mean
  - Advantage over simple mean increases with heterogeneity of Σ’s eigenvalues
- Experiments – how long it takes to detect degradation of rewards in modified environment?
  - **Our test variants (UDT,PDT,MDT) usually win** simple mean, CUSUM and Hotelling

Test-threshold tuning

- Threshold-tuning by bootstrap usually relies on i.i.d data samples (for re-sampling)
- **BFAR**: Bootstrap for False-Alarm Rate control that **can be applied to episodic (non-i.i.d) data**


[Pendulum] noise05
rejected [%]
test
Mean (4%)
CUSUM (23%)
Hot (7%)
UDT (100%)
PDT (100%)
MDT (100%)
1e2
1e3
[HalfCheetah] ccost130
rejected [%]
test
Mean (23%)
CUSUM (10%)
Hot (13%)
UDT (100%)
PDT (100%)
MDT (100%)
10^3
10^4
[Humanoid] len120
rejected [%]
test
Mean (2%)
CUSUM (3%)
Hot (22%)
UDT (51%)
PDT (63%)
MDT (46%)
10^2
10^3
t