① Distributional RL

Typical objective fn: Maximize expected discounted return.

In Value-function based,

Bellman Expectation Eq^n $Q(s,a) = E[R(s,a) + \gamma \sum_{s'} Q(s',a')]$ (return)

Bellman Optimality Eq^n $Q(s,a) = E[R(s,a) + \gamma \max_{a'} Q(s',a')]$

$= E[R(s,a)] + \gamma E[\max_{a'} Q(s',a')]$

In Distributed RL, $Q(s,a) = E[Z(s,a)]$

↳ return, random variable from some dist.

$= E[R(s,a) + \gamma Z(s',a')]$

We want to learn this!

② Algorithm (C51)

ⓐ Objective functions for learning

In Value based, tabular: TD error

In Approx setting: MSE

In value-dist, propose W. distance BUT

In practice, KL divergence used instead of W distance.

ⓑ Defining $Z(s,a)$ (N=51 in C51)

Z defined as discrete dist of N fixed atoms (support)

$Z_\theta(s,a) = z_i$ w.p. $p_i = \frac{e^{\theta_i(s,a)}}{\sum_j e^{\theta_j(s,a)}}$ (softmax)

z_i's bounded in $[V_{min}, V_{max}]$

$z_i = V_{min} + i\,\Delta z$ $\quad \Delta z = \frac{V_{max} - V_{min}}{N-1}$

ⓒ Projection step

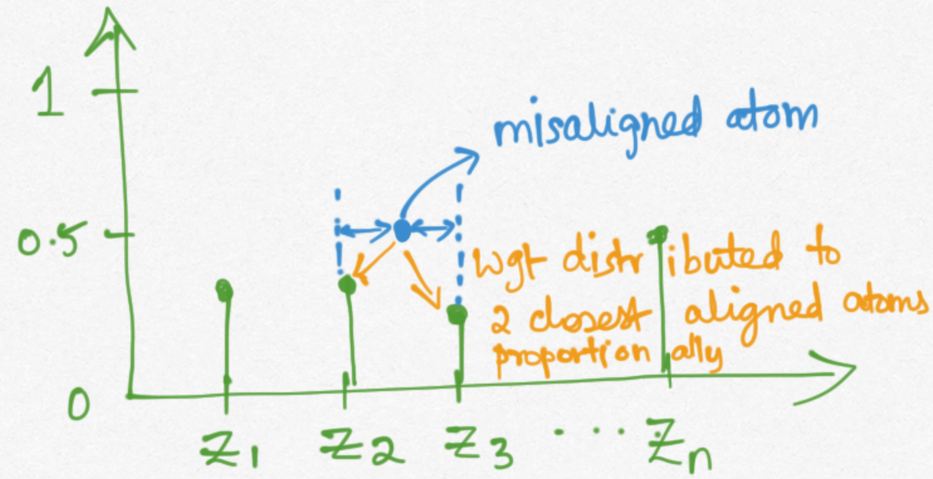
Estimated Z: $Z_i(s,a)$ Target: $R_i + \gamma Z_i(s',a')$

Supports' disjoint, cannot compute KL loss. (goes to ∞)

⇒ Projection: Transfer prob. mass from misaligned target atoms to closest neighboring estimate 'aligned' atoms.

Then compute KL loss. $KL(p\|q)$ b/w target p and estimate q.

Minimize loss using gradient descent.



③ Wasserstein Distance

Dist b/w 2 prob distributions.

$$W(Z_1, Z_2) \equiv \inf_{J \in \Pi(Z_1, Z_2)} \mathop{E}_{(x,y)\sim J} \|x - y\|$$

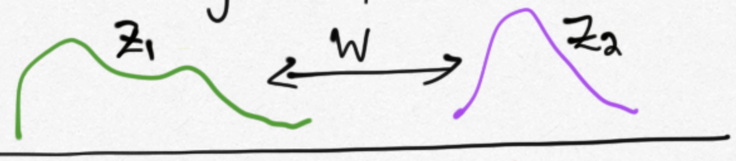
(inf: min)

Π - set of transport plans to move "earth" from one dist to other

J - transport plan w/ min. cost/distance

(x, y) sampled from J

$\|x - y\|$ - dist b/w x & y. Can also be L_p norm to give p-Wasserstein distance



Primal is intractable, joint dist. We can use dual. Dual also has convergence guarantees in policy evaluation case.

$$W(Z_1, Z_2) = \sup_{f \in F_{Lip}} \mathop{E}_{x\sim Z_1}[f(x)] - \mathop{E}_{y\sim Z_2}[f(y)]$$

Or written as: (HOW?)

$$\bar{W}(Z_1, Z_2) = \sup_{s,a} W(Z_1(s,a), Z_2(s,a))$$

This dual form can be proved to be γ-contraction in W in policy eval case.

In control, cannot guarantee.

⑤ Why Should this work?

In value function based,

a) Bellman operator T is γ-contraction mapping

$$\|TF - TG\|_\infty \leq \gamma \|F - G\|_\infty$$

(has convergence properties $F^* = TF^*$, $F_t = TF_{t-1}$)

in tabular setting. Converges to fixed point.

In value distribut", we can use Wasserstein distance, dual form is γ-contraction mapping (PROOF). Only in policy evaluation case.

Banach's fixed pt theorem: $d(T(F), T(G)) \leq \gamma d(F, G)$

(T is contraction mapping in d) d - some distance measure

So, also convergence to fixed point.

[But athors claim W distance cannot be used w/ samples. Hence KL divergence used instead.

But, dual of W distance is in terms of Expectat, which means we can use samples to estimate, & same measure is used in WGANs.]

Implicit Quantile Networks (IQN)

- Prev work (QR-DQN) learnt a distribut^n over returns but still used mean returns for policy.
- IQN learns a risk sensitive policy.

Risk-Sensitive RL - I

- Instead of max E[Returns], max E[Utility].
- E.g.s of Utility : Exp/Log (Returns), Measure that includes variance etc.
- In general, utility f^n satisfies 4 axioms
  - ① Monotonicity : lower cost ⇒ lower risk
  - ② Sub-additivity : Diversification ⇒ lower risk
  - ...
- Can be concave (risk averse) / convex (seeking) or both.

Risk-Sensitive RL - II

- One more axiom of utility f^n is of independence:
  If $X \succ$ (preferred over) Y, then $pX + (1-p)Z > pY + (1-p)Z$
  (Any mixture of X & Z preferred over same mixture over Y & Z).

We can think of utility as mixture of distributions, which we want to learn, to learn a risk-sensitive policy.

(As before, we want to use quantiles to learn distribut^n)

Learning mixture of distributions (MoD)

- MoD can be represented in terms of cdf.
- cdf of MoD is convex combination of its individual distribut^{ns}.

$$F(x) = \sum_i p_i F_i(x)$$

- MoD is also represented as convex comb. of quantiles (in insurance/reinsurance ...) where weighting is some risk f^n. (e.g. VaR)

Distortion Risk Measures (DRM)

- VaR, Conditional VaR are DRMs.
- In General, DRM is a continuous non-decreasing f^n.

$g : [0,1] \to [0,1]$ s.t $g(0)=0$, $g(1)=1$.

Distortion f's.

Dual theory of Choice under Risk

- An alternate objective for risk sensitive policy:
  Instead of maximizing expected utility,
  maximize expectation of distortion risk measure
  $\Rightarrow$ i.e. Distorted Expectation.

Learning a Distorted Expectation

Say $Z(s,a)$ is our usual returns distribut^n.
For some quantile τ, $E[Z_\tau(s,a)]$ is expected return for that quantile.
$\tau \in U(0,1)$

We can reweight this expectat^n w/ β (DRM),

$$Q_\beta(s,a) := E[Z_{\beta(\tau)}(s,a)]$$

[Can think of this as Inverse Transform Theorem: To generate samples from some distribut^n, sample from Unif(0,1), & transform them through inverse CDF (quantile) of the prob. distribut^n.]

$Q_\beta(s,a)$ can be used to follow policy (action-select^n)

$$\pi_\beta(s) = \underset{a \in A}{\text{argmax}}\ Q_\beta(s,a)$$

IQN Loss function

- Similar to QR-DQN we want to minimize loss b/w pairs of quantiles. But we don't have a fixed set of quantiles as in QR-DQN
- Instead, parametric f^n $\phi(\tau)$ used to represent quantiles (embedding layer in NN).
- So we can generate samples, N for τ, N' for τ' & calc. loss.

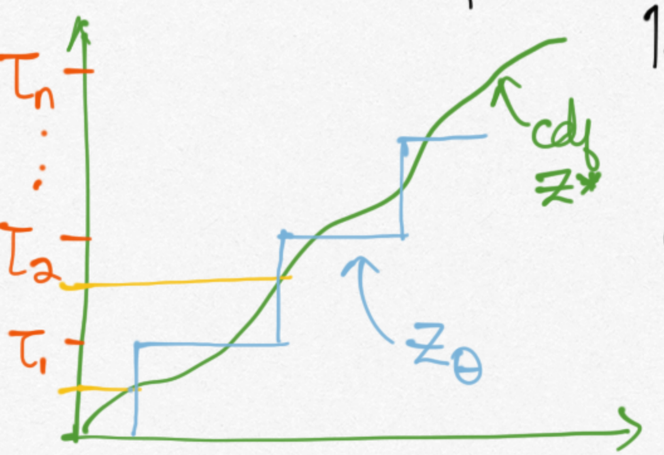
$$L = \frac{1}{N'} \sum_{i=1}^{N} \sum_{j=1}^{N'} \rho^k_{\tau_i}(\delta^{\tau_i, \tau_j'})$$

$\delta^{\tau,\tau'}$ - TD error

- Similarly to select actions, use K samples of $\tilde{\tau}$ & take argmax of mean $Q_{\beta(\tilde{\tau})}(s,a)$
- How to generate these samples? Sample from $U(0,1)$ & reweight using the parametric f^n.

QR-DQN

1. We want to learn distribut^n of return Z
2. We can start with estimate Z_θ (θ: model params) & iteratively improve using bellman operator style updates.
3. It was proved that p-Wass. distance is a γ-contraction mapping & the distributional Bellman operator $T^\pi Z : R + \gamma Z(s',a')$ can converge to a fixed point (using p-Wass in its dual form)
4. $d_p(Z_1, Z_2) \equiv \sup_{s,a} W_p(Z_1(s,a), Z_2(s,a))$
5. Consider some true distribution we want to learn Z^*. If we want to use Wass. dist to learn, Wass dist is the max difference in CDFs' of the 2 distribut^{ns}.



What is the best approx distribut^n?

If we divide cdf into equal prob intervals, say N, then $\tau_i = i/N$. It turns out that best approx Z_θ is at quantile midpoints $\tau' = \frac{\tau_{i-1} + \tau_i}{2}$

So we can learn quantiles using QR.

Quantile Regression

Linear regression estimates "conditional mean" (mean/expectat^n over set of conditions)
If Least absolute deviat^n is used (MAE), we estimate median of target.

QR - we estimate conditional quantiles. (Median is special case, $q = 0.5$)
Given R.V. X: drive time Everett to Seattle
$q \in (0,1)$, $P(X \leq x) = q$
say $P(X \leq 42 \text{ min}) = 0.95$ implies
Prob. of getting to Seattle in ≤ 42 min is 95%

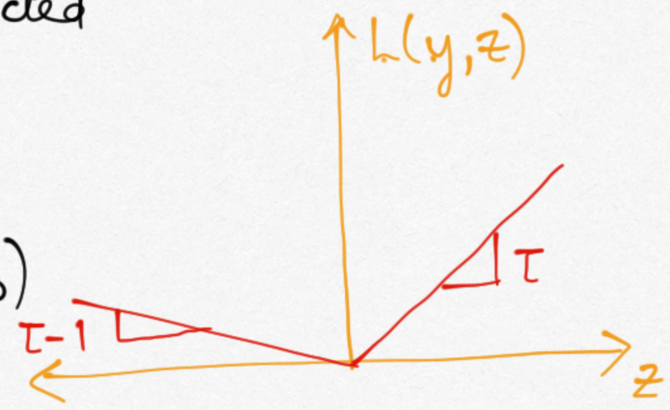
QR Loss (pinball loss)

τ - target quantile, y - actual, z - predicted

$$L_\tau(y-z) = \begin{cases} |(y-z)\tau| & y \geq z \\ |(z-y)(1-\tau)| & y < z \end{cases}$$

or written as $\rho_\tau(u) = u(\tau - \mathbb{I}_{u<0})$

u - residual, $\mathbb{I}$ - indicator fn.



Eg. $\tau = 0.05$, 5%, quantile value = Y of test scores.
Only 5% of students scored below Y
and 95% of students scored above Y.

Let $Y = 10$. If $z = 9$, $L_\tau = (10-9)\cdot(0.05) = 0.05$

$z = 11$, $L_\tau = (11-10)\cdot(1-0.05) = 0.95$

L_τ is higher when we predict on "wrong" side of quantile.

In practice we compute the loss fn between all pairs (θ_i, θ_j) of quantiles of estimated & target distribution.

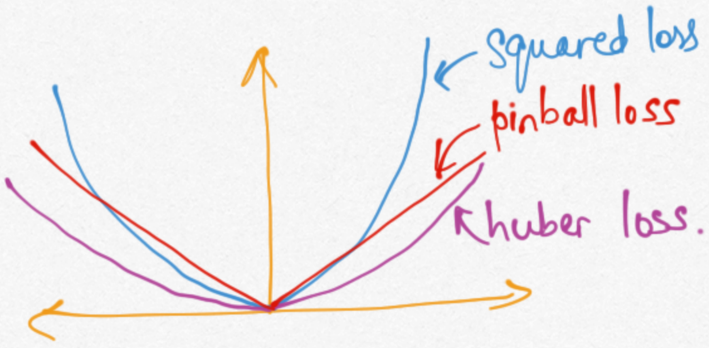
i.e. $\rho^k_{\hat{\tau}_i}(T\theta_j - \theta_i(s,a))$

$T\theta_j$: target (Bellman update)

$\theta_i(s,a)$: current estimate

Huber loss

QR loss is not zero everywhere.
When $u \to 0$, derivative is constant. So authors combine w/ huber loss.



$$L_k(u) = \begin{cases} \frac{1}{2}u^2 & |u| \leq k \\ k(|u| - \frac{1}{2}k) & \text{otherwise} \end{cases}$$

u - residual, $k = 0$ or 1

Quantile huber loss $= |\tau - \mathbb{I}_{u<0}| L_k(u)$