

The Statistical Benefits of Quantile Temporal-Difference Learning for Value Estimation

Discussed by: Harvey Huang

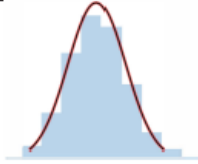
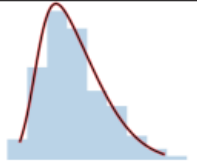
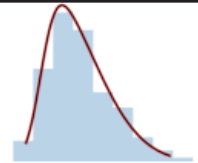
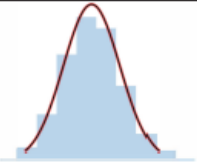
June 2023

Paper

- May 28, 2023 ICML by Google Deepmind.
- Link to paper: <https://arxiv.org/pdf/2305.18388.pdf>

Recap: (temporal-difference) distributional RL

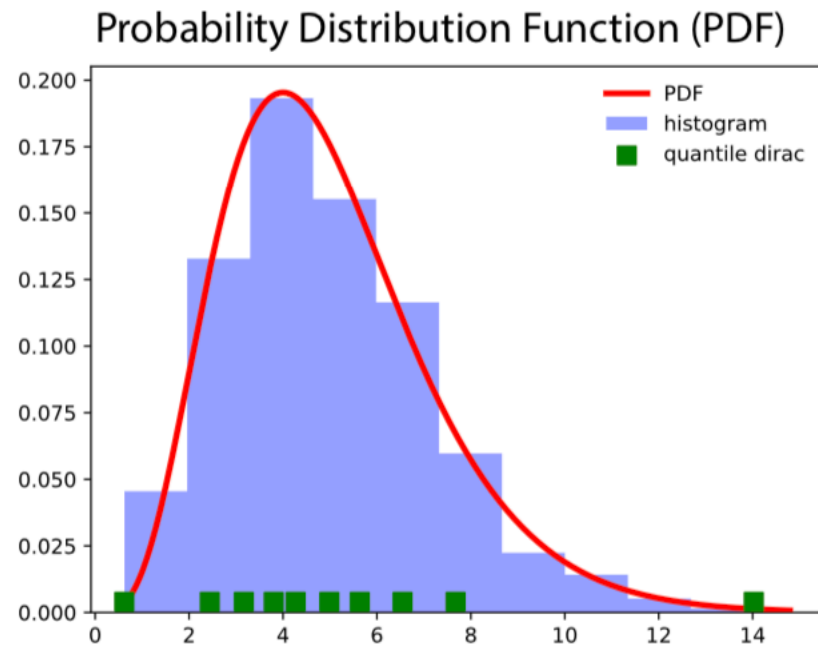
- Broadly speaking, rather than estimating a mean value (scalar) for each state-action pair, we estimate a distribution and then calculate the mean.
 - If greedy policy: action = $\operatorname{argmax}(Q(s,a))$.

$Q(s, a)$	s_0	s_1	\Rightarrow	$Z(s, a)$	s_0	s_1
a_0	$Q(s_0, a_0)$	$Q(s_1, a_0)$		a_0		
a_1	$Q(s_0, a_1)$	$Q(s_1, a_1)$		a_1		

Recap: two practical questions

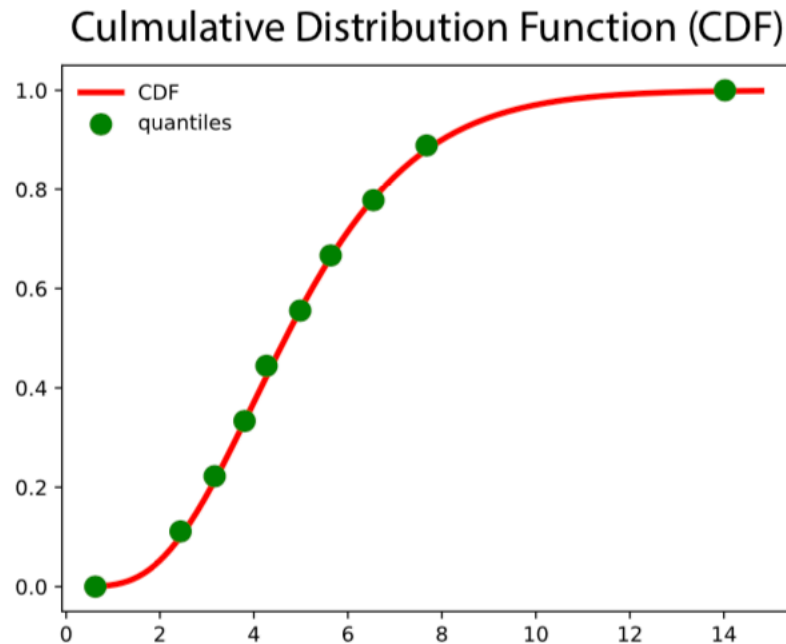
- Q1: what distribution?
 - Parametric
 - E.g., $N(\mu, \sigma)$.
 - Non-parametric
- Q2: how do we apply distributions in the context of RL?
 - E.g., updating an action value \Rightarrow update a distribution?

Recap: non-parametric example, $Z(s, a)$



- To draw a histogram (to approximate PDF):
 - Min, Max
 - Equal-width bins at some locations
 - Bin height
- Density is represented by how ``tall'' a bin is

Recap: non-parametric example, $Z(s, a)$



- To draw a CDF, we can use quantiles
 - You need the position value (x, y) for each green dot

Recap: a broad scope on distributional RL

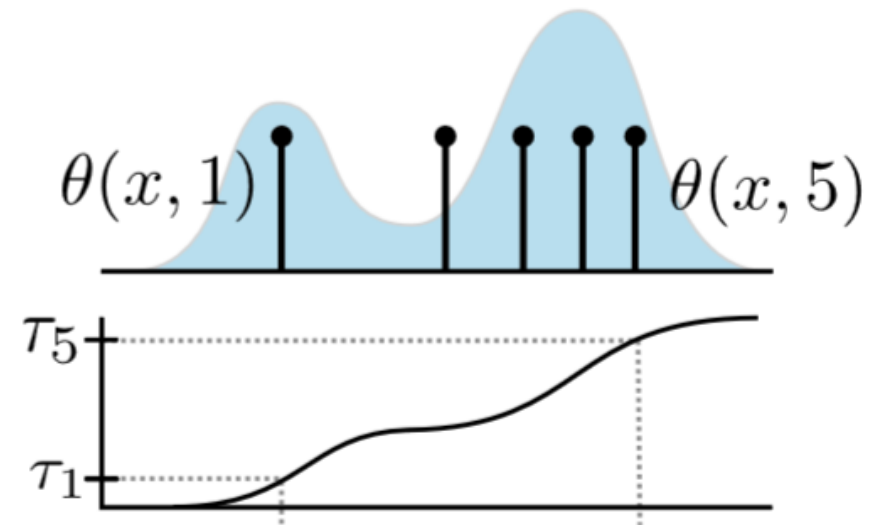
- Suppose that we can find the required statistics.
 - Histogram: min, max, bin location, bin width, bin height
 - CDF: quantile dot (x, y)
- We then have an approximated action-value distribution.
 - Action-value = mean of the distribution.
- So why bother this extra distribution step?
 - Motivation: you get a richer representation of the action value, and hence you can get a more accurate mean estimation.
 - In standard RL, the action-value (mean) is estimated directly.

Motivation of this paper

- “It is commonly hypothesized that the benefits of the distributional approach stem from its interaction with non-linear function approximators such as deep neural networks, rather than statistical reasons.”
 - I.e., distributional-RL works because of deep neural networks; it’s ability to extract rich information.
 - Personally, I think this is trivial but nonetheless.
- What about temporal-difference learning (the base case)?
 - When state and action spaces are limited and small.
 - Can distributional RL (quantile) help?

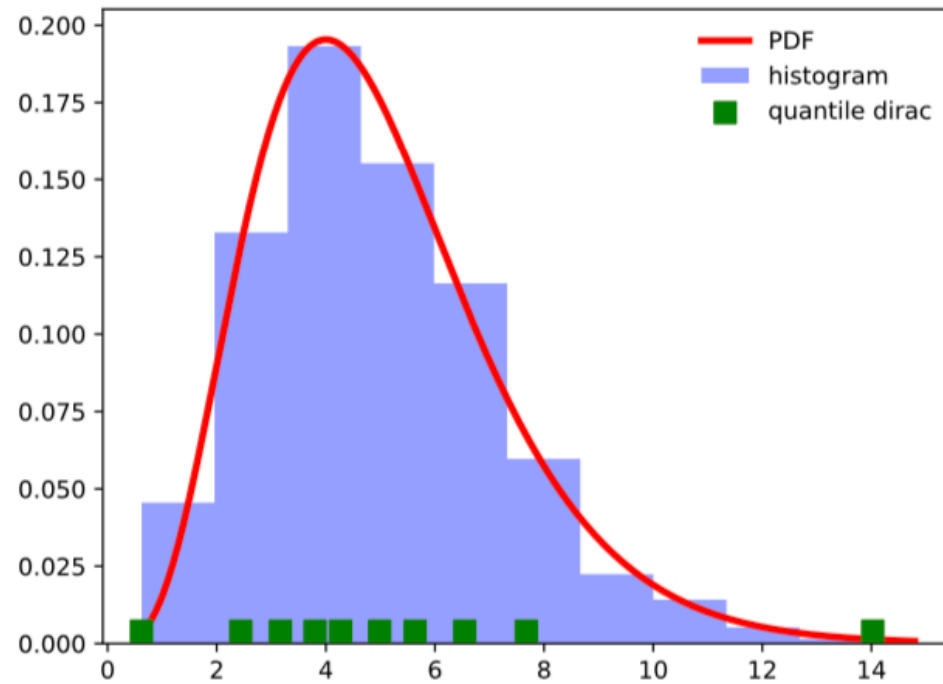
Quantile temporal-difference learning

- Given a state x , and a quantile position τ , we estimate a value θ for this quantile. Then collectively we have a set of statistics that describes the distribution (a list of quantile values).
- τ : 1st, ..., 5th quantile.
- θ : given state x , the quantile value.
- Upper: (dirac delta) PDF; lower: CDF.

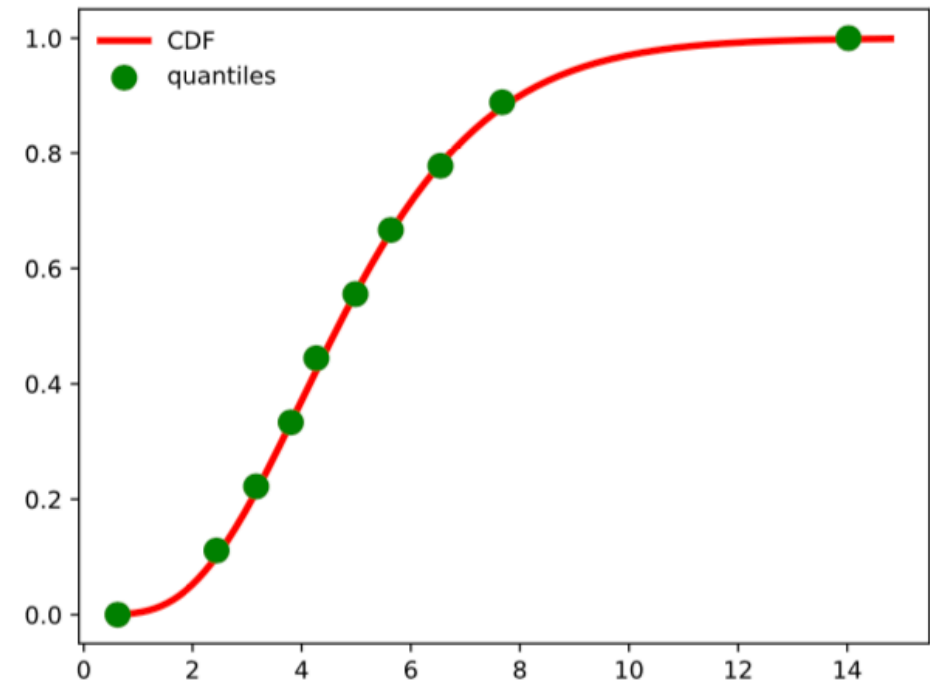


My demo

Probability Distribution Function (PDF)



Culmulative Distribution Function (CDF)



Related to my research: heavy tails

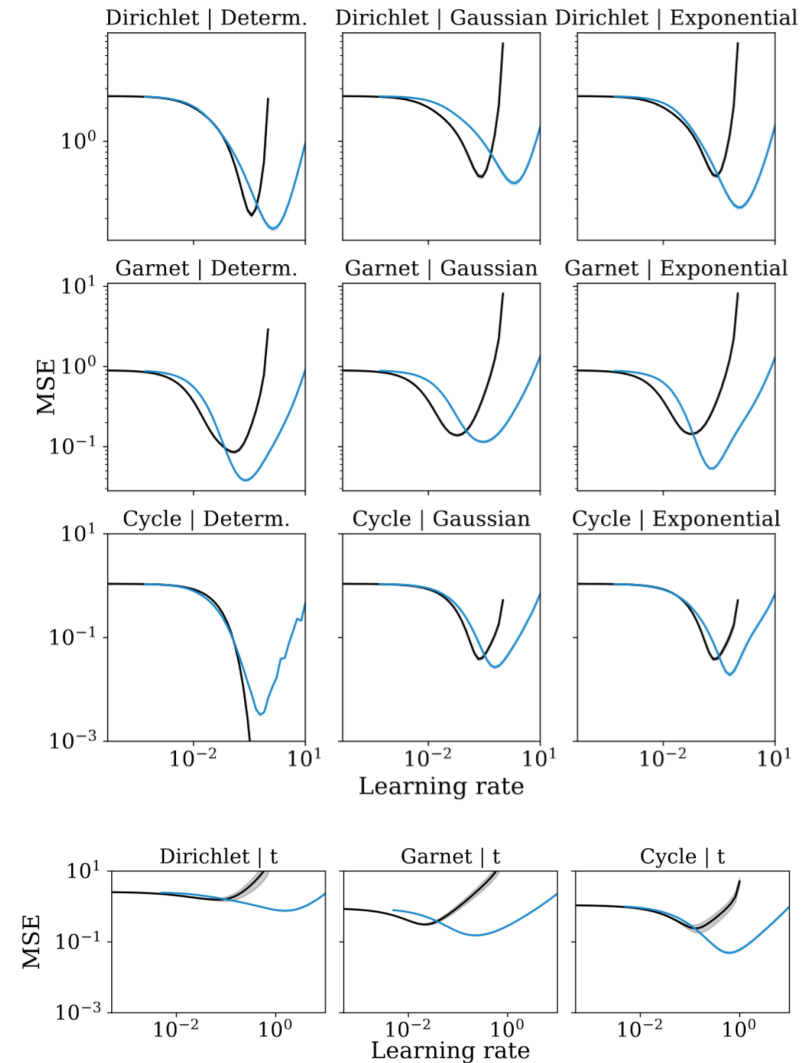
- “...Gastwirth (1966) observing that this approach (quantile) to estimation provides competitive relative efficiency across a wide variety of distributions, including those with heavy tails, where the usual sample-average mean estimator can be inefficient.”
- “...we might conjecture that Quantile Temporal Difference provides an approach to value estimation that is effective across a wide range of environments, particularly those with heavy-tailed reward distributions.”

Method: varying two environment components in Markov Decision Making (MDP) process

- State transition
 - Deterministic cycle structure
 - Sparse stochastic transition (sampled from a Gartner distribution)
 - Dense stochastic transition (sampled from Dirichlet(1, . . . , 1) distributions)
- Reward
 - Deterministic rewards
 - Gaussian rewards (Variance = 1)
 - Exponentially distributed (rate = 1) rewards
 - t-distribution degree of freedom = 2.

(1) TD vs. Quantile TD (QTD)

- Varying agents' learning rate (but keep constant value)
 - X-axis: learning rate
 - Y-axis: MSE (estimated Q vs. true Q after 1000 updates).
 - Black line: TD, blue line: QTD
- Conclusion:
 - QTD is better than standard TD.
- Note the difference between the optimal learning rates...



Why QTD > TD (theory)

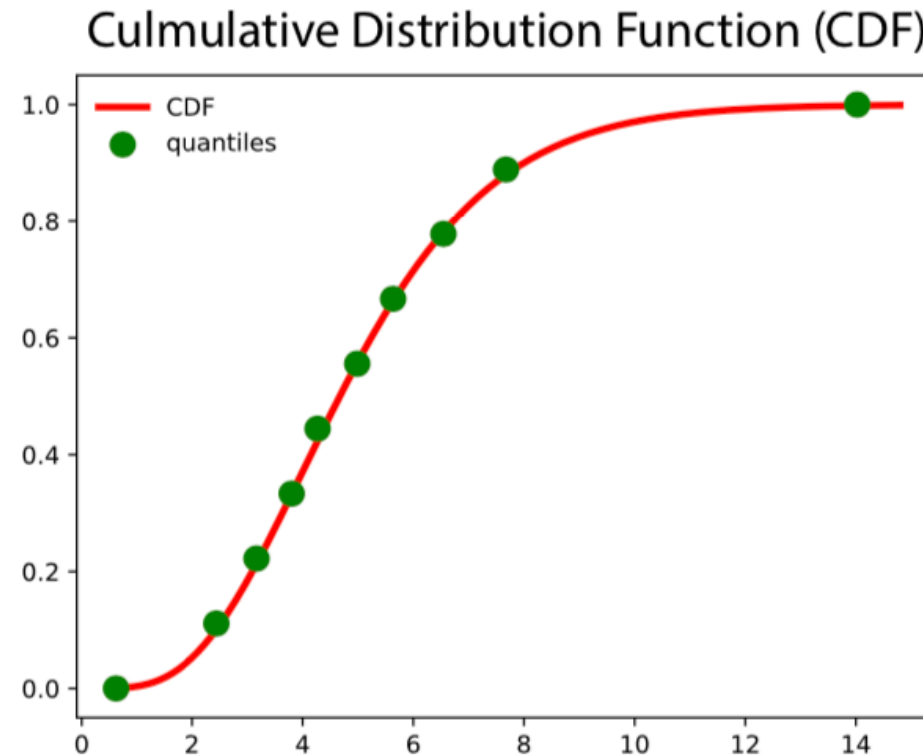
- In conventional TD, you can mathematically prove that ultimately the estimated value function converges to the true value function.
 - Broadly, $|Q_{\text{estimates}} - Q_{\text{true}}| < \text{some constant}$.
- Watkins & Dayan (1992).
- Two conditions:
 - *The learning rates must approach zero, but not too quickly.*
 - *Each state-action pair must be visited infinitely often.*

Why QTD > TD (theory)

- Equivalently in the distribution case, the infinity norm of the distance between two sets.
 - $|\text{quantile_estimates} - \text{quantile_true}|_\infty$
- Note convergence cannot generically be obtained for QTD (or distributional reinforcement learning in general).
- You need sufficiently large number of quantiles to be able to obtain convergence.
 - \Leftrightarrow Practically, you need sufficiently large number of quantiles to get a relatively accurate mean estimate.

Let's pause and think...

- What does it mean if we have “infinite” number of quantiles?



Why QTD $>$ TD (theory)

- First assume bounded rewards.
- Convergence in QTD
 - Unlike classical TD, which has unique convergence (to the true value), QTD converges to a set of convergence points.
 - Possibly non-unique unless the CDF is a strictly increasing function.
 - Mentioned both in this paper and an earlier analysis QTD paper (Jan 2023).
 - Echo my dissertation: <Discussion on limitations of distributional RL>.
- But the authors consider this a secondary issue...

Why QTD > TD (theory)

- The theory analysis also went further for unbounded rewards...
 - But within the Gaussian realm.
- Two highlights:
 - As the number of quantiles \rightarrow infinity, convergence can be obtained.
 - “The heavier the tails of the reward distributions, the slower the convergence may be”.

TD decomposition

- A large learning rate
 - Increases both (1) expected update and (2) mean-zero noise
 - We need to find an optimal learning rate to balance the two.
- “...the magnitude of the noise is potentially unbounded...”
 - If the reward is drawn from an unbounded distribution.

$$\begin{aligned} & V(x) + \alpha(r + \gamma V(x') - V(x)) \\ = & (1 - \alpha)V(x) + \\ & \alpha \underbrace{((T^\pi V)(x))}_{\text{Expected update}} + \underbrace{(r + \gamma V(x') - (T^\pi V)(x))}_{\text{Mean-zero noise}}, \end{aligned}$$

TD decomposition: $\alpha * r$

- If alpha is constant and reward r is unbounded -> no-convergence
 - There is always some probability (albeit small) that reward is +ve/-ve large enough such that the policy end up changing
- Suppose we decay alpha...
 - The question is how?
 - Amplified in heavy tailed reward case.
 - High probability (compare to Gaussian) that a large reward dominates the TD errors.

$$\begin{aligned} & V(x) + \alpha(r + \gamma V(x') - V(x)) \\ = & (1 - \alpha)V(x) + \\ & \alpha(\underbrace{(T^\pi V)(x)}_{\text{Expected update}} + \underbrace{(r + \gamma V(x') - (T^\pi V)(x))}_{\text{Mean-zero noise}}), \end{aligned}$$

QTD decomposition

- Experiences similar tension as in the TD value function.
- However, rewards do not directly influence the updates
 - Updates depend only on the sign (the “1” in the sum) not the magnitude.
- Conclusion:
 - “QTD is more resilient to heavier tails”.

$$\begin{aligned} & \alpha \left(\tau_i - \frac{1}{m} \sum_{j=1}^m \mathbb{1}[r + \gamma \theta(x', j) < \theta(x, i)] \right) \\ &= \alpha \left(\underbrace{\tau_i - \mathbb{P}_x^\pi(\Delta_{iJ}(x, R_0, X_1) < 0)}_{\text{Expected update}} + \right. \\ & \quad \left. \underbrace{\mathbb{P}_x^\pi(\Delta_{iJ}(x, R, X') < 0) - \frac{1}{m} \sum_{j=1}^m \mathbb{1}[\Delta_{ij}(x, r, x') < 0]}_{\text{Mean-zero noise}} \right). \end{aligned}$$

where we write $\Delta_{ij}(x, r, x') = r + \gamma \theta(x', j) - \theta(x, i)$.

Empirically show impact of heavy tail rewards

- **t-distribution** with degree of freedom = 2.
- 128 quantiles.
- Assume a constant learning rate.
- Compare the ratio QTD MSE/TD MSE.
- Confidence bound ± 2 standard error.
- Conclusion:
 - QTD performs better than TD.

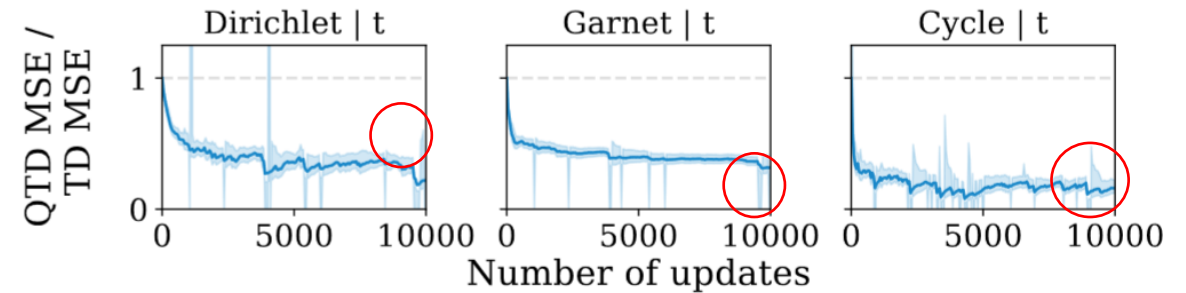


Figure 4. Relative improvement of QTD(128) over TD in mean-squared error against number of updates for all transition structures in Figure 2, with t_2 -distributed rewards.

Broader Conclusion

- “Historically, distributional RL algorithms have often been evaluated in (near-)deterministic environments; this paper also supports the idea that by evaluating algorithms on a wider range of environments, we may obtain a more nuanced view of the strengths and weaknesses of the algorithms at play.”

Key takeaways

- The distributional RL approach does provide a richer information/description about the true action-value.
- Provides benefit in generalizing RL applications.
 - Stochastic state transition/rewards.
- Techniques and conclusions from classical statistics can help in RL.