Revisiting Fundamentals of Model-Free Reinforcement Learning

Raja Grewal

raja_grewal1@pm.me

https://github.com/rgrewa1

https://linkedin.com/in/rgrewa1

Sydney, Australia

November 30, 2021

Abstract

Deep reinforcement learning is rapidly advancing the training of immortal agents that will inevitably continue to surpass all human experts in an ever-increasing number of intellectually intensive activities. This accelerating growth has led to the wide-spread adoption of numerous entrenched practices irrespective of algorithm or environment. In this work we directly examine several deeply held convictions for which we can construct alternatives using the current state-of-the-art model-free algorithms: Twin Delayed DDPG (TD3) and Soft Actor-Critic (SAC).

The first is the ubiquitous use of MSE loss functions for critic evaluation as there exists a myriad of alternatives. We find that its use is acceptable as a starting point, but its choice becomes another tuneable hyperparameter with potential to accelerate performance. Research into asymptotic convergence has also highlighted the inadequacy of the Monte-Carlo approach in aggregating mini-batch losses where the empirical (arithmetic) mean may severely underestimate the true mean if the unknown underlying distribution contains rare, but very large outliers, known as fat tails. Their presence is gauged if the tail exponent is less than unity, which we conclusively find to be case, and the properties of a heuristically estimated shadow mean are analysed. Furthermore, multi-step returns in continuous action spaces are found to inhibit learning due to lack of global policy maximisation amplified by geometric target Q-value dampening, which sets the stage to examine whether their puzzling coupling to replay buffer size persists.

For the overwhelming majority of real-world domains, rewards are multiplicative and scale proportionally with the amount existing cumulatively obtained rewards. For this dynamic, losses have an asymmetrically larger effect on performance compared to equal percentage gains. Attention then shifts to identifying the path that maximises the time-average growth rate with probabilities of outcomes becoming less relevant. To accommodate these situations, we show Q-learning, both stochastic and deterministic policy gradient theorems, soft Q-learning, and soft-policy iteration remain relatively unchanged, though will very likely lead to completely different optimisation priorities. Many of these results are also implicitly valid in a subset of non-stationary regimes incorporating all MDPs. The agent now aims to maximise the future compounding growth through the avoidance of steep losses.

This theoretical development is rigorously examined through the creation of a myriad environments that accommodate multiplicative dynamics. The results are validated with known optimal values for gambles involving coin flips, die rolls, and geometric Brownian motion. The utility of maximising time-average growth is then expanded to far more realistic situations involving multiple simultaneous gambles and concepts regarding cost-effective risk mitigation. Overall, provided the rewards can be accurately parametrised, we extend the applicability of the existing field of reinforcement learning to now encompass all conceivable environments.

The code is publicly available at `https://github.com/rgrewa1/nonergodic-rl` [1].

Keywords Reinforcement Learning · Model-Free · Off-Policy · Critic Loss · Extreme Values · Multi-Step Returns · Non-Markovian Decision Processes · Ergodicity · Multiplicative Dynamics · Time-Average Growth Rate

Acknowledgments The Sydney Informatics Hub and the University of Sydney's high performance computing cluster, Artemis [2], for providing the computing resources that have contributed to the results reported herein.

Notation The agents sequential interaction with the environment is characterised by a tuple representing the history $h_t \equiv s_1 a_1 r_1 ... s_{t-1} a_{t-1} r_{t-1} s_t$ where the trajectory develops as $s_1 \to a_1 \to r_1 s_2 \to a_2 \to r_2 s_3 \to ... \to r_{t-1} s_t$.

Contents

1 Can You Access Yourself in Infinite Parallel Universes?

Changes in wealth, heath, and the life of any random individual or institution are more often than not compounding, and are represented by relative changes between values using rates of returns. The overwhelming majority of contemporary decision theory is however formulated under the assumption of fixed (additive) absolute changes in values [3–11]. This approach is invalid when the true nature of the environment is multiplicative where percentage decreases in value are not reverted by equal percentage increases and vice versa. Designing of optimal risk-taking strategies in these domains requires the abandonment of deeply entrenched methods involving the maximisation of 'expectation' values calculated using a probabilistic approach. Instead, the focus shifts towards finding the optimal path that maximises return under the constraint of avoiding steep losses, quantified by the time-average growth rate.

In this section we outline two motivating examples. The first provides a very simple introduction to why the path taken to identical expectation values even under certainty matters when examined using a multiplicative lens. The second illustrates a simple gamble that an investor behaving perfectly in tune with existing decision theory is virtually certain to go bankrupt if they take a probabilistic approach calculating expected returns at each time step when determining how to allocate risky capital. We discuss these in the context of how a random singular individual would behave if they could replay events multiple times or have only chance to pick a path. Then we present a more formal explanation of why payoffs are more important than probabilities. The section concludes with a discussion on implications and motivations behind this work.

1.1 Path Preferences with Identical Expectations

Assume we start with \$100 at time t_0, if we were to first gain \$50 at t_1 and then lose \$50 at t_2, and then vice versa with equal probability, under additive dynamics, the expectation value would remain \$100 at all times. The existing field of economic decision-making dictates that a 'rational' person should be indifferent between both paths. In reality if you were to go outside and randomly ask people without formal training in STEM or economics which path they would prefer, would you get a 50-50 split? Chances are you would not, as the overwhelming majority of the general public would prefer to first gain \$50 rather than lose \$50 [12–14]. Not satisfied with this empirical fact and similarly puzzling inconsistencies, the fields of behavioural economics and behavioural finance were concocted to explain many of these experimental realities by introducing a large variety of cognitive 'biases' that ultimately resulted in labelling the person on the street as "irrational".

Let us now revisit both paths of this scenario using multiplicative dynamics that aim to maximise compounding growth while avoiding steep losses. The first situation with gaining \$50 then losing \$50 can be represented as $100\% \cdot (1 + 50\%) \cdot (1 - 0.33\%) = 100\%$ where the left hand side indicates change in value from the previous value. The second path is $100\% \cdot (1 - 50\%) \cdot (1 + 100\%) = 100\%$. Both paths yield the same final (nill) change in value. Visually this is shown in Fig. 1. Which one avoids large crashes? The first path increases value by 50% and then suffers a -33% loss. The second path starts with a -50% loss and then requires a 100% gain (doubling) to get back where we started. Assuming equal time intervals for both paths, as $\delta t = t_2 - t_0 \to 0$ the preference is likely to be identical as the expectation approach suggests. However, as δt becomes large, empirical evidence conclusively suggest preference for the path with the smaller drawdown [12–14]. The explanation for this that the -50% takes us far closer to complete irrecoverable ruin (of having \$0) and correspondingly we would need a doubling (100%) of value to get back to where we started. This appears to be a concrete art of human nature honed over millenniums of evolution

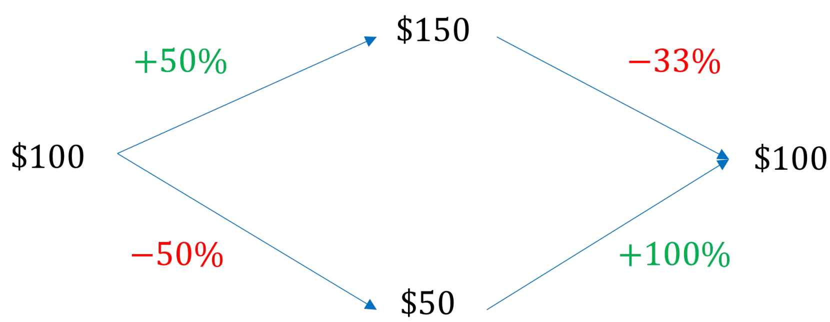

Figure 1: Visual depiction of both paths of the gamble with unchanged expectation values and noticing that to recover from the initial steep loss, a doubling of value is required.

where we rightfully concern ourselves with payoffs, or in other words, prefer “cash in hand”, even though the final outcome remains unchanged.

If we were to modify the absolute change from \$50 $\rightarrow$ \$0, identical preferences would result as the difference between the compounding percentages for both paths would reduce to be negligible. Likewise, as \$50 $\rightarrow$ \$100 there would be even stronger preference for the first path. Therefore, while the two paths are exactly equivalent using additive dynamics, under multiplicative dynamics there can exist distinctly superior paths. Preference is then for the path that maximises reward under the constraint of avoiding ruin for an individual that only has once chance to decide on which direction to take. This is opposed to having the luxury of being able to repeat the event multiple times and taking the averaged outcome, obtained by uniformly averaging the outcome of each repetition, that is, the Monte Carlo approach. The Monte Carlo method is assumed to converge in the infinite repetition limit to the ‘expectation’ value as the repeated trials should ultimately generate and accurate estimate of the underlying, generally unknown, probability distribution.

Notice one curious fact with multiplicative dynamics, at no point did we use probabilities when deciding which path is optimal. This is because probabilities do not matter when a singular individual or institution evaluates which path to take on a gamble if the payoff is non-linear which we will show in the next section. In these situations, probabilities are completely artificial constructions based on what would result if we could replay an event from the start an infinite number of times, with expectation value not necessarily having any connection to the final result a random singular entity achieves. For these situations, the only thing that matters is the payoff structure and all strategies should be constructed based on where in a random world, a random singular individual can conceivably end up in the infinite time limit.

The label “expectation value” $\mathbb{E}[x]$ is hence a misnomer, it should be better named “ensemble average” $\langle x \rangle$ more in line with its actual origin in statistical mechanics [5, 8, 14–17]. Therefore, in gambles where leverage can be controlled to either amplify and reduce the payoff, calculating probabilities and hence expectation values will be shown to not only be entirely meaningless, but an absolutely absurd activity for a random individual. This is because we are solely concerned with how any randomly selected individuals valuation changes over time, not the average final valuation across all individuals. The former is what any sensible person would consider their performance to resemble, the latter is very prone to severe overestimation bias.

1.2 Optimal Leverage for a Simple Coin Flip

A standard example used to highlight failure of conventional wisdom when attempting to maximise reward signals using expectations involves betting on a biased coin [8, 9]. This also highlights that the Gamblers Paradox is not really a paradox as it incorrectly assumes we have access and can pool returns from infinite versions of ourselves. A corollary is that observable world class gambling performance obtained via maximising expectation values is likely nothing more than the living embodiment of survivorship bias, that is, the overwhelming majority of attempts to achieve such success through this method ends in failure, though there exists a very small minority who succeed due to pure 'luck'. Therefore, for any random investor, utilising this approach is a fool's errand.

Consider a gamble where a 'rational' investor initially has a portfolio of worth $V_0 = \$100$ and has the option to place a leveraged bet on a coin flip. The possible changes in wealth for this game are held constant across time where the payoffs at each flip are

$$V_{t+1} = \begin{cases} (1+50\%)l_t V_t + (1-l_t)V_t = (1+0.5\cdot l_t)V_t, & p_u = \frac{1}{2} \\ (1-40\%)l_t V_t + (1-l_t)V_t = (1-0.4\cdot l_t)V_t, & p_d = \frac{1}{2} \end{cases} \tag{1}$$

and $l_t \in \mathbb{R}$ is the amount of leverage the investor can apply and either outcome is equally probable. This represents an option since the investor can select $l_t = 0$ and so takes no risk at time t. Note this clearly indicates multiplicative dynamics as at each time step the worth of the portfolio increases by ratio rather than an additive fixed amount. The returns on investment (payoffs) are then

$$r_{t+1} = \begin{cases} +0.5\cdot l_t, & p_u = \frac{1}{2} \\ -0.4\cdot l_t, & p_d = \frac{1}{2} \end{cases} \tag{2}$$

Given the payoffs remain unchanged and the game can be played indefinitely, the sequence of investment management decisions is then:

1. Should investors consider playing this game at $t = 0$?
2. If so, what direction should they place their bet, long ($l_0 > 0$) or short ($l_0 < 0$)?
3. How much leverage l_0 should they apply?
4. Should investors consider playing this game at $t = 1$?

   $\vdots$

This either continues ad infinitum ($t \to \infty$) or when the game ends where the bounds are $V_t \in [0, \infty)$. The lower bound is strict as once bankrupt the game forcible concludes and there is no possibility of recovery. Let us also define a 'stop-loss' $V_{\min}$ where if $V_\tau \leq V_{\min}$ the investor stops playing at time $t = \tau$ to live to fight another day. The exact value of $V_{\min} = \lambda V_0$ will be dependent on an investors preference for $\lambda \in [0, 1)$ and so we can say $V_t \in (V_{\min} - \varepsilon_\tau, \infty)$ where the lower bound is $V_\tau = V_{\min} - \varepsilon_\tau$ is time dependent. Notice that for some $l_t \in [l_{\min}, l_{\max}]$ where $l_{\min}$ and $l_{\max}$ are such that it impossible to go bankrupt as at worst $V_t \to V_{\min}$, the investor is technically always in the game and can 'bounce back' if the favourable state occurs sufficiently often as $t \to \infty$. Similarly, for leverages outside this bound, losses can not only exceed the predefined stop-loss, but also the initial deposit leaving the investor in debt.

How would a 'rational' investor then answer these questions? They would calculate the expectation value of course! This yields $\mathbb{E}[r_{t+1}] = rl_t = 5\% \cdot l_t \ \forall \ t$. If they are feeling especially fancy they would also calculate the 'standard' deviation of returns $\sigma_{t+1} = 45\% \cdot |l_t| \ \forall \ t$. In this special case $\sigma_{t+1} = \mathrm{MAD}_{t+1}$ as there are only two components. MAD is discussed in detail in Section 3.9. Therefore, since $\mathbb{E}[r_t] \neq 0$, the investor should consider playing the game at all times. Next, as $\mathbb{E}[r_t] > 0$ the investor would clearly consider going long with $l_t > 0$ for all time. The next question is by far the most important, how should they calculate $l_t \ \forall \ t$?

To answer this we consider four types of rational investors increasing in their level of sophistication.

(i) Investor 1: This first class of investors that only undertake in favourable bets and select leverage $|l_t| \leq 1$ based on maximising return while avoiding bankruptcy with certainty. Therefore, they have $|l_t| \in (0, 1]$, and because $V_t > 0 \ \forall \ t$ they will have $V_{\min} = 0$ since they can always 'bounce back'. As these investors believe this is a favourable bet there is no reason it should not be played indefinitely, however for computational purposes we truncate time to a fixed maximum investment horizon T. A single investors compounding return is then

$$V_T = V_0(1 + \bar{g})^T = V_0 \prod_{t=0}^{T} (1 + rl_t) \tag{3}$$

where V_T is their final portfolio value and the time-averaged growth rate is $\bar{g}$. For a fixed leverage $l_t = l$, they 'expect' $\bar{g} = rl$ in the long-term as the horizon $T \to \infty$. Therefore, the choice of leverage appears to be trick question as this appears to be a situation where more leverage is superior and having a stop-loss is not necessary since the game can always be played. The supposed optimal leverage is then simply $l^* = 1$.

(ii) Investor 2: These investors improve on the first by applying the optimal time-dependent leverage l_t^* while always maintaining a fixed $V_{\min} = \lambda V_0$ for $\lambda \in [0, 1)$ to keep their maximum loss capped at all times. By re-balancing at each step, these investors ensure optimal risk-reward while not being stopped-out. To prevent losses from exceeding the stop-loss we institute a constraint on leverage, where for $|l_t| > 1$ the change in portfolio value V_t from a single time step to V_{t+1} can be very sizeable. If the investor is long and wrong, we have danger threshold $l_t > \frac{5}{2}\left(1 - \frac{V}{V_t}\right) > 0$, if short and wrong, the threshold is $l_t < 2\left(1 - \frac{V}{V_t}\right) < 0$, where $V = V_{\min}$ to exceed stop-loss and $V = 0$ to exceed deposit. Therefore for a rational investor not to be stopped out in a single step, they should have time-dependent leverage bounds

$$l_t \in \left(-2\left(1 - \frac{V_{\min}}{V_t}\right), \frac{5}{2}\left(1 - \frac{V_{\min}}{V_t}\right)\right) \tag{4}$$

This shows that as $V_t \to \infty$ they are able to take substantially more risk at a maximum of $l_t \to \frac{5}{2}$ than if $V_t \to V_{\min}$. These investors have therefore successfully constructed open bounds on the leverage they can take at any point. Notice very importantly that the bounds are independent of the probability of either outcome, the only thing that matters are the final payoffs. The question still remains, on what exactly the leverage should be when taking the bet. Since the payoffs are fixed, we formally arrive at the optimisation problem

$$l_t^* = \arg\max_{l_t} V_{t-1}(1 + rl_t) = \arg\max_{l_t} l_t \tag{5}$$

under the constraint of Eq. (4) that must be solved at each step. The solution to this is clearly to approach $l_t^* \to \frac{5}{2}\left(1 - \frac{V_{\min}}{V_t}\right)$ since $r > 0$, that is, take the maximum amount of leverage possible while ensuring they are not stopped out. We also assume leverage is available at no additional cost. These investors therefore simply calculate l_t^* at each step and bet $l_t^* V_{t-1}$ portion of their existing portfolio at each step.

(iii) Investor 3: Improving on the second, these investors apply the same leverage principles but institute a 'rolling' stop-loss wherein

$$V_{\min,t} = \begin{cases} \lambda V_0, & \text{if } V_t \leq V_0 \\ V_0 + \phi(V_t - V_0), & \text{otherwise} \end{cases} \tag{6}$$

attempting to retain a fixed portion $\phi \in [0, 1)$ of their winning at each time step if they are profitable. The general leverage bounds for any fixed binary payout are then

$$l_t \in \left(-\frac{1}{|r^u|} \left(1 - \frac{V_{\min,t}}{V_t} \right), \frac{1}{|r^d|} \left(1 - \frac{V_{\min,t}}{V_t} \right) \right) \tag{7}$$

where the r^u and r^d are the static up and down returns, and the result is again independent of probabilities. For optimal leverage, if $|r_u| > |r_d|$ the upper bound is approached and vice versa. Through this type of judicious risk management, they expect that after they cross the $V_t > V_0$ threshold, they will constantly keep build their portfolio wealth as time goes on since this is favourable bet. For the proposed gamble, half of investors should start at this capital accumulation stage while a portion of the others will also eventually reach this stage.

(iv) Investor 4: The final category of risk-takers involves those for which everything is a variable that they optimise at each time step. Through experience and instinct, they will select optimal the $\lambda_t^*, \phi_t^*, l_t^*$ to maximise returns, expressed very generally at each step as

$$\lambda_t^*, \phi_t^*, l_t^* \in \arg \max_{\lambda_t, \phi_t, l_t} V_{t-1} \left(1 + g_t(\lambda_t, \phi_t, l_t)\right) \tag{8}$$

with it not being clear whether a unique solution exists. The growth rate g_t is the key variable that incorporates changes in wealth. Explicitly, from initiation $t = 0$ the problem is to obtain the set variables

$$\begin{aligned} \lambda_1^*, \phi_1^*, l_1^*, ..., \lambda_T^*, \phi_T^*, l_T^* &\in \arg \max_{\lambda_1, \phi_1, l_1, ..., \lambda_T, \phi_T, l_T} \prod_{t=1}^{T} (1 + g_t(\lambda_t, \phi_t, l_t)) \\ &\in \arg \max_{\lambda_1, \phi_1, l_1, ..., \lambda_T, \phi_T, l_T} \sum_{t=1}^{T} \ln |1 + g_t(\lambda_t, \phi_t, l_t)| \\ &\in \arg \max_{\lambda_1, \phi_1, l_1, ..., \lambda_T, \phi_T, l_T} \sum_{t=1}^{T} (1 + g_t(\lambda_t, \phi_t, l_t)) \end{aligned} \tag{9}$$

since the logarithm is monotonic. The final performance is then measured by the time-averaged (exponential) growth rate $\bar{g}$ defined by

$$\ln |1 + \bar{g}| = \frac{1}{T} \ln \left| \frac{V_T}{V_0} \right| \ \rightarrow \ 1 + \bar{g} = \exp \left[\frac{1}{T} \ln \left| \frac{V_T}{V_0} \right| \right] \tag{10}$$

which has no explicit dependence on any variable, incorporates all effects, and is a clear measure of average change in wealth achieved per time step. To select the optimal set from Eq. (9) across all strategies S, the superior set of variables are determined by strategies $S_i \in S$ where $\bar{g}_i \in \max_{S_j \in S} \bar{g}_j$ which need not be unique. Identically we can identify the optimal set by looking at the final value $\bar{g}_i \in \max_{S_j \in S} V_T^j$. Observe the connection between the time-average and the geometric average rate-of-return $\bar{g} = \sqrt[T]{V_T/V_0} - 1$. This link is crucial to represent the coupling of returns on investment over time.

Another way to represent this is in terms of maximising the future exponential growth rate at each time step

$$g_{t+1} = \frac{\Delta \ln V_t}{\Delta t} \tag{11}$$

which has reduced computational feasibility but can be used for simpler gambles. Next, given that in general $\bar{g} \neq \mathbb{E}[r]$, the difference between them is expressed as

$$\bar{g} = \mathbb{E}[r] + \nu(\lambda, \phi, l, \sigma) \tag{12}$$

where $\nu(\cdot)$ is referred to as the 'volatility tax' [11, 18–24]. This can be interpreted as the reward or punishment associated with certain variables. Clearly for any gamble where $0 \leq \sigma_1 < \sigma_2$, $0 \leq l_1 < l_2$, $0 \leq \lambda_1 < \lambda_2$, and $0 \leq \phi_1 < \phi_2$, we have the following general relations

$$\nu(\sigma_2) < \nu(\sigma_1) \leq 0, \;\; \nu(l_2) < \nu(l_1) \leq 0 \tag{13}$$

$$\nu(\lambda_2) > \nu(\lambda_1) \geq 0, \;\; \nu(\phi_2) > \nu(\phi_1) \geq 0 \tag{14}$$

The first two are expected as large variability in returns due to either the inherent gamble or increasing leverage negatively effects $\bar{g}$. The latter two indicate that less tolerance to loss will positively effect $\bar{g}$. More aggressive risk-taking will lead to a volatility tax 'cost', while having less tolerance to loss will yield a volatility tax 'benefit'. The exact nature and impact of volatility tax on the longer-term return will be entirely dependent on the gamble and the parameters that can be controlled. With this formulation the primary concern is simply the final outcome $\bar{g}$, the success of strategy is then measured by its simulated performance across many trials. This is largely an abstract intractable problem but serves to highlight the decision-making process.

For the gamble at hand, we can optimise Eq. (11) directly assuming no stop-loss or retention ratio, that is, taking the same fixed blind betting strategy across all time as with the first investor. This yields

$$\frac{\mathrm{d}g_{t+1}}{\mathrm{d}l_t} = \frac{\mathrm{d}}{\mathrm{d}l_t}\bigg(p_u \ln|V_t(1 + r_u l_t)| + p_d \ln|V_t(1 + r_d l_t)|\bigg) \tag{15}$$

$$l_t^* = -\frac{(p_u r_u + p_d r_d)}{r_u r_d} = \frac{p_u r_u - p_d|r_d|}{r_u|r_d|} = \frac{p_u}{|r_d|} - \frac{p_d}{r_u} \tag{16}$$

$$p_u \geq \frac{|r_d|\,(l_t r_u + 1)}{r_u + |r_d|} \tag{17}$$

where the last line indicates the minimum up-probability required for any given fixed l_t $\forall$ t to be profitable. Hence for our given parameters they find a fixed $l_t^* = 25\%$ $\forall$ t to match $p_u = \frac{1}{2}$ in order to maximise growth $\bar{g}$ in the infinite time limit while seemingly having a heavily reduced $\mathbb{E}[r] = 1.25\%$ $\forall$ t.

The optimal leverage in Eq. (16) is a very well-known result called the Kelly Criterion [4]. However, its origins are almost 200 years older in the now translated work of Daniel Bernoulli [3]. The ultimate purpose of this leverage is maximising the median wealth across an ensemble of investors of any size while there also exists fractional Kelly betting to maximise across an arbitrary percentile of total investors [11]. Overall, the difference between this investor and the previous three is their distinct focus on solely on the improving geometric average, that is, the evolution of wealth over time as opposed to the 'expected' wealth over an ensemble.

Let us now simulate this gamble with $N = 1,000,000$ random investors for $T = 3,000$ steps from each of the four categories. For each category, all N investors will utilise exactly the same strategy and so the difference in final portfolio values will be purely due to the random sequence of payoffs they experience, in other words, the cards they were dealt. The goal is then to accurately assess how robust each of the strategies are for the average investor in N. To accomplish this, we split the sample into two sub-samples, the first contains the bottom 99.99% of adjusted average values V_T^A while the other consists of the top 0.01% of performers with maximum average V_T^M for each time step. This conservatism avoids upwards bias present when working with random variables since it removes the impact of unlikely cases, a total extreme being $V_T = V_0(1+r_u)^T$. The choice of large N therefore ensures we capture the behaviour any random rational investor, using T steps will be shown to sufficient in determining the long-term outcome, and removing the top 100 investors will give us a more complete picture for performance for $9,999$ out of $10,000$ investors.

For the first type of investor the results for fixed leverage ranging from as little as 10% all the way to betting entirely 100% on favourable bet are shown in Fig. 2. Due to the differences in magnitude, we present results in terms of the log averages $\log_{10} V_t$. Recall that changes in tick increments on log axes represent changes by magnitudes of 10x for each increment.

These are a puzzling results as Fig. 2(a) shows that for $l \gtrsim 75\%$ the $V_T^A \to 0$ with certainty at rates increasing with leverage. The optimal constant leverage is found to be in the range $l^* \in [35\%, 45\%]$ so we take $l^* \approx 40\%$ which is incredibly difficult to predict a priori. The expected optimal leverage of unity leads to complete loss of capital when they 'expected' $V_T^A = V_0 \cdot 1.05^T \sim \$10^{63}$ or $\log_{10} V_T^A \sim 63$. In Fig. 2(b) we see the box plot distribution for log values at maturity across leverages that further reveals even for $l \gtrsim 50\%$ at least half the investors are below the starting value and that only for $l \lesssim 15\%$ are they all guaranteed to make a profit. We therefore refer to $l^* \approx 40\%$ as the optimal leverage and $l_s^* \approx 15\%$ as the safe leverage. We also plot the trajectory from the initial value $(2, 2)$ of the log average maximum value against the log average adjusted value in Fig. 2(c) that reveals how astronomically larger these top values are in magnitude. It is these abnormally large values this that drive the largest contributions to the mean forcing it to not approach zero instantly, and so removing them is crucial to accurately assess performance. Finally, in Fig. 2(d) we show all three MADs and standard deviations (STD) added to the mean, highlighting why STD is inferior to MAD as large outliers hugely inflate volatility.

Overall, we arrive at a completely non-trivial result where the optimal decision for the first investor generates $\bar{g} \approx 0.56\%$ return per step while always holding on to 60% of their capital at all times, while 'expecting' $\mathbb{E}[r] \approx r \cdot 0.40 = 2.00\%$. The volatility tax is $\nu = \bar{g} - \mathbb{E}[r] \approx -1.44\%$. This is an astounding result, the asymmetric effect of losses offset the majority of the 'expected' upside, they receive only 25% of what they predicted. Similarly for the safe leverage $\bar{g}_s \approx 0.32\%$ with $\nu = -0.43\%$ while 'expecting' $r \cdot l_s^* = 0.75\%$, and so steep losses erase 66% of the 'expected' gain with everyone still benefiting from the gamble.

This is not only true for $|r_d| < |r_u|$ but is even more significant for the opposite case. For fixed leverage across time, this is still the optimal play as it maximises time-averaged return in the long-run through the avoidance of steep losses. Effectively, we can say in reality $\bar{g} \ll \mathbb{E}[r]$ and conclude that maximisation of the 'expected' return is not the quantity of interest, rather we only care about the performance of random investor over time. Notice to arrive at this conclusion, at no point were probabilities calculated, only the paths were simulated.

Moving on to the second category of investors that automatically calculate optimal leverage for a range of fixed stop-loss values and are the kinds of people that calculate standard deviations. The results for any $\lambda \in [0, 1)$ for

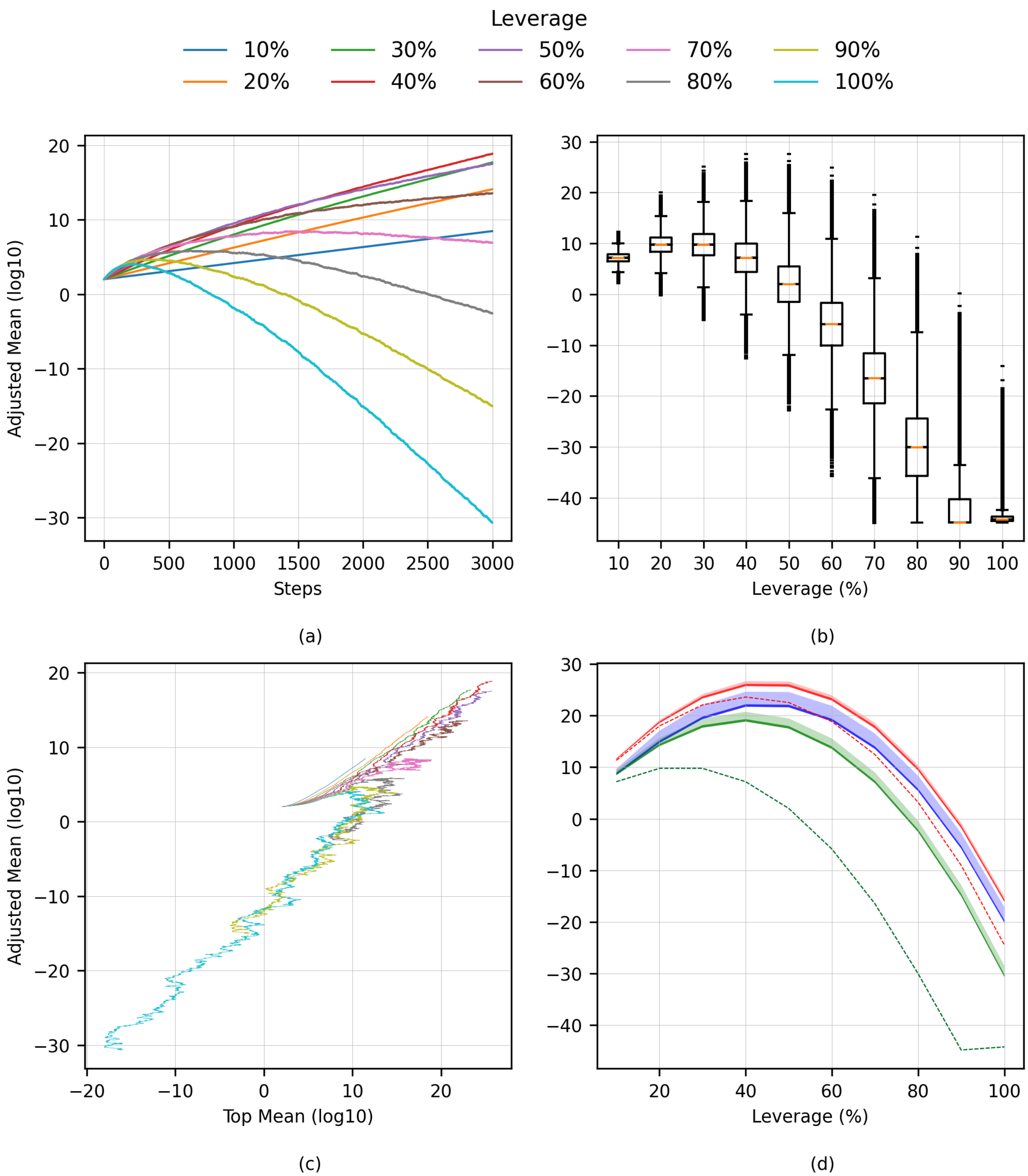

Figure 2: Summary of investor one results: (a) The trajectories of the adjusted (bottom 99.99%) log means for various leverages showing non-ergodicity at display, (b) Box plots of the distribution of adjusted values at maturity of $T = 3,000$ step for various leverages. Note $\sim 10^{-40}$ is not a lower bound, rather it is a numerical accuracy limit, (c) The trajectory of the adjusted mean along with the top (0.01%) log mean for various leverages all initiated at $(2, 2)$. Notice in all cases how astronomically larger the top values are compared to the adjusted values, and (d) Plots the medians (dotted), MAD (dark shading), and STD (light shading) added to the mean for various leverages across all three subgroup complete (blue), top (red), and adjusted (green). Note the medians of adjusted and complete are identical. Observe how STD grossly inflates by several orders of magnitude the true volatility of the gamble in the profitable regions due to the small number of high performers in every group.

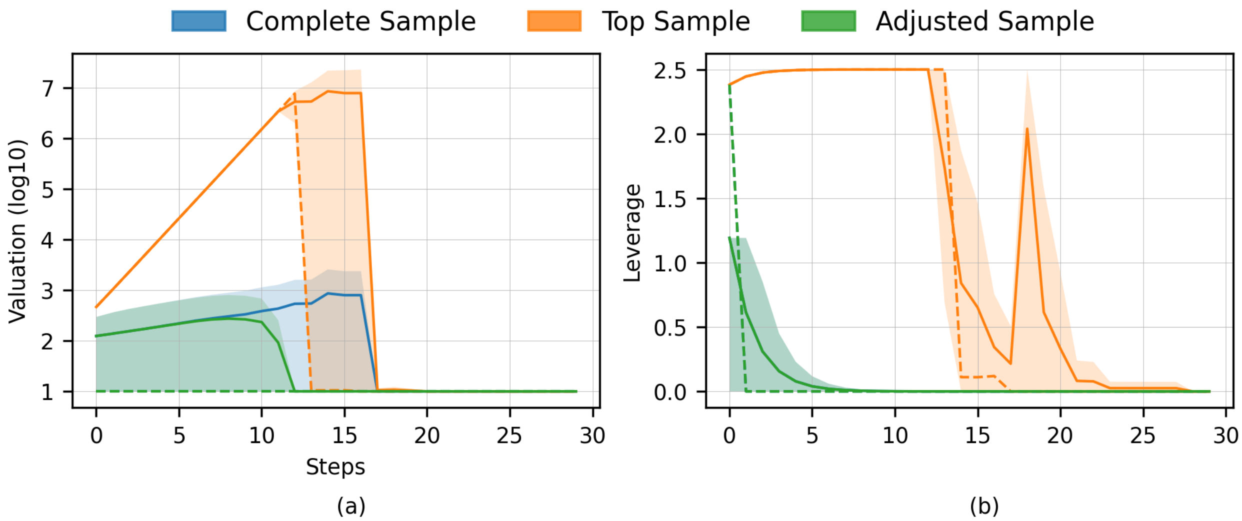

Figure 3: Summary of investor two results: (a) Reveals the mean values for each of the three subgroups highlighting how the top 0.01% are able to misrepresent the complete samples' performance by temporarily greatly skewing it upwards, and (b) The mean leverages for the same groups noting the inevitable crash to zero. The solid lines indicate the (arithmetic) means, dashed lines are the medians, and the shaded bounds represent one MAD about the mean. Note for leverage the adjusted sample is identical to the complete sample.

$V_{\min} = \lambda V_0$ are shown in Fig. 3. We see in Fig. 3(a) that $V_t \to V_{\min}\ \forall\ \lambda$ with complete certainty (zero MAD) within roughly 25 time steps and the approach rate of V_t^A is significantly quicker than V_t^M. Notice that this seemingly optimal strategy is able to intermediately generate enormous profits for the top 0.01%, up to $V_t^M \sim \$10,000,000$, while inevitably these also eventually crash to their stop-loss. While they may 'cash out' at any time, it is 'irrational' to do so since this is a favourable bet, and the game should be played till maturity. This also reveals how heavily such a small minority can raise total average. The changes in optimal leverage in Fig. 3(b) show this again show this behaviour where the maintenance of the maximum leverage by the 0.01% occurs only for roughly the 15 intermediate steps coinciding with the huge valuations. Observing the sample medians, we can notice that they provide far more early indications of the longer-term behaviour as leverage almost immediately drops to zero for half the sample due to the highly leveraged nature of the bet due to the exceeding high leverage on the first step. Therefore, these investors' strategy also fails to consistently increase valuations over time.

Why does this occur? Recall the example in Section 1.1 for why paths to the same valuation matter. In this case we have again equally probable up and down paths, both expressed as $100\% \cdot (1 + 50\%) \cdot (1 - 40\%) = 90\%$ and $100\% \cdot (1 - 40\%) \cdot (1 + 50\%) = 90\%$. Notice the final change in value is a 10% decrease from the initial value. For no change in value we must have $100\% \cdot (1 + 50\%) \cdot (1 - 33\%) = 100\%$ and $100\% \cdot (1 - 40\%) \cdot (1 + 66\%) = 100\%$ if the up and down state occur first respectively as shown in Fig. 4. In a compounding (multiplicative) world, this asymmetry that lies at the heart of everything. We need a 66% gain to recover from a loss, and a -33% loss to revert a gain. Due then to the power of compounding over time, the losses have an asymmetrically larger effect on portfolio worth.

The source of this error lies in the fundamental assumptions of Eq. (5) where they express optimal leverage to be linear in 'expected' return. As such, they lose all sense of the multiplicative nature of this process. This can be considered identical to case of incorrectly using a linear utility function for investor wealth when a logarithmic function in Eq. (11) would be far more appropriate as it correctly penalises large steep losses [5–10].

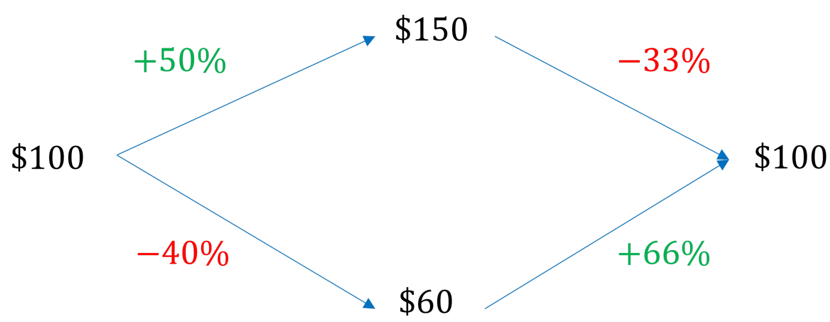

Figure 4: Visual depiction of the simple gamble for two paths to leading to no change in initial value, highlighting the pitfalls of using expectation values for multiplicative processes. Based initially on the example in [8, 9].

The third category of investors proceeds unaware of the performance of the previous two and are the kind of people that use words like “alpha”. Their modified approach is based on calculating optimal leverage as function of the fixed tuple (λ, ϕ) at each step. With this prudent structure they seek to steadily accumulate wealth by betting only a fixed portion of their winnings at each step. In Fig. 5(a-b) we show the results of the grid search using $\lambda \in [0, 1)$ and $\phi \in [0, 1)$ represented with density plot at maturity T. The optimal value are found to be in the vicinity $(\lambda^*, \phi^*) \approx (5\%, 85\%)$. The key variable being then retention level ϕ as there appears to be a steady rise in performance as it increases reaching a maximum at holding on to 85% of your achieved profits at each time step when calculating the leverage for the next step. The stop-loss at 5% of V_0 is also clearly understood as it gives the investors the greatest chances of escaping the $V_t > V_0$ threshold while not being stopped out. Figures for final leverage, MAD and STD look identical in density. The optimal leverage appears to converge around $l_T^* \approx 0.18$ and the adjusted MAD for this grid search is once again comparable if not and order of magnitude larger than adjusted averages as with first investor, highlighting that there is a very real possibility of still going bankrupt.

This however stands at stark contrast to Fig. 5(c-d) for median investor wealth using this strategy. Here we observe downright abysmal performance wherein any retention less than 90% is leads to unchanged wealth for more than half the sample. The safe (median) leverage is found at the boundary $(\lambda^*, \phi^*) \approx (5\%, 95\%)$ to be $l_{s,T}^* \approx 0.12$. Hence we extrapolate that retaining the majority of your wealth and only betting an infinitesimal amount would be advised. We can therefore observe that maximising the median is more pivotal than the mean.

So how did the first three investors perform? The first two would be catastrophically crushed by an entirely risk averse individual that keeps the \$100 in their pocket. The third will achieve outstanding adjusted mean performance of $V_T^A \sim \$1 \cdot 10^{18}$ with $\bar{g} \approx 0.54\%$ and $\nu_T = \bar{g} - rl_T^* \approx -0.37\%$. Despite this, they still 5x underperform the very simple strategy with fixed optimal leverage $l^* = 0.40$ with no stop-loss or retention that achieves $V_T^A \sim \$7 \cdot 10^{18}$ with $\bar{g} \approx 0.56\%$ and $\nu \approx -1.44\%$.

In terms of maximising the median wealth, the third investor obtains an adjusted median wealth of $V_{s,T} \sim \$6 \cdot 10^3$ with $\bar{g}_s = 0.06\%$ and $\nu_T = \bar{g}_s - rl_{s,T}^* \approx -0.54\%$. Which also losses to a fixed safe leverage $l_s^* = 15\%$ holding $V_T^A \sim \$10^{11}$ with $\bar{g}_s \approx 0.32\%$ and $\nu = -0.43\%$.

We emphasise how this seemingly small difference in $\bar{g}$ compounds over T to result in such a large difference. For $T = 5,000$, investor three underperforms by 100x compared to the optimal mean leverage which is then reduced to $l^* \approx 35\%$. What we see is that as $T \to \infty$, leverage l^* decreases as we can see in the time evolution of Fig. 2(a) which is a simple resolution of the Gambler Paradox.

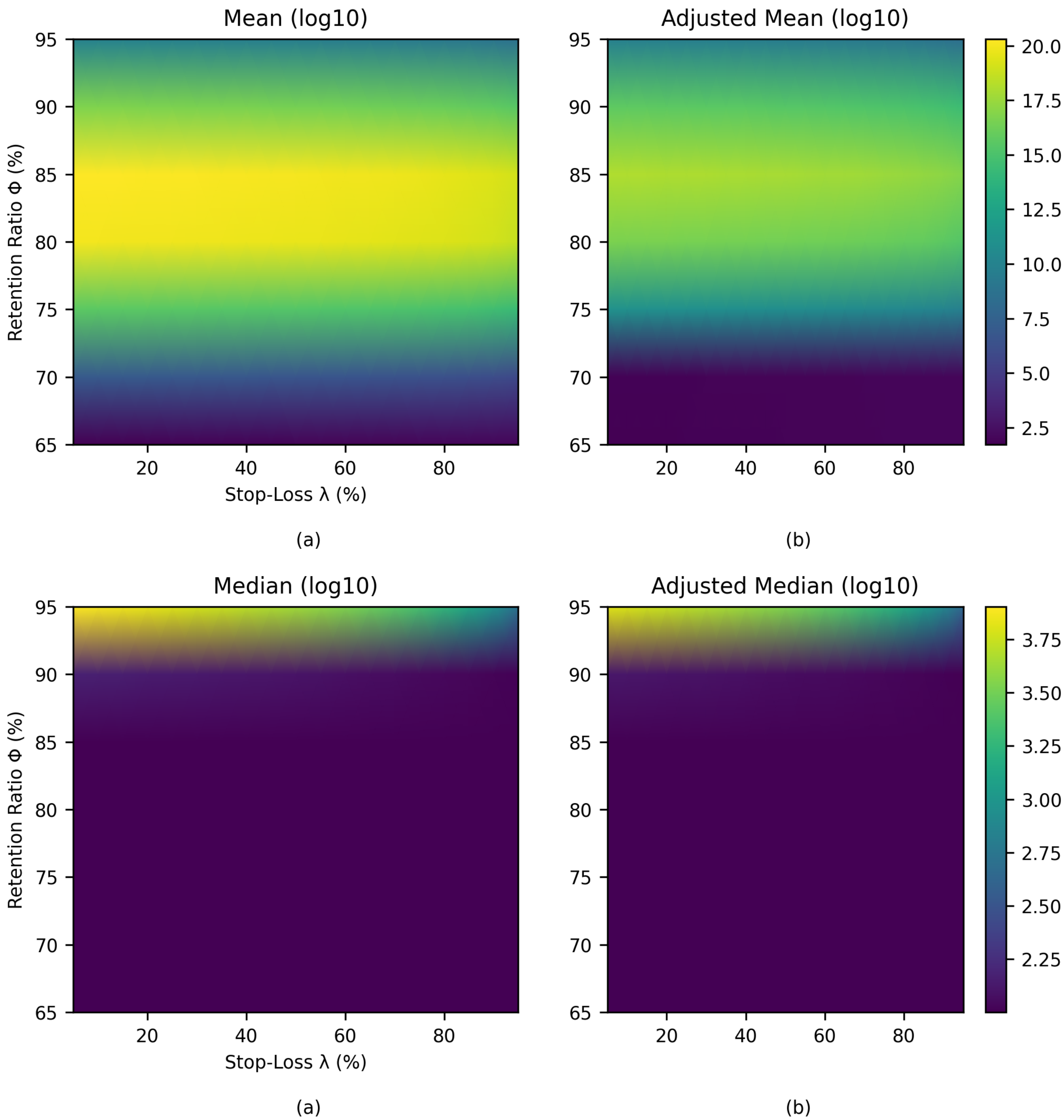

Figure 5: Investor three final valuations as a function of retention ratio and stop-loss at maturity of 3,000 steps summarized in terms of valuations. Top row: mean (a) and adjusted mean (b). Bottom row: median (c) and adjusted median (d). Observing that maintaining a fixed retention ratio of 85% of all winnings at each time step and betting a portion of the remainder appears to generate the highest mean valuations across all stop-loss levels. While the medians reveal that only incredibly conservative approaches of greater than 95% retention result in a clear wealth gain. Note each row shares a common scale.

What we see here is that the highest valuation is achieved by maximising $\mathbb{E}[r] + \nu$ across all time. This underperformance is puzzling, but the explanation has to do with short-cut learning, that is, the sophisticated investor gets the correct result, but achieve it using the wrong logic. Their (rightful) success is due entirely to their ironclad (fully automated) risk management, where even on a favourable bet, they have minimal tolerance for steep losses and so limit their leverage so that they accumulate wealth slowly. This prevents them from losing value even though the strategy they have implemented is identical to the completely failing second investor.

We now turn our attention to the final fourth investor that solves the abstract Eqs. (9, 11). The simple betting strategy may form a non-unique solution to these equation. It is this type of decision-making we are interested in obtaining as the expectation value-based methods have been shown to wholly inappropriate. For the optimal fixed $l_t^* = 25\%$, they obtain $V_T^A \sim \$10^{16}$ with $\bar{g} \approx 1.09\%$ and $\nu = \bar{g} - \mathbb{E}[r] \approx -0.16\%$. This is strategy has the lowest volatility tax and so would be the optimal strategy for any random investor in the infinite time limit. This result effectively generalises and solves the problem outlined in [8, 9].

Why does the simple blindly betting 25% of wealth at each step with no stop-loss or profit retention strategy outperform all others? The reason is that the first three investors do not have any understanding on how to take risks in a compounding world. By conflating probabilities with payoffs, the root of their problem lies again in using expectation values which are only valid in additive environments. Probabilities have absolutely no bearing on the real-world performance when the change in value is multiplicative. In these cases, when a random investor calculates an expectation value, they are implicitly announcing via megaphone that they have the ability to pool results across the entire sample N and receive the average which we have shown can be grossly inflated by the performance of the 'lucky' top 0.01%. This is presumably accomplished by the singular random investor travelling to N parallel universes and forging a contractual agreement with all themselves to split the winnings evenly amongst each selves. However, for investors confined to a singular reality, the only feature of any concern is the payoff structure with strategies evaluated using time-average growth rate $\bar{g} = \mathbb{E}[r] + \nu$ of the bottom 99.99% as it accurately reveals the outcome for any random singular investor.

Regarding this optimal leverage from the Kelly criterion, in the density plots contained within Fig. **??** its values are shown for a range of up and down returns for various probabilities. We see observe that only for the most minute of down returns coupled with very high up returns do we expect optimal leverages in vicinity of unity for equal probability. As we reduce the chance of a up move, the optimal leverage becomes negative, that is, go short, while for higher probabilities we can almost universally expect long positions to be optimal. Recall again that these optimal leverages are for the median sample so that at least 50% of participants will maximise wealth with this approach [11].

Note we do not form explicit risk-reward ratios such as Sharpe, as these are entirely worthless and misleading metrics for past performance. This is because they are independent of leverage, do not account for higher order statistical moments skew and kurtosis, and assume a equivalent importance for both variables where the whole purpose of this example to show the far greater importance of volatility. Furthermore, STD is not an accurate measure of volatility for the majority of underlying probability distributions. Accounting for these factors [14] reveals that high Sharpe ratios, if anything, are a much better predictor for who will go bust in the future rather than achieve superior performance. Ideally, the only metric that should be used to compared performance across any and all strategies that is manipulation proof, correctly penalises for leverage and volatility, and is effortless to communicate is the time-averaged growth rate $\bar{g}$ in Eq. (10) [11, 18–25]. Reporting of past performance via any other measure should merit scepticism.

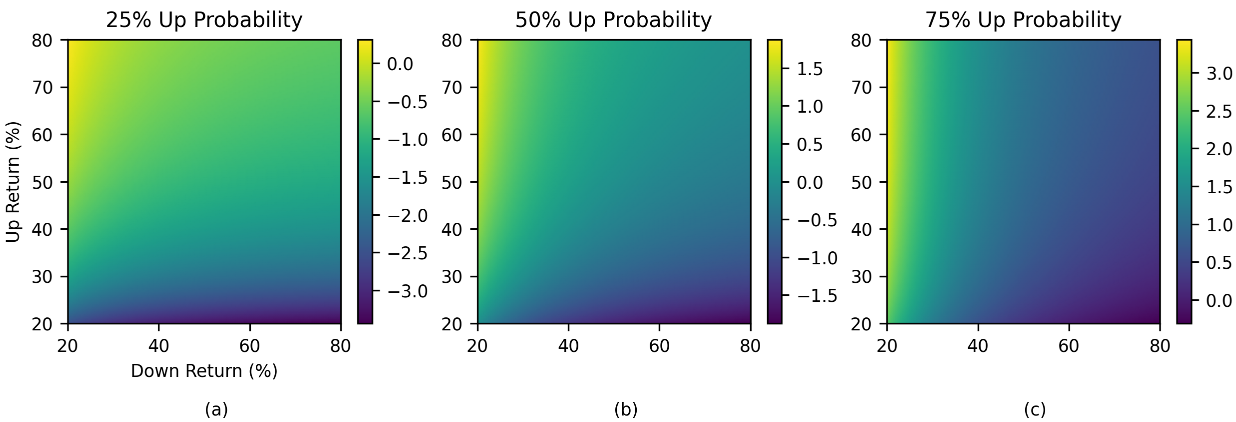

Figure 6: Investor four optimal leverages using the Kelly criterion in Eq. (16) for a range of up/down returns and probability of up moves. Each leverage is designed to maximise the time-average growth rate or geometric mean of the median investor [11].

While we could further explore this simple toy model, such as the interesting short-term profitability for roughly the first 200-300 steps, the actual concern is real-world problems and whether this approach would generalise to other compounding domains. By fully acknowledging the multiplicative nature of the problem and shedding all pre-existing notions of value maximisation, whether a robust methodology exists is an open question.

1.3 Conflating Probabilities with Payoffs

Formally the error in Eq. (5) can be expressed by the inseparability of probability [14]. Given the probability mass function $\pi(\cdot) : \mathcal{A} \to [0, 1]$ and measurable payoff function $y : \mathbb{R} \to \mathbb{R}$, for a subset $\mathcal{A}' \subset \mathcal{A}$ we have both the continuous and discrete cases

$$\int_{\mathcal{A}'} \mathrm{d}x\ \pi(x)y(x) \neq \int_{\mathcal{A}'} \mathrm{d}x\ \pi(x)y\left(\int_{\mathcal{A}'} \mathrm{d}x\right) \equiv \mathbb{E}_{x\sim\pi(x)}\left[y(x)\right] \tag{18}$$

$$\sum_{x\in\mathcal{A}'} \pi(x)y(x) \neq \sum_{x\in\mathcal{A}'} \pi(x)y\left(\frac{1}{N}\sum_{x\in\mathcal{A}'} x\right) \equiv \mathbb{E}_{x\sim\pi(x)}\left[y(x)\right] \tag{19}$$

if the payoff function $y(x)$ is non-linear by Jensen's inequality [26] where the right-hand side is the payoff of an average event weighted by the probability of the event. By non-linear we mean that the response to a change of any magnitude $x_2 - x_1 \neq y(x_2) - y(x_1)$, with the discrepancy becoming much worse as the difference expands. The payoff function, or exposure, is in general a non-linear function, while probabilities on their own are nothing more than reductionist beliefs. Since any investor only cares about their end performance which is non-linear in payoff, risk management and leverage allocation should never operate in probability space [11, 14, 18–21].

The only situation where the expectation value can be effectively used is strictly when the payoff function is fixed where $y(x) = \theta(x - K)$ is the standard general Heaviside step function. This situation clearly represents the realm of additive dynamics which forms the entire basis of contemporary decision theory.

The question is then whether this simplification can potentially be used in multiplicative environments. Suppose we are interested in calculating the expected non-linear payoff exceeding a point $K \in \mathbb{R}$, we define the two integrals

$$I_1 \equiv \int_K^\infty \mathrm{d}x \ \pi(x) y(x) \tag{20}$$

$$I_2 \equiv y(K) \int_K^\infty \mathrm{d}x \ \pi(x) = p(x \geq K) y(K) \tag{21}$$

where I_1 is the true outcome and I_2 is the pseudo-expected value. The substitution of integral I_2 for I_1 can be considered valid only for thin-tailed distributions defined by criterion $\eta \equiv \lim_{K \to \infty} \frac{I_1}{I_2} = 1$ [14]. This essentially implies that $p(x \geq K) = p(x = K) + p(x > K) \approx p(x = K)$ as the speed of the probability of extreme values crashing to zero is not offset by their absolute magnitude. The normalised and centred Gaussian distribution does satisfy this limit condition if at most the payoff is linear $y(x) = x$.

Distributions with $\eta > 1$ are defined as having fat tails where large deviations from the mean are far less likely but exist on astronomically larger scales. These distributions operate with an inverted logic, the typical behaviour in the vicinity of the mean is treated as the noise, while the signal is the exceedingly rare extreme event that entirely defines the long-term behaviour of the environment. An example of this is selling uncapped insurance, most of the time the insurer will generate steady premium with minimal volatility, however when the flood comes, all past performance will be entirely erased by one event. The key fact of fat tailed distributions is that after obtaining more empirical data, you can only ever revise them to be more 'fatter', never the opposite.

Unfortunately, not only is virtually every single conceivable payoff function in the real-world is non-linear, the overwhelming majority of underlying distributions in reality, at the bare minimum, have kurtosis in excess of the Gaussian [14]. While for very small changes in the $\delta x = x_2 - x_1$ we may be able at best to take a perturbative approach using linear Taylor approximation of the payoff so that $\delta x \approx y^{(1)}(x_2) - y^{(1)}(x_1)$. This however is not useful in the long run as the linear payoff hides the true nature of the process. Furthermore, since $\pi(x)$ can only become less thin as more data is collected, $\frac{\mathrm{d}\eta}{\mathrm{d}t} \geq 0$ and so $I_1 \geq I_2$ with difference only ever permitted to increase over time for all environments.

This can be seen through simple explanation in Section 1.1 where the final return is

$$y_{t_2}(x) = \begin{cases} 1 + \frac{x}{V_{t_1}}, & \text{if } V_{t_1} = \$100 - x \\ 1 - \frac{x}{V_{t_1}}, & \text{if } V_{t_1} = \$100 + x \end{cases} \tag{22}$$

If the absolute change in value $x = \$50 \to \$0$ the difference between both responses clearly becomes negligible and so using the expectation value to conclude both paths are equally preferable can be considered accurate. Whereas if $x = \$50 \to \$100$, we have $|\delta y_{t_2}(x)| \gg 0$ so we cannot simplify this payoff to be non-linear and therefore we have a distinct preference for the path avoiding the steep loss.

Regarding the simple gamble in Section 1.2 we have the non-linear (absolute) payoff at each step

$$y_{t+1}(r_u, r_d, \lambda_t, \phi_t, l_t) = \begin{cases} V_t - V_{\min,t} + V_t \cdot r_u l_t(\lambda_t, \phi_t), & p_u = \frac{1}{2} \\ V_t - V_{\min,t} + V_t \cdot r_d l_t(\lambda_t, \phi_t), & p_d = \frac{1}{2} \end{cases} \tag{23}$$

and the expectation value can similarly be utilised only if both returns $r \to 0$ which ensures $y_{t+1}(\cdot) - y_t(\cdot) \to 0 \ \forall \ t$ regardless of leverage if there is a stop-loss. The first three investors do not understand this requirement, despite $r_u + r_d > 0$ and $p_u \geq p_d$, the gamble is not additive, rather it scales with V_t. If the outcomes were fixed $V_t \cdot r_u l_t = R_u$

and $V_t \cdot r_d l_t = R_d \; \forall \; t$ then maximising expectations would be perfectly adequate.

The correct approach is in recognising this and taking the approach of the fourth investor that finds the parameters that simply maximise the time-average growth rate $\bar{g}$ using Eq. (9) as it incorporates all effects. No attempt should be made to disentangle probabilities with non-linear payoffs. Therefore, we pose that the default stance should be to never use expectation values to model the outcome of any process, contrasting sharply with the contemporary prescription from decision theory. Nature and human civilisation are governed by multiplicative processes that by definition are non-additive since they scale dependent on the existing value. The actual response to changing of underlying variables can then easily be $I_1 \gg I_2$ if the variability in returns is significant.

1.4 Implications and Anecdotal Outcomes

We unscientifically proposed and aggregated the responses to the following gamble:

"If I were to give you \$100 *and then offer you game where I flip a fair coin, if heads you make* 50%, *if tails you lose* 40%, *and let's say we play this game for* 3,000 *rounds. Would you play the game? If no, you get to keep the* \$100 *and continue with your day. If yes, as percentage, how much of your money would you bet at the start of each round and why?"*

Furthermore, if the respondents were familiar with leverage, they were informed they could also take unlimited leverage at no additional cost. In total, depending on you measure it, we received 155 very informal responses.

In our experience the overwhelming majority of respondents would agree to play the game with fixed leverage $l = 1$ in line with the first investor as most were (thankfully) unfamiliar with the concept of leverage. This group consisted of people with very diverse backgrounds, with and without tertiary education. Most held undergraduate degrees in finance, economics, and a whole host of STEM fields from leading universities. Individuals with more advanced training, often in the form of postgraduate level education in finance or STEM recognised this as a 'favourable' gamble and universally proposed to take constant leverage $l \to \frac{5}{2}$ with no stop-loss $\lambda = 0$ across all time. Respondents with active financial markets experience also came to the exact same conclusion. Both of these groups then form a variant of the second investor. We received no responses resembling the strategy of the third investor.

Finally, a sizeable portion of people also choose not to play the game opting to keep the free \$100 as they in one form or another disliked the odds of the gamble. Very often they were unable to even express any reasoning for their risk aversion, only that they did not want to participate. A major difference between the composition of this group and the former groups was the overall lack of any formal training or education in economics, finance, or any of the STEM fields. The results for this game are of course known and discussed thoroughly in Section 1.2.

It should also never be forgotten that the first known causality of this process was the author in April 2020. Understanding this error was the principal motivation for the entire Master's degree. This work should be interpreted as nothing other than a monument to that catastrophic mistake.

A very frequent rebuttal is of the form:

"*While this is a very interesting result, it is really nothing more than a trick question about compounding. It does not significantly affect much, however it would make a very good interview question.*"

This game offers a laser-guided surgical exposé of the most deep-rooted underlying decision-making abilities of any individual. It is not about whether they just understand compounding, rather it's about whether the person realises that the consequences of decisions they make now can have an everlasting (non-linear) influence on the final outcome. While everyone would say they understand this, actions speak louder than words, and failing to operate in line with this supposed understanding for such a simple gamble reveals huge inconsistency.

As virtually every single process in the real-world is governed by fat tails with multiplicative non-linear payoffs, not with fixed additive payoffs even remotely resembling bell curves, there is no good reason to believe that these people are operating correctly in practice. Furthermore, in the majority of cases, the confidence exhibited by respondents mimicking the first two investors was so immense that if it could be converted to electricity, a dozen of these people could cleanly power the planet.

Empirically we have also determined that the speed at which someone comes to the conclusion to dismiss these results is very positively correlated with their age and professional experience. We find immediate dismissal to be more common for participants over the ages of 29-33. A likely reason for their 'knee-jerk' reaction is that they are simultaneously finding out that not only is their belief system about formalised risk-taking incomplete, but that the implication of this is that any prior success they have achieved using this incomplete method is quite likely due to luck rather than skill. They utilised the same strategies as everyone else was taught, where others might have failed, they potentially achieved success, but likely only due to the cards they were dealt. C'est La Vie.

The source of this error is not the individual, rather the formal education they have received is lacking crucial content [5–11, 18–24]. They were taught probability theory and (thin tail) statistics but were not told that neither is particularly useful when making personal decisions that cannot be reversed. Probability lets them indulge in fantasises of escaping to more favourable parallel universes, thin tails raise them in a safety net where actions do not have long-term consequences. Most are also completely unaware of the concepts of non-ergodicity and time-averages.

This is clearly seen since the group with the least education had best performance while taking zero risk. Recall the final outcomes of the first two investors in this case would be \$0, making the risk averse individuals' with \$100 performance literally '∞%' superior. While a portion of these people would never play the game, it is unlikely this includes all of them. We refer to the portion that would choose to play if the odds appeared (subjectively) better to them as 'risk averse'.

We hypothesise that since the risk averse person has no defective education, they are forced to totally surrender their decision-making to their subconscious brain, or 'gut' instinct. The human brain is not foolish enough conflate payoffs with probabilities, a skill acquired over millenniums of evolution. Its underlying goal is to survive and is very acutely aware of the fact that it has only one life, hence its ability to assess risk may involve intuitively processing Eqs. (15-17). While it is doubtful they are thinking in terms of optimal leverage, perhaps they abstractly recognise in Eq. (17) that for $l = 1$, they instinctively require better odds $\left(p_u \geq \frac{2}{3}\right)$. These people are not "irrational", far from it, in fact the quality of their inexplicable human instincts is quite exceptional.

Another way to see this is by building on the analogy in [14]. When walking through the forest alone, the risk-averse investor has a tendency to mistake stones for bears but absolutely never the opposite. The person calculating expectations correctly realises the probability of encountering a bear is minuscule and so walks with a concern weighted by this infinitesimal chance. If they do encounter a bear, the only defence they have are a collection of A3 pages stamped with logos of old buildings. This of course is not an issue if they can travel to the overwhelming majority of parallel universes where the bear is not present. We repeat again for the final time, for multiplicative dynamics, probabilities do not matter!

A major implication rigorously discussed in [14] is that this largely invalidates psychology research. This field, not known for its mathematical prowess, almost always assumes in Eqs. (20-21) that $I_1 = I_2$ and therefore mainly uses I_2 when researching human decision-making. One key claim they have is the cognitive bias where empirically, people assume higher probabilities for rare events than what model parameters would predict, implying they are excessivelly risk averse [11–14, 24]. We now know that characterising this as a mistake can be catastrophic outside of textbooks since payoffs are non-linear with fat tails. All conclusions they have arrived at using this simplification and countless others must be re-examined from scratch. Therefore, given that the whole profession just might turn out to be pseudoscience, its tendency to frequently label people as "irrational" may be very incorrect.

A point worth emphasising is that generations of Sveriges Riksbank Prize recipients in highly authoritative positions, openly announced, and proudly taught countless students that the overwhelming majority of the public they would encounter in the lives were "irrational". When in fact it appears their models were never seriously questioned as the unadjusted empirical data seems correct [5–14]. This may add considerable weight to the common belief that the average person has a far firmer grip on reality than academics.

Keep in mind that modern psychologists have the legal authority to involuntarily administer medications and provide advice that is taken seriously in court rooms which is truly terrifying as one day they may be seen as no different to the witch doctors of old. There is however another field of concern that is far more intertwined with every person's life. Most Western governments mandate all citizens to have an externally managed superannuation or pension scheme where portion of their salary is given to 'professional' money managers, with the alternative of managing your own wealth penalised with increasing fees. As many of these managers often underperform compared to extremely low-fee passive market products, the average random citizen is in real trouble since a sizeable portion of their retirement savings is entirely dependent on these managers.

Let us first reflect on the fact that these managers, their subordinates, their applicants, their superiors, and even their regulators' entire education is based on maximising expectation values. What could possibly go wrong?

Next lets recognise that for the very simple gamble in Section 1.2 based on just maximising the value of portfolio that has binary returns, if they did not implement a very carefully optimised rolling stop-loss and took the approach of investor two by applying enough leverage to not be wiped out by a single move, after $3,000$ steps they would have 'expected' a valuation of $\mathbb{E}[V_T] = \left(1 + \frac{5}{2} \cdot 5\%\right)^{3000} \sim 10^{153}$. However, within about 30 steps they would all, without exception, be at their stop-loss. At this current time, we are unaware of any profession where a tolerance for error using the default textbook approach of 153 orders of magnitude would be considered acceptable. For reference, there are estimated to be 10^{80} atoms in the observable universe [27], and the size of the search tree required to recursively identify the winning strategy for the board games of Chess and Go are approximately 10^{124} and 10^{360} respectively.

Despite this failure, recall that for roughly the first 20 time steps the top 0.01% would achieve performance of $V_T^M \sim 10^7$ entirely due to the cards they were dealt, before eventually joining their peers at their stop-loss. Note that at no point have we specified the unit of time, it could be seconds, minutes, days, months, years, or anything. If we use larger time increments these individuals will increasingly appear to be 'legendary' investors, while in practice the fundamental basis of their strategy is exactly identical to the other 99.99%.

Financial markets in the real world are of course far more complicated, there are huge amounts of assets, returns are variable, and simulation is difficult. However, the principle of focusing solely on maximising $\bar{g}$ as opposed to $\mathbb{E}[r]$ remains valid throughout. The choice of leverage to take in any bet that does not have a fixed constant payoff is obviously non-trivial, but step one is recognising the multiplicative nature of the process.

Therefore, we present a heuristic that could be used to evaluate anyone that professionally manages money based on their response to the simple gamble. This should not be considered financial advice, but neither should anything they recommend be either. We classify their responses in terms of the previous investor categories. Responses mimicking the first two investors are obviously inadequate and so we focus on the third:

(i) Investor 3: So-called "alpha generating strategies" are of little value unless they are constructed with a clearly demonstrated an understanding of multiplicative dynamics. However, never bet against them. The simulations reveal the terrifying (mean) levels of success they can achieve by using prudent and completely automated risk management principles. They can accomplish this while not having any better understanding of the gamble than the previous two investors.

As they attempt to predict ex-ante $\mathbb{E}[\alpha]$, unless they are solely assessed on forecasting ability, they then have to actually allocate capital to capture it in a timely manner. Typically, the weights they select for each security in a portfolio are proportional to each assets 'expectation' which obviously does not correctly model the non-linearity of the payoff. Their performance is then measured by an ex-post $\hat{\alpha}$ defined relative to obtained return $\hat{r}$, this can be taken as the excess to a benchmark r_B, so that $\hat{\alpha} = \hat{r} - r_B$. However, the quantities $\mathbb{E}[\alpha]$ and $\hat{\alpha}$ are not even remotely equivalent.

This is the prescription detailed by all contemporary financial education programs offered globally. The same theories and models are taught, high distinctions and charters are awarded for accurately reciting these ideas, with candidates then proceeding to act in the real-world in line with these beliefs validated now by A3 pages.

One should therefore be very sceptical of investors dedicated to seeking "alpha" as most of them have likely been taught to use expectation values in a compounding world. Therefore, they essentially operate on the assumption that the payoff they receive from taking actions in world around them is the (probability-weighted) average of what they predict, when in fact, they are confined to a singular future [11, 18–24]. The "expected alpha" they chase, using their definition, cannot be replicated for a portfolio constructed with variable leverage.

This would be akin to extremely competent brain surgeons not knowing the function of the brain. Does it matter if they always get the surgery correct? Not really. But if they make a mistake, devoid is their knowledge of the consequences.

A potential entirely self-admitted candidate for this type is [28] who outlines many principles with "Life Principle 5.6" being "make your decisions as expected value calculations" along with many other probability-based claims. Surely, he would get this simple gamble correct? If not, his prior unmatched performance would therefore highlight his beyond outstanding world-leading risk management, while being somewhat inept at risk-taking. His selection of stop-losses and retention ratios are phenomenal, though his long/short predictions should not be blindly replicated or taken at face value. An alternative explanation is that these "principles" are brilliant disinformation campaign to prevent the formation of competitors.

Regardless, suppose someone at this pinnacle level of achievement is not only publicly conflating probabilities with payoffs, but has proceeded to write a 600 page book receiving countless awards and limitless amounts of praise from highly authoritative sources, that ultimately serves to concretely display their misunderstanding. What are then the chances exponentially less prominent individuals, world-wide, that have a fiduciary responsibility to manage large portions of people's life-long retirement savings also get it wrong?

The correct behaviour, instead, should be to make investment management decisions today that allows each of them embrace the one irreversible, unpredictable, volatile, and unknown singular fate that awaits them all [11, 22, 24]. We will discuss this further in due time in Section 13 and Appendix C.5.

(ii) The risk averse person that does not want to gamble with provided odds: Commend them for having better risk-taking instincts than the first three investors.

(iii) All other responses: Since the gamble is simple, have an open mind and simulate the outcome.

Furthermore, despite all the perils of using expectation values, there exists an even more dangerous quantity, standard deviation σ. While an investor using expectations without prudent risk management will go bust, individuals using variance to quantity volatility will not only go bust but will take counterparties down with them.

This occurs because variance is incredibly unstable under fat tails and all subsequent computations that utilise variance as an input are also then contaminated. Utilisation of variance minimising machine learning tools are not robust to shocks. This includes all forms of regression even with ridge and lasso regularisation. GARCH predictions are spurious as they cannot adapt to environments with large natural kurtosis without forcibly truncating the data. Value at risk measures combing both standard deviation and probability are ineffective. Similarly, one can misinterpret the natural existence of fat tails to be conclusive evidence of heteroscedasticity. Pearson correlations are at best uninformative. Non-parametric methods lead to even worse out-of-sample robustness. See [14] for further details.

1.5 Path to a Permanent Solution

Two valid criticisms of the preceding text are whether the toy problem generalises to more real-world scenarios and that we have not provided a systematic method to solve Eq. (9) for any environment. Obviously, the simple optimisation in Eqs. (15-17) to find $l^* = 25\%$ is not possible in systems with huge numbers of continuous parameters. Furthermore, in Appendix A we discuss in an identical fashion another 'simple' gamble involving numerically finding the optimal leverage for payoffs based on the rolling of fair dice that will be elaborated on in Section 11.4.

This type of analysis has also been done in the more useful case of finding optimal leverage for a financial security following geometric Brownian motion (GBM). GBM is often considered as the fundamental basis for most theoretical derivative pricing models in mathematical finance [29]. Using stochastic calculus and Itô's Lemma they found the explicit inverted parabolic relation $\bar{g} = \mathbb{E}[r] + \nu = lr - (l\sigma)^2/2$ revealing the importance of selecting the correct leverage [5, 15, 16]. Clearly then we have $l^* = r/\sigma^2$ implying that $\bar{g}_{\max} = r^2/2\sigma^2$ which is reminiscent to the exponent of the log-normal probability density function. Note that while $\mathbb{E}[r] \propto l$, we have $\nu \propto -l^2$, which is probably the shortest explanation in history for the dangers of leverage.

GBM is also a key model for representing self-reproducing entities, which may be considered as the definition of life as the dynamics it induces are of interest to those concerned with living systems from biology to economics [16]. They then determine very interesting long and medium-term behaviour of the simulated paths in terms of the speeds at which they will ultimately converge to $\bar{g}$.

In the real-world where all payoffs are non-linear and there are infinite possible multiplicative problems, the optimisation problem in Eq. (9) can be very generally expressed as

$$x_1^*, ..., x_T^* \in \arg\max_{x_1,...,x_T} \sum_{t=1}^{T} (1 + g_t(x_t)) \tag{24}$$

where $x_t \in \mathbb{R}^n$ incorporates the range of all possible n actions that can be taken to modify the existing state of the system at each time step t in order to maximise the future rewards. Maximisation of the time-average growth rate is clearly a formidable problem and appears that it needs to be solved on a case-by-case basis. They key point is that at time $t = 0$ we must solve across all time steps from $t = 1 \to T$ to properly maximise $\bar{g} = \mathbb{E}[r] + \nu$.

Therefore while [5–11, 14–16, 18–24] have presented a concrete general case for the need for a reformulation of existing decision theory, the majority of the practical applications of their suggestions need to be explicitly constructed using Eq. (24) and then effectively solved. This is no easy task and requires bespoke domain knowledge for each problem. To encourage this adoption, for over a decade they have presented their findings in hope to reform and re-educate existing practitioners on the validity multiplicative dynamics, and more importantly, on the perils of using additive dynamics in a compounding world. Mainstream acceptance however has been at best mixed, despite an ever-growing community forming under the brand Ergodicity Economics. Regardless, we hope momentum continues to build incorporating multiplicative dynamics in environments where it is necessary to correctly model the scenario.

One example of this in practice is the financial performance of the "Universa Tail Hedge" portfolio by Universa Investments L.P. where they have consistently outperformed the SPX benchmark by 3.6% p.a. since the funds inception in March 2008 [24]. Furthermore, in absolute terms, for March 2020 the year-to-date return on capital of their "tail hedge" insurance was $4,144\%$ [24] (designed to offset the broader market decline in the period). Importantly, this was all performed with zero market timing or active directional positioning. Next, the relatively recent endorsement [8] by the world-renowned late theoretical physicist Murray Gell-Mann will ideally serve as an accelerator for its adoption given that his existing contributions (to fields such as Quantum Chromodynamics) will be taught almost certainly, at the very least, till the end of time. Out of everyone, someone of this calibre has the potential to hold a candle to combined intellect [30, 31] of every economist that has ever lived regarding the foundations of economics.

The lack of widespread admission of the failings of the contemporary methods would be especially valid for senior practitioners that dominate and control their fields as this would involve acknowledging their entire careers have been built on shaky foundations, which is in line with our crude empirical findings in Section 1.4. This inability to decouple probabilities from payoffs not only effects them, and those around them, but can have devastating consequence for human civilisation since no matter how "data-driven" they are, expectation values do not capture the non-linearity of the outcome for any decision.

For example, returning again to pension funds management. OECD data [32, 33] reveals that despite turbulent markets, total global assets under management annually rose 9% to USD$34.2 trillion at end-2020. For the countless managers of these funds, regardless of how well-meaning their intentions are and how seriously they take their fiduciary responsibility, the moment their decision-making process incorporates probability estimates, they essentially become delusional. No longer are they concerned with each client's financial wealth, rather they aspire to maximise the wealth of the average of each individual client. The difference is subtle, but the impact astronomical. Unwittingly, the bulk of these retirement savings are effectively being mismanaged by people acting as schizophrenics. Much the same can be said for the countless other trillions being managed in more discretionary funds. While it is perverse that this has continued for so long, this situation breeds unimaginably profitable opportunities. Their existence will permit the construction of strategies that will appear non-sensical to them, however, are capable generating inconceivably high returns for a given level of risk. One non-unique method involves a careful combination of largely a passive market portfolio and a small extremely convex insurance portfolio [11, 18–24].

At the same time, we are also left with the intractable Eq. (24) and are unable to offer any systematic and tangible alternatives to the contemporary approach. Therefore, we require a method to generally enforce the principles of multiplicative dynamics that is compatible with maximising compounding growth. To be applicable to any environment, it also needs to be self-learning and fully autonomous to analyse the cause and effects of each x_t. This is necessary as directly solving stochastic partial differential equations as with GBM in [5, 15, 16] is not possible, let alone tractable in general. Successful construction of this approach also demands performance that eventually exceeds of all existing practitioners in each real-world domain.

The remainder of this work is dedicated to accomplishing this formidable task. To achieve this goal, a general problem-solving approach is required. Reading extremely carefully the words under Eq. (24) again, 'take correct actions to modify the existing state to maximise future rewards' — this is the language of reinforcement learning.

2 Introduction

Reinforcement learning is generally formulated under the assumptions of ergodic Markov decision processes (MDPs) that are stationary through time. One interpretation of ergodicity is that it implies the time average reward of an agent's trajectory through state-action space is exactly equal to the expectation value of that reward. This simplification is useful to formulate the theory and prove numerous convergence criterions [34]. The difference between classical dynamics programming methods and reinforcement learning is that in the latter, large scale approximation methods are necessary as exact modelling of the MDPs becomes intractable [34–37]. Performance in environments across various algorithms is generally compared by additive dynamics using the averaging cumulative summations of the rewards the agent receives per time step in each evaluation episode. This approach combined with the use of deep neural networks acting as universal function approximators has led to highly promising advancements over the last decade with gains across the board [34, 37–41].

The most well-known of these include classic Atari video games [42–51], the board games of Chess, Go and Shogi [52–57], StarCraft II [58], and the protein folding problem [59–61]. For continuous action spaces pertaining largely to continuous locomotive control tasks [62–70], actor-critic methods combining advancements in Q-learning [71–73] with stochastic or deterministic [74–80] policies have steadily matured. An alternative approach using the principle of maximum causal entropy [81] has led to the soft actor-critic algorithms [82–88] incredibly robust results requiring minimal hyperparameter tuning. In parallel, for these continuous action spaces, on-policy model-free methods [89–91] have also achieved decent performance, and augmented random search [92] has once again proven to be remarkably effective in simpler environments.

During the Q-learning phase, the critic minimises the difference between the Q-value and a target Q-value obtained using the Bellman equation. The vast majority of literature constructs this off-policy critic loss through aggregating the (mean-square) Bellman error. This use of the MSE loss function is considered by [93] to be a "reasonable choice" while acknowledging there is a 'lack of a good understanding for this measure', but regardless, believes that "most conclusions would hold for different measures". There does not appear to exist any such analysis experimentally validating these claims. The use of the MSE functional form is likely due to several reasons, firstly the overwhelming majority of machine learning is built upon this variant of L_2-norm which traces back to its ability in finding optimal coefficient for linear regression. Secondly, the compatible function theorem and several historical proofs have used MSE from which further developments solidified its prevalence. There likely also exists many other reasons unaware to us at this time. Based on recent advances in non-negative matrix factorisation (NMF) proving the benefit of other loss functions [94], there exists a possibility that similar gains may be achieved.

An additional feature that is globally utilised throughout all statistics and machine learning is when constructing of the empirical mean, it is assumed to converge by the strong law of large numbers to the true unknown population mean in the infinite sample limit. All samples in the real-world are finite and so this convergence never formally occurs, regardless the (equal-weighted) arithmetic mean is still utilised. This approach is appropriate if underlying distribution is well-behaved, meaning that it has thin-tails, and a true mean exists. If it is fat-tailed, not only may the variance be undefined, but so might the mean. To combat this, we need to first test whether a distribution is fat-tailed and if so, a 'shadow' mean should be estimated as a more faithful representation of the true mean [14, 95–97]. This is a very recent development and its applications to statistics and machine learning remain a very open question that merits our investigation.

Off-policy learning is characterised through the use of an experience replay buffer $\mathcal{D}$ [98]. Mini-batch sampling is often either uniform $U(\mathcal{D})$ or with prioritisation proportional to absolute TD-error [99]. The vast majority of algorithms generally use a fixed buffer containing the 10^6 most recent transitions, mainly due to its historical use in [43]. For discrete action spaces size of the buffer is found to be highly impact on overall learning especially when combined with other features [100, 101]. The size of the buffer is interpreted by [101] as the degree of 'off-policy-ness' as more data allows the agent to search further back in history and therefore reduces the chance of overfitting.

Using a modified Rainbow agent called Dopamine [50] on classic Atari games, [101] finds performance increases as both the size of buffer increases and the age of the oldest policy reduces which is intuitive as more recent transitions are higher performing. Therefore, there exists a trade-off between buffer size and how relatively 'on-policy' the transitions are. They also show that prioritised experience replay is gives no additional benefits with increasing buffer size which is an incredibly unintuitive result as one would expect preferential sampling to be very useful. Interestingly, [101] determines multi-step returns [71] to be absolutely crucial using ablative trials while acknowledging there exists no theoretical or well-grounded explanation its importance. Any bootstrapping is superior to the vanilla one-step case when simultaneously increasing buffer size, with the optimal level found via grid search. One explanation for this puzzling result is that gains achieved from increasing the buffer size are positively correlated with the variance of target returns and therefore increasing steps achieves this randomness. There does not appear to exist an equivalent analysis on whether their results hold in continuous action domains.

Under additive dynamics, an algorithms overall performance is evaluated using the summation of the rewards received at each time step. This approach is not suitable for a large class of environments where losses have an asymmetrically larger effect on performance than identical magnitude gains. This is particularly crucial in situations where the agent is operating in mission critical scenarios where time order matters and the agent has only 'one chance' as opposed to assembly line production tasks. To model such environments, multiplicative dynamics [5, 8] must be used where evaluation episode performance is expressed as cumulative compounding returns of the rewards received at each time step. Much of the applications for such environments are in non-ergodic domains where the time average is not equal to expectation value or more accurately, the ensemble value.

Pioneering work on such tasks has been done over the last decade [5, 8, 14–16] with direct applications to finance and economics [6, 7, 9, 10, 18–24, 102–107] with experimental psychology validation into optimal human decision-making [12–14, 24]. This type of modelling is crucial in environments such as medicine, supply chains, guidance systems, economic policy, financial portfolio management, and systems control where we may want to encode the asymmetric effect of large negative rewards. The crux of this approach is that the correct way to represent non-ergodic domains is with multiplicative dynamics using compounding products of returns over time as opposed summations.

These ideas were initially correctly posited in 1965 by the well-known Kelly Criterion [4] obtained using information theory and used for making optimal bets based on prior performance where the probability of bankruptcy approaches unity as games of maximum (expected) rate of return are repeated. This certainty of bankruptcy is the well-known Gambler's Paradox [108], which was also a prime motivator for development of the causal conditioned entropy methods [81] used to derive the soft actor-critic algorithm where time ordering is crucial [82–88]. The current paradigm however is entirely built on a parallel development in 1952 that bases decisions on incorrectly maximising expectations using Markowitz (or Modern) Portfolio Theory [109–111]. Kelly emphasised maximising the time average reward by avoiding steep losses, while Markowitz based risk preferences on completely subjective utility functions that are dependent on personal circumstances, see [5–10] for the complete history.

A direct application of this is seen by the fact that the average economic growth of a country is not equal to the economic growth of a random citizen over time. Economics is entirely concerned with the former and implicitly states that it is exactly equal at all times to the latter [9, 10, 106], and then proceeds to casually build an entire discipline around this assumption. The responsibility for this mistake traces to using a 300-year-old result of Bernoulli that contained a conceptual error [3], and then subsequently a failure to pick up on its 200-year-old (minimally advertised) correction by Laplace [8]. The failure of contemporary academics managing global economies and world trade to recognise this glaring weakness, amongst several other crippling issues [112], is quite alarming.

As an aside, the Markowitz approach also throughout utilises probabilities bounded within the closed interval $[0, 1]$ to construct expectations. Results from the path formulation of quantum mechanics, reveal that probabilities meaningfully exist outside both bounds of this domain in nature [113, 114]. The interpretation of these values in terms of Bayesian inference where decisions on whether information gathering and utilising systems should take a bet is determined by whether it can be first settled, and then to isolate all possible realities [113]. Therefore, it is debatable whether expectation values should be used at all if one does not normalise across the true complete probability space. Accounting for this may offer a path to reconciliation where the additive case approaches multiplicative dynamics.

Existing work on extending the applicability of Q-learning to non-ergodic state process beyond MDPs can be written in the language of Feature Reinforcement Learning and state aggregation [115–119]. Much of this is inspired by earlier work on incorporating partially observed MDPs (POMDPs) and other non-Markovian decision processes (non-MDPs) into reinforcement learning [120–123]. Other more recent approaches in this area is discussed in [124–127]. Using extreme state aggregation [117], any non-MDP can be modelled as a finite-state MDP if there exists a feature map aggregating different histories to states. Construction of this feature mapping is non-trivial in general but correctly reduces to all the well-known results if the underlying process is an MDP. One way to incorporate non-ergodicity while retaining much of the existing machinery [119] involves introducing a new class of Q-Value Uniform Decision Processes (QDPs) specifying several constraints. Under these constraints they show that Q-learning under the additive dynamics case converges to the optimal action-value using slight modifications of the usual methods and so this approach can be used in a subset of non-stationary domains called QDPs. One unknown is that the ability of the QDP approach to perform while using function approximators such as deep neural networks which is essential to many practical domains.

Then by modifying the existing formulation of model-free reinforcement learning to be compatible with maximising compounding growth rate, we will be able to construct full autonomous, self-learning, and risk-reward maximising algorithms that can operate in multiplicative domains. This is necessary as the path to artificial general intelligence (AGI) will be of little value if these AIs also get simple gambles such as those in Section 1.2 incorrect since reinforcement learning as a field is also entirely based on maximising additive sums of all predicted future rewards. Therefore, through making fairly straight-forward modification to well-known algorithms, it might be possible to reformulate the way in which these agents learn. Perhaps modifying their risk-taking to be more consistent with reality [12–14] may lead to them engaging in interesting activities.

As the field of reinforcement learning matures and eventually begins to enter the real-world environments where the cardinality of state and action spaces expands immensely, concerns regarding computational efficiency of agent training and operating will begin to emerge. Generally, the efficiency of training is not of pivotal concern as it can be parallelised over dedicated supercomputing clusters. Agent operation and inference one the other hand is what will occur in practice. For applications where the agent operates using a battery, by minimising power consumption, operating time will be extended which will inevitably reduce costs. There are two broad methods to reduce power: 1. Increasing hardware computational efficiency, and 2. More efficient agent learning algorithms. The first is outside the scope of this work. The second is our focus and is expected to naturally occur as time progresses. For the special case of agents operating in environments separated by distinct phases or stages, we propose 'linking' several agents sharing the same state space but different action spaces. Over millions of agent decisions per agent, the reduction in computational resources is likely to steadily accumulate resulting in overall lower energy consumption permitting lengthier operation that would be especially important in mission-critical situations.

This project is structured with Section 3 presenting a comprehensive review of background material. Section 4 outlines differences between additive and multiplicative dynamics in the context of non-ergodicity. Sections 5 and 6 modify critic Q-learning and actor policies to incorporate multiplicative dynamics respectively. Section 7 incorporates these changes into the soft actor-critic algorithm. In Section 8 we provide an introduction to a new framework for efficient reinforcement learning for segregated action spaces. Section 9 provides a brief recap of motivations and related work justifying our originality. Experiments are conducted in Sections 10-13 covering several of the research areas. An overall discussion summarising all findings is presented in Section 14.

In Appendix B the two utilised agent algorithms are presented. Appendix C provides a succinct outline of potential applications of multiplicative dynamics utilising reinforcement learning. Finally, in Appendix D a brief summary of the limits of our models in the context of delivering realistic capabilities.

3 Background

3.1 Preliminaries

Standard reinforcement learning is formulated by considering a infinite-horizon Markov decision process $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$ forming a sequence of states $(s_t)_{t\geq 0}$, actions $(a_t)_{t\geq 0}$, and rewards $(r_t)_{t\geq 0}$ experienced by an agent at each time step $t \in \mathbb{Z}^+$. The agents' behaviour is characterised by observing a state $s \in \mathcal{S}$, selecting the next action $a \in \mathcal{A}$ to take from state s based on its current policy $\pi : \mathcal{S} \to \mathcal{A}$, arriving at the new environment state $s' \in \mathcal{S}$, and also receiving a bounded reward from this transition $r : \mathcal{S} \times \mathcal{A} \to [r_{\min}, r_{\max}]$ where $r \in \mathcal{R} \subset \mathbb{R}$. The Markov property implies there exists an initial state distribution $p_1(s_1)$ and a stationary transition conditional distribution satisfying $p(s_{t+1}|s_1 a_1 ... s_t a_t) = p(s_{t+1}|s_t a_t)$. The discount factor $\gamma \in [0, 1)$ determines the priority given to shorter-term rewards. The environment also provides at each time step a Boolean done flag signifying whether the game or episode has concluded. Model-free learning is characterised situations where the agent does not explicitly utilise the transition probability distribution P, rather it takes a trial-and-error approach such as Q-learning.

The policy used to select actions in the MDP could be either stochastic $\pi_\phi(a_t|s_t) : \mathcal{S} \to P(\mathcal{A})$ or deterministic $\mu_\phi(s_t) : \mathcal{S} \to \mathcal{A}$ where the policy is generally expressed with a vector of n parameters $\phi \in \mathbb{R}^n$. The reward for all future time steps from a time t is written as an additive discounted cumulative future reward $R_t \equiv \sum_{k=t}^{\infty} \gamma^k r(s_k, a_k)$. For stochastic policies, the state value and action-value functions are defined to be $V^{\pi_\phi}(s) \equiv \mathbb{E}\left[R_1|s;\pi_\phi\right]$ and $Q_{\pi_\phi}(s,a) \equiv \mathbb{E}\left[R_1|s,a;\pi_\phi\right]$ respectively. The objective to then maximise becomes $J(\phi) = \mathbb{E}_{s\sim\rho^{\pi_\phi}, a\sim\pi_\phi}\left[R_t\right] = Q_{\pi_\phi}(s,a)$. In discrete action spaces, a ϵ-greedy approach is generally taken with regards to action selection. For continuous action spaces we are able to explicitly optimise the policy gradient $\nabla_\phi J(\phi)$.

The density at state s', after transitioning for t time steps from a state s is represented by $p(s \to s', t, \pi_\phi)$. The (improper) discounted state distribution representing the marginals for the trajectory distribution induced by the policy is $\rho^{\pi_\phi}(s,a) \equiv \int_{\mathcal{S}} \sum_{t=1}^{\infty} \gamma^{t-1} p_1(s) p(s \to s', t, \pi_\phi) \mathrm{d}s$. Therefore, by definition, agent learning is a non-stationary process as the policy improves $\pi'_\phi \leftarrow \pi_\phi + \alpha(\pi'_\phi - \pi_\phi)$ where $\alpha > 0$ is the actor learning rate. All proofs of optimal convergence meanwhile are derived under the MDP assumption that transition probabilities do not vary across time. It should be noted that while this inconsistency is present throughout the field, we are still able to train functioning agents, but the true impact of such an approximation is not well understood.

3.2 Policy Gradient Theorems

For continuous action spaces, there exists two well-known policy gradient methods: stochastic and deterministic. The first is very well-known for a policy $\pi(a|s)$ so the gradient of the objective is

$$\nabla_\phi J(\phi) = \mathbb{E}_{s\sim\rho^{\pi_\phi}, a\sim\pi_\phi}\left[\nabla_\phi \ln \pi_\phi(a|s) Q_{\pi_\phi}(s,a)\right] \tag{25}$$

where the monotonic log likelihood is called the score function or eligibility vector [74]. Deterministic policies are characterised by $\pi_\phi(a|s) \to \mu_\phi(s)$ and modify the density $\rho^{\pi_\phi} \to \rho^{\mu_\phi}$. For continuous action spaces, instead of greedy action selection $\mu_\phi(s') = \arg\max_a Q_{\mu_\phi}(s', a)$ at each step which leads to global maximisation, a more efficient approach is to move policy $\mu_\phi(s)$ in the direction of $\nabla Q_{\mu_\phi}(s, \mu_\phi(s))$ for each transition. This lets us approximate the gradient of the actor learning objective as

$$\nabla_\phi J(\phi) \approx \mathbb{E}_{s\sim\rho^{\mu_\phi}}\left[\nabla_\phi Q_{\mu_\phi}(s,\mu_\phi(s))\right] \tag{26}$$

$$= \mathbb{E}_{s\sim\rho^{\mu_\phi}}\left[\nabla_\phi \mu_\phi(s)\nabla_a Q_{\mu_\phi}(s,\mu_\phi(s))|_{a=\mu_\phi(s)}\right] \tag{27}$$

where the chain rule is used in the second line [75]. A challenge with both these approaches is determining how to accurately estimate the Q-value at each optimisation step as it directly coupled the policy.

3.3 Actor-Critic Methods

In actor-critic architectures the above policy gradient algorithms are split into two coupled components. The actor updates the policy parameters ϕ by performing the gradient ascent in Eqs. (25-26). To address the concern regarding coupling with Q-value estimation, a critic is introduced using different parameters θ so that $Q_\theta(s,a) \leftarrow Q_{\pi_\phi}(s,a)$. By the compatible function approximation theorem, the reparameteristaion is only exact if i) θ is linear in π_ϕ, and ii) θ is obtained by minimising the MSE error $\sim (Q_{\pi_\phi} - Q_\theta)^2$ [74, 75, 128]. These assumptions are very often relaxed, such as when using deep neural networks as universal function approximators. In practice this leads to brittle weights but works well in practice after several additional stability improvements such as introducing target networks [76, 80, 84, 86, 87].

To learn critic values for both stochastic and deterministic actors, Q-learning is used as a form of temporal difference control [72]. Using the deterministic policy as an example, we know from the Bellman equation [129, 130] that the valuation for the current state-action pair $Q_\theta(s,a)$ is related to the value of the next subsequent state-action pair $Q_\theta(s',a')$. This allows us to construct a target state-action value

$$Q_{\bar\theta}(s,a) \leftarrow \mathbb{E}_{s'\sim\rho^{\mu_{\bar\phi}}}\left[r\left(s,\mu_{\bar\phi}(s)\right) + \gamma Q_{\bar\theta}\left(s',\mu_{\bar\phi}(s')\right)\right] \tag{28}$$

where the target parameters weights $\bar{w}$ are obtained through delayed Polyak averaging $\bar{w} \leftarrow \tau w + (1-\tau)\bar{w}$ with $\tau \ll 1$. The critic then minimises the difference between these two theoretically equivalent values. The overwhelming majority of literature represents this with standard MSE loss (Bellman error) objective

$$J(\theta) = \mathbb{E}_{s\sim\rho^{\mu_\phi}}\left[\left(Q_{\bar\theta}\left(s,a\right) - Q_\theta\left(s,\mu_{\bar\phi}(s)\right)\right)^2\right] \tag{29}$$

This approach led to famous the Deep Deterministic Policy Gradient (DDPG) algorithm [76] along with several other modifications over the years [77–79]. The contemporary stand-out successor is the Twin Delayed DDPG (TD3) algorithm [80]. TD3 improves on DDPG in three key areas: 1. Introduces clipped double-Q learning and uses the minimum of two Q-values for target values to address critic overestimation bias. 2. Delays policy, target policy, and target critic updates to occur every second step. 3. Adds noise to target policy as a form of regularisation to prevent the formation of brittle policies.

Off-policy learning for TD3 utilises a experience replay buffer $\mathcal{D}$ containing tuples (s,a,r,s') of all past steps across all training episodes up to a maximum buffer size. Actor and critic optimisation per respective gradient step involves uniformly sampling a mini-batch of N transitions $U(\mathcal{D})$ from the buffer to construct

$$J(\theta) = \mathbb{E}_{(s,a,r,s')\sim U(\mathcal{D})}\left[\left(Q_{\bar\theta}(s,a) - Q_\theta(s,a)\right)^2\right] \tag{30}$$

$$J(\phi) = -\mathbb{E}_{(s,a,r,s')\sim U(\mathcal{D})}\left[Q_\theta(s,a)\right] \tag{31}$$

where the negative in the second line is to force gradient ascent. The vectors of parameters ϕ and θ are then adjusted to minimise the empirical losses. Note two explicit assumptions, firstly that using MSE is preferred over a myriad of other loss functions that potentially might offer greater resistance to outliers. The Monte-Carlo or law of large numbers approach is used where as $N \to \infty$ the sampling of $U(\mathcal{D})$ approaches the true unknown distribution provided the underlying data is i.i.d. and stationary. In most situations this does not hold since in practice a mini-batch of size of a few hundred transitions is always used and policy optimisation by definition is non-stationary process and hence the underlying data is not i.i.d. over training time.

The Kolmogorov theorem on the strong law of large numbers for non-i.i.d. data does however still guarantee this approaches validity provided the mini-batch variable has finite variance [131]. It is worth pointing out that this finite variance assumption applies directly to the true unknown distribution, not the mini-batch as finite sampling will always produce finite moments [14]. For example, is they underlying distribution is Cauchy then it has both undefined variance and mean, if it is Pareto then it also can have undefined variance (if tail index $\alpha < 2$) and mean (if $\alpha < 1$). This will be discussed further in Section 3.6.

3.4 Soft Actor-Critic

A alternative formulation based on conditional energy-based models involves the combined use of soft values $V_\theta^{\text{soft}}(s)$ and $Q_\theta^{\text{soft}}(s,a)$, and stochastic policies $\pi_\phi(a|s)$ to maximise a causal entropy objective [81]. Soft generally refers to the value functions being defined by application of softmax functions over all actions whether in discrete or continuous action spaces [81]. As deterministic polices such as TD3 heuristically explore via the injection of noise when selecting actions, stochastic polices have the ability to navigate multi-modal objectives that are common in robotic environments [83]. For example, in situations where two actions appear equally attractive, the policy will commit equal probability mass rather than deterministic selection.

The additive discounted cumulative future reward from a time t is then $R_t = \sum_{k=t}^{\infty} \gamma^k \left(r(s_k,a_k) + \alpha H\left(\pi_\phi(\cdot|s_k)\right)\right)$ using the objective $J(\phi) = \mathbb{E}_{s\sim\rho^{\pi_\phi}, a\sim\pi_\phi}\left[R_t\right]$. The temperature α is a automatically tuned entropy regularising hyperparameter that controls the relative weighting or importance given to stochastic behaviour and $H(x) = \mathbb{E}_{x\sim P(x)}[-\ln P(x)]$ is the entropy at each state.

The policy is very generally represented as $\pi_\phi(a|s) \propto \exp\left(\frac{1}{\alpha} Q_\theta^{\text{soft}}(s_t,a)\right)$ with the definitions

$$Q_\theta^{\text{soft}}(s_t,a) \equiv r_t + \mathbb{E}\left[\sum_{k=t} \gamma^{k-t}\left(r(s_k,a_k) + \alpha H\left(\pi_\phi(\cdot|s_k)\right)\right)\right] \tag{32}$$

$$V_\theta^{\text{soft}}(s_t) \equiv \alpha \ln \int_{\mathcal{A}} \mathrm{d}a' \; e^{\frac{1}{\alpha} Q_\theta^{\text{soft}}(s,a')} = \alpha \ln \mathbb{E}_{Z_\theta}\left[\frac{\exp\left(\frac{1}{\alpha} Q_\theta^{\text{soft}}(s_t,a')\right)}{Z_\theta(a')}\right] \tag{33}$$

where $Z_\theta(a)$ is the partition function normalising the distribution being independent of policy and does not contribute to the gradients [81, 84, 85]. Notice that $V_\theta^{\text{soft}}(s_t)$ is by definition a softmax function. The optimal policy is very generally expressed by

$$\pi_\phi(a|s) \equiv \exp\left(\frac{1}{\alpha}\left(Q_\theta^{\text{soft}}(s,a) - V_\theta^{\text{soft}}(s)\right)\right) \tag{34}$$

with convergence guaranteed using fixed-point iteration [81]. The soft Q-value also satisfies the soft Bellman equation

$$Q_\theta^{\text{soft}}(s_t, a) = r_t + \gamma \mathbb{E}_{s' \sim \rho^{\pi_\phi}} \left[V_\theta^{\text{soft}}(s') \right] \tag{35}$$

with the (hard) Bellman equation recovered in the zero entropy policy limit $\alpha \to 0$ [81]. Note we can also rewrite $V_\theta^{\text{soft}}(s_t) = \mathbb{E}_{a \sim \pi_\phi} \left[Q_\theta^{\text{soft}}(s_t, a_t) - \alpha \ln \pi_\phi(a_t|s_t) \right]$.

Stochastic action sampling from the policy $\pi_\phi(\cdot, s_t)$ is defined over some fixed distribution $a_t \sim f_\phi(\epsilon_t; s_t)$. In the case of a Gaussian, the number of outputs nodes of the ϕ network is $2 \times \dim(\mathcal{A})$ where each actions probability density function is completely defined by a unique mean and standard deviation. One additional modification we require for the sampled actions to be numerically bounded involves a change of variable. Consider action sampling from an unbounded distribution $v_t \sim f_\phi(\epsilon_t; s_t)$ such as a Gaussian, to enforce strict symmetric bounds we can use $a = \tanh(v)$ and the new density can be expressed as $\pi(a|s) = \nu(v|s)|\det(da/dv)|^{-1}$ where the Jacobian is $da/dv = \text{diag}(1 - \tanh^2(v))$ [84, 86]. The transformed log-likelihood is then given by

$$\log \pi(a_t|s_t) = \log \nu(v_t|s_t) - \sum_{j=1}^{|\mathcal{A}|} \log \left(1 - \tanh^2 v_{t,j}\right). \tag{36}$$

One method of improving the policy $\pi_\phi \to \pi'_\phi$ using an information-theoretic approach is by directly minimising the Kullback-Leibler divergence $\mathrm{D_{KL}} \left(\pi_\phi(\cdot|s_t) \;\middle|\middle|\; \exp\left(\frac{1}{\alpha} Q_\theta^{\text{soft}}(s_t, \cdot)\right)\right)$ to give

$$\pi'_\phi(\cdot|s_t) = \arg\max_{\pi_\phi} \mathbb{E} \left[Q_\theta^{\text{soft}}(s_t, a_t) - \alpha \ln \pi_\phi(\cdot|s_t) \;\middle|\; \pi_\phi \right] \tag{37}$$

which is reminiscent of an advantage function with the baseline being the average across actions soft Q-value.

For automatic entropy adjustment for the α coefficient, we solve the constraint problem of max $\mathbb{E}_{a_t \sim \pi_\phi}[R_t]$ under $\mathbb{E}_{a_t \sim \pi_\phi}[-\ln \pi_\phi] \geq \overline{H}$. The minimum desired expected entropy of any environment is heuristically set as the cardinality of the action space $\overline{H} = -|\mathcal{A}|$ [86]. Using gradient descent from convex optimisation as an approximation, after updating to π'_ϕ, the optimal solution is $\alpha_t^* = \arg\min_{\alpha_t} \mathbb{E}_{a_t \sim \pi'_\phi} \left[-\alpha_t(\ln \pi'_\phi + \overline{H})\right]$ [86, 132].

The Soft Actor-Critic (SAC) algorithm can be expressed at every step the agent takes as sequentially updating three parameters with following objectives to be minimised

$$J(\theta) = \mathbb{E}_{(s,a,r,s') \sim U(\mathcal{D})} \left[\frac{1}{2} \left(Q_{\overline{\theta}}^{\text{soft}} - Q_\theta^{\text{soft}} \right)^2 \right] \tag{38}$$

$$J(\phi) = \mathbb{E}_{(s,a,r,s') \sim U(\mathcal{D})} \left[\alpha \ln \pi_\phi - Q_\theta^{\text{soft}} \right] \tag{39}$$

$$J(\alpha) = \mathbb{E}_{(s,a,r,s') \sim U(\mathcal{D})} \left[-\alpha \left(\ln \pi_\phi + \overline{H} \right) \right] \tag{40}$$

where the target action-value function in Eq. (38) is constructed using Eq. (35). SAC also utilises the clipped double-Q learning feature of TD3 for updating both ϕ and θ networks. SAC has the strong advantage of minimal hyperparameters pertaining largely to the neural network architecture. SAC-Discrete [88] is also available that requires minimal changes to policy optimisation where $\pi_\phi(a|s)$ outputs a probability rather than a density.

3.5 Robust Critic Evaluation

The use of the MSE loss function in Eqs. (30) and (38) is deeply entrenched in all algorithms. We propose to investigate the effect of 10 different loss functions. Firstly we examine the effect of MSE and higher even powers where $\mathbb{E}_{U(\mathcal{D})}\left[(Q_{\bar{\theta}} - Q_{\theta})^{2+n}\right]$ for $n = 0, 2, 4, 6$. The effects of this loss amplification is to see whether giving substantially larger weights to outliers is beneficial to learning. Outliers in this case represent situations in the mini-batch where the difference between a particular samples current value of a state and its target value is excessively large. Amplifying these effects will force the optimiser to heavily modify the responsible parameters in the neural network. The other six loss functions presented in [94, 133–138] represent different degrees of outlier detection and reduction.

There has been considerable research into the merit these functions in NMF factorisation with a summary and literature review available in [94]. Briefly, NMF is formulated with $V - WH$ where V is the actual (fixed) matrix and WH is the learned representation, the difference is then minimised. While these dictionary learning methods have largely been shelved in preference for deep learning, [94] reveals how truncated Cauchy NMF is vastly superior to existing methods while being computational intensive. We propose to loosely connect this with the difference $Q_{\bar{\theta}} - Q_{\theta}$. An issue is that $Q_{\bar{\theta}}$ is also a learned quantity that varies across time unlike the matrix V. Therefore, TD3 performance should be more stable due to delaying target network updates to occur every second step unlike SAC.

Another argument for shifting away from MSE also based on its exponent is that it resembles standard deviation. An extremely insightful discussion in [14, 139] traces the origin of standard deviation (STD) to its historical roots. They find that preference of STD to mean absolute deviation (MAD) in the statistical sciences is due to a dispute in the 1920s where STD is shown to be 12.5% more asymptotically efficient than MAD if and only if the underlying data set is normally distributed. Under the presence of even minuscule ‘fat tails’, MAD is proven to be overwhelmingly more efficient. Relevant to our discussion, MAD functionally resembles the mean absolute error (MAE) loss function. Since the underlying agent data is also very unlikely to be Gaussian there exists a strong basis to investigate MAE.

Explicitly, we investigate the following functions arranged in increasing levels of outlier suppression

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{MSE}_n\left(Q_{\bar{\theta}}, Q_{\theta}\right)\right] = \frac{1}{N}\sum_{i=1}^{N}(Q_{\bar{\theta}}^i - Q_{\theta}^i)^{2+n}, \qquad \text{for } n = 0, 2, 4, 6 \tag{41}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{Huber}\left(Q_{\bar{\theta}}, Q_{\theta}\right)\right] = \frac{1}{N}\sum_{i=1}^{N}\begin{cases} \frac{1}{2}(Q_{\bar{\theta}}^i - Q_{\theta}^i)^2, & \text{if } |Q_{\bar{\theta}}^i - Q_{\theta}^i| < 1 \\ |Q_{\bar{\theta}}^i - Q_{\theta}^i| - \frac{1}{2}, & \text{otherwise} \end{cases} \tag{42}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{MAE}\left(Q_{\bar{\theta}}, Q_{\theta}\right)\right] = \frac{1}{N}\sum_{i=1}^{N}|Q_{\bar{\theta}}^i - Q_{\theta}^i| \tag{43}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{HSC}\left(Q_{\bar{\theta}}, Q_{\theta}\right)\right] = \frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{1 + (Q_{\bar{\theta}}^i - Q_{\theta}^i)^2} - 1\right) \tag{44}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{Cauchy}\left(Q_{\bar{\theta}}, Q_{\theta}, \omega\right)\right] = \frac{1}{N}\sum_{i=1}^{N}\ln\left(1 + \left(\frac{Q_{\bar{\theta}}^i - Q_{\theta}^i}{\omega}\right)^2\right) \tag{45}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{TCauchy}\left(Q_{\bar{\theta}}, Q_{\theta}, \omega, \xi\right)\right] = \frac{1}{N}\sum_{i=1}^{N}\begin{cases} \ln\left(1 + \omega^{-2}\left(Q_{\bar{\theta}}^i - Q_{\theta}^i\right)^2\right), & \text{if } 0 \leq \left(Q_{\bar{\theta}}^i - Q_{\theta}^i\right)^2 \leq \omega^2\xi \\ \ln\left(1 + \xi\right), & \text{otherwise} \end{cases} \tag{46}$$

$$\mathbb{E}_{U(\mathcal{D})}\left[\mathrm{CIM}\left(Q_{\bar{\theta}}, Q_{\theta}, \sigma\right)\right] = \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{1}{\sqrt{2\pi\sigma}}\exp\left(\frac{-(Q_{\bar{\theta}}^i - Q_{\theta}^i)^2}{2\sigma^2}\right)\right) \tag{47}$$

where HSC, TCauchy, and CIM refer to hypersurface cost-based, truncated Cauchy, and correntropy induced metric

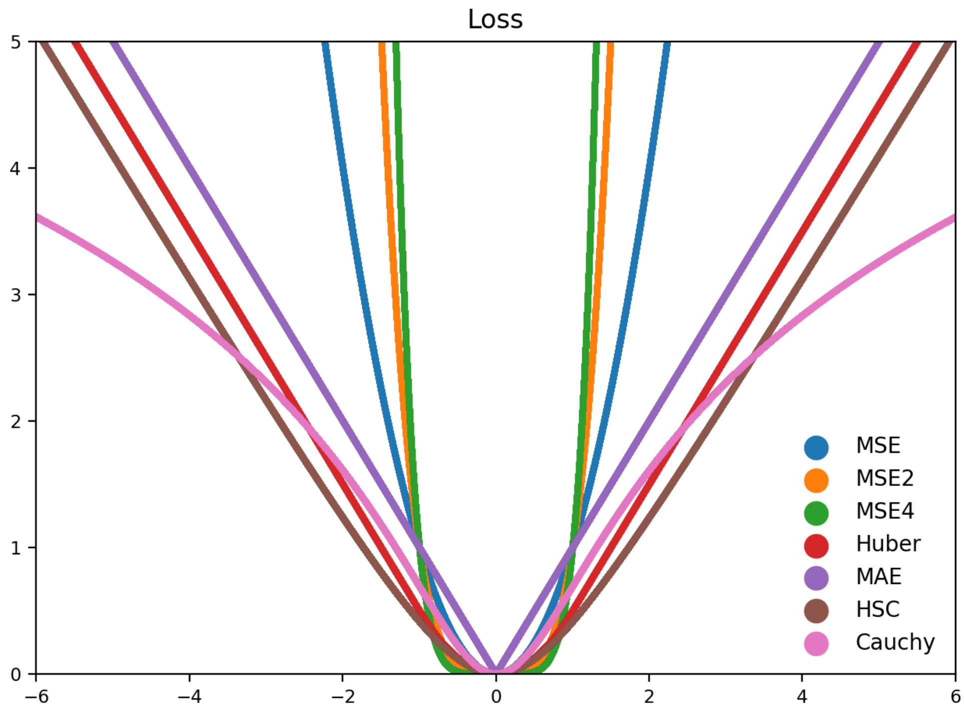

Figure 7: Several of the critic loss function in Eqs. (41-47).

loss functions respectively.

The scale parameters are empirically evaluated at each training step with the Cauchy scale parameter ω obtained iteratively via the Nagy algorithm [136] with

$$\omega_{t+1} = \omega_t \cdot \left[\left(\frac{1}{N} \sum_{i=1}^{N} \left(1 + \left(\frac{Q^i_{\bar{\theta}} - Q^i_{\theta}}{\omega_t} \right)^2 \right)^{-1} \right)^{-1} - 1 \right]^{1/2} \tag{48}$$

where $\omega_0 > 0$ and is functionally similar to the previous scale multiplied by a harmonic mean. The CIM kernel size σ is calculated as the empirical standard deviation of the mini-batch [138] with

$$\sigma_t^2 = \frac{1}{N} \sum_{i=1}^{N} \left((Q^i_{\bar{\theta}} - Q^i_{\theta}) - \mu_t \right)^2 \tag{49}$$

where $\mu_t = \frac{1}{N} \sum_{i=1}^{N} (Q^i_{\bar{\theta}} - Q^i_{\theta})$ is the empirical mean. Behaviour of these parameters can be independently evaluated using any loss function. The Cauchy truncation level ξ in Eq. (46) is empirically determined [94] using the 3σ rule where mini-batch samples $|(Q^i_{\bar{\theta}} - Q^i_{\theta}) - \mu_t| > 3\sigma_t$ exceeding this value are set to zero $|Q^i_{\bar{\theta}} - Q^i_{\theta}| = 0$ and ignored. In Fig. 7 we provide visualisations for some of the functions.

3.6 Preasymptotics and the Tail Exponent

A key assumption present throughout machine learning and statistics discussed in Section 3.3 is that the Monte-Carlo approach is valid if the data is i.i.d. by the law of large numbers. Reinforcement learning by definition is non-i.i.d. but the Kolmogorov theorem on the strong law of large numbers reveals that convergence is still guaranteed provided the true unknown underlying distribution Ω has finite variance σ^2 [131]. If this is true for a random variable $X \sim \Omega$, we have the condition

$$\frac{1}{M}\sum_{i=1}^{M} X_i - \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_{X\sim\Omega}[X_i] \underset{M\to\infty}{\rightarrow} 0 \tag{50}$$

and so in practice the universal standard is to simply utilise the Monte-Carlo approach

$$\mathbb{E}_{X\sim\Omega}[X] = \frac{1}{M}\sum_{i=1}^{M} X_i \tag{51}$$

This is especially important in machine learning where parameter optimisation is based entirely on either minimising aggregate losses or maximising aggregate gains. Furthermore, this is usually performed using mini-batch learning $N \subset M$ as it is far more computationally efficient. The mini-batch is often uniformly sampled $N \sim U(M)$ at each learning iteration to ensure overall it is reflective of the actual sample M as the number of iterations increases.

A gigantic and monumental question that appears to have never been seriously asked regarding this method is whether this remains valid for finite M [14]. As the real-world contains only finite sample sizes, the convergence may never formally occur. This problem is even more critical in machine learning as we utilise $N < M$ and so the equality is even more debatable. Essentially, we obtain Ω_N, treat it as an accurate and efficient reflection of Ω_M, which is then assumed to be a perfect representation of the true Ω. In Eq. (51) the left is referred to as the true population mean μ and the right is called the empirical mean $\bar{x}$. Clearly empirical means are not empirical for $N < M \ll \infty$.

The question is then in what regimes does utilising $\bar{x}$ as substitute for μ remain appropriate and to what degree, that is, what is the preasymptotic behaviour of $\bar{x}$? This problem is one of the primary focuses of [14, 95–97] which provides explicit answers to all these and many more questions. Their focus is mainly on the statistical consequences of fat tail, namely distributions Ω with far larger kurtosis than the Gaussian. As discussed in Section 1.3, these probability density functions are solely defined by their infrequent extreme outliers and are more accurate representations of almost all real-world systems. Hence using $\bar{x}$ is uninformative as 'empirically' it will be heavily biased by the noise and fail accurately account for shocks.

Detailed discussion on power laws (fat-tailed distributions) and the Generalised Pareto distribution (GPD) is outside the scope of this work and so we present the final results without proof [140–143]. What is relevant is that the tail exponent α governs the fatness of the tails, lower implies less thin, and is the characteristic feature of a power law. Furthermore, the moment of order p for $X \sim$ GPD only exists if $\alpha > p$. Hence if $\alpha < 1$ all moments: mean, variance, skew, kurtosis, and higher are formally undefined and cannot be estimated even in the limit $M \to \infty$. One extreme is the Gaussian with $\alpha \to \infty$, while Cauchy has $\alpha < 1$. Note MAD only requires the mean to exist.

A very important point is that since sample is finite, even if $\alpha < 1$, we can still 'empirically' calculate all moments. These point estimates are therefore entirely misleading and lead to a very false sense of confidence about the nature of distribution Ω_M let alone Ω. A more conservative approach to evaluating them involves first determining the tail exponent α for the sample, only then can we crudely gauge the true nature of Ω.

Assuming a strictly right-tailed fat distribution capped with $X_i \geq 0 \;\forall\; i$, we are very likely to obtain $\bar{x} \ll \mu$ for $M < \infty$ as seen in Fig. 8. Using this estimate will eventually lead to a rude awakening when a outlier naturally appears. To counter this gross underestimation [14, 95] propose first constructing a new distribution Ω'_M using the M known samples that exhibits the correct right-tail behaviour. The mean of this distribution $\mathbb{E}_{X\sim\Omega'_M}[X] = \mu_s$ called the 'shadow' mean is then considered a far more accurate representation of the true mean μ.

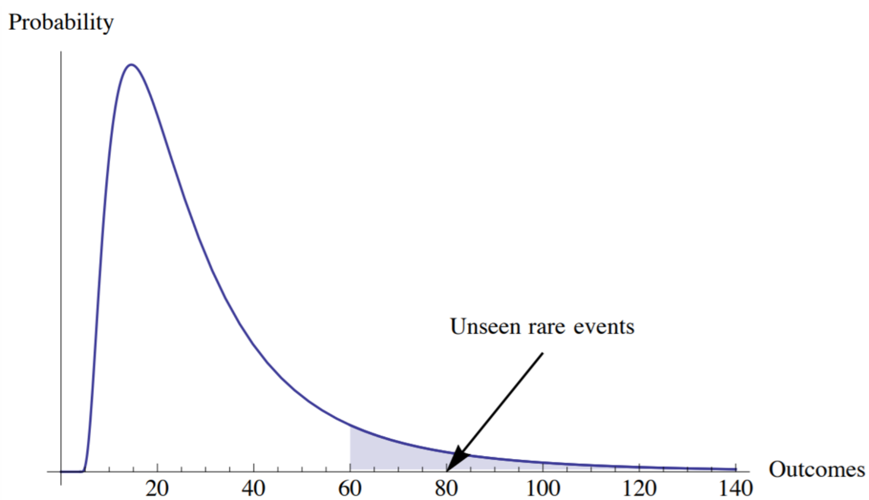

Figure 8: Typical structure of a fat-tailed distribution where the absence of rare large outliers in the obtained finite sample leads to severely underestimated empirical means. Adapted from [14].

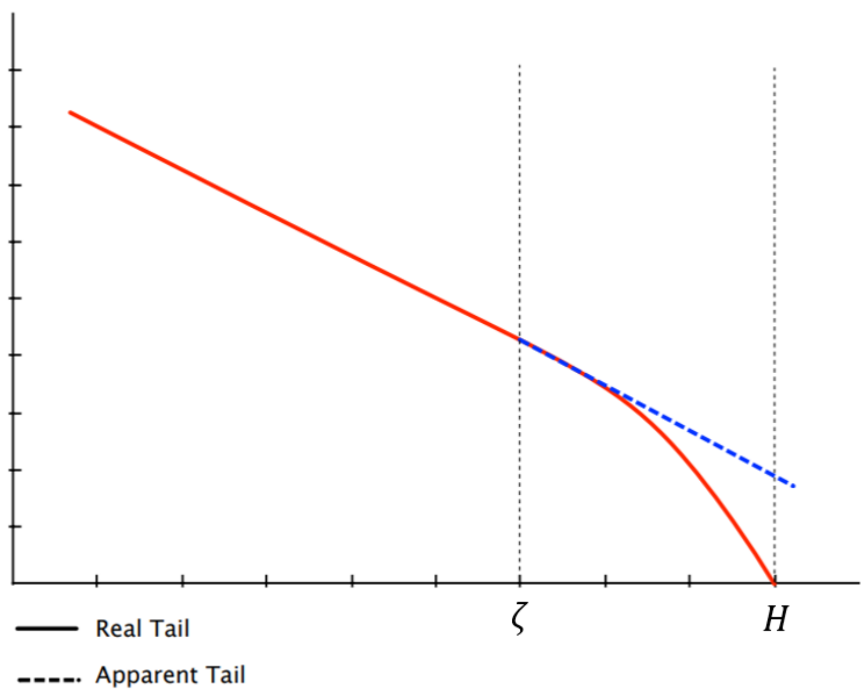

Figure 9: Graphical representation of why a very large finite upper support $H \gg \zeta$ is required to ensure both numerical stability and realistic power law decay when estimating a shadow mean from a finite sample. For all practical purposes the existence of H is ignored as divergence is only evident when we approach the limit, hence using the apparent tail is considered a reasonable approximation. Adapted from [14].

To derive Ω'_M for the finite set of observations $\omega = \{X_1, ..., X_M\}$ first define the observed maximum $\zeta = \max(\omega)$. Next we require the finite support $X \in [L, H]$ where $L \geq 0$ and H is set to be exceedingly large but finite so that the probability of observing it is minuscule so that $\zeta \ll H < \infty$ shown in Fig. 9. Another feature is defining a threshold $L^* \geq L$ where we are uninterested in the region $X < L^*$ as these occurrences are treated as noise.

For random variable with a known support $X \in [L, H]$ define a smooth function $\varphi(X)$ requiring $\varphi \in C^\infty$ (differentiable to all degrees), $\varphi^{-1}(\infty) = H$, and $\varphi(L) = \varphi^{-1}(L) = L$, parameterised non-uniquely as

$$\varphi(X) = L - \ln\left|\frac{H - X}{H - L}\right| = L + \ln\left|\frac{H - L}{H - X}\right| \tag{52}$$

Then define a new random variable $Z \equiv \varphi(X)$ with bounds $Z \in [L, \infty)$ and for very large H we can approximate $Z \approx X$ given the first-order Taylor series expansion $\ln|y| \approx y - 1$ about $y = 1$. Therefore, the tail structures in Fig. 9 of the bounded X (real tail) and unbounded Z (apparent tail) are identical up till the vicinity of huge H. Hence the goal is to model the tail of Z and then convert back to X with the inverse

$$X = \varphi^{-1}(Z) = H + (L - H)e^{\frac{L-Z}{H}} \tag{53}$$

as a non-unique method to estimate the shadow moments of the distribution. Note any of the moments of the Z distribution do not formally exist as it is formally unbounded.

Next as we are only interested in the right tail, we focus strictly on values exceeding a interest threshold u where $u = L^* \geq L$. Then define a random variable w as the excess $w \equiv X - u$. In the limit $u \to \infty$, the cumulative $D(w; \alpha, \varsigma)$ and probability $w \sim d(w; \alpha, \varsigma)$ density functions for $w \geq 0$ can be approximated by a GDP where

$$D(w; \alpha, \varsigma) = \begin{cases} 1 - \left(1 + \frac{w}{\alpha\varsigma}\right)^{-\alpha}, & \text{if } \alpha < \infty \\ 1 - e^{-\frac{w}{\varsigma}}, & \text{otherwise} \end{cases} \tag{54}$$

$$d(w; \alpha, \varsigma) = \frac{1}{\varsigma}\left(1 + \frac{w}{\alpha\varsigma}\right)^{-\alpha-1}, \quad w \in [L^*, \infty) \tag{55}$$

with tail exponent $\alpha \in (-\infty, \infty)$ and the scale parameter $\varsigma \in (0, \infty)$ estimated from the data using a several possible approaches [140–143]. In this case the we have the equivalence of the of both distributions $X \sim f(X; \alpha, \varsigma)$ and $Z \sim d(Z; \alpha, \varsigma)$ where

$$\int_{L^*}^{\varphi^{-1}(\infty)} \mathrm{d}X \; f(X; \alpha, \varsigma) = \int_{L^*}^{\infty} \mathrm{d}Z \; d(Z; \alpha, \varsigma) = 1 \tag{56}$$

by definition. Note in particular for both cases the excess difference from the threshold L^* is taken to be compatible with the formulation of the GDP. To find $f(X; \alpha, \varsigma)$ we solve the integral equation by substituting Eq. (52) into Eq. (55), recalling $\varphi^{-1}(L) = L$, and using a modified Eq. (55) to forrm the ansatz

$$f(X; \alpha, \varsigma) = \frac{H}{\varsigma(H - X)}\left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H - L}{H - X}\right|\right)^{-\alpha-1}, \quad X \in [L^*, H] \tag{57}$$

with integration yielding

$$\begin{aligned} \lim_{B \to H} \int_{L^*}^{B} \mathrm{d}X \; f(X; \alpha, \varsigma) &= -\lim_{B \to H} \left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H - L^*}{H - X}\right|\right)^{-\alpha}\Bigg|_{X=L^*}^{X=B} \\ &= 1 - \lim_{B \to H} \left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H - L^*}{H - B}\right|\right)^{-\alpha} \end{aligned}$$

$$= 1 - \left(1 + \frac{H}{\alpha\varsigma}\left(\ln|H - L^*| - \lim_{B\to H} \ln|H - B|\right)\right)^{-\alpha} \tag{58}$$

which converges to unity for $\alpha > 0$. For our purposes as we are not overly interested in the behaviour in the vicinity of H, we can also assume the second term approaches zero before the limit is reached where $\lim_{B\to E} \frac{H-L^*}{H-B} \to \infty$ for $\zeta \ll E \ll H < \infty$. Therefore, as $f(X\alpha,\varsigma)$ and $d(Z;\alpha,\varsigma)$ are one-to-one transformations on each other, they must share the same tail exponent and scale parameter.

The shadow moments of order p of a fat-tailed random variable X conditional on $X \geq L^*$ are then generally

$$\mathbb{E}_{X\sim\Omega'_M}[X^p|X > L^*] \equiv \int_{L^*}^{H} \mathrm{d}X\ X^p f(X;\alpha,\varsigma) \tag{59}$$

with our desired shadow mean being

$$\begin{aligned}
\mathbb{E}_{X\sim\Omega'_M}[X|X > L^*] &= \int_{L^*}^{H} \mathrm{d}X\ X f(X;\alpha,\varsigma) \\
&\overset{(a)}{=} -\frac{H}{\varsigma}\lim_{B\to H}\int_{H-L^*}^{H-B} \mathrm{d}t\ \frac{(H-t)}{t}\left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H-L}{t}\right|\right)^{-\alpha-1} \\
&\overset{(b)}{=} \lim_{B\to H}\left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H-L^*}{t}\right|\right)^{-\alpha} \\
&\quad\times\left[\alpha(H-L^*)e^{\frac{\alpha\varsigma}{H}}\left(\frac{\alpha\varsigma}{H} + \ln\left|\frac{H-L^*}{t}\right|\right)^{\alpha}\Gamma\left(-\alpha, \frac{\alpha\varsigma}{H} + \ln\left|\frac{H-L^*}{t}\right|\right) - H\right]\Bigg|_{t=H-L^*}^{t=H-B} \\
&= \lim_{B\to H}\left[\alpha(H-L^*)e^{\frac{\alpha\varsigma}{H}}\left(\frac{\alpha\varsigma}{H}\right)^{\alpha}\Gamma\left(-\alpha, \frac{\alpha\varsigma}{H} + \ln\left|\frac{H-L^*}{t}\right|\right) - H\left(1 + \frac{H}{\alpha\varsigma}\ln\left|\frac{H-L^*}{t}\right|\right)^{-\alpha}\right]\Bigg|_{t=H-L^*}^{t=H-B} \\
&\overset{(c)}{=} H - \alpha(H-L^*)e^{\frac{\alpha\varsigma}{H}}\left(\frac{\alpha\varsigma}{H}\right)^{\alpha}\Gamma\left(-\alpha, \frac{\alpha\varsigma}{H}\right) \\
&\overset{(d)}{=} H - (H-L^*)\left[1 - e^{\frac{\alpha\varsigma}{H}}\left(\frac{\alpha\varsigma}{H}\right)^{\alpha}\Gamma\left(1-\alpha, \frac{\alpha\varsigma}{H}\right)\right] \\
&= L^* + (H-L^*)e^{\frac{\alpha\varsigma}{H}}\left(\frac{\alpha\varsigma}{H}\right)^{\alpha}\Gamma\left(1-\alpha, \frac{\alpha\varsigma}{H}\right)
\end{aligned} \tag{60}$$

where $\Gamma(s,x)$ for strictly $s > 0$ and $x \geq 0$ is the upper incomplete gamma function defined as

$$\Gamma(s,x) \equiv \int_{x}^{\infty} \mathrm{d}y\ y^{s-1}e^{-y} \tag{61}$$

and in (a) we perform the change of variable $t = H - X$ and so $\mathrm{d}t = -\mathrm{d}X$, (b) involves trivial integration by inspection, or alternatively, plebeians can perform lengthy and cumbersome manipulation of known results from tables [144] wherein

$$\begin{aligned}
\int \mathrm{d}x\ \frac{1-x}{x(1-\ln|x|)^d} &= (1-\ln|x|)^{1-d}\left(\frac{1}{d-1} - e\int_{1}^{\infty}\mathrm{d}t\ \frac{e^{-(1-\ln|x|)t}}{t^d}\right) + C \\
&= (1-\ln|x|)^{1-d}\left(\frac{1}{d-1} - e(1-\ln|x|)^{d-1}\Gamma(1-d, 1-\ln|x|)\right) + C \\
&= \frac{(1-\ln|x|)^{1-d}}{d-1} - e\Gamma(1-d, 1-\ln|x|) + C
\end{aligned} \tag{62}$$

with e as Euler's number, in (c) for the upper bound we use the same logic as Eq. (58) and also the limit

$$\lim_{B\to E}\Gamma\left(-\alpha, \frac{\alpha\varsigma}{H} + \ln\left|\frac{H-L^*}{H-B}\right|\right) \to 0 \tag{63}$$

for $L^* \ll \zeta \ll E \ll H$ as the upper support need never be reached, while the lower bound results in all logarithms yielding zero, and finally (d) uses the known recurrence relation $\Gamma(1+a,x) = a\Gamma(a,x) + x^a e^{-x}$ taught throughout kindergarten [145]. In what follows for simplicity we assume $\varsigma = 1$ as its computationally intense estimation [140] is not practical for mini-batch reinforcement learning over millions of training iterations.

Using then the provided sample ω, the shadow mean of the known distribution Ω'_M is expressed as

$$\mathbb{E}_{X\sim\Omega'_M}\left[X|X>L^*\right] = L^* + (H - L^*)e^{\frac{\alpha}{H}}\left(\frac{\alpha}{H}\right)^{\alpha}\Gamma\left(1-\alpha, \frac{\alpha}{H}\right) \tag{64}$$

where $\mathbb{E}_{X\sim\Omega'_M}\left[X|X>L\right] = \mathbb{E}_{X\sim\Omega'_M}\left[X\right]$. Furthermore, given that by construction we should have $\frac{\alpha}{H} \ll 1$, the fact that numerical integration of the usual (complete) gamma function $\Gamma(s)$ is far more common and hence easier to perform, and the known relation with the lower incomplete gamma function $\gamma(s,x)$ where $\Gamma(s) = \Gamma(s,x) + \gamma(s,x)$, it may be easier to numerically evaluate the difference

$$\Gamma\left(1-\alpha, \frac{\alpha}{H}\right) = \Gamma\left(1-\alpha\right) - \int_0^{\frac{\alpha}{H}} \mathrm{d}y\ y^{-\alpha}e^{-y} \tag{65}$$

$$\approx \int_0^{\infty} \mathrm{d}y\ y^{-\alpha}e^{-y} \tag{66}$$

where great care must be taken when using the approximation as it should only be considered if $\frac{\alpha}{H} \to 0$ with certainty. Therefore, a smaller α (larger kurtosis of sample ω) correctly increases the value obtained from integration by both reducing the impact of $\gamma(s,x)$, and resulting in greater divergence of $y^{-\alpha}e^{-y}$ near the origin.

This entire approaches rests on α, commonly represented in literature with the scale parameter as $\gamma = \alpha^{-1}$, and whose estimation is no easy task as there are bountiful methods for obtaining an empirical value. Extreme value theory offers general approaches involve using MLE or the method of moments [140]. In ω let $X_{1,M} \leq ... \leq X_{M,M}$ be the order statistics, then define

$$H_{k,M}^{(j)} \equiv \frac{1}{k}\sum_{i=0}^{k-1}\left(\ln|X_{M-i,M}| - \ln|X_{M-k,M}|\right)^j \tag{67}$$

where $X_{M-k,M}$ is an empirically determined intermediate order statistics used as threshold for when the power law dominates the distribution. Importantly, for asymptotic reasons we take $1 \leq k < M$ so that for $k \to \infty$, $k/n \to 0$ as $n \to \infty$ by the strong law of large numbers [140]. However, since we are constrained with requiring $0 < \gamma < 1$, the simplest method is the Hill estimator [146] where

$$\begin{aligned}\hat{\gamma} = H_{k,M}^{(1)} &= \frac{1}{k}\left[\ln\left|\prod_{i=0}^{k-1} X_{M-i,M}\right| - k\ln|X_{M-k,M}|\right] \\ &= \ln\left|\left(\prod_{i=0}^{k-1} X_{M-i,M}\right)^{\frac{1}{k}}\right| - \ln|X_{M-k,M}|\end{aligned} \tag{68}$$

observing that this represents the difference between the natural logarithm of geometric means. Another perspective can be seen by first defining the geometric mean $G_k = \left(\prod_{i=0}^{k-1} X_{M-i,M}\right)^{\frac{1}{k}} \geq X_{M-k,M}$ by definition. We then write the ratio $G_k/X_{M-k,M} = 1 + R_k/X_{M-k,M}$ where $R_k > 0$ can be interpreted as an excess 'return'. Hence we are able to express

$$\hat{\gamma} = H_{k,M}^{(1)} = \ln\left|\frac{G_k}{X_{M-k,M}}\right| \geq 0 \tag{69}$$

$$e^{\hat{\gamma}} = e^{H_{k,M}^{(1)}} = 1 + \frac{R_k}{X_{M-k,M}} \geq 1 \tag{70}$$

and so what the Hill estimator is essentially measuring is the exponential 'return' of the geometric average of extreme values relative to the selected intermediate order statistic $X_{M-k,M}$. Notice that this is identical to the time-average growth rate seen in Eq. (10) being the most accurate method of measuring performance.

The more advanced method of moments yields the following estimator

$$\hat{\gamma} = H_{k,M}^{(1)} + 1 - \frac{1}{2}\left(1 - \frac{\left(H_{k,M}^{(1)}\right)^2}{H_{k,M}^{(2)}}\right)^{-1} \tag{71}$$

to then obtain $\hat{\alpha} = \hat{\gamma}^{-1}$ [147, 148]. This entire process rests on the correct selection of the k extreme values. Selecting small k causes large variance in $\sigma_\gamma^2 \sim \gamma^2$ while for large k the entire sample is unlikely to adhere to a power law. The choice of k is then dependent on the properties of the underlying process X and appears a very subjective decision. There are however a plethora of methods of doing so that are not exact but offer a more objective approach, however these are also subject to other bias. Many of these approaches are require expensive computational optimisation and so are not feasible for mini-batch learning.

There is though far more simpler approach that is consistent with the derivation of Eq. (64) as the GPD assumption assumes a very strong power law in a region where it totally dominates the behaviour. At this point the distribution can be simplified by approximating it as Pareto with $\hat{\gamma}$ estimated as the slope of a Zipf or Pareto quantile (Q-Q) plot that is expected to be nearly linear [149–152]. This is done by plotting

$$(r_k, s_k) = \left(\ln\left|\frac{M+1}{k}\right|, \ln|X_{M-k+1,M}|\right), \qquad k = 1, ..., M \tag{72}$$

with its gradient approximately equal to $\hat{\gamma}$. A more refined version involves allowing the power law to only dominate at the infinite limit where the generalised quantile plot becomes

$$(r_k, s_k) = \left(\ln\left|\frac{M+1}{k}\right|, \ln|X_{M-k+1,M} \cdot H_{k,M}^{(1)}|\right), \qquad k = 1, ..., M \tag{73}$$

which is ultimately linear for small k. The slope is estimated with the usual linear regression coefficient

$$\hat{\gamma} = \frac{\sum_{i=1}^{k}(r_i - \bar{r})(s_i - \bar{s})}{\sum_{i=1}^{k}(r_i - \bar{r})^2} \tag{74}$$

noticing that by construction we will always obtain $\hat{\gamma} \geq 0$ as we are using ordered statistics.

Furthermore, using Zipf plots suppose we construct an estimate using $k = b \ll c$ points we will generally find that $\hat{\gamma}_b \geq \hat{\gamma}_c$ or $\hat{\alpha}_b \leq \hat{\alpha}_c$ if the sample actually contains some extreme values. This is because the values are sorted in terms of largest to smallest and so including fewer values will lead to steeper gradients as there are fewer points $r_1, ..., r_k$ with tighter spacing. In other words, we will see comparable (albeit a little less) rise, for less run, causing a greater incline in the line of best fit. Instead of then tuning for appropriate k, if we take the complete sample $k = M$ we are able to construct the upper bound for the true value $\hat{\alpha} \leq \hat{\alpha}_M$. Additionally, if $\hat{\alpha}_M < 1$ using Eq. (72), then it likely must also be the case using Eq. (73). Hence if $\hat{\alpha}_M < 1$, it will be satisfactory to state that the sample contains extreme values and so justification for using the shadow mean is strong.

Suppose we have either determined the optimal intermediate $X_{M-k,M}$ or have used either of the Zipf plots to obtain $\hat{\gamma} = \hat{\alpha}^{-1}$. For $\hat{\alpha} \in (0, \infty)$ use of Eq. (64) is considered acceptable. If $0 < \alpha < 1$ use of Eq. (51) is entirely inappropriate as the sample appears to exhibit fat tails with an undefined traditional mean and so only the shadow mean should be utilised. If $\hat{\alpha} \leq 0$, then we have no choice but to estimate with the empirical mean in Eq. (51). Overall, we then have the deterministic shadow mean estimate

$$\mu_s(L^*, H, \hat{\alpha}) \equiv L^* + (H - L^*) e^{\frac{\hat{\alpha}}{H}} \left(\frac{\hat{\alpha}}{H} \right)^{\hat{\alpha}} \int_{\frac{\hat{\alpha}}{H}}^{\infty} \mathrm{d}y \; y^{-\hat{\alpha}} e^{-y} \tag{75}$$

In machine learning it is an open question as to whether this approach could potentially be utilised. Most common are situations in supervised learning where $X \sim (Y - \bar{Y})^2$ are 'empirical' losses from a mini-batch and then aggregated using Eq. (51). In this case clearly $X_i \geq 0 \; \forall \; i$ and so we have a situation with unbounded right-tailed skew. Perhaps this preasymptotic approach may lead to superior results.

In reinforcement learning we have critic and soft-critic losses of the form $X \sim (Q_\theta - Q_{\bar{\theta}})^2$ which certainly might be amenable to this approach. The fact that this represents the difference between quantities that are only separated by one or two learning iterations, it is likely that it might have somewhat stable properties. In this case, the extreme values will be isolated solely to samples in the mini-batch where valuations are most divergent. Using the shadow mean would then amplify the impact of these outliers and force the optimisation process to place greater emphasis on adjusting the network parameters to account for them. This can therefore be considered somewhat analogous to the case in Eq. (41) for $n > 0$ amplifying the effect of these outliers. This approach however is far more sophisticated as it does not inflate the smaller values.

In this case the backpropagation of the graph will centre on adjusting parameters so the tail exponent $\hat{\alpha}$ increases leading to a reduced shadow mean μ_s. Furthermore, for each mini-batch, a systematic scheme for estimating the supports $[L^*, H]$ is required. Setting $L^* = L = 0$ intuitively appears to the strictest requirement as therefore we are interested in minimising all critic errors regardless of how minor they are in a given mini-batch. Selection of H is a trickier endeavour, as we require $\zeta \lll H < \infty$, we must utilise both the known sample maximum ζ and the our domain knowledge to estimate its value.

Theoretically, uniformly sampling experiences $\left(s_j, a_j, r_j, s'_j\right)$ for $j = 1, ..., N$ from the replay buffer $\mathcal{D}$ during training iteration t, the critic loss $X_{j,t} = (r_j + \gamma Q_{\bar{\theta},t}(s'_j, a') - Q_{\theta,t}(s_j, a_j))^2$ are naturally uncapped $X \in [0, \infty)$ and form the mini-batch $\omega_t = \{X_{1,t}, ..., X_{N,t}\}$. However we also expect as the number of training iterations increases that $X_{j,t+\delta t} \leq X_{j,t}$ for $\delta t \gg 0$ as indication of successful agent learning. Hence we demand that the upper support is dependant on both time and the known largest loss $\zeta_t = \max(\omega_t)$ contained in the mini-batch $H_t = H(t, \zeta_t)$. We can further simplify this approach by assuming that ζ_t completely encapsulates the time-dependence so that $H_t = H(\zeta_t)$. Recall again that the exact value of the finite maximum loss H is not required as we do not ever expect it to be approached. Therefore a very simple approach would be to utilise a constant linear multiplier $H_t = \kappa_t \zeta_t$ for $\kappa_t \in (1, \infty)$ and its exact value is empirically determined on a case-by-case basis.

One crude method of potentially gauging a lower bound κ_t^l would be to estimate the time-dependent value required for equivalence between both empirical and shadow means

$$\frac{1}{N}\sum_{j=1}^{N} X_{j,t} = L^* + (\kappa_t^l \zeta_t - L^*) e^{\frac{\hat{\alpha}_t}{\kappa_t^l \zeta_t}} \left(\frac{\hat{\alpha}_t}{\kappa_t^l \zeta_t}\right)^{\hat{\alpha}_t} \int_{\frac{\hat{\alpha}_t}{\kappa_t^l \zeta_t}}^{\infty} \mathrm{d}y \; y^{-\hat{\alpha}_t} e^{-y} \tag{76}$$

Numerically solving this transcendental equation would likely find $\kappa_t^l \leq 1$ if ω_t is very fat-tailed with $0 < \hat{\alpha}_t \ll 1$. Then examining the time-dependent structure of κ_t^l for successful agents would allow us to gain insight on the overall process. Generally a larger κ_t^l would necessarily demand a higher estimate of the finite support $H_t = \kappa_t \zeta_t$ where $\kappa_t^l \ll \kappa_t \; \forall \; t$. Overall, this is a ad hoc procedure and in practice setting a large constant for all time $\kappa = \kappa_t$ is likely the most efficient approach.

Additionally, this more formal approach can be improved as presenting results in terms of κ_t^l is likely uninformative given its volatility and fact the maximum value is scarcely analysed. Instead to convey the scale, we should express the multiplier in terms of the empirical mean at each time step where

$$\kappa_{\text{eqv}} = \kappa^l \zeta \left(\frac{1}{N}\sum_{j=1}^{N} X_j\right)^{-1} \tag{77}$$

Regarding actor losses, deterministic policy gradient are of the form $X \sim -Q_\theta$ and soft-policies are $X \sim \alpha \ln \pi_\phi - Q_\theta$. The former seems applicable if one applies the negative factor at the final step, while latter is unlikely to be bounded by zero. Finally, entropy temperature $X \sim -\alpha \left(\ln \pi_\phi + \overline{H}\right)$ might also be feasible with this approach as we can again factor out the negative multiplier.

Therefore, to utilise this approach there are several hyperparameters to tune in contrast with global standard in Eq. (51). For actors and each of the twin critics (and also temperature if using SAC) we need to determine suitable L^* and H, and also have a systematic procedure to identify the intermediate order statistics $X_{M-k,M}$ for each network or utilise the Zipf plot. None of these selections are remotely trivial. Use of Zipf plots therefore is likely the superior choice as the computations are routine. An additional complication when mini-batch learning over many iterations is that these selections must be done at each iteration and so the procedure needs to be computational efficient.

3.7 Multi-Step Returns

Multi-step targets or returns [71] up to the m-step ahead of current step t are expressed as

$$Q_{\bar{\theta},t}^{(m)}(s,a) \equiv \sum_{k=0}^{m-1} \gamma^k r_{t+k} + \gamma^m Q_{\bar{\theta}}\left(s_{t+m}, a'\right) \tag{78}$$

where the action a' is sampled from either the SAC or TD3 policies. One advantage to this approach is that tuning for appropriate m can potentially rapidly accelerate learning [34, 48, 71, 101, 153, 154].

Practically, for each training step, this performed by treating the uniformly sampled experiences with each forming the tuples $\left(s_j^{(n-1)}, a_j^{(n-1)}, r_j^{(n-1)}, s_j^{(n)}\right)$ for $j = 1, ..., N$ where N is the size of the mini-batch. For each experience j, we then backwardly reconstruct the history from the replay buffer for steps $1 \rightarrow n-1 \leq m$ in order to aggregate the historical sum $\sum_{k=1}^{n-1} \gamma^k r_k$ and also obtain the 'current' state-action pair $(s^{(1)}, a^{(1)})$ to estimate the value $Q_\theta(s^{(1)}, a^{(1)})$.

Hence the usual Bellman error objective in Eq. (41) to be minimised is again

$$J(\theta) = \frac{1}{N}\sum_{j=1}^{N}\left[\left(\sum_{k=1}^{n-1}\gamma^k r_{j,k} + \gamma^n Q_{\bar{\theta}}(s_j^{(n)}, a_j')\right) - Q_\theta(s_j^{(1)}, a_j^{(1)})\right]^2 \tag{79}$$

One important subtly of this formulation during off-policy learning is what occurs if the agents' sequential interaction with environment is terminated at the mth step or earlier for any particular episode. Now if for any j, the episode is terminated at $s_j^{(n-1)}$, and so $s_j^{(n)}$ does not formally exist, therefore we forcibly set $Q_{\bar{\theta}}(s_j^{(n)}, a_j') = 0$.

Another consideration is that when sampling from the replay buffer, the situation if less than total m-steps exist in a previous episode based on how early the tuple occurs. In other words, if $n < m$ for any j, we have no alternative to using only n-steps. Therefore as we increase the number of desired m-steps, the difference $m - \bar{n} \geq 0$ will likely continue to expand where $\bar{n} \equiv \frac{1}{N}\sum_{j=1}^{N} n_j$ is the effective average number of multi-steps across the mini-batch in any given training iteration.

3.8 History Decision Processes

Consider a general environment with finite observations $o \in \mathcal{O}$ where $\mathcal{S} \in \mathcal{O}$ which is typical for partially observed non-Markovian decision processes. Often due to the complexity or the sheer size of $\mathcal{O}$, the agent is unable to directly learn from this space and so has to utilise $\mathcal{S}$ when interacting with the environment. To perform this task the agent must have access to a mapping from the complete observation space to the operating access space [115–119].

As the agent continues through time in each episode, it constructs tuples of histories $h_t \in \mathcal{H}_t : (\mathcal{O}\times\mathcal{R}\times\mathcal{A})^{t-1}\times\mathcal{O}$ where $h_t \equiv o_1 a_1 r_1 ... o_{t-1} a_{t-1} r_{t-1} o_t$ or $h_t \equiv h_{t-1} a_{t-1} r_{t-1} o_t$. Explicitly, the trajectory sequentially develops as $o_1 \rightarrow a_1 \rightarrow r_1 o_2 \rightarrow a_2 \rightarrow r_2 o_3 \rightarrow ... \rightarrow a_{t-1} \rightarrow r_{t-1} o_t$. Note this definition is different to the one presented in [117–119] but the results remain valid. The set of all finite histories is denoted by $\mathcal{H}^* = \bigcup_t \mathcal{H}_t$ and the empty set is ϵ. Histories also need not be unique, but as the agent becomes more successful, we typically expect the length of these histories to increase in environments where there is no time limit. The transition probability at any time t to the next state o_{t+1} is a function of the history-action pair (h_t, a_t) not the state-action pair (s_t, a_t). This leads to the notion of history-based decision process (HDPs) with transition probabilities $P : \mathcal{H}^* \times \mathcal{A} \rightsquigarrow \mathcal{O} \times \mathcal{R}$ where $\rightsquigarrow$ denotes a stochastic mapping [119].

This allows us to define $R_t \equiv \sum_{k=t}^{\infty} \gamma^k r(s_k, a_k)$ for stochastic policies where the state value and action-value functions are $V^\pi(h_t) \equiv \mathbb{E}\left[R_t | h_t; \pi\right]$ and $Q_\pi(h_t, a) \equiv \mathbb{E}\left[R_t | h_t, a; \pi\right]$. The optimal values are $V^*(h_t) = \max_\pi V_\pi(h_t)$ and $Q^*(h_t, a_t) = \max_\pi Q_\pi(h_t, a_t)$ where $\pi^* \in \arg\max_\pi V_\pi(\epsilon)$. Note that the optimal policy may not be unique and so is denoted as an element. New Bellman equations can be written

$$Q_\pi(h_t, a_t) = \sum_{o_{t+1}, r_t} P(o_{t+1}, r_t | h_t, a_t)\left[r_t + \gamma V_\pi(h_{t+1})\right] \tag{80}$$

$$V_\pi(h_t) = Q_\pi(h_t, \pi(h_t)) \tag{81}$$

$$Q^*(h_t, a_t) = \sum_{o_{t+1}, r_t} P(o_{t+1}, r_t | h_t, a_t)\left[r_t + \gamma V^*(h_{t+1})\right] \tag{82}$$

$$V^*(h_t) = \max_{a_t \in \mathcal{A}} Q^*(h_t, a_t) \tag{83}$$

$$\pi^*(h_t) \in \arg\max_{a_t \in \mathcal{A}} Q^*(h_t, a_t) \tag{84}$$

where they are considered pseudo-recursive and not self-consistent as length of h_{t+1} is larger than h_t and so an algorithm based on frequency of visits cannot be used [117]. Utilising this approach is impractical as the cardinality $|\mathcal{H}^*|$ is extremely large since the histories are unlikely to ever repeat. To handle this situation [117] introduce a surjective aggregation feature mapping $\phi : \mathcal{H} \to \mathcal{S}$ so that $s = \phi(h) \in \mathcal{S}$ where $\mathcal{S}$ is by definition finite and small enough to work within. The history is then reduced $h_t \equiv h_{t-1}a_{t-1}r_{t-1}o_t \to h_t \equiv h_{t-1}a_{t-1}r_{t-1}s_t$ where the (s_t, a_t) pair is sufficient. This obviously describes a MDP however [117] find its applicability to be far more general.

For the feature map ϕ, the transition probabilities are constructed via marginalisation

$$P_\phi(s_{t+1}, r_t|h_t, a_t) = \sum_{o_{t+1}:\phi(h_t a_t r_t o_{t+1})=s_{t+1}} P(o_{t+1}, r_t|h_t, a_t) \tag{85}$$

where $P_\phi \in$ MDP and therefore is stationary if $\exists\ p : P_\phi(s_{t+1}, r_t|h_t, a_t) = p(s_{t+1}, r_t|s_t, a_t)\ \forall\ \phi(h_t) = s_t$ where $p : \mathcal{S} \times \mathcal{A} \rightsquigarrow \mathcal{O} \times \mathcal{R}$. The exact aggregation for $P_\phi \in$ MDP can then be shown to yield $V_\pi(h_t) = V_\pi(s_t)$, $Q_\pi(h_t, a_t) = Q_\pi(s_t, a_t)$ where $\pi(h_t) = \pi(s_t)$, and $V^*(h_t) = V^*(s_t)$, $Q^*(h_t, a_t) = Q^*(s_t, a_t)$ where $\pi^*(h_t) = \pi^*(s_t)$ [117]. This allows us to recover the well-known Bellman equations

$$Q_\pi(s_t, a_t) = \sum_{s_{t+1}, r_t} p(s_{t+1}, r_t|s_t, a_t)\left[r_t + \gamma V_\pi(s_{t+1})\right] = \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathbb{E}\left[r_t|s_t, a_t\right] \tag{86}$$

$$V_\pi(s_t) = Q_\pi(s_t, \pi(s_t)) \tag{87}$$

$$Q^*(s_t, a_t) = \sum_{s_{t+1}, r_t} p(s_{t+1}, r_t|s_t, a_t)\left[r_t + \gamma V^*(s_{t+1})\right] = \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathbb{E}\left[r_t|s_t, a_t\right] \tag{88}$$

$$V^*(s_t) = \max_{a_t \in \mathcal{A}} Q^*(s_t, a_t) \tag{89}$$

$$\pi^*(s_t) \in \arg\max_{a_t \in \mathcal{A}} Q^*(s_t, a_t) \tag{90}$$

that are identical to standard reinforcement learning and are widely amenable to optimal solutions using the usual iterative frequency-based convergence [34, 35, 155–157].

For the general case with transition probabilities P not necessarily forming MDPs, approximate aggregation results reveal that they can also be modelled by MDPs [117]. They find several insightful results with the most relevant to our discussion being that assuming

$$|Q^*(h, a) - Q^*(\tilde{h}, a)| \leq \epsilon\ \forall\ \phi(h) = \phi(\tilde{h})\ \forall\ a, \tag{91}$$

the bounds for the optimal value functions for some $\epsilon, \gamma > 0$ can be proven to be

$$|Q^*(h, a) - Q^*(s, a)| \leq \frac{\epsilon}{1 - \gamma} \tag{92}$$

$$|V^*(h) - V^*(s)| \leq \frac{\epsilon}{1 - \gamma} \tag{93}$$

with the condition $\epsilon = 0$ if $\pi^*(h) = \pi^*(s)$. Determining the exact structure of this feature map ϕ however is in general a highly non-trivial process requiring significant calibration for each environment. They go further discussing extreme state aggregation wherein any process P can theoretically be represented using small finite-state MDPs. In particular they also present an argument explaining the surprisingly robust performance of MDP-based Q-learning methods when applied to non-MDP domains is very likely due to the generality of the law of large numbers [131] in

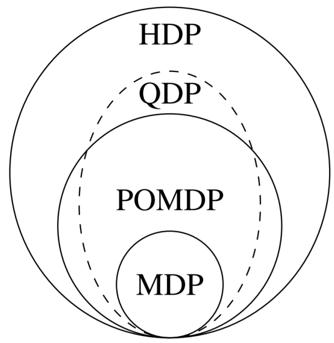

Figure 10: Q-Value Uniform Decision Processes (QDPs) in terms intersections with a broader class of more well-known processes. Adapted from [119].

the infinite sampling limit.

To convert HDPs into practical algorithms we must introduce additional definitions and constrains [119]. The state process p_h for an $\phi(h) = s$ is similarly

$$p_h(s_{t+1}, r_t|s_t, a_t) = \sum_{o_{t+1}:\phi(h_t a_t r_t o_{t+1})=s_{t+1}} P(o_{t+1}, r_t|h_t, a_t) \tag{94}$$

where for any $h \neq \tilde{h}$ clearly $p_h(s', r'|s, a) \neq p_{\tilde{h}}(s', r'|s, a)$ in general. This encodes the nature of non-MDPs where path-dependent histories are essential to future decisions and can incorporate non-stationary domains. If the state process is a MDP then p_h is independent of history and standard convergence result in Eqs. (86-90) follow.

To keep the state process history-dependent while making Q-learning independent of history, [119] introduce a subset of HDPs called Q-Value Uniform Decision Processes (QDPs) shown in Fig. 10 that incorporate all MDPs, have a non-empty intersection with all POMDPs, and are defined by a state-uniformity condition. The condition specifies that for any action, if any two histories $h, \tilde{h}$ map to the same state s, that is, $\phi(h) = \phi(\tilde{h}) = s$, then the optimal Q-value of the underlying HDP of these histories is identical $Q^*(h, a) = Q^*(\tilde{h}, a)$ [119]. This class is defined by Eq. (91) where $\epsilon = 0$ as $\pi^*(h) = \pi^*(s)$, and therefore by Eq. (92) the Q-values $Q^*(h, a) = Q^*(s, a)$ are state-uniform. QPDs then allow modelling of non-stationary domains, which accurately represents agent learning, and can be interpreted as history-independent Q-values of POMDPs.

The proof of QDP Q-learning convergence assumes that the state-process is ergodic where in all states are reachable under any policy from the current state after sufficiently many steps, meaning different unique histories can all reach the same state while the underlying HDP process P is still clearly non-ergodic [119]. This point is worth repeating, only the theoretical optimal values need to be equivalent at convergence that may never be reached, all intermediate values need not be equal at any fixed point or time. Also note for practical purposes the history of an agent is already stored within the experience replay buffer.

3.9 Agent Performance Evaluation

The current reward aggregation paradigm in all of reinforcement learning utilises additive dynamics when assessing the performance of the agent over an episode [34]. This means that the final cumulative reward is composed of the independent summation of rewards received at every time step $\Gamma^+(s_T, a_{T-1}|h_T) = \sum_{t=1}^{T} r_t$ where the "+" exponent indicates additive scheme. Improvements from tweaking existing algorithms, and comparison between models are generally measured using this method under two approaches. The first is generally a tabular presentation of averaged scores and standard deviations

$$\overline{\Gamma^+} \equiv \frac{1}{n}\sum_{i=1}^{n}\Gamma^+(s^i_T, a^i_{T-1}|h^i_T) \tag{95}$$

$$\sigma^+ \equiv \left(\frac{1}{n}\sum_{i=1}^{n}\left(\Gamma^+(s^i_T, a^i_{T-1}|h^i_T) - \overline{\Gamma^+}\right)^2\right)^{1/2} \tag{96}$$

across a range of environments using the final trained agent across n runs with each generating a unique history h^i_t. The second approach is similar to the first and usually graphically displays the average scores and standard deviations across evaluation episodes occurring at fixed intervals during agent training to better highlight performance over training time.

Before proceeding we clearly define what is meant by comparable performance. In practice this is done by conducting N evaluation episodes every fixed training interval using the identical parameters weights ϕ_r, θ_r for policies π_{ϕ_r} or μ_{ϕ_r} and action-values Q_{θ_r}. This process is then repeated for M trials where the agent is trained from scratch again likely generating different weights ϕ, θ each time due to random initialisation where the theoretical optimal weights are only reached in the infinite time limit guaranteeing identical parameters. Ultimately this process results in $n = NM$ unique histories at each evaluation interval whose performances are summarised using Eqs. (95-96). The reason for this design is that agent performance over multiple runs is notorious for being brittle and sensitive to hyperparameter selection [158, 159].

This universally accepted procedure is perfectly reasonable provided that the nature of the environment can be characterised by summing independent uncorrelated reward signals at each time step, and if the volatility in agent learning can be represented by a "standard" deviation. In general, it is unlikely that accurate performance measurement in all conceivable environments can be reduced to this way. Furthermore, quantifying the scale of volatility of a metric using standard deviation (STD) is only acceptable to alternative measures under strict requirements. As discussed in Section 3.5, a origin story for near-universal use of STD in all areas of science is presented in [14, 139] where they reveal it is only the preferred measure if the underlying data is also normally distributed. For all other situations, mean absolute deviation (MAD) is superior as it is far more asymptotically efficient, meaning that it is more robust to 'fat tails', and it does not require the variance of the true unknown underlying distribution to be finite. Therefore, it would be unwise to blindly assume the Gaussian relationship $\Gamma^+(s^i_T, a^i_{T-1}|h^i_T) \sim \mathcal{N}\left(\overline{\Gamma^+}, (\sigma^+)^2\right)$. Mean deviation MAD defined as

$$\mathrm{MAD}^+ \equiv \frac{1}{n}\sum_{i=1}^{n}\left|\Gamma^+(s^i_T, a^i_{T-1}|h^i_T) - \overline{\Gamma^+}\right| \tag{97}$$

should therefore be used as the default volatility measure instead. Noting in particular the bound $\sigma^+ \geq \mathrm{MAD}^+$. Other common names for MAD are mean absolute error (MAE), and average deviation (AVEDEV) used in Microsoft Excel. It is also functionally equivalent to the L_1-norm used throughout machine learning as in Eq. (43).

4 Non-Ergodicity in Reward Accumulation

The additive reward scheme is not in general appropriate for accurately modelling rewards in all environments. There exists a vast array of environments where the amount of risk-taking (leverage) can either be amplified or reduced in order to magnify or shrink potential rewards, where time order matters, and losses have an asymmetrically larger effect on performance compared to equally sized gains. These are domains where simply maximising the total reward is not the objective, rather the goal is to maximise the total reward while avoiding steep losses. Much of the applications for such environments are in non-erogdic domains where the time average is not equal to expectation or ensemble value as seen in Section 1.

To construct reward signals for agents in multiplicative dynamic environments there are three possibilities dependent on what information the agent receives. Firstly, as these situations involve change in valuations across time steps, an initial portfolio value $V_0 \neq 0$ must be specified as a baseline from which all future returns are measured against. The signal received by the agent at a time t can then be in terms of: absolute reward received r_t so the new valuation is $V_t = (V_{t-1} + r_t)/V_{t-1} = 1 + r_t/V_{t-1}$, directly as a growth percentage g_t so that $V_t = V_{t-1}(1 + g_t)$, or as a multiplier λ_t that gives $V_t = V_{t-1} \times \lambda_t$. In what follows, this work will focus solely on the first type of signal.

This formulation allows the valuation to crash towards to zero if substantial negative rewards are encountered. This allows us create a 'game over' or bankruptcy criterion where for example, if $V_\tau \leq V_{\min}$, the episode ends at $t = \tau$ with history $h_\tau = h_{\tau-1}a_{\tau-1}r_{\tau-1}s_\tau$. This is essentially the stop-loss from Section 1 but we refrain from using this terminology to keep the discussion general. To evaluate agent performance, we then measure the compounding changes in valuation over all time steps in a evaluation episode. For both dynamics we can write cumulative rewards up to a final step t as

$$\Gamma_t^+(s_t, a_{t-1}|h_t) = \Gamma_t^+ \equiv \sum_{k=1}^{t} r(\Gamma_{k-1}^+, s_k, a_k) = \Gamma_{t-1}^+ + r_t \tag{98}$$

$$1 + \Gamma_t^\times(s_t, a_{t-1}|h_t) = \left(1 + \Gamma_t^\times\right) \equiv \prod_{k=1}^{t} \frac{\Gamma_{k-1}^+ + r(\Gamma_{k-1}^+, s_k, a_k)}{\Gamma_{k-1}^+} = \left(1 + \Gamma_{t-1}^\times\right) \cdot \frac{\Gamma_{t-1}^+ + r_t}{\Gamma_{t-1}^+} \tag{99}$$

for $t > 0$ with the initial value $\Gamma_0^+ = V_0 > V_{\min}$. Note to ease computations we can also decompose the product $\Gamma_t^\times = \exp\left[\sum_{k=1}^t \ln|1 + r_k/\Gamma_{k-1}^+|\right] - 1$. This allows explicit encoding of the 'game over' condition at $t = \tau$, if $r_\tau \leq \left(V_{\min} - \Gamma_{\tau-1}^+\right)$ then the episode is over and we set compounding return to be $\Gamma_{t\geq\tau}^\times = 0$.

We strongly highlight that in general, the rewards $r(\Gamma_{k-1}^+, s_k, a_k)$ do not necessarily equal the standard rewards discussed in Section 3.1 so that $r(\Gamma_{k-1}^+, s_k, a_k) \neq r(s_k, a_k)$. This how we explicitly encode past performance and the concept of leverage into future rewards. Functionally, the tuple (Γ_{t-1}^+, s_t) act as a new effective 'state' of the system combining the previous cumulative reward with the current environment.

For Eq. (99) to be a valid representation we must have the crude relation where, in general, $|r(\Gamma_{k-1}^+, s_k, a_k)| \not\ll \Gamma_{k-1}^+$, meaning that the reward at an time $t = k$ should be somewhat comparable the cumulative reward. This is achieved when the agent has access to Γ_{k-1}^+ as another state of the environment and so the 'aggressiveness' of the next action to maximise the reward is dependent on the existing value. We define a successful policy in the limit $t \to \infty$ as one that exhibits $\Gamma_{t-1}^+ \to \infty$ where the agent is more comfortable in taking more risky actions that have greater potential in maximising the reward but are also correspondingly capable of causing larger absolute losses.

For a fixed successful policy we then have strict requirement

$$\lim_{t\to\infty} |r(\Gamma^{+}_{t-1}, s_k, a_t)| \gg |r(\Gamma^{+}_{t-1}, s_k, a_t)| \tag{100}$$

to ensure the reward appropriately scales with the cumulative value due partly to more aggressive actions taken for the same underlying state. We refer to this as the asymptotic reward scaling condition. Environments where this condition cannot be envisioned to occur are not multiplicative and should be treated as additive.

We can also rewrite the product in the form of compounding returns as

$$\left(1+\Gamma^{\times}_{t}\right) \equiv \prod_{k=1}^{t}\left(1+\frac{r(\Gamma^{+}_{k-1}, s_k, a_k)}{\Gamma^{+}_{k-1}}\right) = \left(1+\Gamma^{\times}_{t-1}\right)\cdot\left(1+\frac{r(\Gamma^{+}_{k-1}, s_k, a_k)}{\Gamma^{+}_{t-1}}\right) \tag{101}$$

$$\ln\left|1+\Gamma^{\times}_{t}\right| \equiv \sum_{k=1}^{t}\ln\left|1+\frac{r(\Gamma^{+}_{k-1}, s_k, a_k)}{\Gamma^{+}_{k-1}}\right| = \ln\left|1+\Gamma^{\times}_{t-1}\right| + \ln\left|1+\frac{r(\Gamma^{+}_{t-1}, s_t, a_t)}{\Gamma^{+}_{t-1}}\right| \tag{102}$$

Note without any loss of generality we can use $\Gamma^{\times}_{t} \leftarrow V_0\left(1+\Gamma^{\times}_{t}\right)$ when comparing overall performance. Regarding the two other types of reward signals we can also equivalently write $\left(1+\Gamma^{\times}_{t}\right) = \prod_{k=1}^{t}(1+g_t) = \prod_{k=1}^{t}\lambda_t$.

For any general environment the agents performance for episodes is ergodic if it satisfies the Birkhoff theorem

$$V_0(1+\bar{g}) \equiv \underbrace{\lim_{t\to\infty}\ \frac{1}{t}\int_0^t \mathrm{d}t\ \Gamma(s^j_t, a^j_{t-1}|h^j_t)}_{\text{Time average of }\Gamma} \stackrel{?}{=} \underbrace{\lim_{n\to\infty}\ \frac{1}{n}\sum_{i=1}^{n}\Gamma(s^i_t, a^i_{t-1}|h^i_t)}_{\text{Expectation value of }\Gamma} \equiv \mathbb{E}\left[\Gamma\right] \tag{103}$$

where the equality implies the systems is ergodic [9, 160, 161]. The left-hand side states that for any random history h^j_t we calculate the cumulative reward in the infinite time limit or till the episode ends and take the average over time. The right-hand side represents the ensemble average across infinite histories at any points in time. To test whether this holds in general, take $h^j_t = h_{t-1}a_{t-1}r_{t-1}s_t$ with $h^j_T = h^j_t a_t r_t s_T$ meaning the game ends at the next step. Simplifying notion $\Gamma(s^j_t, a^j_{t-1}|h^j_t) = \Gamma(h^j_t)$, for both the additive and multiplicative dynamics we have

$$\frac{1}{T}\left(\Gamma^{+}(h^j_t) + r_T\right) \stackrel{?}{=} \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\Gamma^{+}(h^i_t) \tag{104}$$

$$\frac{1}{T}\left(\left(1+\Gamma^{\times}(h^j_t)\right)\times\frac{\Gamma^{+}(h^j_t)+r_T}{\Gamma^{+}(h^j_t)}\right) \stackrel{?}{=} \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\left(1+\Gamma^{\times}(h^i_t)\right) \tag{105}$$

As we are free to select any h^j_t arbitrarily, by contradiction, it is unlikely that either equality holds in general for any non-trivial environment. Hence under both dynamics, agent learning is very likely a non-ergodic process where the time-average performance of any single path should not be assumed to exactly equal the average performance of all paths at any time. The difference is again the volatility tax $\nu = \bar{g} - \left(\frac{\mathbb{E}[\Gamma]}{V_0} - 1\right)$.

Returning to the discussion in Section 3.9 regarding how in practice $n = NM$ where M is the amount total training trials and N is the amount of evaluations episodes per trial. We can see that as $n \to \infty$, if M is held constant the ergodic case is far more likely to be approached compared to if N is held constant. This is again due to the fact that the agent policy ϕ and action value θ parameters are shared for each M. For all theoretical results in this work, whenever such limits are considered we assume $N = 1$ and $M \to \infty$ as this is more in line with reality.

Regarding evaluation on exactly the same environments via $\overline{\Gamma}$ and $\mathrm{MAD}(\Gamma)$ we can also hypothesise several results comparing agent training on both the dynamics. We expect the following three occurrences

$$\mathrm{MAD}(\Gamma^{+}) \geq \mathrm{MAD}(\Gamma^{\times}) \tag{106}$$

$$\mathrm{MAD}_u(\Gamma^{+}) \leq \mathrm{MAD}_u(\Gamma^{\times}) \tag{107}$$

$$\mathrm{MAD}_d(\Gamma^{+}) \geq \mathrm{MAD}_d(\Gamma^{\times}) \tag{108}$$

where the up-side and down-side MADs are calculated considering only the $\Gamma_j \geq \overline{\Gamma}$ and $\Gamma_j \leq \overline{\Gamma}$ respectively. This is because multiplicative dynamics is more suitable for steady growth expect there to be less overall volatility while also anticipating there exists more positive, and fewer negative deviations from the mean. Overall, this is consistent with stable increasing learning while avoiding of steep losses.

4.1 Ergodic Special Case

There is one scenario where the system can become ergodic if the agent has learned the optimal deterministic policy $\mu(s) \to \mu^*(s)$ at some time $t \to t^*$. During on- or off-policy model-free learning for all times $t \geq t^*$, Then the agent will take identical actions when presented identical states. If the game has no natural end and the agent and can continue for $t \to \infty$, the agent states and actions will convergence to a optimal sequence. Under this special case, for some critical time $t_c \gg t^*$ we can say that

$$\lim_{T\to\infty} \frac{1}{T - t_c} \int_{t_c}^{T} \mathrm{d}t \; \Gamma(h_t^j) \approx \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \Gamma(h_{t\geq t_c}^i) \tag{109}$$

where the approximation is very loose and assumes that majority of the cumulative rewards are obtained from times $t > t_c$. Another way this can be seen is through the situation where longer-term rewards are more significant for both dynamics

$$\Gamma^{+}(h_T^j) = \Gamma^{+}(h_{t<t_c}^j) + \Gamma^{+}(h_{t\geq t_c}^j) \approx \Gamma^{+}(h_{t\geq t_c}^j) \tag{110}$$

$$1 + \Gamma^{\times}(h_T^j) = \Big(1 + \Gamma^{\times}(h_{t<t_c}^j)\Big) \cdot \Big(1 + \Gamma^{\times}(h_{t\geq t_c}^j)\Big) \approx 1 + \Gamma^{\times}(h_{t\geq t_c}^j) \tag{111}$$

for all histories. Given the purpose of multiplicative dynamics is to penalise losses significantly we can also likely assume that at least $t_c^{\times} \geq t_c^{+}$ in general. This then allows us to assume ergodicity amongst all histories wherein we can crudely state that $\lim\limits_{t\to\infty} \frac{1}{t}\Gamma(h_t^j) \approx \lim\limits_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n}\Gamma(h_T^i)$ as both limits are functionally equivalent. Therefore we obtain in the limit $\nu \to 0$ for any randomly selected history h_j.

This situation could be considered an implicit assumption made in all existing literature results since it reasonable under stationary MDP convergence proofs. The current formulation of reinforcement learning using additive reward signals in MDP environments can therefore be described as ergodic if we assume the agent has learned and is operating with an optimal deterministic policy at all times. For non-MDP processes, stochastic policies, and sub-optimal policies, all bets are off, and the system should be considered non-ergodic.

5 Q-Learning with Multiplicative Dynamics

Recall standard Q-learning the agent performs an action-value update to the (s, a)-estimate using the rule

$$Q_{t+1}(s,a) \leftarrow (1-\alpha_t(s,a))\, Q_t(s,a) + \alpha_t(s,a) \left(r_t(s,a) + \gamma \max_{a'} Q_{t+1}(s', a) \right) \tag{112}$$

where $\alpha_t(s, a) > 0$ is the time- and (s, a)-dependent learning rate under a ϵ-greedy policy using Bellman's principle of optimality [129, 130]. Essentially, the second term performs a small adjustment to the Q-value in the direction of optimal value. The convergence of this approach to unique optimal fixed point is very well-known result [34, 35, 72, 93, 155–157].

For additive dynamics we can adjust $r_t(s, a) \leftarrow r_t(s, a) + \Gamma^+_{t-1}$ where Γ^+_{t-1} is a known constant and all the results in [117, 119] remain unchanged. This is valid as the next reward to be maximised is treated independent from the previous cumulative rewards.

5.1 Model-Free Return Maximisation

To incorporate multiplicative dynamics into this formulation we must resort to using the QDPs discussed in Section 3.8 in order to retain the histories of rewards while keeping the evaluation of Q-values dependent only on the (s, a)-pair in Markovian fashion.

Recalling the definitions in Section 3.7 that the rewards are now also dependent on the cumulative previous additive reward where $r(\Gamma^+_{t-1}, s_t, a_t) \leftarrow r(s_t, a_t)$ in order to internally control leverage by factoring the existing value along with the usual environment. This way the agent is able to keep track of final outcome of all past actions summarised with a single scalar. For this to be occur the asymptotic reward scaling condition in Eq. (100) must universally hold. As Γ^+_{t-1} is always known at any time t we are free to define a new environment state $\xi_t = s_t \cup \Gamma^+_{t-1}$ with the fixed cardinality relation $|\xi_t| = |s_t| + 1$ and formally $\xi_t \in \Xi : \mathcal{S} \times \mathcal{R}$.

The agent history is then $h_t \equiv \xi_1 a_1 r_1 ... \xi_{t-1} a_{t-1} r_{t-1} \xi_t$ where $h_t \in \mathcal{H}_t : (\Xi \times \mathcal{R} \times \mathcal{A})^{t-1} \times \Xi$ remains unchanged and the trajectory develops as $\xi_1 \to a_1 \to r_1 \xi_2 \to a_2 \to r_2 \xi_3 \to ... \to r_{t-1}\xi_t$. Practically, the only change required is the optimisation procedure also uses the current additive reward as an input which treated as an additional state. In what follows we will always utilise the notion $r_t = r(h_t, a_t)$ for convenience, it should never be confused for $r_t = r(s_t, a_t)$ and it will be made clear where the latter applies.

To reformulate Q-learning to be compatible with multiplicative dynamics and prove convergence we must derive results from the most basic levels again. The value functions are then defined as $V^\pi(h_t) \equiv \mathbb{E}\left[1 + R^\times_t | h_t; \pi\right]$ and $Q_\pi(h_t, a) \equiv \mathbb{E}\left[1 + R^\times_t | h_t, a; \pi\right]$, the optimal values are $V^*(h_t) = \max_\pi V_\pi(h_t)$ and $Q^*(h_t, a_t) = \max_\pi Q_\pi(h_t, a_t)$ where $\pi^* \in \arg\max_\pi V_\pi(\epsilon)$. In this case the discounted future compounding rewards are

$$\begin{aligned}
1 + R^\times_t &\equiv \left(1 + \Gamma^\times_{t-1}\right) \cdot \frac{\Gamma^+_{t-1} + r_t}{\Gamma^+_{t-1}} \cdot \frac{(\Gamma^+_{t-1} + r_t) + \gamma r_{t+1}}{\Gamma^+_{t-1} + r_t} \cdot \frac{(\Gamma^+_{t-1} + r_t + \gamma r_{t+1}) + \gamma^2 r_{t+2}}{\Gamma^+_{t-1} + r_t + \gamma r_{t+1}} \cdot ... \\
&= \left(1 + \Gamma^\times_t\right) \cdot \frac{\Gamma^+_t + \gamma r_{t+1}}{\Gamma^+_t} \cdot \frac{(\Gamma^+_t + \gamma r_{t+1}) + \gamma^2 r_{t+2}}{\Gamma^+_t + \gamma r_{t+1}} \cdot ... \\
&= \left(1 + \Gamma^\times_t\right) \cdot \prod_{k=1}^{\infty} \frac{\Gamma^+_{t-1+k} + \gamma^k r_{t+k}}{\Gamma^+_{t-1+k}}
\end{aligned} \tag{113}$$

where $\Gamma^+_{t-1+k} = \Gamma^+_{t-1} + \sum_{\lambda=0}^{k-1} \gamma^\lambda r_{t+\lambda}$ is the discounted additive reward up to time step $t = k - 1$, the current return

$\Gamma_t^{\times}$ is completely known, and in Section 5.2 we show that the initial value $\Gamma_0^{+} = V_0 \geq \gamma > V_{\min}$ is required. We can then rewrite this more clearly as

$$
\begin{aligned}
1 + R_t^{\times} &\equiv \left(1 + \Gamma_t^{\times}\right) \cdot \left(1 + \frac{\gamma r_{t+1}}{\Gamma_t^{+}}\right) \cdot \left(1 + \frac{\gamma^2 r_{t+2}}{\Gamma_{t+1}^{+}}\right) \cdot \left(1 + \frac{\gamma^3 r_{t+3}}{\Gamma_{t+2}^{+}}\right) \cdot \ldots \\
&= \left(1 + \Gamma_t^{\times}\right) \cdot \prod_{k=1}^{\infty} \left(1 + \frac{\gamma^k r_{t+k}}{\Gamma_{t-1+k}^{+}}\right) \qquad (114)
\end{aligned}
$$

For the other two reward signals, we can express this equivalently as $\left(1 + R_t^{\times}\right) = (1 + g_t) \prod_{k=1}^{t} (1 + \gamma^k g_{t+k})$ or $\left(1 + R_t^{\times}\right) = \lambda_t \prod_{k=1}^{t} \gamma^k \lambda_{t+k}$. This represents the general compounding return that the agent seeks to maximise over time through learning an optimal policy π^*. A more practical way to characterise this is

$$
\begin{aligned}
1 + R_t^{\times} &\equiv \left(1 + \Gamma_t^{\times}\right) \cdot \exp\left[\sum_{k=1}^{\infty} \ln \left| 1 + \frac{\gamma^k r_{t+k}}{\Gamma_{t-1+k}^{+}} \right| \right] \qquad (115) \\
&= \left(1 + \Gamma_t^{\times}\right) \cdot e^{\sum_{k=1}^{\infty} \psi_{t+k}} \qquad (116)
\end{aligned}
$$

which represents the objective as (exponentially) continuously compounding growth with the logarithmic return in each time period being ψ_t. The total future return is then the summation of all these compounding rates $\Psi_t = \sum_{k=1}^{\infty} \psi_{t+k}$. Furthermore, given the logarithm and exponential are monotonically increasing functions, the optimisation process can be greatly simplified as we can treat the agent decision at every time step individually. This way maximisation of Eq. (115) involves simply maximising each ψ_t at each time t separately for all time. In terms of optimal policy we explicitly have $\pi^* \in \arg\max_{\pi} \Psi_t$.

We can then directly write the proportionality

$$
\begin{aligned}
1 + R_t^{\times} &\propto \left(1 + \frac{\gamma r_{t+1}}{\Gamma_t^{+}}\right) + \left(1 + \frac{\gamma^2 r_{t+2}}{\Gamma_{t+1}^{+}}\right) + \left(1 + \frac{\gamma^3 r_{t+3}}{\Gamma_{t+2}^{+}}\right) + \ldots \\
&\propto \sum_{k=1}^{\infty} \left(1 + \frac{\gamma^k r(h_{t+k}, a_{t+k})}{\Gamma_{t-1+k}^{+}}\right) \qquad (117)
\end{aligned}
$$

where we retain the addition of unity at each step to highlight the reward is a return relative to an existing value and therefore is bounded $\gamma^k r_{t+k} \geq \left(V_{\min} - \Gamma_{t-1+k}^{+}\right)$ with equality implying $(1 + R_t^{\times}) = 0$ 'game over' at time $t + k$. We have therefore reframed the desired Eq. (24) in the language of reinforcement learning.

Observe that by doing this we have left multiplicative dynamics and returned back to the realm of additive dynamics given we now maximise a summation [34]. However, we continue to refer to this situation as multiplicative due to three reasons: 1. The rewards at each time step are coupled to the cumulative prior rewards forming a relative return and so performance is not measured on absolute terms independently, rather it is designed to maximise rate of growth, 2. The state $\xi_t = s_t \cup \Gamma_{t-1}^{+}$ explicitly incorporates the existing cumulative value as a factor determining the reward, and 3. The objective is still ultimately a compounding rate of return that is proportional to a summation.

Then by using an iterative process at each time step we can numerically approximate using the Bellman equation in Eq. (28) all future returns at any time step to be

$$
1 + \frac{Q_\pi(h_{t+k}, a_{t+k})}{\Gamma_{t-1+k}^{+}} \leftarrow 1 + \frac{\mathbb{E}_{a_{t+k+1} \sim \pi}\left[r\left(h_{t+k}, a_{t+k}\right) + \gamma Q_\pi\left(h_{t+k+1}, a_{t+k+1}\right)\right]}{\Gamma_{t-1+k}^{+}} \qquad (118)
$$

where the value functions are not equivalent to the additive case, however since they are an artificial construction, we are free to define them as we choose. Notice that we can also write the right-hand side as

$$\frac{\Gamma^+_{t+k} + \gamma V_\pi(h_{t+1+k})}{\Gamma^+_{t-1+k}} = \left(1 + \frac{r_{t+k}}{\Gamma^+_{t-1+k}}\right) + \frac{\gamma V_\pi(h_{t+1+k})}{\Gamma^+_{t-1+k}} = 1 + \frac{r_{t+k} + \gamma V_\pi(h_{t+1+k})}{\Gamma^+_{t-1+k}} \tag{119}$$

where $\Gamma^+_{t+k} = \Gamma^+_{t-1+k} + r_{t+k}$. This can be interpreted as the sum of both the one-holding period return and the forecasted discounted perpetuity ratio for all later periods. The 'game over' criterion is also built-in where if $\Gamma^+_t \leq V_{\min}$ the episode is terminated. The pseudo-Bellman relations in Eqs. (80-84) can then be written as

$$1 + \frac{Q_\pi(h_t, a_t)}{\Gamma^+_{t-1}} = \sum_{o_{t+1}, r_t} P(o_{t+1}, r_t | h_t, a_t) \left[1 + \frac{r_t + \gamma V_\pi(h_{t+1})}{\Gamma^+_{t-1}}\right] \tag{120}$$

$$V_\pi(h_t) = Q_\pi(h_t, \pi(h_t)) \tag{121}$$

$$1 + \frac{Q^*(h_t, a_t)}{\Gamma^+_{t-1}} = \sum_{o_{t+1}, r_t} P(o_{t+1}, r_t | h_t, a_t) \left[1 + \frac{r_t + \gamma V^*(h_{t+1})}{\Gamma^+_{t-1}}\right] \tag{122}$$

$$V^*(h_t) = \max_{a_t \in \mathcal{A}} Q^*(h_t, a_t) \tag{123}$$

$$\pi^*(h_t) \in \arg\max_{a_t \in \mathcal{A}} Q^*(h_t, a_t) \tag{124}$$

To make this amenable to Q-learning we change notation not to confuse iterative updates of a Q-value estimate from $Q_{t+1}(h, a) \leftarrow Q_t(h, a)$ for any history-action pair (h, a) defined as as $(h_i, a_i) \leftarrow (h_t, a_t)$ with known $\Gamma^+_{i-1} \leftarrow \Gamma^+_{t-1}$. Standard Q-learning in Eq. (112) is then modified to

$$\left(1 + \frac{Q_{t+1}(h, a)}{\Gamma^+_{i-1}}\right) \leftarrow (1 - \alpha_t(h, a)) \left(1 + \frac{Q_t(h, a)}{\Gamma^+_{i-1}}\right) + \alpha_t(h, a) \left(1 + \frac{1}{\Gamma^+_{i-1}} \left(r_t(h, a) + \gamma \max_{a'} Q_{t+1}(h', a)\right)\right) \tag{125}$$

which forms a general non-stationary HDP using a ϵ-greedy policy. This can also be crudely expressed as

$$\psi_{t+1} \leftarrow \psi_t + \frac{\alpha_t(h, a)}{\Gamma^+_{i-1}} \left(r_t + \gamma \max_{a'} Q_{t+1}(h', a) - Q_t(h, a)\right) \tag{126}$$

where the second term on the right is the improvement in the exponential compounding rate in one time step from a single learning update. To make this tractable we convert the standard results in [119] to multiplicative rewards and use their following assumptions:

1. The state-process is ergodic meaning that all states are reachable under any policy from the current state after sufficiently many steps. Strictly speaking this is false due to the 'game over' criterion, however if $\Gamma^+_{i-1} + r_i \gg V_{\min}$ for any conceivable $r_i < 0$, the agent can afford to go any state for some $\delta t > 0$ where $\delta t \to 0$ as $\Gamma^+_{i-1+\delta t} \to V_{\min}$. Therefore if we initialise the value $\Gamma^+_0 = V_0 \gg V_{\min}$ this assumption may be considered reasonable. Another way to state this is $|r_i| \ll V_0 \; \forall \; r_i < 0$, that is, the change in maximum absolute downwards change in valuation from a single step is small relative to the initial value. No such constraint is required $\forall \; r_i > 0$. Practically this can be enforced if each step represents a very small change in time and so there is a limit on the maximum change in cumulative additive value that can reasonably occur.

2. The rewards are bounded $r \in [r_{\min}, r_{\max}]$ which is standard to ensure stable convergence. In our case the lower bound also varies with time where $r_{i,\min} > V_{\min} - \Gamma^+_i$ otherwise the episode ends.

3. The state-process is a QDP where $Q^*(h,a) = Q^*(\tilde{h},a)$ for some feature map $\phi(h) = \phi(\tilde{h}) = \xi$, and therefore $Q^*(h,a) = Q^*(\xi,a)$. We do not explicitly specify the feature map ϕ but assume one exists. This is a reasonable assumption since the scope of this work encapsulates only keeping track of the reward history while making no use of prior states and actions for future decisions. Importantly, this condition allows $Q(h_i,a_i) \neq Q(\xi_i,a_i)$ for all intermediate action-values and so is very flexible.

To prove the convergence of Eq. (125) by repeated updates we re-purpose standard methods [35, 93, 119, 155–157]. Without loss of generality, we can reparameterise the $Q_t(\xi,a) \leftarrow \left(1 + \frac{Q_t(\xi,a)}{\Gamma^+_{i-1}}\right)$ as it is a artificially constructed value that we seek to maximise. The update rule is then rewritten as

$$Q_{t+1}(\xi,a) = (1 - \alpha_t(\xi,a))\, Q_t(\xi,a) + \alpha_t(\xi,a) \left(T^\pi_{h_t} Q\right)(\xi,a) \tag{127}$$

where we define $T^\pi_{h_t}$ to be the Bellman history-based operator for a non-stationary policy π that generally incorporates both the decision process history h_i and the history of all learning step sizes $\alpha_t(\xi,a)$ with the condition $\alpha_t(\xi_i,a_i) = 0 \;\forall\; (\xi,a) \neq (\xi_i,a_i)$. Application of this operator to a (ξ,a)-pair yields

$$\begin{aligned}
\left(T^\pi_{h_t} Q\right)(\xi,a) = 1 + \frac{Q_{\pi,t+1}(\xi_i,a_i)}{\Gamma_{i-1}} &= \sum_{\xi_{i+1},r_i} p_{h_i}(\xi_{i+1},r_i|s_i,a_i) \left[1 + \frac{r_i + \gamma V_\pi(\xi_{i+1})}{\Gamma^+_{i-1}}\right] && (128)\\
&= \sum_{\xi_{i+1}} p_{h_i}(\xi_{i+1}|s_i,a_i) \mathbb{E}\left[1 + \frac{r_i + \gamma V_\pi(\xi_{i+1})}{\Gamma^+_{i-1}} \,\middle|\, \xi_i, a_i\right] && (129)\\
&= \mathbb{E}_{p_{h_i}}\left[1 + \frac{1}{\Gamma^+_{i-1}}\left(r_i + \gamma \max_{a'} Q_t(\xi_{i+1},a)\right) \,\middle|\, T^\pi_t\right] && (130)
\end{aligned}$$

where in the final line we assume a ϵ-greedy policy pi and T^π_t is a complete history of the algorithm including h_i and all the steps $(\alpha_k)_{k\leq t}$. In Section 5.2 we prove under a strict criteria the convergence of Eq. (127) where $\left(T^\pi_{h_t} Q\right)(\xi,a) \propto Q_{t+1}(\xi,a) \to Q^*(\xi,a)$ with probability 1 (w.p.1.) as infinite updates $t \to \infty$ are applied.

5.2 Proof of Convergence and Uniqueness

The convergence and uniqueness of multiplicative Q-learning is demonstrated by first converting Eq. (127) to a standard form [157]. Consider a stochastic process $(\alpha_t(\xi,a), \Delta_t, F_t), t > 0$ for $t \in \mathbb{Z}^+$ where $\alpha_t(\xi,a), \Delta_t, F_t : s,a \to \mathbb{R}$. Let $T^\pi_{h_t}$ be a sequence of increasing σ-fields such that $\alpha_0(\xi,a)$ and Δ_0 are $T^\pi_{h_0}$-measurable and $\alpha_t(\xi,a)$, Δ_t and F_{t-1} are $T^\pi_{h_t}$-measurable $\forall\; t$. Begin by defining

$$\Delta_t \equiv Q_{\pi,t+1}(\xi,a) - Q^*_t(\xi,a) \tag{131}$$

$$F_t \equiv \left(T^\pi_{h_t} Q\right)(\xi,a) - Q^*_t(\xi,a) \tag{132}$$

$$\Delta_{t+1} \equiv (1 - \alpha_t(\xi,a))\, \Delta_t(\xi,a) + \alpha_t(\xi,a) F_t \tag{133}$$

A sequence $(Q_t(\xi,a))_{t\in\mathbb{N}}$ then generated by the iteration of Eq. (127) represented using Eqs. (131-133) converges to the optimal action-value under multiplicative dynamics $Q^*(\xi,a) = Q^*(h,a)$ of a QDP state-process w.p.1. if the following conditions are satisfied:

1. The agent state space $\xi \in \Xi : \mathcal{S} \times \mathcal{R}$ is finite.

2. The discount factor is bounded $\gamma \in [0,1)$ for all steps.

3. Infinite learning updates are possible where $\Delta_{t+1} \to \Delta_\infty$ in the limit $t \to \infty$.

4. The Robbins-Monro (RM) conditions

$$\sum_{t=0}^{\infty} \alpha_t(\xi, a) = \infty, \text{ and } \sum_{t=0}^{\infty} \alpha_t^2(\xi, a) < \infty \tag{134}$$

for learning rates are satisfied which requires $\alpha_t(\xi, a) \in (0, 1]$ and $\alpha_t(\xi, a) = 0 \ \forall \ (\xi, a) \neq (\xi_t, a_t)$ [162, 163]. This also requires the state-process to be ergodic and the step size asymptotically decreases to converge to a fixed point though never ceases $\alpha_t(\xi, a) \neq 0$ learning in order to avoid local maxima.

5. There exists a $Q^*(\xi, a)$ such that

$$|| \left(T^\pi_{h_t} Q\right)(\xi, a) - Q^*(\xi, a)||_\infty \leq \gamma ||Q_{\pi,t}(\xi, a) - Q^*(\xi, a)||_\infty \ \forall \ t \tag{135}$$

to prove that $Q^*(\xi, a)$ is a unique fixed point of the contraction T^π_h and converges to the optimal solution in the limit $t \to \infty$. This condition follows from the usual Banach's fixed-point theorem applied to MDPs.

6. The F_t term satisfies in expectation

$$||\mathbb{E}_{p_{h_i}}[F_t | T^\pi_t]||_\infty \leq \kappa ||\Delta_t||_\infty + c_t \tag{136}$$

where $\kappa \in [0, 1]$ and $c_t \to 0$ w.p.1. as $t \to \infty$.

7. The noise is bounded if the conditional variance of F_t satisfies

$$\text{Var}(F_t | T^\pi_t) \leq \kappa \left(1 + ||\Delta_t||_\infty\right)^2 \tag{137}$$

where κ is a constant.

The first condition is satisfied as it is the primary reason for converting a general HDP to a tractable QDP where $\mathcal{S}$ is a small finite subset of $\mathcal{S} \in \mathcal{O}$. The third condition condition will be assumed valid in line with all literature derivations despite there existing clear episode termination criteria.

In practice RM conditions are violated as constant iterative step sizes $\alpha_t(\xi, a) = \alpha \ \forall \ t$ are usually used. This simplification works well as π_ϕ is non-stationary and that when using a mini-batch for learning, the policy parameters converge $\phi \to \phi^*$ as $t \to \infty$ with the variance of convergence proportional to α^2 [93].

To prove the fifth condition we must first show that $T^\pi_{h_t}$ is a max-norm contraction and that the fixed point equation $\left(T^\pi_{h_t} Q\right)(\xi, a) \propto Q^\pi_{t+1}(\xi, a)$ has a unique solution for L-Lipschitzian

$$|| \left(T^\pi_{h_t} Q\right)(\xi, a) - \left(T^\pi_{h_t} Q'\right)(\xi, a)|| \leq L ||Q_{\pi,t}(\xi, a) - Q'_{\pi,t}(\xi, a)|| \tag{138}$$

where L is called the contraction error and $T^\pi_{h_t}$ is called a non-expansion if $L \leq 1$ or a contraction if $L < 1$. This is done by proving Banach's fixed point theorem for multiplicative dynamics where $\left(T^\pi_{h_t} Q\right)(\xi, a) \propto Q_{\pi,t+1}(\xi, a) \to Q^*(\xi, a)$ at a geometric rate as $t \to \infty$ expressed as

$$||Q_{\pi,n}(\xi, a) - Q_\pi(\xi, a)|| \leq \gamma^n ||Q_{\pi,0}(\xi, a) - Q_\pi(\xi, a)|| \tag{139}$$

which then poses two additional questions: 1. Whether the $Q(\xi, a)$ pair is a fixed point of T_h^π in action-value space, and 2. Whether the Q-values $Q(\xi, a) = Q'(\xi, a)$ implying the (ξ, a)-pair can be uniquely represented.

For a fixed history h_t, the operator $T_{h_t}^\pi$ for a ϵ-greedy policy is shown to be a contraction mapping by

$$
\begin{aligned}
&|| \, (T_{h_t}^\pi Q)(\xi, a) - \left(T_{h_t}^\pi Q'\right)(\xi, a)||_\infty \\
&\qquad = \max_{\xi, a} \left| \mathbb{E}_{p_{h_i}} \left[1 + \frac{1}{\Gamma_{i-1}^+} \left(r_i + \gamma \max_{a'} Q_t(\xi', a) \right) \middle| T_t^\pi \right] - \mathbb{E}_{p_{h_i}} \left[1 + \frac{1}{\Gamma_{i-1}^+} \left(r_i + \gamma \max_{a'} Q_t'(\xi', a) \right) \middle| T_t^\pi \right] \right| \\
&\qquad \overset{(a)}{=} \frac{\gamma}{\Gamma_{i-1}^+} \max_{\xi, a} \left| \mathbb{E}_{p_{h_i}} \left[\max_{a'} Q_t(\xi', a) \middle| s, a \right] - \mathbb{E}_{p_{h_i}} \left[\max_{a'} Q_t'(\xi', a) \middle| s, a \right] \right| \\
&\qquad \overset{(b)}{\leq} \frac{\gamma}{\Gamma_{i-1}^+} \max_{\xi, a} \max_{\xi'} \left| \max_{a'} Q_t(\xi', a) - \max_{a'} Q_t'(\xi', a) \right| \\
&\qquad \overset{(c)}{\leq} \frac{\gamma}{\Gamma_{i-1}^+} \max_{\xi, a} \, \left| Q_t(\xi, a) - Q_t'(\xi, a) \right| \\
&\qquad \overset{(d)}{=} \frac{\gamma}{\Gamma_{i-1}^+} ||Q_t(\xi, a) - Q_t'(\xi, a)||_\infty
\end{aligned} \tag{140}
$$

in which (a) for a fixed history h_i has the same $\Gamma_{i-1}^+ \geq \Gamma_0^+ > V_{\min}$ and the QDP assumption assures the expectation depends only on the (ξ, a)-pair, (b) establishes a upper bound by removing the expectation, (c) increases the upper bound by no longer demanding the smallest optimal maximal difference, and (d) is the usual max-norm contraction $||x||_\infty = \sup_{x \in \chi} (f(x)) = \max_{x \in \chi} |x|$ by definition.

For Eq. (140) to be valid, clearly we must have $\Gamma_{i-1}^+ > V_{\min} \geq \gamma \;\forall\; i$ to ensure $\gamma < \Gamma_{i-1}^+$ which enforces the bound for multiplicative dynamics. This is clearly an artificial construction on the dimensionless reward scheme we must use for convergence. The impact of this is subtlety will be highly dependent on the domain. One interpretation for this requirement is that the reward scheme could be defined in units of the discount factor where the zero point is γ. Regardless, this allows us to say

$$
|| \, (T_{h_t}^\pi Q)\,(\xi, a) - \left(T_{h_t}^\pi Q'\right)(\xi, a)||_\infty \leq \gamma ||Q_{\pi,t}(\xi, a) - Q'_{\pi,t}(\xi, a)||_\infty \tag{141}
$$

which is now in the desired form in Eq. (138). To prove that this represented unique fixed points we follow the discussion in [93] where we can show for subsequent update steps $k > 0$ that

$$
\begin{aligned}
||Q_{t+k}^\pi(\xi, a) - Q_t^\pi(\xi, a)|| &= || \left(T_{h_{t-1+k}}^\pi Q\right)(\xi, a) - \left(T_{h_{t-1}}^\pi Q\right)(\xi, a)|| \\
&\leq \gamma ||Q_{\pi,t-1+k}(\xi, a) - Q_{\pi,t-1}(\xi, a)||_\infty = \gamma || \left(T_{h_{t-2+k}}^\pi Q\right)(\xi, a) - \left(T_{h_{t-2}}^\pi Q\right)(\xi, a)|| \\
&\;\;\vdots \\
&\leq \gamma^t ||Q_{\pi,k}(\xi, a) - Q_{\pi,0}(\xi, a)||_\infty
\end{aligned} \tag{142}
$$

as shown in Eq. (139). Next use the triangle inequality and recall $\sum_{\lambda=0}^\infty \gamma^\lambda = (1-\gamma)^{-1}$ to get

$$
\begin{aligned}
||Q_{\pi,k}(\xi, a) - Q_{\pi,0}(\xi, a)|| &\leq \sum_{j=1}^k ||Q_{\pi,j}(\xi, a) - Q_{\pi,j-1}(\xi, a)||_\infty \\
&\leq \sum_{j=1}^k \gamma^{j-1} ||Q_{\pi,1}(\xi, a) - Q_{\pi,0}(\xi, a)||_\infty
\end{aligned}
$$

$$\leq \frac{1}{1-\gamma}||Q_{\pi,1}(\xi,a) - Q_{\pi,0}(\xi,a)||_\infty \tag{143}$$

Therefore

$$||Q_{\pi,t+k}(\xi,a) - Q_{\pi,t}(\xi,a)|| \leq \frac{\gamma^t}{1-\gamma}||Q_{\pi,1}(\xi,a) - Q_{\pi,0}(\xi,a)||_\infty \tag{144}$$

is of the form of a standard Cauchy sequence in the limit of infinite learning updates $t \to \infty$. Due to exponential suppression by $\gamma \in [0,1)$ this leads to for $k > 0$ the well-known convergence

$$\lim_{t\to\infty} ||Q_{\pi,t+k}(\xi,a) - Q_{\pi,t}(\xi,a)|| = 0 \tag{145}$$

allowing us to state $Q_{\pi,t+k}(\xi,a) \to Q_\pi(\xi,a)\ \forall\ k > 0$ that is the optimal value. Next recall the definition $(T^\pi_{h_t}Q)(\xi_i,a_i) \propto Q_{\pi,t+1}(\xi_i,a_i)$ and the take the limit on both sides

$$\lim_{t\to\infty} ||(T^\pi_{h_t}Q)(\xi_i,a_i)|| = \lim_{t\to\infty}\left|\left|1 + \frac{Q_{\pi,t+1}(\xi_i,a_i)}{\Gamma_{i-1}}\right|\right| = \lim_{t\to\infty}\left|\left|1 + \frac{Q_{\pi,t}(\xi_i,a_i)}{\Gamma_{i-1}}\right|\right| = 1 + \frac{Q_\pi(\xi_i,a_i)}{\Gamma_{i-1}} \tag{146}$$

to see that $Q_\pi(\xi,a)$ must be a fixed point of the L-Lipschitzian continuous contraction $T^\pi_{h_t}$. Finally, regarding the uniqueness of Q-values for any (ξ_i,a_i)-pair we can solve the equation that

$$\begin{aligned}||(T^\pi_h Q)(\xi_i,a_i) - (T^\pi_h Q')(\xi_i,a_i)|| &= \frac{1}{\Gamma_{i-1}}\left|\left|Q_\pi(\xi_i,a_i) - Q'_\pi(\xi_i,a_i)\right|\right| \\ &\leq \frac{\gamma}{\Gamma_{i-1}}||Q_\pi(\xi_i,a_i) - Q'_\pi(\xi_i,a_i)|| \\ (1-\gamma)\,||Q_\pi(\xi_i,a_i) - Q'_\pi(\xi_i,a_i)|| \leq 0 &\end{aligned} \tag{147}$$

so either $||Q_\pi(\xi,a) - Q'_\pi(\xi,a)|| = 0$ or $Q_\pi(\xi,a) = Q'_\pi(\xi,a)$ (unique). To show that it is the latter case we once again construct the bound

$$\begin{aligned}||Q_{\pi,t}(\xi,a) - Q'_\pi(\xi,a)|| &= ||\left(T^\pi_{h_{t-1}}Q\right)(\xi,a) - \left(T^\pi_{h_{t-1}}Q'\right)(\xi,a)|| \\ &\leq \gamma||Q_{\pi,t-1}(\xi,a) - Q'_\pi(\xi,a)||_\infty = \gamma||\left(T^\pi_{h_t-2}Q\right)(\xi,a) - \left(T^\pi_{h_t-2}Q'\right)(\xi,a)|| \\ &\leq \gamma^t||Q_{\pi,0}(\xi,a) - Q'_\pi(\xi,a)||_\infty \\ &\leq \frac{\gamma^t}{1-\gamma}||Q_{\pi,0}(\xi,a) - Q'_\pi(\xi,a)||_\infty\end{aligned} \tag{148}$$

which forms a Cauchy sequence and by the same logic as earlier we can prove that as $t \to \infty$ we must have $Q_{\pi,t}(\xi,a) \to Q_\pi(\xi,a)\ \forall\ t$ and therefore $Q_\pi(\xi,a) = Q'_\pi(\xi,a)$ is a unique fixed point.

Condition six requires that the as the agent samples from the underlying HDP, the expected difference between the application of the Bellman history-based operator $(T^\pi_{h_t}Q)(\xi,a)$ and the optimal Q-value $Q^*(\xi,a)$ is finite and bounded as we apply infinite updates. Application of Eq. (141) directly into Eq. (136) gives

$$\begin{aligned}||\mathbb{E}_{p_{h_i}}[F_t|T^\pi_{h_t}]||_\infty &= ||\mathbb{E}_{p_{h_i}}[(T^\pi_{h_t}Q)(\xi,a) - Q^*_t(\xi,a)|T^\pi_t]||_\infty \\ &\overset{\text{(a)}}{\leq} ||\mathbb{E}_{p_{h_i}}[Q_{\pi,t+1}(\xi,a) - Q^*_t(\xi,a)|\xi,a]||_\infty \\ &\overset{\text{(b)}}{\leq} \gamma||Q_{\pi,t}(\xi,a) - Q^*_t(\xi,a)||_\infty \\ &= \gamma||\Delta_t||_\infty\end{aligned} \tag{149}$$

as for (a) recall the reparameterisation $Q_t(\xi_i, a_i) \leftarrow \left(1 + \frac{Q_t(\xi_i,a_i)}{\Gamma^+_{i-1}}\right)$ and for (b) $\mathbb{E}_{p_{h_i}}[Q_{\pi,t+1}(\xi, a)|\xi_i, a_i] = Q_{\pi,t+1}(\xi, a)$ since the Q-value is already an expectation as seen in Eq. (129). Therefore condition six is satisfied since $\gamma \in [0, 1)$ as one would expect by our definition of F_t.

The final condition ensures the conditional variance between difference between $(T^{\pi}_{h_t}Q)(\xi, a)$ and $Q^*(\xi, a)$ is also finite and bounded as we apply infinite updates. This is seen with

$$
\begin{aligned}
\mathrm{Var}(F_t|T^{\pi}_t) &= \mathbb{E}_{p_{h_i}}[F_t^2|T^{\pi}_t] - \mathbb{E}_{p_{h_i}}[F_t|T^{\pi}_t]^2 \\
&\overset{(a)}{\leq} \frac{1}{4}\left(\max(F_t) - \min(F_t)\right)^2 \\
&\overset{(b)}{=} \frac{1}{4}\max_{s,a}\ \left|\left(T^{\pi}_{h_t}Q\right)(\xi, a) - Q^*_t(\xi, a)\right|^2 \\
&\overset{(c)}{\leq} \frac{\gamma}{4}||Q_{\pi,t}(\xi, a) - Q^*_t(\xi, a)||^2_{\infty} \\
&\leq \frac{\gamma}{4}||\Delta_t||^2_{\infty}
\end{aligned} \tag{150}
$$

where (a) utilises Popoviciu's inequality [164, 165] on variances $\mathrm{Var}(X) \leq \frac{1}{4}\left(\max(X) - \min(X)\right)^2$, (b) uses the fact the $\min(F_t) = 0$ when $(T^{\pi}_{h_t}Q)(\xi, a) \propto Q^*(\xi, a)$, and (c) is again the reparameterisation. An alternative to construct the strict upper bound in (a) may involve using the far more general Bhatia-Davis inequality [166] $\mathrm{Var}(X) \leq (\max(X) - \mathrm{Mean}(X))\,(\mathrm{Mean}(X) - \min(X))$. Hence condition seven is also satisfied as $\gamma \in [0, 1)$.

Therefore we have shown that for multiplicative dynamics if $\Gamma^+_{t-1} > V_{\min} \geq \gamma\ \forall\ t$ then $Q_t(\xi, a) \to Q^*(\xi, a)$ w.p.1. as the number of updates increases indefinitely. This convergence is valid if we assume the state-process is ergodic and forms at the very least a MDP. However, unlike the standard stationary MDP proof, our formulation uses the contraction operator $T^{\pi}_{h_t}$ which is dependent on history and allows to proof to scale to state-processes that are QDPs which can be non-stationary. Formally, if $Q^*(h, a) = Q^*(\tilde{h}, a)$ for some feature map $\phi(h) = \phi(\tilde{h}) = \xi$, then $Q^*(h, a) = Q^*(\xi, a)$. This concludes the proof for the convergence of Q-learning using multiplicative dynamics that valid for a large class of decision problems including MDPs.

5.3 Clipped Double Q-Learning

The utility of using two Q-values was first in the context of Double-Q learning to enhance learning stability in the tabular regime [73]. This approach combined with deep neural networks to more expressively represent action-values was successfully used to obtain seminal results in classic Atari video games [42–48], the board games of Chess, Go and Shogi [52–55]. In actor-critic methodologies discussed in Section 3.3 this approach was combined with actors to generate Q-value estimates [71, 75, 76]. A constant trend seen is the overestimation bias in the Q-values due to accumulating propagation of errors as learning continues. To combat this [80] introduced clipped double Q-learning that uses the minimum of two Q-values for target values which works extremely well compared to all prior formulations as seen across locomotive continuous control tasks [80, 84, 86, 87, 167].

With some modification of the proofs in Section 5.2 and using the methods of [80] we can show the convergence of clipped double Q-learning under multiplicative dynamics assuming the state-process is once again a QDP. For the twin Q-values $Q^A_{\pi,t}$ and $Q^B_{\pi,t}$ where the contraction $T^{\pi}_{h_t}$ operates uniquely on the minimum of two. The optimal action a^* for both Q-values at the same state s_i is defined by using only the first Q-value where for a ϵ-greedy policy

$a^* = \arg\max_a Q^A_{\pi,t}$. Therefore, using

$$\left(T^{\pi}_{h_t}Q\right)(\xi, a) = 1 + \frac{1}{\Gamma^+_{i-1}} \min\left(Q^A_{\pi,t+1}(\xi_i, a^*), Q^B_{\pi,t+1}(\xi_i, a^*)\right) \tag{151}$$

and the same reparameterisation $Q_t(\xi_i, a_i) \leftarrow \left(1 + \frac{Q_t(\xi_i, a_i)}{\Gamma^+_{i-1}}\right)$. We then modify Eqs. (131-133) to yield

$$\Delta_t \equiv Q^A_{\pi,t+1}(\xi, a) - Q^*_t(\xi, a) \tag{152}$$

$$\begin{aligned} F_t &\equiv \left(T^{\pi}_{h_t}Q\right)(\xi, a) - Q^*_t(\xi, a) \\ &= \left(T^{\pi}_{h_t}Q\right)(\xi, a) - Q^*_t(\xi, a) + \left(T^{\pi}_{h_t}Q^A\right)(\xi, a) - \left(T^{\pi}_{h_t}Q^A\right)(\xi, a) \\ &= F^Q_t + c_t \end{aligned} \tag{153}$$

$$\Delta_{t+1} \equiv \left(1 - \alpha_t(\xi, a)\right)\Delta_t(\xi, a) + \alpha_t(\xi, a)F_t \tag{154}$$

where $F^Q_t = \left(T^{\pi}_{h_t}Q^A\right)(\xi, a) - Q^*_t(\xi, a)$ is the standard Q-learning in Eq. (132) which we have proven to converge in Eqs. (149-150). The second term $c_t = \left(T^{\pi}_{h_t}Q\right)(\xi, a) - \left(T^{\pi}_{h_t}Q^A\right)(\xi, a)$ is purposely designed to resemble Eq. (136) and so $c_t \to 0$ w.p.1. which is expected given we expect $Q^A_{\pi,t+1}(\xi, a^*)$ and $Q^B_{\pi,t+1}(\xi, a^*)$ to eventually converge to be equal as $t \to \infty$. Formally we show this by defining $\Delta^{AB}_t \equiv Q^A_{\pi,t}(\xi, a) - Q^B_{\pi,t}(\xi, a)$ so that

$$\begin{aligned} \Delta^{AB}_{t+1} &= \Delta^{AB}_t + \alpha_t(\xi, a)\left(\left(T^{\pi}_{h_t}Q\right)(\xi, a) - Q^A_{\pi,t}(\xi, a) - \left(\left(T^{\pi}_{h_t}Q\right)(\xi, a) - Q^B_{\pi,t}(\xi, a)\right)\right) \\ &= \Delta^{AB}_t + \alpha_t(\xi, a)\left(Q^B_{\pi,t}(\xi, a) - Q^A_{\pi,t}(\xi, a)\right) \\ &= \left(1 - \alpha_t(\xi, a)\right)\Delta^{AB}_t \end{aligned} \tag{155}$$

which clearly proves $c_t \to 0$ convergence as the learning rate decreases. All other aspects of Q-learning convergence remain unchanged and so we can state that $Q^A_t(\xi, a) \to Q^*(\xi, a)$ w.p.1. under the same criteria as before. Noting that in our derivation at no point did we explicitly assume additive or multiplicative dynamics and so can be used for either. Hence, we have extended the applicability of clipped double Q-learning for both dynamics and confirmed that it is functional for history-dependent non-stationary QDP domains.

5.4 Multi-Step Targets

Multi-step targets discussed in Section 3.7 can be easily extended to both dynamics. Firstly, the additive dynamics case adjusts Eq. (78) to include the cumulative additive episodic sum so that the target Q-value becomes

$$Q^{(m)}_{\bar{\theta}}(s_t, a_t) \equiv \Gamma^+_{t-1} + R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(s_{t+m}, a'\right) \tag{156}$$

where $R^{(m)}_t = \sum_{k=0}^{m-1} \gamma^k r_{t+k}$. For multiplicative dynamics there are two ways to achieve this. The first involves calculating the compounding return for $m > 1$ as

$$\Gamma^{\times}_{t+m-2} \equiv \prod_{k=0}^{m-2} \frac{\Gamma^+_{t-1+k} + \gamma^k r_{t+k}}{\Gamma^+_{t-1+k}} = \exp\left[\sum_{k=0}^{m-2} \ln\left|1 + \frac{\gamma^k r_{t+k}}{\Gamma^+_{t-1+k}}\right|\right] \tag{157}$$

where at each m-step bootstrapping, $\Gamma^+_{t-1+k} + \gamma^k r_{t+k} > V_{\min} \geq \gamma$ must be explicitly confirmed. This then allows us to write the target value as

$$1 + \frac{Q^{(m)}_{\bar{\theta}}(\xi_t, a_t)}{\Gamma^+_{t+m-2}} \equiv \Gamma^\times_{t+m-2} \cdot \frac{\Gamma^+_{t+m-2} + \gamma^{m-1}\left(r_{t+m-1} + \gamma Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right)\right)}{\Gamma^+_{t+m-2}} \tag{158}$$

as the product of compounding growth rates at each step with the estimated Q-value being the heavily discounted future value. Calculating this at each learning step when randomly sampling from an experience replay buffer $\mathcal{D}$ for a relatively large mini-batch size will be computationally expensive as there are two products to calculate while confirming each episode is not terminated at each bootstrapping step due to the 'game over' condition.

Alternatively, we can use a simplification that is far more computationally efficient but strictly speaking is not a multiplicative process. Instead of products, using Eq. (119) as a guide we can construct

$$1 + \frac{Q^{(m)}_{\bar{\theta}}(\xi_t, a_t)}{\Gamma^+_{t+m-2}} \approx \left(1 + \frac{R^{(m)}_t}{\Gamma^+_{t-1}}\right) + \frac{\gamma^m Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right)}{\Gamma^+_{t-1}} = 1 + \frac{R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right)}{\Gamma^+_{t-1}} \tag{159}$$

as the sum of the m-holding period return and the forecasted discounted perpetuity ratio for all later periods. It is unlikely that these two methods would produce identical learning outcomes since intermediate rewards in $R^{(m)}_t$ could trigger the 'game over' criterion. An exception to this is obviously the usual $m = 1$ case

$$1 + \frac{Q^{(1)}_{\bar{\theta}}(\xi_t, a_t)}{\Gamma^+_{t-1}} \equiv 1 + \frac{Q_{\bar{\theta}}(\xi_t, a_t)}{\Gamma^+_{t-1}} \leftarrow 1 + \frac{r_t + \gamma Q_{\bar{\theta}}\left(\xi_{t+1}, a'\right)}{\Gamma^+_{t-1}} \tag{160}$$

with no additional bootstrapping that can be considered the multiplicative analogue of Eq. (28). For actor-critic methods, the critic optimisation in Eq. (30) can then be written

$$\begin{aligned} \frac{Q^{(m)}_{\bar{\theta}}(\xi_t, a_t)}{\Gamma^+_{t+m-2}} - \frac{Q_\theta(\xi_t, a_t)}{\Gamma^+_{t-1}} &\approx \left(1 + \frac{R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right)}{\Gamma^+_{t-1}}\right) - \left(1 + \frac{Q_\theta\left(\xi_t, a_t\right)}{\Gamma^+_{t-1}}\right) \\ &= \frac{1}{\Gamma^+_{t-1}}\left(R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right) - Q_\theta\left(\xi_t, a_t\right)\right) \end{aligned} \tag{161}$$

as the difference to be minimised. This is the key point, for off-policy learning with a mini-batch using an MSE loss function we have the objectives

$$J(\theta^+) = \mathbb{E}_{U(\mathcal{D}^+)}\left[\left(R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(s_{t+m}, a'\right) - Q_\theta\left(s_t, a_t\right)\right)^2\right] \tag{162}$$

$$J(\theta^\times) \approx \mathbb{E}_{U(\mathcal{D}^\times)}\left[\left(\frac{1}{\Gamma^+_{t-1}}\left(R^{(m)}_t + \gamma^m Q_{\bar{\theta}}\left(\xi_{t+m}, a'\right) - Q_\theta\left(\xi_t, a_t\right)\right)\right)^2\right] \tag{163}$$

with uniform sampling from the experience replay buffers returning the tuples $\left(s_t, a_t, R^{(m)}_t, s_{t+m}\right) \sim U(\mathcal{D}^+)$ and $\left(\Gamma^+_{t-1}, s_t, a_t, R^{(m)}_t, s_{t+m}\right) \sim U(\mathcal{D}^\times)$ respectively, with multiplicative case restricting sampling to include only cases where $\Gamma^+_{t-1} > V_{\min} \geq \gamma$ and so only learns from 'living' states $\mathcal{D}^\times \subseteq \mathcal{D}^+$, or in terms of cardinality $|\mathcal{D}^\times| \leq |\mathcal{D}^+|$. Recall from Section 3.7 the details regarding how to each of the tuples are actually formed through the reconstructing the histories from the buffer.

Overall, this formulation is crucial for many environments where the agent is not permitted to go 'temporarily bankrupt' in order to maximise returns, it must strictly find the optimal policy under the constraint of staying 'alive' at all times. This type of behaviour is required for all situations where there may exists alternative paths to the same destination. It should be noted that this approach will then by definition be more sample inefficient as it does not use the complete buffer, this will be especially important during the early stages when the agent is repeatedly failing.

Each sample within the mini-batch will have a unique Γ^+ and so the difference between the two aggregation schemes is non-trivial. The largest contributions for the additive case will be when the absolute difference is large, while for multiplicative, it will be for those samples that have the largest difference in returns. There is no reason to assume that the largest contributors will be shared between the dynamics.

6 Policy Gradients with Multiplicative Dynamics

Policy gradients introduced in Section 3.2 come in two varieties, stochastic $\pi_\phi(a|s)$ and deterministic $\mu_\phi(s)$ action sampling when the agent is provided a state. These are essential if we desire to use actors capable of performing more complex manoeuvres than the ϵ-greedy approach. In both cases, the action-values are reparameterised $Q_{\pi_\phi}(s,a) \to Q_\theta(s,a)$ to enhance stability by reducing coupling.

For additive dynamics, as with Q-learning, all existing literature results holds as we simply scale the rewards $r_t \leftarrow r_t + \Gamma^+_{t-1}$. In this section we prove that for multiplicative dynamics policy gradient theorems also remain relatively unchanged as there are no temporal differences between Q-values at any point. The results will again resemble Eqs. (162-163) with the functional forms of both dynamics looking similar but experimental results will likely be very different.

6.1 Stochastic Actors

For stochastic policies the objective to maximise $J(\pi_\phi) = \mathbb{E}\left[1 + R^\times_t|\pi_\phi\right]$ through gradient ascent using $\nabla_\phi J(\pi_\phi) = \mathbb{E}\left[\nabla_\phi \pi_\phi(a|s)\left(1 + R^\times_t\right)\right] = \mathbb{E}\left[\nabla_\phi \ln \pi_\phi(a|s)\left(1 + R^\times_t\right)|\pi_\phi\right]$ or in other words $\phi^* = \arg\max_\phi \mathbb{E}\left[\left(1 + R^\times_t\right)|\pi_\phi\right]$ where $R^\times_t$ is defined in Eqs. (113-115). Using the usual (improper) discounted state visitation distribution for policy π_ϕ as ρ^{π_ϕ} which can be interpreted as representing the marginals for the trajectory distribution, the policy gradient becomes

$$\begin{aligned}\nabla_\phi J(\pi^\times_\phi) &= \mathbb{E}_{\xi_t \sim \rho^{\pi_\phi}, a_t \sim \pi_\phi}\left[\nabla_\phi \pi_\phi(a_t|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right] \\ &= \mathbb{E}_{\xi_t \sim \rho^{\pi_\phi}}\left[\nabla_\phi \ln \pi_\phi(a_t|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\middle|\, \pi_\phi\right] \end{aligned} \tag{164}$$

where the future return is due to the Bellman equation in Eq. (118) being of the correct monotonic structure to maximise $R^\times_t$, but not an exact representation of the compounding growth rate.

To prove that this theorem works in multiplicative dynamics we must explicitly show that gradient ascent works $\nabla_\phi J(\pi^+_\phi) \propto \nabla_\phi \pi_\phi(a_t|s_t) Q_{\pi_\phi}(s_t, a_t)$ as seen in Eq. (25). This is done by using the Bellman equation and unrolling the Q-values following the original derivation [34, 74]. We first remove the expectation value and operate in a discrete tabular state-action space. We also denote the additive cumulative reward $\Gamma^+_{t-1}(s_{t-1}, a_{t-2}|h_{t-1}) = \Gamma^+_{t-1}$ as a constant at each step for stochastic sampling $a \in \mathcal{A}$ since it is solely based on a known history. Observe then

$$\begin{aligned}\nabla_\phi J(\pi^\times_\phi) &= \mathbb{E}_{a_t \sim \pi_\phi}\left[\nabla_\phi \pi_\phi(a|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right] \\ &= \sum_a \left[\nabla_\phi \pi_\phi(a|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right) + \pi_\phi(a|\xi_t)\nabla_\phi\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right] \\ &\overset{\text{(a)}}{=} \sum_a \left[\nabla_\phi \pi_\phi(a|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right. \\ &\qquad\left. + \pi_\phi(a|\xi_t)\nabla_\phi \sum_{\xi_{t+1}}\left[p_{h_t}(\xi_{t+1}|\xi_t, a_t)\left(1 + \frac{r(\xi_t, a') + \gamma Q_{\pi_\phi}(\xi_{t+1}, a')}{\Gamma^+_{t-1}}\right)\right]\right] \\ &\overset{\text{(b)}}{=} \sum_a \left[\nabla_\phi \pi_\phi(a|\xi_t)\left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right. \end{aligned}$$

$$
\begin{aligned}
&\quad +\pi_\phi(a|\xi_t)\nabla_\phi \sum_{\xi_{t+1}} \left[p_{h_t}(\xi_{t+1}|\xi_t, a_t) \left(1 + \frac{\gamma Q_{\pi_\phi}(\xi_{t+1}, a')}{\Gamma^+_{t-1}}\right)\right]\Bigg] \\
&\overset{(c)}{=} \sum_a \Bigg[\nabla_\phi \pi_\phi(a|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right) + \pi_\phi(a|\xi_t) \sum_{\xi_{t+1}} \Big[p_{h_t}(\xi_{t+1}|\xi_t, a_t) \\
&\quad \times \sum_{a'} \Bigg[\nabla_\phi \pi_\phi(a'|\xi_{t+1}) \left(1 + \frac{\gamma Q_{\pi_\phi}(\xi_{t+1}, a')}{\Gamma^+_{t-1}}\right) \\
&\qquad + \pi_\phi(a'|\xi_{t+1}) \nabla_\phi \sum_{\xi_{t+2}} \left[p_{h_{t+1}}(\xi_{t+2}|\xi_{t+1}, a_{t+1}) \left(1 + \frac{r(\xi_{t+1}, a'') + \gamma^2 Q_{\pi_\phi}(\xi_{t+2}, a')}{\Gamma^+_{t-1}}\right)\right]\Bigg]\Big]\Bigg] \\
&= \sum_a \Bigg[\nabla_\phi \pi_\phi(a|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right) + \pi_\phi(a|\xi_t) \sum_{\xi_{t+1}} \Big[p_{h_t}(\xi_{t+1}|\xi_t, a_t) \\
&\quad \times \sum_{a'} \Bigg[\nabla_\phi \pi_\phi(a'|\xi_{t+1}) \left(1 + \frac{\gamma Q_{\pi_\phi}(\xi_{t+1}, a')}{\Gamma^+_{t-1}}\right) \\
&\qquad + \pi_\phi(a'|\xi_{t+1}) \nabla_\phi \sum_{\xi_{t+2}} \left[p_{h_{t+1}}(\xi_{t+2}|\xi_{t+1}, a_{t+1}) \left(1 + \frac{\gamma^2 Q_{\pi_\phi}(\xi_{t+2}, a')}{\Gamma^+_{t-1}}\right)\right]\Bigg]\Big]\Bigg]
\end{aligned} \tag{165}
$$

which clearly leads to infinite recursion. In (a) we use the Bellman equation seen in Eq. (129), (b) we assume $\nabla_\phi \left(r(\xi_t, a')/\Gamma^+_{t-1}\right) = 0$ as action sampling for constant rewards is random which is not exactly true since policy learning is a non-stationary endeavour, and (c) we apply the Bellman equation again further revealing the recursion.

As usual we define $p(\xi_t \to \xi', k, \pi_\phi)$ to be the probability of $\xi_t \to \xi'$ in k time steps under policy π_ϕ. We can summarise this recursion by repeated application of the Bellman equation with

$$
\nabla_\phi J(\pi^\times_\phi) = \sum_{\xi_{t+1}} \left(\sum_{t=0}^{\infty} p(\xi_t \to \xi_{t+1}, t, \pi_\phi)\right) \sum_{a_t} \nabla_\phi \pi_\phi(a|\xi_{t+1}) \left(1 + \frac{\gamma^t Q_{\pi_\phi}(\xi_{t+1}, a)}{\Gamma^+_{t-1}}\right) \tag{166}
$$

where the policy quantifies Eq. (118). When performing gradient ascent, the impact of constant values does not affect the optimisation process. Hence if modify the unrolled returns such that

$$
\gamma^t \left(1 + \frac{Q_{\pi_\phi}(\xi_{t+1}, a)}{\Gamma^+_{t-1}}\right) \leftarrow \left(1 + \frac{\gamma^t Q_{\pi_\phi}(\xi_{t+1}, a)}{\Gamma^+_{t-1}}\right) \tag{167}
$$

there will be no change on the final results, only the speed of convergence especially if there are no local maximums. The γ^t term geometrically decreases for subsequent bootstrapping and so poses no issues. It would be incorrect to exactly factor out γ^t as inclusion of a γ^{-t} would lead to divergence given $\gamma \in [0, 1)$ when $t \to \infty$. For practical purposes, policy learning occurs at a fixed step and so this will still ensure the weights ϕ are updated in the correct direction regardless.

This allows us to utilise existing machinery and construct the usual density

$$
\rho^{\pi_\phi}(\xi') \equiv \int_{\mathcal{S}} \mathrm{d}\xi \sum_{t=1}^{\infty} \gamma^{t-1} p_1(\xi) p(\xi \to \xi', t, \pi_\phi) \tag{168}
$$

so that

$$
\nabla_\phi J(\pi^\times_\phi) \approx \nabla_\phi \int_{\mathcal{S}} \rho^{\pi_\phi}(\xi) \int_{\mathcal{A}} \mathrm{d}a \mathrm{d}\xi \; \pi_\phi(a_t|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)
$$

$$= \int_{\mathcal{S}} \rho^{\pi_\phi}(\xi) \int_{\mathcal{A}} \mathrm{d}a\mathrm{d}\xi \; \nabla_\phi \pi_\phi(a_t|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)$$

$$= \mathbb{E}_{\xi_t \sim \rho^{\pi_\phi}, a_t \sim \pi_\phi} \left[\nabla_\phi \ln \pi_\phi(a_t|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right] \tag{169}$$

where the approximation is due to Eq. (167). This is of the required form [34, 74, 75] and note we have purposely retained the $(1 + R^\times)$ structure to highlight that the policy is maximising the future discounted (exponentially) continuous compounding return in Eq. (115).

Effectively this is saying that as long as the underlying parameters policy $\phi^\times$ are updated in the direction of $\nabla_\phi \ln \pi_\phi(a_t|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t,a_t)}{\Gamma^+_{t-1}}\right)$, the optimal policy is approached. The difference between this and the standard $\nabla_\phi J(\pi^+_\phi) \propto \nabla_\phi \pi_\phi(a_t|s_t) Q_{\pi_\phi}(s_t, a_t)$ result is simply here we have represented the value as return over the existing cumulative episodic return. Therefore, objectives to be maximised for off-policy stochastic actors are then

$$J(\phi^+) = \mathbb{E}_{U(\mathcal{D}^+)} \left[\ln \pi_\phi(a_t|s_t) Q_{\pi_\phi}(s_t, a_t)\right] \tag{170}$$

$$J(\phi^\times) \approx \mathbb{E}_{U(\mathcal{D}^\times)} \left[\ln \pi_\phi(a_t|\xi_t) \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\right] \tag{171}$$

where the discussion in Section 5.4 on the differences between both dynamics is applicable. Overall, we see that multiplicative dynamics maximises the future return relative to the existing value while additive dynamics simply maximises the future value. The former then can be interpreted as specifically prioritising policies that avoid steep losses at all time steps even though they may not yield the largest valuations.

6.2 Deterministic Actors

Deterministic polices have the advantage of being more computationally efficient as they remove random policy distribution sampling at each gradient step. The objective to maximise is $J(\mu_\phi) = \mathbb{E}\left[1 + R^\times_t | \mu_\phi\right]$ through gradient ascent using $\nabla_\phi J(\mu_\phi) = \mathbb{E}\left[\nabla_\phi \mu_\phi \left(1 + R^\times_t\right)\right]$ or equivalently $\phi^* = \arg\max_\phi \mathbb{E}\left[\left(1 + R^\times_t\right) | \mu_\phi\right]$. Similarly the (improper) discounted state visitation distribution for policy μ_ϕ is ρ^{μ_ϕ}, the deterministic policy gradient is then

$$\nabla_\phi J(\mu^\times_\phi) \approx \mathbb{E}_{\xi_t \sim \rho^{\mu_\phi}} \left[\nabla_\phi \left(1 + \frac{Q_{\mu_\phi}(\xi_t, \mu_\phi(\xi_t))}{\Gamma^+_{t-1}}\right)\right] \tag{172}$$

where the approximation is again due to local maximisation and we must then show $\nabla_\phi J(\mu^+_\phi) \propto \nabla_\phi Q_{\pi_\phi}(s_t, a_t)$ as seen in Eq. (26). This is done similarly to the stochastic case but does not require action sampling $a \in \mathcal{A}$. To avoid hassles with changing order of integrals, we use a discrete tabular version of the original derivation [75]. The additive cumulative reward $\Gamma^+_{t-1}(s_{t-1}, \mu_\phi(s_{t-2})|h_{t-1}) = \Gamma^+_{t-1}$ is also a constant at each step given the history is completely known. We then express the policy gradient as

$$\begin{aligned}
\nabla_\phi J(\mu^\times_\phi) &\approx \nabla_\phi \left(1 + \frac{Q_{\mu_\phi}(\xi_t, \mu_\phi(\xi_t))}{\Gamma^+_{t-1}}\right) \\
&\overset{(a)}{=} \nabla_\phi \sum_{\xi_{t+1}} \left[p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t)) \left(1 + \frac{r(\xi_t, \mu_\phi(\xi_t)) + \gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right)\right] \\
&\overset{(b)}{=} \nabla_\phi \frac{r(\xi_t, \mu_\phi(\xi_t))}{\Gamma^+_{t-1}} + \nabla_\phi \sum_{\xi_{t+1}} \left[p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t)) \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right)\right]
\end{aligned}$$

$$
\begin{aligned}
&\overset{(c)}{=} \frac{\nabla_\phi \mu_\phi(\xi_t) \nabla_a r(\xi_t, a_t)|_{a_t=\mu_\phi(\xi_t)}}{\Gamma^+_{t-1}} \\
&\quad + \sum_{\xi_{t+1}} \Bigg[\nabla_\phi \mu_\phi(\xi_t) \nabla_a p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t))|_{a_t=\mu_\phi(\xi_t)} \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right) \\
&\qquad + p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t)) \nabla_\phi \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right)\Bigg] \\
&\overset{(d)}{=} \nabla_\phi \mu_\phi(\xi_t) \nabla_a \left(\frac{r(\xi_t, a_t)}{\Gamma^+_{t-1}} + \sum_{\xi_{t+1}} \left[p_{h_t}(\xi_{t+1}|\xi_t, a_t) \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma^+_{t-1}}\right)\right]\right)\Bigg|_{a_i=\mu_\phi(\xi_i)} \\
&\quad + \sum_{\xi_{t+1}} p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t)) \nabla_\phi \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right) \\
&= \nabla_\phi \mu_\phi(\xi_t) \nabla_a \sum_{\xi_{t+1}} \left(p_{h_t}(\xi_{t+1}|\xi_t, a_t) \left(1 + \frac{r(\xi_t, a_t) + \gamma Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma^+_{t-1}}\right)\right)\Bigg|_{a_i=\mu_\phi(\xi_i)} \\
&\quad + \sum_{\xi_{t+1}} p_{h_t}(\xi_{t+1}|\xi_t, \mu_\phi(\xi_t)) \nabla_\phi \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right) \\
&\overset{(e)}{=} \nabla_\phi \mu_\phi(\xi_t) \nabla_a \left(1 + \frac{Q_{\mu_\phi}(\xi_t, a_t)}{\Gamma^+_{t-1}}\right)\Bigg|_{a_i=\mu_\phi(\xi_i)} + \Lambda(\xi_t, \mu_\phi(\xi_t))
\end{aligned} \tag{173}
$$

where we define the infinitely recursive summation as $\Lambda(\xi_t, \mu_\phi(\xi_t))$. In (a) the Bellman equation in Eq. (129) is used, (b) the rewards are independent of the next state probability distributions, (c) we use the chain rule on the first term and both the product rule on the second term, (d) terms are grouped according to the respective gradients, and (e) the first term is aggregated since it is a constant at a fixed action and the second term includes all additional unrolling via the Bellman equation. This term can be rewritten as

$$
\Lambda(\xi_t, \mu_\phi(\xi_t)) = \sum_{\xi_{t+1}} p(\xi_t \to \xi_{t+1}, 1, \mu_\phi) \nabla_\phi \left(1 + \frac{\gamma Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right) \tag{174}
$$

with $p(\xi_t \to \xi', k, \mu_\phi)$ again to be the probability of $\xi_t \to \xi'$ in k time steps. The recursion can be seen through

$$
\begin{aligned}
\Lambda(\xi_t, \mu_\phi(\xi_t)) &= \nabla_\phi \mu_\phi(\xi_t) \nabla_a \sum_{\xi_{t+1}} \left(p(\xi_t \to \xi_{t+1}, 1, \mu_\phi) \left(1 + \frac{r(\xi_t, a_t) + \gamma Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma^+_{t-1}}\right)\right)\Bigg|_{a_i=\mu_\phi(\xi_i)} \\
&\quad + \sum_{\xi_{t+1}} p(\xi_t \to \xi_{t+1}, 1, \mu_\phi) \sum_{\xi_{t+2}} p(\xi_{t+1} \to \xi_{t+2}, 1, \mu_\phi) \nabla_\phi \left(1 + \frac{\gamma^2 Q_{\mu_\phi}(\xi_{t+2}, \mu_\phi(\xi_{t+2}))}{\Gamma^+_{t-1}}\right) \\
&= \nabla_\phi \mu_\phi(\xi_t) \nabla_a \sum_{\xi_{t+1}} \left(p(\xi_t \to \xi_{t+1}, 1, \mu_\phi) \left(1 + \frac{r(\xi_t, a_t) + \gamma Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma^+_{t-1}}\right)\right)\Bigg|_{a_i=\mu_\phi(\xi_i)} \\
&\quad + \sum_{\xi_{t+1}} p(\xi_t \to \xi_{t+1}, 2, \mu_\phi) \nabla_\phi \left(1 + \frac{\gamma^2 Q_{\mu_\phi}(\xi_{t+1}, \mu_\phi(\xi_{t+1}))}{\Gamma^+_{t-1}}\right) \\
&\;\vdots \\
&= \sum_{\xi_{t+1}} \sum_{t=1}^{\infty} p(\xi_t \to \xi_{t+1}, t, \mu_\phi) \nabla_\phi \mu_\phi(\xi_t) \nabla_a \left(1 + \frac{\gamma^t Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma^+_{t-1}}\right)\Bigg|_{a_{t+1}=\mu_\phi(\xi_{t+1})}
\end{aligned} \tag{175}
$$

Therefore we can write

$$\nabla_\phi J(\mu_\phi^\times) = \sum_{\xi_{t+1}} \sum_{t=0}^{\infty} p(\xi_t \to \xi_{t+1}, t, \mu_\phi) \nabla_\phi \mu_\phi(\xi_t) \nabla_a \left(1 + \frac{\gamma^t Q_{\mu_\phi}(\xi_{t+1}, a_{t+1})}{\Gamma_{t-1}^+}\right)\Bigg|_{a_{t+1}=\mu_\phi(\xi_{t+1})} \tag{176}$$

As before, the impact of constant values does not affect the optimisation process and so we use the approximations in Eqs. (167) which allows us to then construct the usual density

$$\rho^{\mu_\phi}(\xi') \equiv \int_{\mathcal{S}} \mathrm{d}\xi \sum_{t=1}^{\infty} \gamma^{t-1} p_1(\xi) p(\xi \to \xi', t, \mu_\phi) \tag{177}$$

and so finally

$$\begin{aligned}
\nabla_\phi J(\mu_\phi^\times) &\approx \nabla_\phi \int_{\mathcal{S}} \mathrm{d}s\ \rho^{\mu_\phi}(s) \left(1 + \frac{Q_{\pi_\phi}(s, \mu_\phi(s))}{\Gamma_{t-1}^+}\right) \\
&= \mathbb{E}_{\xi_t \sim \rho^{\mu_\phi}} \left[\nabla_\phi \left(1 + \frac{Q_{\pi_\phi}(s, \mu_\phi(s))}{\Gamma_{t-1}^+}\right)\right] \qquad (178) \\
&= \mathbb{E}_{\xi_t \sim \rho^{\mu_\phi}} \left[\frac{\nabla_\phi \mu_\phi(\xi_t) \nabla_a Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma_{t-1}^+}\Bigg|_{a_t=\mu_\phi(\xi_t)}\right] \qquad (179)
\end{aligned}$$

which is of the desired form [75]. We also assume without proof the limiting case theorem that the stochastic policy actor approaches the deterministic policy actor as variance is reduced

$$\lim_{\sigma \to 0} \nabla_\phi J(\pi_{\phi,\sigma}^\times) = \nabla_\phi J(\mu_\phi^\times) \tag{180}$$

which is reasonable as the functional form for the multiplicative gradients is identical to the stochastic gradients up to a constant and scaled by cumulative reward. Therefore we do not expect these predefined known values to have any impact on limiting convergence.

The parameters $\phi^\times$ are therefore updated in the direction of $\nabla_\phi \left(1 + \frac{Q_{\pi_\phi}(\xi_t, a_t)}{\Gamma_{t-1}^+}\right)$ with difference between this and the standard $\nabla_\phi J(\pi_\phi) \propto \nabla_\phi Q_{\pi_\phi}(s_t, a_t)$ being now using return over the existing cumulative episodic return. Therefore, for popular off-policy algorithms such as DDPG [76] and TD3 [80] the deterministic actors maximise the following objectives

$$J(\phi^+) \approx \mathbb{E}_{U(\mathcal{D}^+)} \left[Q_{\pi_\phi}(s_t, \mu_\phi(s_t))\right] \tag{181}$$

$$J(\phi^\times) \approx \mathbb{E}_{U(\mathcal{D}^\times)} \left[\left(1 + \frac{Q_{\pi_\phi}(\xi_t, \mu_\phi(\xi_t))}{\Gamma_{t-1}^+}\right)\right] \tag{182}$$

with the same implications as discussed earlier. Both are approximations due to formulation of deterministic policy gradients not performing global maximisation of Q-values [75], while the second requires an additional approximation due to Eq. (167). Notice that for both the stochastic and deterministic policy gradients, if we performed the simple reparameterisation $Q(s, a) \leftarrow \left(1 + \frac{Q(\xi, a)}{\Gamma_{t-1}^+}\right)$, the existing derivations [34, 74, 75] would have been completely valid and left unchanged, allowing us to effortlessly arrive at the final results. This shortcut however would not have elucidated the strict requirements in Eq. (167) that are needed to represent the gradients in a tractable manner. Furthermore, with the explicit proof we are unable to find any interesting hidden structure.

7 Maximum Causal Entropy with Multiplicative Dynamics

The energy based polices discussed in Section 3.4 are built around soft updates using an maximum entropy objective [81–88]. Modification of this approach to incorporate multiplicative dynamics uses a very similar method to that in Section 5.1. We also explicitly assume the underlying processes is a QDP and therefore we formally extend the soft-actor critic algorithm to this much larger class of process already encompassing all MDPs.

The discounted future compounding rewards with state-dependent entropies are

$$
\begin{aligned}
1+R_t^{\times} &\equiv \left(1+\Gamma_{t-1}^{\times}\right)\cdot\frac{\Gamma_{t-1}^{+}+\left(r_t+\alpha H\left[\pi(\cdot|\xi_t)\right]\right)}{\Gamma_{t-1}^{+}}\cdot\frac{\left(\Gamma_{t-1}^{+}+r_t\right)+\gamma\left[r_{t+1}+\alpha H\left(\pi(\cdot|\xi_{t+1})\right)\right]}{\Gamma_{t-1}^{+}+r_t}\cdot\ldots \\
&= \left(1+\Gamma_t^{\times}\right)\cdot\prod_{k=1}^{\infty}\left(1+\frac{\gamma^k\left[r_{t+k}+\alpha H\left(\pi(\cdot|\xi_{t+k})\right)\right]}{\Gamma_{t-1+k}^{+}}\right) \\
&= \left(1+\Gamma_t^{\times}\right)\cdot\exp\left[\sum_{k=1}^{\infty}\ln\left|1+\frac{\gamma^k\left[r_{t+k}+\alpha H\left(\pi(\cdot|\xi_{t+k})\right)\right]}{\Gamma_{t-1+k}^{+}}\right|\right] && (183) \\
&\propto \sum_{k=1}^{\infty}\left(1+\frac{\gamma^k\left[r_{t+k}+\alpha H\left(\pi(\cdot|\xi_{t+k})\right)\right]}{\Gamma_{t-1+k}^{+}}\right) && (184)
\end{aligned}
$$

where $\Gamma_{t-1+k}^{+}=\Gamma_{t-1}^{+}+\sum_{\lambda=0}^{k-1}\gamma^{\lambda}r_{t+\lambda}$ do not include entropies, the current return $\Gamma_t^{\times}$ is completely known, and we will show again that the initial value $\Gamma_0^{+}=V_0>V_{\min}\geq\gamma$ is required. The final proportionality is based on same logic as in Q-learning in Eq. (117). One important difference is that now the bound is $\gamma^k r_{t+k}\geq\left(V_{\min}-\Gamma_{t-1+k}^{+}-\gamma^k\alpha H\left(\pi(\cdot|\xi_{t+k})\right)\right)$ where the equality again ends the episode. This is a particularly robust mechanism to train agents as it adds an additional state-dependent noise to the obtained reward signal, leading to overall much less brittle learning. In other words, this artificially simulates volatility in returns when learning a policy to encourages exploration. The automatically tuned temperature parameter α then governs the magnitude of randomness, where ideally we desire it to increase as the agents simulated performance increase and vice versa.

7.1 Soft Learning

The maximum entropy objective can then be written by in terms of $\pi^*\in\arg\max_{\pi^*\in\Pi}\left(1+R_t^{\times}\right)$. Formally, first we define the entropy augmented reward $r_{t+k}^{\pi}\equiv r_{t+k}+\alpha H\left(\pi(\cdot|\xi_{t+k})\right)$ that so that

$$
1+R_t^{\times}\propto\sum_{k=1}^{\infty}\left(1+\frac{\gamma^k r_{t+k}^{\pi}}{\Gamma_{t-1+k}^{+}}\right) \tag{185}
$$

which is of the exact form in Eq. (117) and so much of the earlier results apply. Recall the definition for soft values functions in Eqs. (32-33) which we modify to

$$
\begin{aligned}
1+\frac{Q_{\pi}^{\text{soft}}(\xi_t,a_t)}{\Gamma_{t-1}^{+}} &\equiv \mathbb{E}_{\xi_{t+1}\sim\rho^{\pi},a_{t+1}\sim\pi}\left[\sum_{k=1}^{\infty}\left(1+\frac{\gamma^k r_{t+k}^{\pi}}{\Gamma_{t-1}^{+}}\right)\right] && (186) \\
1+\frac{V_{\theta}^{\text{soft}}(\xi_t)}{\Gamma_{t-1}^{+}} &\equiv \alpha\ln\int_{\mathcal{A}}\mathrm{d}a'\ \exp\left[\left(\frac{1}{\alpha}\left(1+\frac{Q_{\pi}^{\text{soft}}(\xi_t,a')}{\Gamma_{t-1}^{+}}\right)\right)\right] && (187) \\
&= \alpha\ln\mathbb{E}_{Z_{\pi}\sim a'}\left[\frac{1}{Z_{\pi}(a')}\exp\left(\frac{1}{\alpha}\left(1+\frac{Q_{\pi}^{\text{soft}}(\xi_t,a')}{\Gamma_{t-1}^{+}}\right)\right)\right] && (188)
\end{aligned}
$$

Next, we use the state distribution ρ^π to define again the approximate (soft) Bellman equation

$$1+\frac{Q^{\text{soft}}_\pi(\xi_{t+k},a_{t+k})}{\Gamma^+_{t-1+k}} \leftarrow 1+\frac{1}{\Gamma^+_{t-1+k}}\mathbb{E}_{\xi_{t+k+1}\sim\rho^\pi,a_{t+k+1}\sim\pi}\left[r^\pi\left(\xi_{t+k},a_{t+k}\right)+\gamma Q^{\text{soft}}_\pi\left(\xi_{t+k+1},a_{t+k+1}\right)\right] \tag{189}$$

where r^π_{t+k} is again a known constant and so in the limit $\alpha\to 0$ this result is exactly equivalent to Eq. (118). Based on the MDP additive dynamics case set out in [82–84, 86], we can introduce a soft Bellman history operator $\mathcal{T}^\pi_{h_t}$ that is valid under all previous QDP assumptions to perform soft policy evaluation. This is defined as

$$\left(\mathcal{T}^\pi_{h_t}Q^{\text{soft}}\right)(\xi,a)=1+\frac{Q^{\text{soft}}_\pi(\xi_t,a_t)}{\Gamma^+_{t-1}}=1+\frac{\mathbb{E}_{\xi_{t+1}\sim\rho^\pi,a_{t+1}\sim\pi}\left[r^\pi\left(\xi_t,a_t\right)+\gamma Q^{\text{soft}}_\pi\left(\xi_{t+1},a_{t+1}\right)\right]}{\Gamma^+_{t-1}} \tag{190}$$

To prove under a strict criteria the convergence $\left(\mathcal{T}^\pi_{h_t}Q^{\text{soft}}\right)(\xi,a)=Q^{\text{soft}}_{t+1}(\xi,a)\to Q^{\text{soft}*}(\xi,a)$ w.p.1. as infinite updates $t\to\infty$ we must introduce an additional condition that the cardinality of the action space $|\mathcal{A}|<\infty$ is finite to ensure the entropy term is bounded. Now combining this condition, the QDP assumptions, and the conditions in Section 5.2, we see that if condition five can be proven then we can utilise all but the prior results without any additional changes to prove that soft Q-learning convergences to the optimal soft Q-value in QDP domains.

To show this we adapt the ideas in [82, 83]. Suppose for a fixed history h_t and fixed point (ξ_t,a_t) the max-norm exists $\epsilon=\frac{1}{\Gamma^+_{t-1}}||Q_{\pi,t}(\xi,a)-Q'_{\pi,t}(\xi,a)||_\infty$, if then $Q_{\pi,t}\le\epsilon+Q'_{\pi,t}$ and similarly $Q_{\pi,t}\ge-\epsilon+Q'_{\pi,t}$ then by the definition of the max-norm

$$||\left(\mathcal{T}^\pi_{h_t}Q\right)(\xi,a)-\left(\mathcal{T}^\pi_{h_t}Q'\right)(\xi,a)||_\infty\le\gamma\epsilon=\frac{\gamma}{\Gamma^+_{t-1}}||Q_{\pi,t}(\xi,a)-Q'_{\pi,t}(\xi,a)||_\infty \tag{191}$$

and therefore by the same constraints on Γ^+_{t-1} we have

$$||\left(\mathcal{T}^\pi_{h_t}Q\right)(\xi,a)-\left(\mathcal{T}^\pi_{h_t}Q'\right)(\xi,a)||_\infty\le\gamma||Q_{\pi,t}(\xi,a)-Q'_{\pi,t}(\xi,a)||_\infty \tag{192}$$

hence all other convergence results follow, proving that there exists a unique optimal solution to soft Q-learning that is greedily approached via soft value iteration.

Furthermore, if we very generally represent the policy distribution as general energy-based policy of the form of a Boltzmann distribution [17] where

$$\pi(a_t|\xi_t)\equiv\exp\left[\frac{1}{\alpha}\left(1+\frac{Q^{\text{soft}}_\pi(\xi_t,a_t)}{\Gamma^+_{t-1}}\right)-\ln|Z_\pi(\xi_t)|\right] \tag{193}$$

the soft policy improvement can be defined by the objective to be maximised

$$J(\pi)\approx\mathbb{E}_{\xi_t\sim\rho^\pi,a_t\sim\pi}\left[\left(1+\frac{Q^{\text{soft}}_\pi\left(\xi_t,a_t\right)}{\Gamma^+_{t-1}}\right)\right] \tag{194}$$

with the approximation due to Eq. (189). Using the Kullback-Leibler divergence we can improve the policy $\pi(a_t|\xi_t)$ in Eq. (193) to $\pi(a_t|\xi_t)\to\pi'(a_t|\xi_t)$ with

$$\begin{aligned}\pi'(\cdot|\xi_t)&=\arg\min_{\pi'\in\Pi}\mathrm{D_{KL}}\left(\pi'(\cdot|\xi_t)\;\middle|\middle|\;\pi(\cdot|\xi_t)\right)\\&=\arg\min_{\pi'\in\Pi}\mathbb{E}_{a_t\sim\pi'}\left[\ln\pi'(a_t|\xi_t)-\left(\frac{1}{\alpha}\left(1+\frac{Q^{\text{soft}}_\pi(\xi_t,a_t)}{\Gamma^+_{t-1}}\right)-\ln|Z_\pi(\xi_t)|\right)\right]\end{aligned}$$

$$
\leq \arg\min_{\pi\in\Pi} \mathbb{E}_{a_t\sim\pi}\left[\ln\pi(a_t|\xi_t) - \left(\frac{1}{\alpha}\left(1+\frac{Q^{\text{soft}}_{\pi}(\xi_t,a_t)}{\Gamma^{+}_{t-1}}\right) - \ln|Z_\pi(\xi_t)|\right)\right] \tag{195}
$$

$$
= 0 \tag{196}
$$

as we are always free to select $\pi'(a_t|\xi_t) = \pi(a_t|\xi_t)$ and since the expectation is state-independent by definition of the policy in Eq. (193). We can then rewrite this more clearly with the normalising definition of the value function in Eq. (188) as

$$
\mathbb{E}_{a_t\sim\pi'}\left[\left(1+\frac{Q^{\text{soft}}_{\pi}(\xi_t,a_t)}{\Gamma^{+}_{t-1}}\right) - \alpha\ln\pi'(a_t|\xi_t)\right] \geq \alpha\mathbb{E}_{a_t\sim\pi'}\left[\ln|Z_\pi(\xi_t)|\right] = \left(1+\frac{V^{\text{soft}}_{\pi}(\xi_t)}{\Gamma^{+}_{t-1}}\right) \tag{197}
$$

which defines policy improvement per time step. The equality holds when $\pi'(a_t|\xi_t) = \pi(a_t|\xi_t)$, hence we have the update rule for the soft value

$$
\left(1+\frac{V^{\text{soft}}_{\pi}(\xi_t)}{\Gamma^{+}_{t-1}}\right) = \mathbb{E}_{a_t\sim\pi}\left[\left(1+\frac{Q^{\text{soft}}_{\pi}(\xi_t,a_t)}{\Gamma^{+}_{t-1}}\right) - \alpha\ln\pi(a_t|\xi_t)\right] \tag{198}
$$

The soft Bellman equation in Eq. (189) can be converted using the fact $Q^{\text{soft}}_{\pi}(\xi_t,a_t) = \mathbb{E}_{a_{t+1}\sim\pi}\left[V^{\text{soft}}_{\pi}(\xi_{t+1})\right]$ to yield

$$
1+\frac{Q^{\text{soft}}_{\pi}(\xi_t,a_t)}{\Gamma^{+}_{t-1}} = 1+\frac{r(\xi_t,a_t)+\gamma\mathbb{E}_{\xi_{t+1}\sim\rho^\pi}\left[V^{\text{soft}}_{\pi}(\xi_{t+1})\right]}{\Gamma^{+}_{t-1}} \tag{199}
$$

To show that this can generalise to policy iteration and allows the soft Q-values to monotonically improve we can apply this fact to the Bellman equation in Eq. (199) repeatedly and construct the

$$
\begin{aligned}
1+\frac{Q^{\text{soft}}_{\pi}(\xi_t,a_t)}{\Gamma^{+}_{t-1}} &= 1+\frac{r(\xi_t,a_t)}{\Gamma^{+}_{t-1}} + \frac{\gamma}{\Gamma^{+}_{t-1}}\mathbb{E}_{\xi_{t+1}\sim\rho^\pi}\left[V^{\text{soft}}_{\pi}(\xi_{t+1})\right] \\
&\leq 1+\frac{r(\xi_t,a_t)}{\Gamma^{+}_{t-1}} + \frac{\gamma}{\Gamma^{+}_{t-1}}\mathbb{E}_{\xi_{t+1}\sim\rho^\pi}\left[\mathbb{E}_{a_{t+1}\sim\pi'}\left[\left(1+\frac{Q^{\text{soft}}_{\pi}(\xi_{t+1},a_{t+1})}{\Gamma^{+}_{t-1}}\right) - \alpha\ln\pi'(a_{t+1}|\xi_{t+1})\right]\right] \\
&\vdots \\
&\leq 1+\frac{Q^{\text{soft}}_{\pi'}(\xi_t,a_t)}{\Gamma^{+}_{t-1}}
\end{aligned} \tag{200}
$$

hence $\pi(a_t|\xi_t) \to \pi'(a_t|\xi_t)$ monotonically improves the soft Q-value. The proof that $\pi(a_t|\xi_t) \to \pi^*(a_t|\xi_t)$ to a unique optimal policy is left unchanged from [84, 86]. The arguments discussing the connection of this approach to policy gradients in Section 6 shown in [83] also remain valid.

Therefore, for off-policy learning with a mini-batch using an MSE loss function the critic objectives to be minimised for both dynamics are

$$
J(\theta^+) = \mathbb{E}_{U(\mathcal{D}^+)}\left[\left(R^{(m)}_t + \gamma^m Q^{\text{soft}}_{\bar\theta}\left(s_{t+m},a'\right) - Q^{\text{soft}}_{\theta}\left(s_t,a_t\right)\right)^2\right] \tag{201}
$$

$$
J(\theta^\times) \approx \mathbb{E}_{U(\mathcal{D}^\times)}\left[\left(\frac{1}{\Gamma^{+}_{t-1}}\left(R^{(m)}_t + \gamma^m Q^{\text{soft}}_{\bar\theta}\left(\xi_{t+m},a'\right) - Q^{\text{soft}}_{\theta}\left(\xi_t,a_t\right)\right)\right)^2\right] \tag{202}
$$

where the targets $Q^{\text{soft}}_{\bar\theta}\left(s_{t+m},a'\right)$ are calculated be applying Eq. (198) into Eq. (199). The stochastic policy objectives to be maximised are then also

$$J(\phi^{+}) = \mathbb{E}_{U(\mathcal{D}^{+})}\left[Q_{\theta}^{\mathrm{soft}}(s_t, a_t) - \alpha \ln \pi_{\phi}(a_t|s_t)\right] \tag{203}$$

$$J(\phi^{\times}) = \mathbb{E}_{U(\mathcal{D}^{\times})}\left[\left(1 + \frac{Q_{\theta}^{\mathrm{soft}}(\xi_t, a_t)}{\Gamma_{t-1}^{+}}\right) - \alpha \ln \pi_{\phi}(a_t|\xi_t)\right] \tag{204}$$

with same implications for the differences between the dynamic discussed in Section 5.4 for critics and Section 6 for actors. Overall, we have shown that energy-based policies, specifically the soft actor critic algorithm under the assumption of QDPs can be used to maximise the objective in Eq. (183) with minor modifications. The updating of critic values were approximated using Eq. (189) while the policy update procedure was were left unchanged, this is unlike the requirement for policy gradients in Eq. (167). One key difference to policy gradients that effectively maximise the return from a given action, is that this structure maximises the advantage, namely the difference between the return from a given action and the average return across all actions.

8 Energy Efficient Agent Inference

Real-world environments have near-infinite, but ultimately finite number possible states and can be accurately represented using HDPs discussed in Section 3.8. These are also accompanied by a finite number of known possible actions the agent has control over at each time interval. As discussed earlier, this problem is often intractable and so we generally simplify the system to be a QDP, crudely considered a history-independent POMDP [119].

Furthermore, for the combined results of Sections 5-7 to be valid, one key requirement is the both the state space $\Xi : \mathcal{S} \times \mathcal{R}$ (where recall $|\Xi| = |\mathcal{S}| + 1$) and action space $\mathcal{A}$ are finite. In terms of the cardinality of these spaces $(|\mathcal{S}|, |\mathcal{A}|)$ for locomotive continuous control tasks, they are quite small for the simulations to be practical. The MuCoJo environments [62] tested in [80, 84, 86, 87] reveal how exponentially more difficult training agents become as you increase the both spaces. The more contemporary PyBullet engine [70] interfaced through the OpenAI gym [63] also features a range of $(|\mathcal{S}|, |\mathcal{A}|) = (4, 1) \to (44, 17)$ with the highly non-linear difficulty scaling similarly seen in [167]. The DeepMimic simulations [66] can range from $(|\mathcal{S}|, |\mathcal{A}|) = (197, 36) \to (418, 94)$ and result in far more realistic and natural movements of trained agents with an uncanny resemblance to humans.

A more realistic example would be that of the popular esport video game of StarCraft II wherein the trained agent AlphaStar competes in real-time against human players and achieves a rank exceeding 99.8% of the community [58] . It achieves such success by operating with constraints very similar to that of the reaction time of top-tier human players and vision constraints identical to any human player. Due to the vision of each player being limited to their particular position and a crude 'mini-map' revealing the general positioning only if they have structures or units in a region of the complete map, therefore it is an imperfect game approximated as a MDP. Each state leads to a set of sequential action decisions: action type (several hundred), who to issue action to (units and or structures), where to target, where to observe and act next, and where to move the agents camera view. Ultimately this leads to approximately 10^{26} possible choices at each step. Interestingly, it achieves such high-levels success while having on average fewer actions per minute than humans which is often considered a key performance metric as it is correlated with the amount of multi-tasking a player performs in order to favourably fine-tune each outcome.

All these systems consist of two separate processes. The first is the training of the agent that can occur at a central location with extensive computational throughput and so while increasing learning speed via algorithmic efficiency is highly sought after, it can also be achieved using with superior quality and or larger quantities of hardware. These dedicated supercomputing clusters are notoriously well-known for their high levels power consumption, leading to immense heat generation, and subsequently requiring constant cooling to prevent, at best thermal throttling, or at worst catastrophic hardware failure coupled with extensive damage to components. Next, once these agents have been trained, decision-making involves simply processing the input state and then taking the optimal action according to the trained policy.

Modern reinforcement learning extensively utilised the use of deep neural networks due to their remarkable ability to act as universal function approximator that allow far superior parametrisation compared to the tabular approach used prior [34]. Regardless of network architecture, training involves improving all network parameters using backpropagation against a target value [168–171], evaluation involves forward propagation through up to or less than all the network parameters.

The simplest architecture is a fully-connected feed-forward neural network has known number of inputs I and outputs O, n hidden layers indexed $k = 1, ..., n$ with each layer having h_k nodes, and features the inclusion of a singular bias unit in all but the output layer that are not connected to the previous layer. The total number of trainable weights in this setup are

$$N = \underbrace{Ih_1 + \sum_{k=1}^{n-1} h_k h_{k+1} + h_n O}_{\text{Fully Connected}} + \underbrace{\sum_{k=1}^{n} h_k + O}_{\text{Bias Connections}} \tag{205}$$

Reinforcement learning using the SAC and TD3 agent learning algorithms presented in Appendix B consist of three feed-forward neural networks, of which two are identically structured critic Q-value approximators, and the other is a single actor policy parametrisation. All three networks consist of two hidden layers, but the value and policy networks have differing quantities of inputs and outputs. Using multiplicative dynamics, we have total parameters

$$N_Q = h_1(|\Xi| + |\mathcal{A}| + h_2 + 1) + h_2 + 1 \tag{206}$$

$$N_\pi^\lambda = h_1(|\Xi| + h_2 + 1) + \lambda|\mathcal{A}|(h_2 + 1) + h_2 \tag{207}$$

where $\lambda = 1, 2$ for TD3 and SAC actor policies respectively. Therefore, for each agent, the total number of parameters needed to be trained are

$$\begin{aligned} N^\lambda &= 2N_Q + N_\pi^\lambda \\ &= h_1(3|\Xi| + 2|\mathcal{A}| + 3h_2 + 3) + \lambda|\mathcal{A}|(h_2 + 1) + 5h_2 + 2 \end{aligned} \tag{208}$$

$$N^{\text{SAC}} = 768|\Xi| + 1,026|\mathcal{A}| + 198,658 \tag{209}$$

$$N^{\text{TD3}} = 1,200|\Xi| + 1,101|\mathcal{A}| + 362,702 \tag{210}$$

using the hyperparameters in Table 5. Importantly, training all these parameters can occur at dedicated supercomputers. Once trained, only forward propagation through the policy network is required which is a substantially easier task as it essentially breaks down to matrix multiplication from the input to output layer. This agent receives state $\xi \in \Xi$ and takes actions $a \in \mathcal{A}$ to maximise the reward signal indefinitely. In this case the number of parameters to be utilised for inference are

$$N_\pi^{\text{SAC}} = 256|\Xi| + 514|\mathcal{A}| + 66,048 \tag{211}$$

$$N_\pi^{\text{TD3}} = 400|\Xi| + 301|\mathcal{A}| + 120,700 \tag{212}$$

and so linearly scale with the size of both state and action spaces. For example, inference of the DeepMimic simulations will have parameter count ranges of $134,984 \to 221,372$ and $210,336 \to 316,194$ for SAC and TD3 respectively. While any shallow learning model with this many parameters almost certainly would lead to overfitting, deep learning has remained triumphant in its ability to achieve world-class accuracy [171]. Furthermore, any modern or even somewhat dated computing hardware can effortlessly process these calculations, however, continuous operation will again consumer power, output heat, and naturally demand cooling.

In comparison, AlphaStar has $139 \cdot 10^6$ trainable parameters while only $55 \cdot 10^6$ parameters are utilised for inference [58]. AlphaStar also performed this forward propagation on slightly dated computing hardware that can be purchased from any consumer electronics store. This is a remarkable feat, the ability for a regular computer to defeat professional gamers in arguably the most globally competitive real-time strategy game, heralds far greater implications than the better known successes of AlphaGo and its boardgame counterparts.

Suppose now we are designing agents to operate in environments where repair, part-replacement, or any contact post-deployment is not possible. Furthermore, let's assume the agent is installed on battery power and so the maximum operating time of the device is directly proportional to is built-in power level. There is also interplay with cooling, a device that utilises large amounts of continuous electron flux will require greater heat dissipation than the same device with lower flow. Coolant regulation also may utilise the on-board fixed power such as fans. This is a highly coupled problem, the desire is then to design agents that ultimately consume the least power for the equivalent performance, that is, the best bang for the buck.

The question of measuring energy consumed by a neural network can be simplistically be considered proportional to the number of multiply-accumulate (MAC) operations as a proxy for the number of floating-point operations, combined with the number of weights to model the number of main memory accesses [172, 173]. As the weights have to be loaded from memory, they have a high relative energy cost relative to MAC operations [174]. Therefore, reducing the number of parameters to forward propagate through can reduce power consumption during inference [175–178]. There are numerous methods of doing so such as pruning [174, 176], compression [179], and compacting [180] the architecture.

Therefore, for reinforcement learning agents operating in these extreme environments, by truncating the number of model parameters for inference we can extended their useful life while operating on a fixed battery. While such applications are likely to only emerge in the somewhat distant future once highly practical and fully functioning agents are used throughout the real-world, we believe it is crucial to begin developing the theory given the accelerating progress in reinforcement learning is only going to increase.

Our purpose is also to construct complete end-to-end agents that can operate with zero external inputs. A further key requirement we limit ourselves to however is that the agent must achieve at worst equivalent performance to the 'inefficient' agent while using fewer nodes. Obviously, there may be cases where we may achieve great power savings for slightly reduced performance, these more interesting situations are outside the scope of our analysis.

8.1 Multi-Stage Policy Control

The open question is then in what environments are we able to reduce the number of parameters for agent inference without affecting performance. Using SAC and TD3 in Eqs. (211-212) as examples, we can either reduce $|\Xi|$, $|\mathcal{A}|$, or both. Suppose then all parameters are still required for operation, but not all of them at all times. In such situations assume we are able to segregate environment into p separate phases or stages indexed $\varrho = 1, .., p$.

Consider the original agent requiring inference through N_π parameters, and that the number of parameters required for any stage p are however $N_\pi^p < N_\pi \;\forall\; p$. In what environments would this be possible? This would involve situations where certain states, actions, or both are not required for inference.

A prime example of this in the future would probes for deep space exploration. Contemporarily, their simplified operation involves: Earthly launch through atmosphere, reaching escape velocity, autonomously charting course to destination using $G_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$, deploying probes, and all while communicating back to mission control. This is an extreme environment where the agent has access to a relatively fixed battery as the utility of solar energy will be diminished at large distances.

Extending the operational life of this probes is equivalent to increasing the amount of scientific data they can uniquely provide, unique because the alternative of sending another probe to their location would take time. Another advantage of end-to-end agents would be that at no time would mission control be required to intervene, hence communication can be limited to be solely one-way. It is much easier for a object to target Earth which travels on a fixed trajectory, then it is for us to target a distant probe, further reducing power consumption. This effect would reduce heat output and require less coolant, enabling either lighter vessels or more equipment. Note that the vacuum of space prohibits any heat transfer other than infrared photon emissions.

Given the feature of distinct sequential stages the probe must always adhere to, we can effectively use different agents for each phase. Furthermore, for simplicity we consider only irreversible transitions to the next stage with no possibility of the reverse. One agent can govern take-off, another controls upper atmosphere manoeuvrers, the next manages the journey, and the final organises the delivery of probes to their optimal destinations. This procedure could also be adopted for other *more explosive* applications.

Regarding states and actions, each phase is likely to share common features while also containing unique variables. For example, states and actions pertaining to launch tied to events such as wind, weather, and atmospheric composition, are not necessary after this phase. This segregation of responsibilities is a somewhat trivial problem, but complexity arises on how to teach the agent to learn to recognise the transitions.

One of the major challenges in applying reinforcement learning to the real-world has to do with designing of an accurate reward function or mechanism [34]. The agents' singular goal is to maximise this value, hence how it is parametrised according to a particular environment state is the basis of its value. Many tasks naturally to provide clear reward signals, for example, for stock prices we can simply calculate the difference between states as the changes in price and hence profit, allowing us to design agents that maximise the portfolio value of the securitises (states) it has access to control. Other tasks are not nearly as simple, something as straight forward as catching a ball has myriad of possible reward functions with it not being clear how to quantitatively distinguish between subtle actions.

For our purposes, we therefore assume an accurate reward function also exists as learning when to transition to phases will be entirely contingent on the total cumulative reward. Considering only environments where state-action tuples can be separated into distinct phases, a general framework for constructing all manner of guidance systems from simple goal-based systems to far more complex and intelligent tools is needed in order to make the problem tractable and or efficient by isolating the relevant state-action spaces.

We present a model for building a three-stage agent with distinct (reversible) phases: standby, active, and engage. We call this prototype method SARAE^n. The acronym is for state-action-reward, then if reward meets a threshold, the agent initiates an 'active' phase (distinct second set of state-actions) which directly leads to a engage phase (distinct third set of state-actions). The $n = 2$ index denotes that the threshold creates only two different decision loops, the $n = 1$ case reduces to the usual existing formulation. This naming is clearly a modification of the well-known SARSA algorithm [34]. Applications involve rockets for deep-space exploration and atmospheric entry, self-driving cars, and items that have exponentially larger potential to explosively change the landscape. The blueprints and schematics for this algorithm will be reveal sometime in the future with an outlined application presented in Appendix C.6. The key note is on how the agent is able to determine what stage it is in and able to take optimal actions for this situation.

9 Related Work

There are numerous areas where existing research has been done and served as inspiration for our results. Most of these related areas were discussed in their corresponding Sections 2-3. In this brief section we provide a succinct review of theses motivations and the locations where our work aims to address.

There appears to be no prior work in the area trying different critic loss functions other than the standard MSE in reinforcement learning. Whether it be model-based or model-free algorithms, everyone appears to use Bellman (MSE) error with the authoritative [93] stating in 2006 that similar results are to be expected for other loss functions. Our analysis imagines the use of different losses to be analogous to the penalty applied to an agent when it makes a perceived error in valuing a given state-action pair. An a large outlier in this case is interpreted to be large 'mistake' relative to the rest of the mini-batch, this outlier then correspondingly forms a larger portion of the aggregated loss to be minimised, and so the agent places more emphasis on adjusting its future parameters to minimise the impact of the same outlier occurring again. Recall this perceived error is nothing more than an artifact of Eqs (28). The target is simply composed of the current known reward and the discounted current Q-value that utilising delay parameter updates, it is essentially a moving target, unlike the fixed true known values that exist in NMF. The question is then whether Eqs. (41-47) that offer different degrees of outlier smoothing change the final result. While all these have been thoroughly tested in NMF with outstanding success [94, 133–138], there does not appear to exist conclusive results in reinforcement learning. The use of stronger functions will therefore be an empirical test of whether relatively large mistakes are important to learning.

On the topic of shadow means there is nothing but emptiness in the literature regarding its utility in not just machine learning, but all of statistics [14, 95, 96]. The utility of this approach for practical purposes has yet to be tested, since as it stands, it is a theoretical construct for estimating the true population mean for a fat-tailed distribution. Its estimation can be split into two components. First we need to estimate the tail coefficient $\hat{\alpha}$, and if $\hat{\alpha} < 1$, the shadow mean estimate can be used. There is ample literature on tail estimation expressed mainly in the field of Extreme Value Theory (EVT) [140]. As discussed in Section 3.1, of the plethora of choices, one is to determine the appropriate intermediate order statistic either using the Hill estimate [146], or more advanced [140, 147, 148] method of moments or MLE techniques. Another avenue is to directly observe the gradient of a Zipf plot of certain number of descending order statistic [149–152]. EVT is a mature field but is yet to offer an objective approach to not only estimating $\hat{\alpha}$, but even agreeing on the validity of some of these approaches. In our case, the first test is whether the estimated $\hat{\alpha} < 1$ which will indicate critic losses are fat-tailed and have no true mean when modelled using GPDs, and then whether the shadow mean an be utilised by the agent to achieve greater success. Therefore, while [14, 95, 96] have shown it to be useful in redefining the way we examine extreme events, applications to reinforcement learning and most of statistics are non-existent.

Multi-step returns have a very long history of being used to accelerate agent learning, but so far only for discrete action spaces [34, 48, 71, 101, 153, 154]. Whether they function similarly for off-policy continuous action domains is not well-known in the history of reinforcement learning. Unlike discrete domains where the Q-value is explicitly used to determine the next action via ϵ-greedy polices, for TD3 and SAC, the Q-values are used to update the policy usually in the direction of maximal value. This is similar to the discrete case but due to the addition of large amounts of noise added across the mini-batch it is unclear what the final outcome will become. One of the scarce analysis of this situation is presented in [154] where they incorporate multi-step returns into DDPG and find better final performance and learning speed for all multi-steps compared to the usual single-step. They also determine its performance for certain environments becomes comparable to the state-of-the-art TD3 algorithm.

Next, the exponentially more interesting phenomenon is the coupling of multi-step returns with the size of the experience replay [101] being an extremely puzzling occurrence. It is incredibly difficult to formulate a well-grounded theoretical explanation for this effect as there is no clear reason why tuning for m-steps and size of the buffer would improve performance across the board.

There has also been zero attempt to incorporate true multiplicative dynamics into reinforcement learning. While the reformulation of contemporary decision theory is discussed in detail in [5–11, 14–16, 102–107], designing of reinforcement learning algorithms that can fully autonomously self-learn these principles is unknown.

As discussed in Section 8.1, one of the challenges with reinforcement learning is designing accurate reward functions. One environment that naturally lends itself well to this requirement is the trading of financial assets. This is because the reward for a holding a position in a particular asset per step can be calculated as the difference between prices, or in other words, the difference between states. Applications to finance are discussed in [40, 41]. What is common in these approaches is the utilisation of Q-values and the Bellman equation to maximise the absolute dollar reward per step. In [40], DQN is used to maximise positions in the Russian equity index using five states to describe it at any time (OHLC and volume) with three corresponding discrete actions (buy, hold, and sell) and so the reward. What is common in these types of environments are: usually only a single asset universe is simulated, rewards are absolute, and historical prices are used (which can lead to overfitting).

What we know however is that wealth is multiplicative and compounds over time, hence changes should always be represented as rates of returns as discussed thoroughly in Section 1. The purpose of the agent should then be to maximise the long-run geometric average growth rate of wealth. On top of this, the real-world consists of countless tradeable assets and so the agent must learn to dynamically take positions in any of these assets at any point in time with simulation using geometric Brownian motion possibly assisting in constructing far more robust agents. These are environments where we require the asymptotic reward scaling condition in Eq. (100) to universally hold. To the best of our knowledge there does not exist any publicly available open-source environments that admit this sort of structure. Therefore, construction of these environments is imperative.

Overall, of the four areas of investigation, none have been thoroughly examined, most have never been considered. Analysis of different critic losses interpreted as whether large mistakes are important for learning does not appear to have been considered in detail. Shadow means and tail exponents appear to vacant in not only machine learning literature but are usually omitted from the general study of statistics despite most distributions in the real-world being fat tailed. Multi-step returns have been thoroughly analysed for discrete action spaces, yet there does not appear to exist enough analyses for continuous action spaces that are far more relevant to the real-world. Multiplicative dynamics is non-existent in majority of all discussions conducted on Earth even though it is the definition of how wealth, health, and life evolve for any random individual or institution.

10 Additive Experiments

This section contains experimental results pertaining to the investigations contained in Section 3. We analyse different critic loss functions, shadow means, and multi-step returns using model-free off-policy algorithms in continuous action spaces. These will be conducted on four robotic locomotive control tasks from the Roboschool environments imported to the PyBullet engine [70] interfaced through the OpenAI gym [63] framework. These environments represent much more difficult task while being more computationally efficient compared to the MuCoJo environments [62] used throughout the last decade [70]. They also present several other advantages such as being open-source, free to use, and continue to receive regular updates. The downside is that the results we obtain are not directly comparable to the majority of already published literature, but they will be perfectly comparable to all future results such as those in [154, 167] and so we consider this choice appropriately forward-looking. First succinct a overview of both the SAC and TD3 algorithms used for our purposes is presented, followed by the experimental results for the three research areas coupled with discussion of the results.

10.1 Algorithms, Resources, and Environments

The primary tools of this work, the SAC and TD3 models are presented in Algorithms 1 and 2. In Table 5 we provide exact specifications for the myriad of internal hyperparameters for each algorithm. Furthermore, to enhance readability we have omitted some components such as checking for selecting of shadow mean limits, multi-step limits, number of trials, conducting the evaluation episodes, warm-up steps for random initialisation seeds, and the intricacies of done flags for starting new episodes.

Observe again that TD3 has three additional hyper parameters controlling the nature of exploration noise. We also generalise the sampling distribution to be $\mathcal{S}$ that is usually set to be Gaussian but can also easily be Laplace. Laplace would result in far more frequent sampling of values closer to the mean and can be interpreted as being 'more' deterministic. It would be interesting to test this unbounded fixed distribution to examine what policies SAC learns. Overall, the presented modified versions of TD3 and SAC are capable of answering both our questions regarding varying critic loss functions $L(Q_{\bar{\theta}}, Q_{\theta})$ shown in Eqs. (41-47) and multi-step targets with $m \geq 1$.

For shadow means in Section 3.1 we take the simple approach of Eq. (72) using the entire mini-batch. In this case if $\hat{\alpha} < 1$ we can crudely confirm at the very least that the sample probably does not have a true finite mean without any comment on whether the shadow mean representation in Eq. (64) is a valid. If the shadow mean estimate is not be suitable for learning, that does not imply use of the global standard empirical mean in Eq. (51) is appropriate. For $\mu_s(L_i^*, H_i, \hat{\alpha}_i)$ we will set L^* as the lowest value and H as ten times largest value in the mini-batch respectively.

In terms of hardware, we use a local prototyping environment and a high-performance computing (HPC) cluster. The local setup involves AMD Ryzen 7 5800X, Nvidia RTX 3070, 64GB RAM, and a Samsung SSD 980 Pro. The Artemis HPC cluster is utilised, specifically, numerous Nvidia V100's to perform multiple experiments in a parallel manner [2]. Regarding software, the local environment was tested using Pop!_OS, Ubuntu, Arch, Windows, while Artemis uses CentOS 6.9. All code is written using the Python programming language with local and HPC using versions 3.9.9 and 3.8.2 respectively. The deep learning algorithms are executed using PyTorch [181] (with CUDA [182]) versions 1.9.1 (CUDA 11.1) locally and with Artemis requiring compilation of version 1.9.0 (CUDA 10.2) using MAGMA 2.5.3 [183–185] due to its GNU C Library (glibc) being outdated. The additional Python packages needed are detailed in the requirements.txt file available on GitHub [1].

Name	Environment ID	States $\lvert\mathcal{S}\rvert$	Actions $\lvert\mathcal{A}\rvert$	Warm-up Steps
Hopper	HopperBulletEnv-v0	15	3	10^3
Walker	Walker2DBulletEnv-v0	22	6	10^3
Cheetah	HalfCheetahBulletEnv-v0	26	6	10^4
Humanoid	HumanoidBulletEnv-v0	44	17	10^4

Table 1: Additive environments to be tested using PyBullet [70]. Warm-up steps represents the number of initial training steps to consist of purely random actions in order to encourage robust learning from different initial conditions given the replay buffer experiences sampled.

We will examine four PyBullet environments outlined in Table 1 that represent increasing levels of difficulty. For all experiments, in the language of Section 3.9, we conduct $M = 10$ randomly initialised trials of $3 \cdot 10^5$ training steps with $N = 10$ evaluation episodes each occurring at every $1,000$ training steps for each algorithm. This results in 100 estimates for the reward the agent achieves in the environment at each interval. In all cases, the mean and MAD about the mean of these values will be plotted with all values within these regions treated as a within error and indistinguishable.

Loss parameters for all trials are also aggregated including 10 pairs of twin critic losses, twin Cauchy scales ω, twin CIM kernel sizes σ, 10 learned SAC entropy temperatures, twin tail exponents $\hat{\alpha}$, twin shadow means of critic losses μ_s, and twin equivalence multipliers κ_{eqv}.

The Cauchy scales ω in Eq. (48) can be interpreted as being proportional to the harmonic mean of the mini-batch critic losses, large increases in this variable imply less smoothing and presence of outliers. The CIM kernel sizes σ in Eq. (49) are the (population) standard deviation of the mini-batch critic losses. The entropy temperatures for SAC are all initialised at unity and indicate the amount of state-dependent entropy (noise) added to each reward in order optimise learning. Lower values indicate less noise and is more in line with traditional Q-learning with deterministic polices similar to TD3, while higher values generally imply the algorithm is using more stochastic polices where the agent requires more noise to successfully explore potentially superior different routes. The indexes $\hat{\alpha}$ from Eq. (74) are inversely proportional to the level of kurtosis where smaller values indicate fatter tails. The shadow means in Eq. (75) attempt to infer a more accurate critic mean under fat tails. The equivalence multiplier κ_{eqv} in Eq. (77) indicates what multiple of the empirical mean set to be the finite improbable upper bound estimate for the shadow mean is required for equvilence with the empirical mean.

There are several limitations to our experiments:

- Total number of training steps is limited to only $3 \cdot 10^5$. This is found to be reasonable for our purposes, but definitive claims cannot be made until millions of steps are carefully examined. Furthermore, the number of evaluation episodes could be increased from 10 to a larger number for more accurate aggregation.

- We are unable to train the agent to learn using the shadow mean in Eq. (75) as the derivative of the gamma function is not yet implemented in PyTorch 1.9. Hence, we are unable to backpropagate the critic shadow losses to train the agent. Implementing this feature is outside the scope of this analysis and so our coverage of shadow means will be confined only to their value relative to the empirical means.

- For multi-step returns we only train with only odd bootstraps, while it is unlikely there is any interesting behaviour limited strictly to even multi-steps, these will not examined. Additionally, the coupling of multi-step returns and the experience replay buffer is not analysed given the astronomical computational requirements.

10.2 Empirical Critic Losses

For much of the experiments, the figures speak for themselves and describing every plot is not conducive to enhancing understanding. Instead, we provide a general discussion and the conclusion based on the results. For all environments, in Figs. 19-23 we plot the results from the evaluation episodes and the training parameters using critic loss functions from Eqs. (41-47).

Focusing first on the outlier smoothing loss functions in order of increasing strength: MSE, HUB, MAE, HSC, CAU and TCAU. We instantly observe the strongest functions CAU/TCAU have the weakest performance in terms of score for both algorithms. For this reason, we omit plots of CIM as it yields even worse results, generally no evidence of any learning. As expected, due to outlier suppression, the empirical critic means for MSE were an order of magnitude larger, while surprisingly, CAU/TCAU was quite large despite its logarithmic dampening.

The Cauchy scales for TD3 are fairly indistinguishable, while for SAC, CAU/TCAU contains noticeably larger outliers within the mini-batch. The real story behind the underperformance of CAU/TCAU is told through its orders of magnitude larger kernel sizes indicating huge relative inhomogeneity throughout the mini-batch. Given that these elevated levels of variance persist, it indicates that the suppression of early outliers leads to their persistence in the future. This leads to a negative feedback loop where the agent continues to make mistakes as it was not able to properly correct them earlier. Furthermore, by taking the (equal-weighted) arithmetic means across the mini-batch for backpropagation we are inappropriately placing equal emphasis on all losses, when in fact we should be targeting the ones disproportionately responsible for the volatility. More evidence is presented in the SAC entropy temperatures wherein CAU/TCAU requires significantly higher action-dependent noise injection to learn, implying more exploration and less exploitation compared to the other functions. TCAU however is seen to be noticeably worse which leads us to conclude that truncating large mistakes is determinantal to learning.

Score performance, Cauchy scales, and entropy temperatures of MSE, HUB, MAE, and HSC all appear to be within the margin of error, with no clear winner across all environments. Their critic losses are all less than MSE which is to be expected, while kernel sizes appear elevated. The reason for the latter is unclear, but the suggestion is that agent learns to correct large (amplified) MSE errors more quickly due to their implicitly higher weighing in the mini-batch, leading to greater homogeneity within the mini-batch.

Next, we focus on outlier amplifying functions in order of increasing strength: MSE, MSE2, MSE4, and MSE6. We observe deteriorating score with increasing intensity. The critic losses are also clearly larger with more strength while the Cauchy scales also increase. The kernel sizes are very interesting as variance for higher powers appear not to be distinctly larger than MSE throughout the experiments. This is unexpected as one would expect higher variance due to mini-batch containing much more amplified errors. Ending entropy temperatures also increase with amplification which can be interpreted as suggesting the agent requires the greater penalty for optimising the advantage function.

Ultimately, we can derive several conclusions, large mistakes are crucial to long-term learning but greatly amplifying them reduces performance. Hence, choice between MSE, HUB, MAE, and HSC is yet another addition to the already countless hyperparameters that need to be specified to the agent for any given environment.

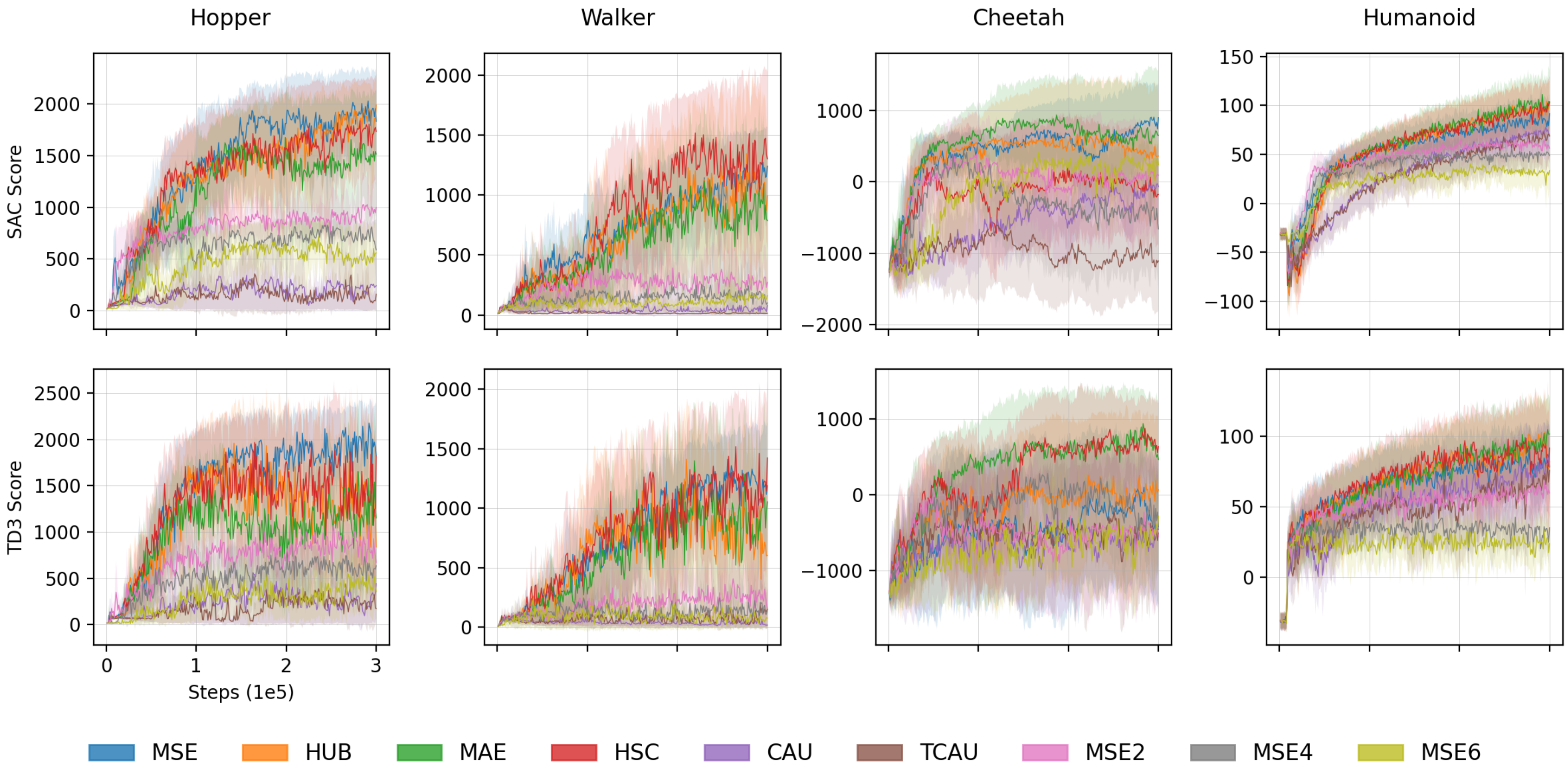

Figure 11: Evaluation episode scores for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

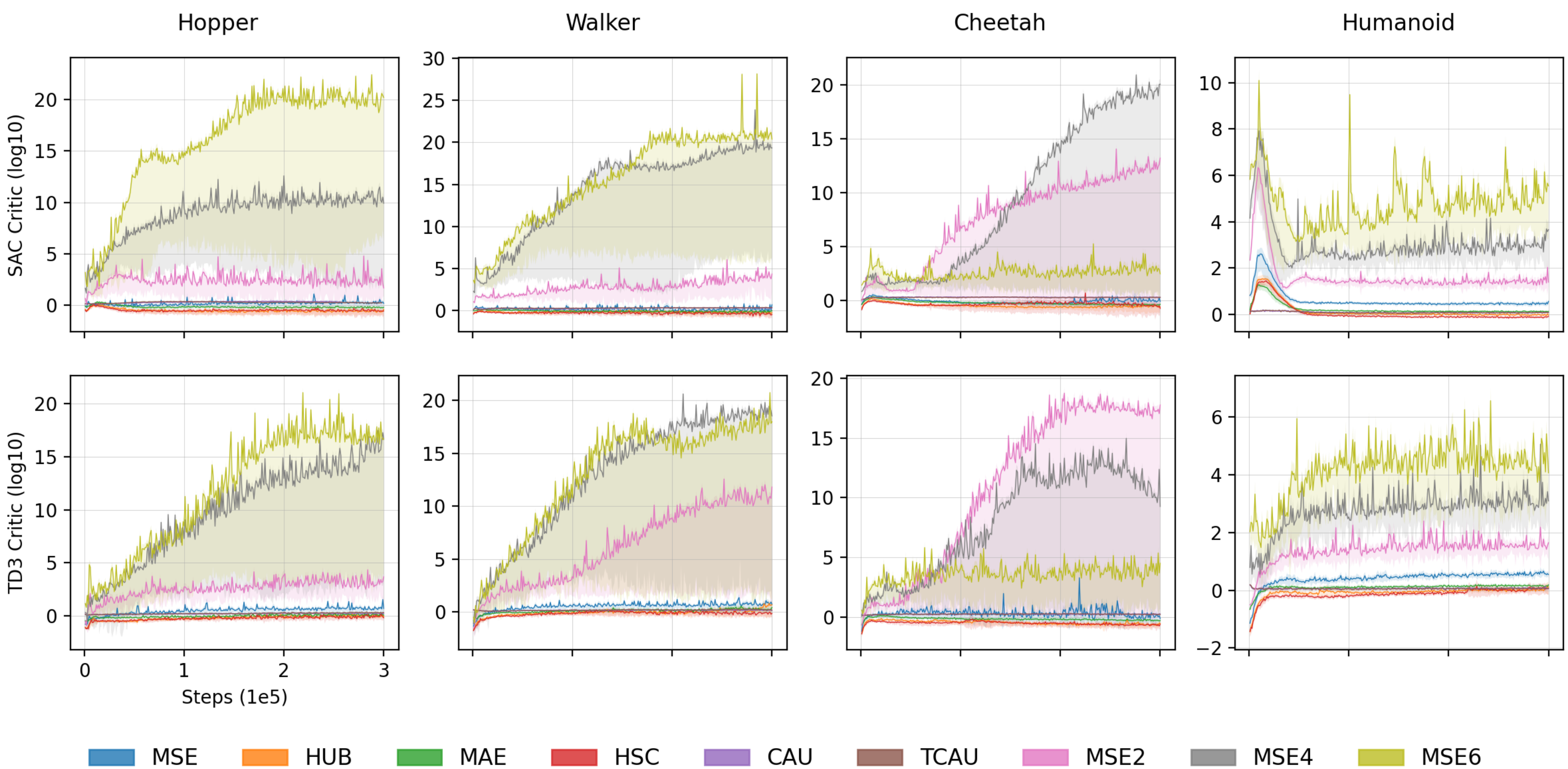

Figure 12: Empirical mini-batch (arithmetic) mean critic losses for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

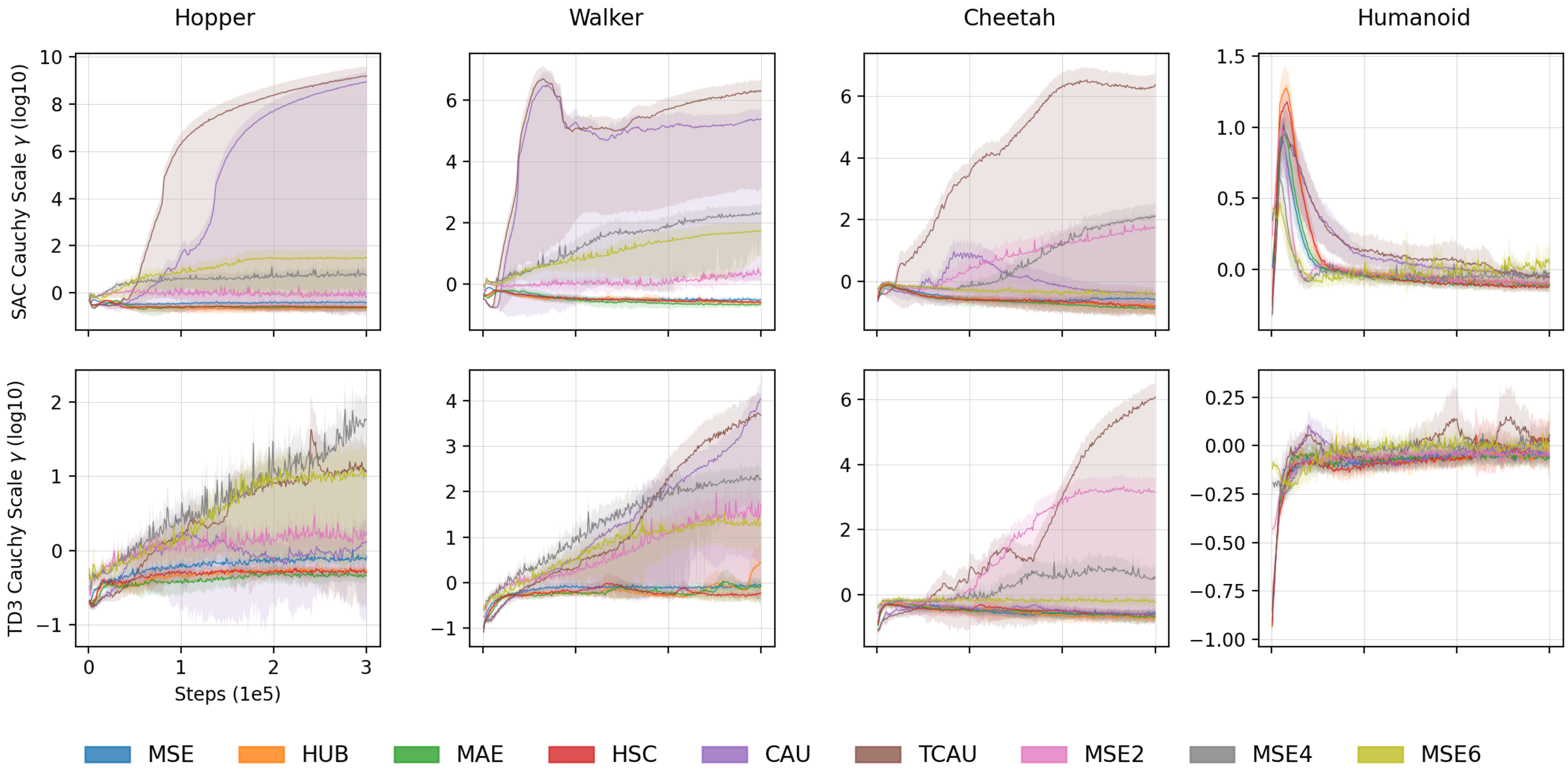

Figure 13: Cauchy scale parameters for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

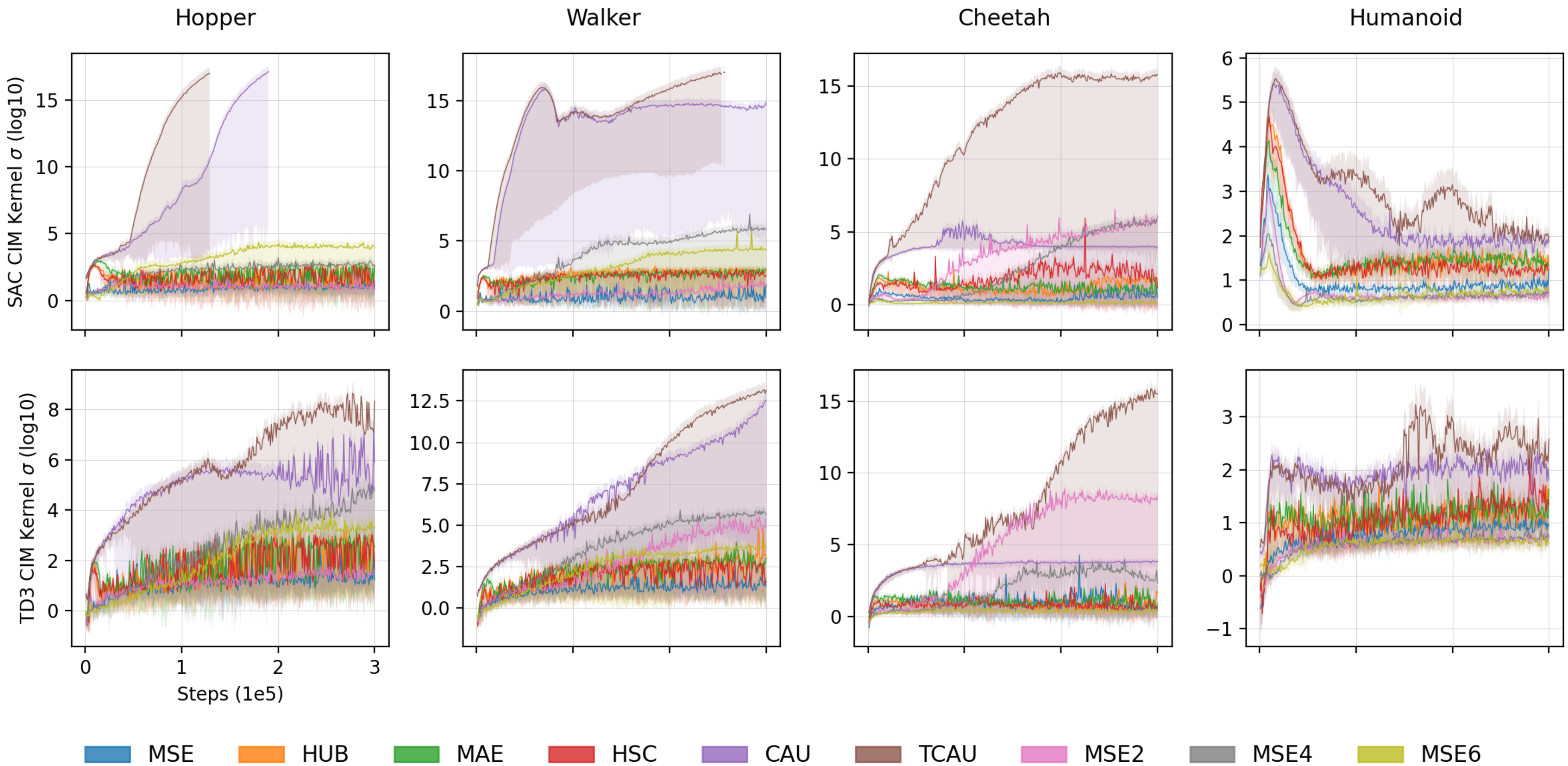

Figure 14: CIM kernel sizes for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean. Note missing values indicate divergence to infinity.

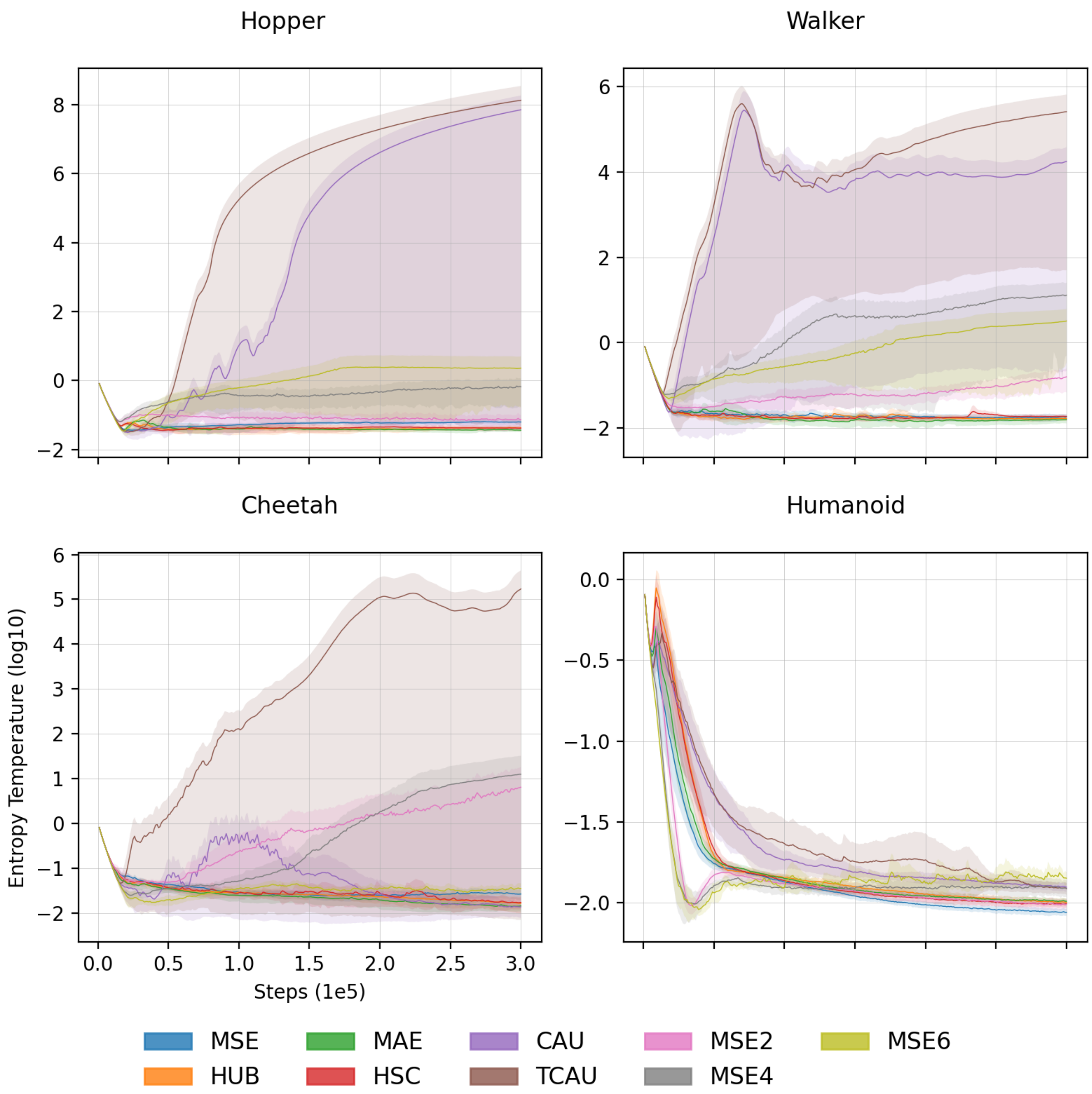

Figure 15: Entropy temperatures for various critic loss functions across additive environments for SAC algorithm. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

10.3 Critic Shadow Means

Continuing with the previous discussion, analysis of the tail structure of the mini-batch critic losses are presented in Figs. 24-26.

We focus on critic loss smoothing functions in order of increasing outlier suppression: MSE6, MSE4, MSE2, MSE, HUB, MAE, HSC, CAU, and TCAU. The tail exponents for all are below unity throughout training. We also receive confirmation that MSE2, MSE4 and MSE6 correctly amplify errors relative to their average value given their miniscule exponents. This astoundingly implies that as the mini-batch size approaches infinity, the sample would not have a formally defined mean under the GPD assumption. The agents' learning of Q-tables appears to be an incredibly fat-tailed phenomenon and aggregating mini-batch losses 'empirically' likely leads to severe underestimation of the true scale of errors made. The presence of extreme values should therefore not be equally weighted, rather far greater emphasis should be placed on reducing them as their impact on the sample is far more determinantal. Furthermore, recall also that to construct tail estimates we have used Zipf plots containing all ordered statistics, using fewer would most likely lead to even "fatter" tails.

The very little-known world of extreme value theory appears to be far more relevant to reinforcement learning than anyone could have ever conceived prior to seeing these results. Overall, training agents with empirical mean is likely not optimal given their severe underestimation of the true (unknown shadow) mean of the sample.

Our shadow mean estimates are inversely proportional to the tail exponents and reveal the possible scale of the population means. In call cases, they are a few orders of magnitude larger while are of identical structure to the empirical means which is to be expected. Directly training the agent with these substitute estimates is not currently feasible but the results would certainly be fascinating. We can however investigate the equivalence multipliers where the maximum theoretical upper (infinitesimally probable) estimates of the underlying distributions would need to be as multiple of the empirical mean for both means to be equal. For outlier suppressing functions we find a few orders of magnitude larger estimates are required. Therefore, given that we have previously seen very large fluctuations, the choice of upper bound should be far more improbable, and hence much larger that the empirical mean.

Overall, these results let us claim that while the mini-batch clearly should not be aggregated using empirical means, we do not currently have a functioning alternative that can be technically implemented. Regardless, the existing shadow mean formulation generates estimates that are likely far more indicative of the true nature of the agents' errors.

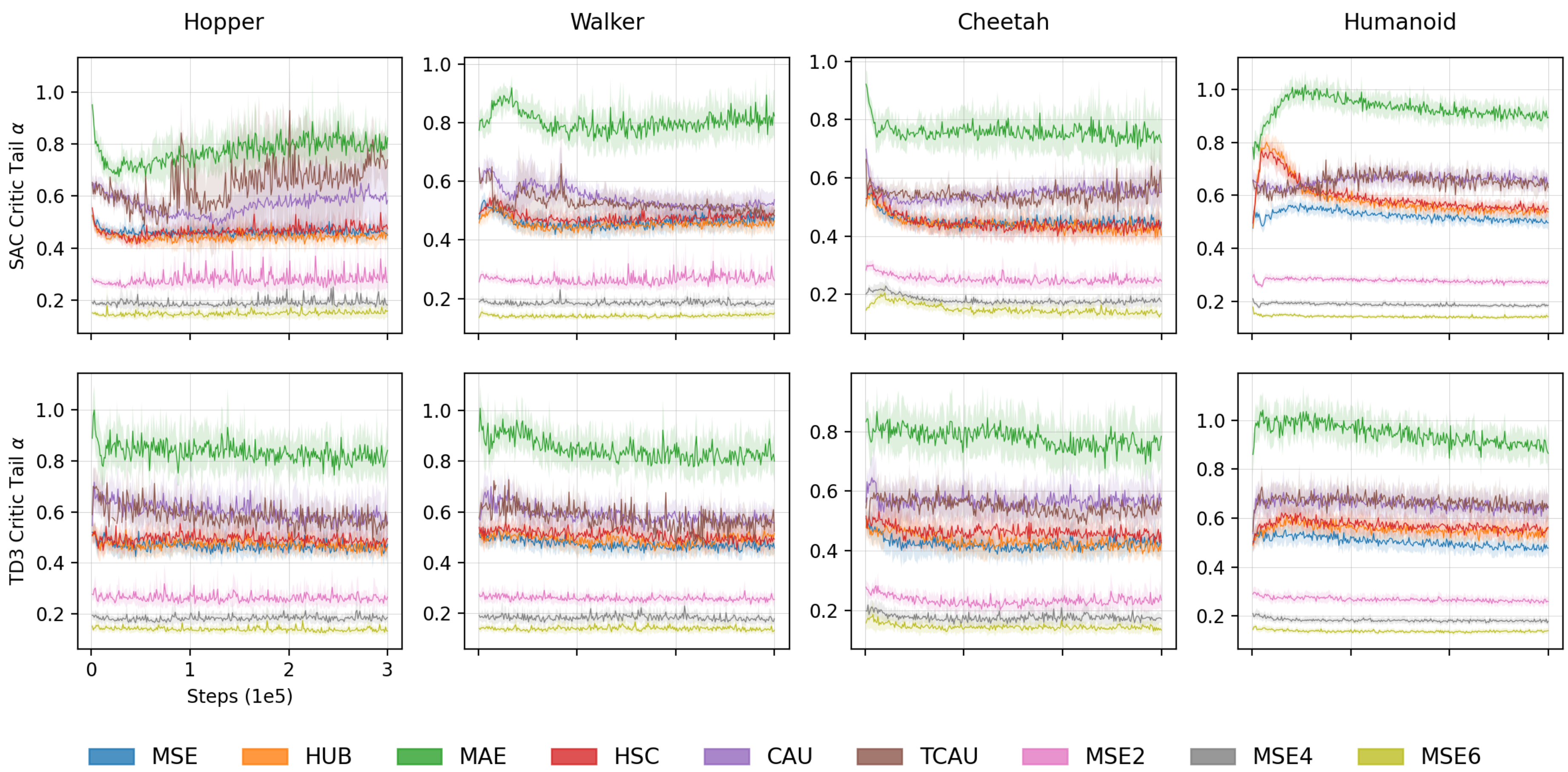

Figure 16: Tail exponents for mini-batch critic losses for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

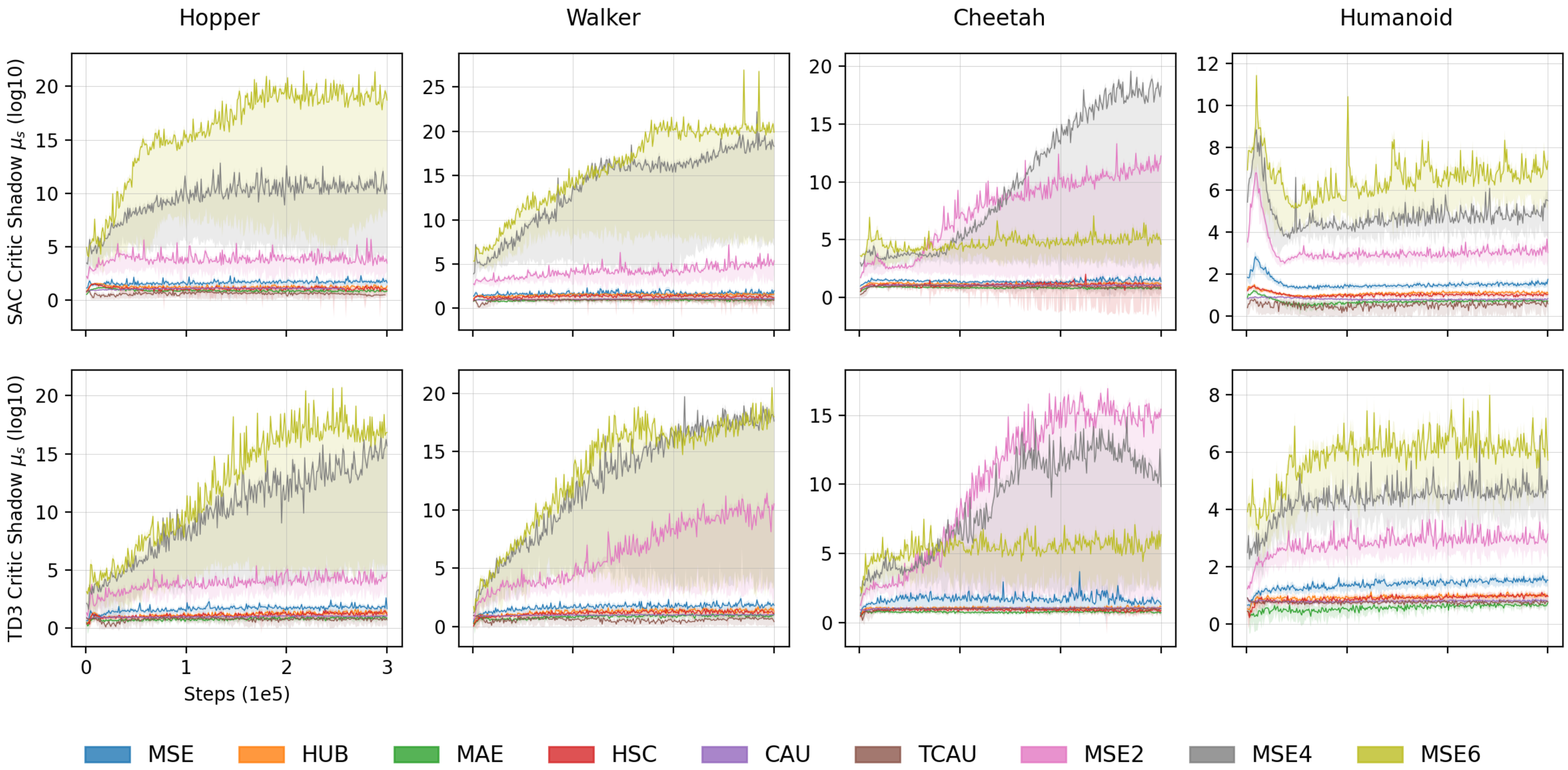

Figure 17: Shadow mean estimated for mini-batch critic losses for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

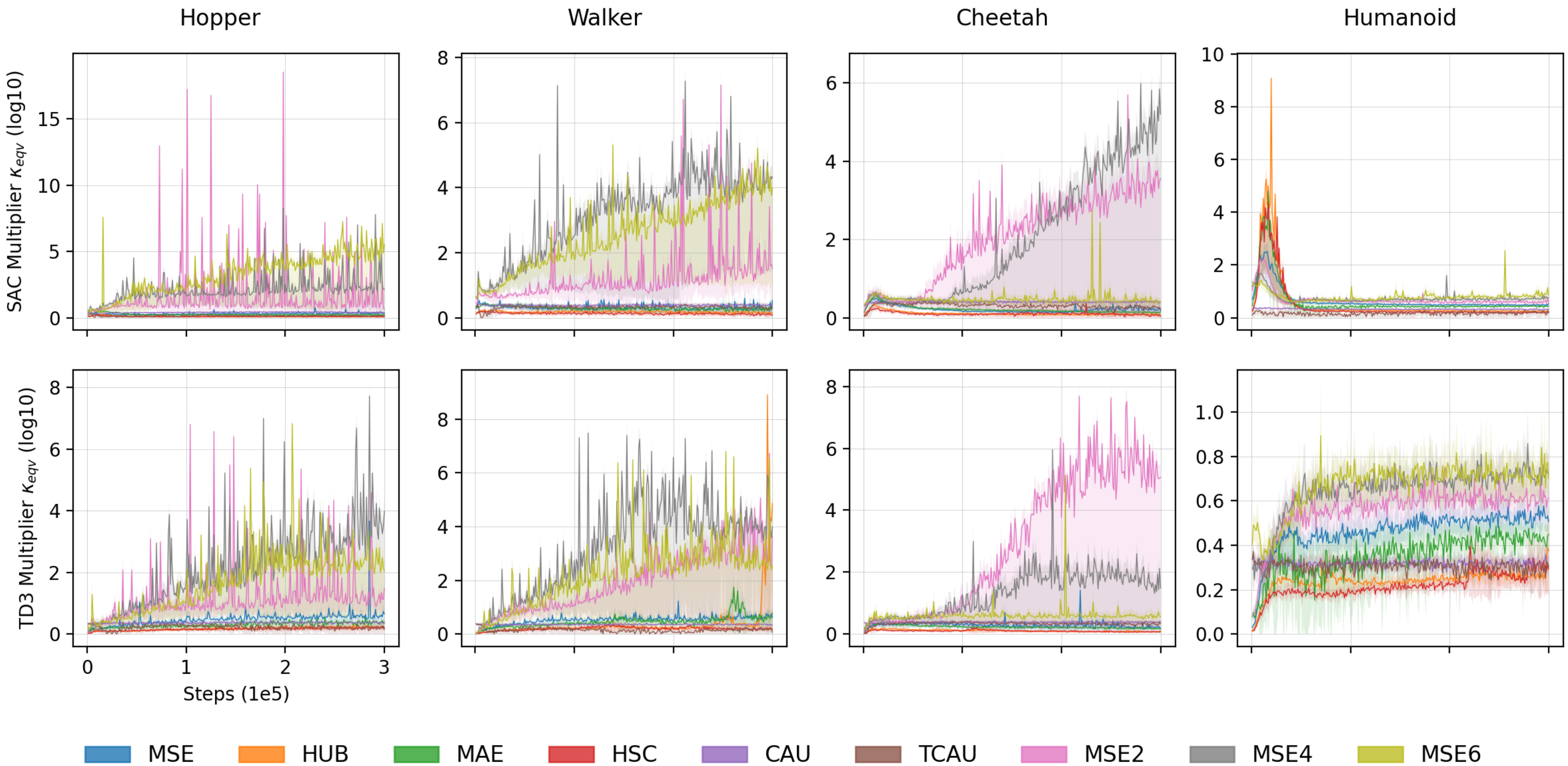

Figure 18: Equivalence multipliers for empirical and shadow means of mini-batch critic losses for various critic loss functions across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

10.4 Bootstrapping Targets

The unexpected findings on multi-step return for odd $m = 1, 3, 5, 7, 9$ in continuous action spaces using MSE critic loss function for TD3 and SAC are shown in Figs. 19-26.

We observe the agent is either incapable of learning or does so at an extremely diminished rate for $m > 1$ for most algorithms and environments. For validation, the results for $m = 1$ are confirmed to be in line with those in [154, 167]. In [154], they also show results for $m > 1$ for DDPG and find it is almost universally superior to the single-step case. Unfortunately, they neither provide error bounds, or an open-source codebase. Furthermore, their findings appear to be extremely ideal to convey their message as it is very interesting to observe that every single novel method they present is superior to the standard DDPG. Overall, the difference between our TD3 and their DDPG results should not be so vast.

One possible explanation for this discrepancy that allows us to heavily improve our TD3 agent performance involves purposefully making error in the bootstrapping procedure. When reconstructing the agents history from the replay buffer in Eq. (79), if were to utilise the current Q-value estimate $Q_\theta(s^{(1)}, a^{(n-1)})$ instead of $Q_\theta(s^{(1)}, a^{(1)})$, that is, consecutive actions are used while states are multi-stepped, the agent is able to learn far more successfully. We do not present these results as they are formally incorrect but serve to highlight a very interesting phenomenon.

It also appears that as you increase the complexity of the environment, multi-steps tend to perform better relative to no bootstrapping. This may be due to a combination of an intrinsic advantage of multi-steps and reduced performance of the $m = 1$ case. This reveals that while multi-steps are functional, they should only really be considered for advanced situations.

Recall that unlike discrete action spaces where each action has finite number of unique positions, continuous action have technically infinite unique values. Due to this difference, straightforward ϵ-greedy policies based on argmax of Q-values that lead to global maximisation are no longer feasible. Instead actor-critic methods with deterministic policy gradients are used that update the policy $\mu_\phi(s)$ in the direction of maximising the Q-value $Q_\phi\left(s^{(1)}, \mu_\phi(s^{(1)})\right)$ (or similarly with soft actor-critic advantage maximisation). The Q-values estimates then continue to be updated through the minimisation of $\left(\sum_{k=0}^{m-1} \gamma^k r_{t+k} + \gamma^m Q_{\bar{\theta}}\left(s^{(m)}, a'\right) - Q_\theta\left(s^{(1)}, a^{(1)}\right)\right)^2$. Ultimately this approximation works well for single-step $m = 1$ learning. It is conceivable that as $m \to \infty$, the error in Q-values makes θ parameter updates less accurate, which then leads to inferior ϕ updates given the policy is not a global maximisation. This leads to a negative feedback loop over time given that policy optimisation is not an exact process. The evidence is seen in the scores since the $m > 1$ cases clearly display increasing levels of learning over time but generally do so at an extremely reduced pace.

This hypothesise is further validated by examining the mean critic loss backpropagated by the agent. We observe the $m > 1$ cases have many orders of magnitude larger values implying huge inaccuracy in Q-values (Bellman errors). Note also that higher errors are also to be expected due to the presence of the cumulative historical reward $\sum_{k=0}^{m-1} \gamma^k r_{t+k}$ term rather than the usual singular r_t reward. The geometric dampening $\gamma^m Q_{\bar{\theta}}\left(s^{(m)}, a'\right)$ also likely contributes to reduced learning given less direct emphasis is placed on improving the Q-values. Importantly, recall again that the critic losses for the same sampled experience should theoretically be identical across all bootstraps by the Bellman equation, with massive differences then suggesting unsuccessful learning.

The Cauchy scales and kernel sizes also appear to increase as you increase bootstrapping given again that they contain a larger portion of fixed known values while the estimated target Q-value that is prone to error is heavily discounted. The behaviour of entropy temperature is also very interesting since they all begin rapidly declining as per usual, though larger m rapidly begin to increase again. This implies that when attempting to maximise rewards, the SAC agent requires far greater exploration noise for larger m to learn policies compared to lower m. Once again, this can be explained by the fact that heavily discounted future target Q-values form a reduced portion of the combined critic loss, and so its impact on learning how to accurately value states are lessened. Hence, we can state that to counter this lower signal, the agent is forced to add additional entropy to rewards to simulate artificial volatility in valuations.

We find that the tail exponents for all m-steps are indistinguishable to each other and all below unity. Overall, this is a good sign as it adds further weight to treating empirical means as underestimates as it reveals that the underlying distribution $\forall\ m$ are of similar structure in terms of the extent of the extreme values which is exactly what we would expect as they are meant to be identical. The higher shadow mean estimates naturally lead to larger equivalence multipliers for larger bootstraps.

Therefore, learning with multi-step targets appears to be a disadvantage in continuous action spaces which is at odds with the closed source results presented in [154]. We hypothesise that since policy optimisation is not globally performed over the entire action space as this is computationally not feasible, the agent is unable to learn at the same pace as the single-step case. This problem is further exacerbated by the geometric dampening of target Q-values causing reduced emphasis on improving the Q-tables directly. While multi-step returns have the potential to rapidly accelerate learning with discrete actions, this is an exponentially simple task since the existence of a finite number of possible unique positions permits easier policy parametrisation. The continuous case, by definition, has infinite unique positions with their implications to maximising the reward in an environment being far more delicate.

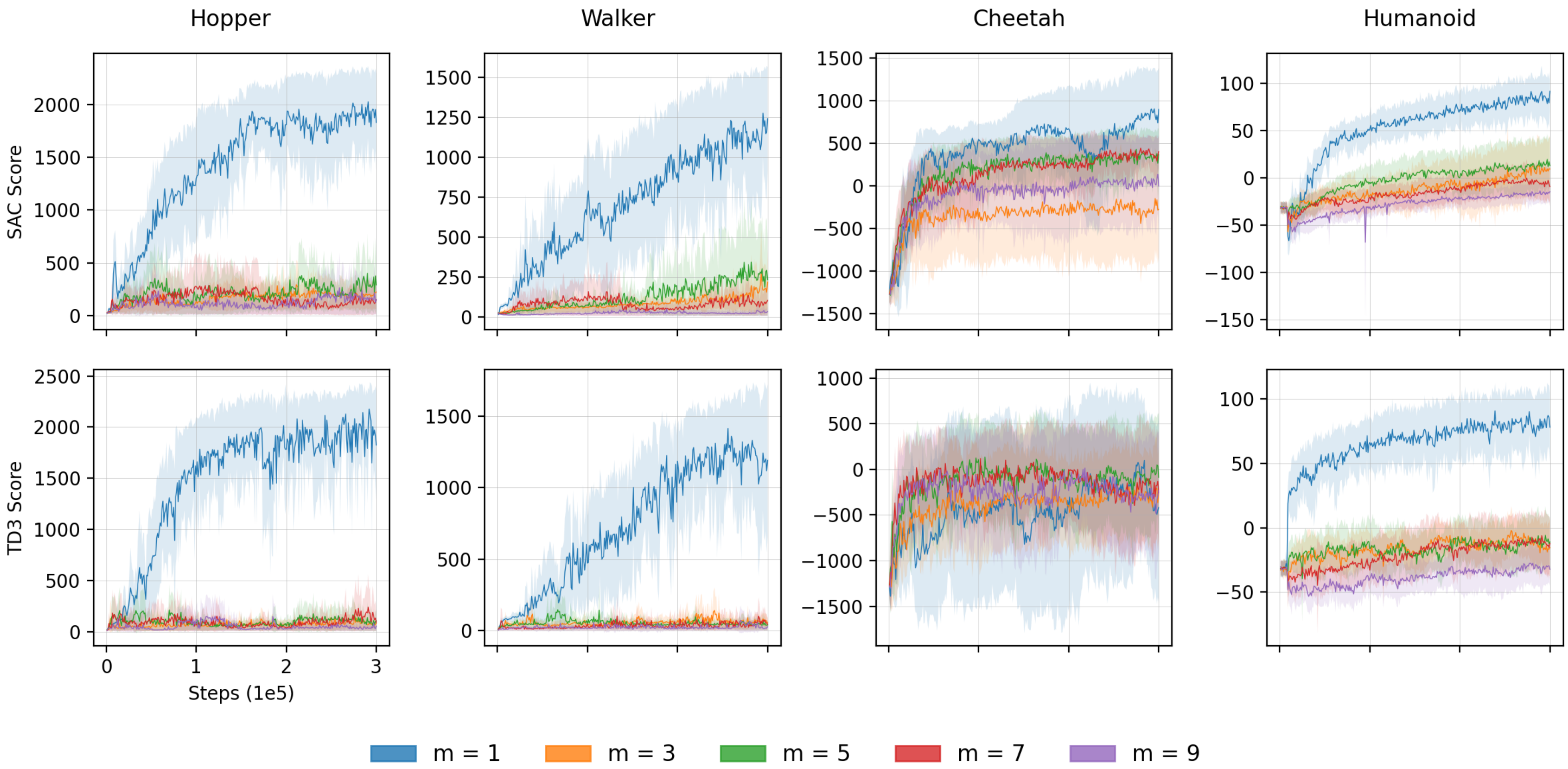

Figure 19: Evaluation episode scores for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

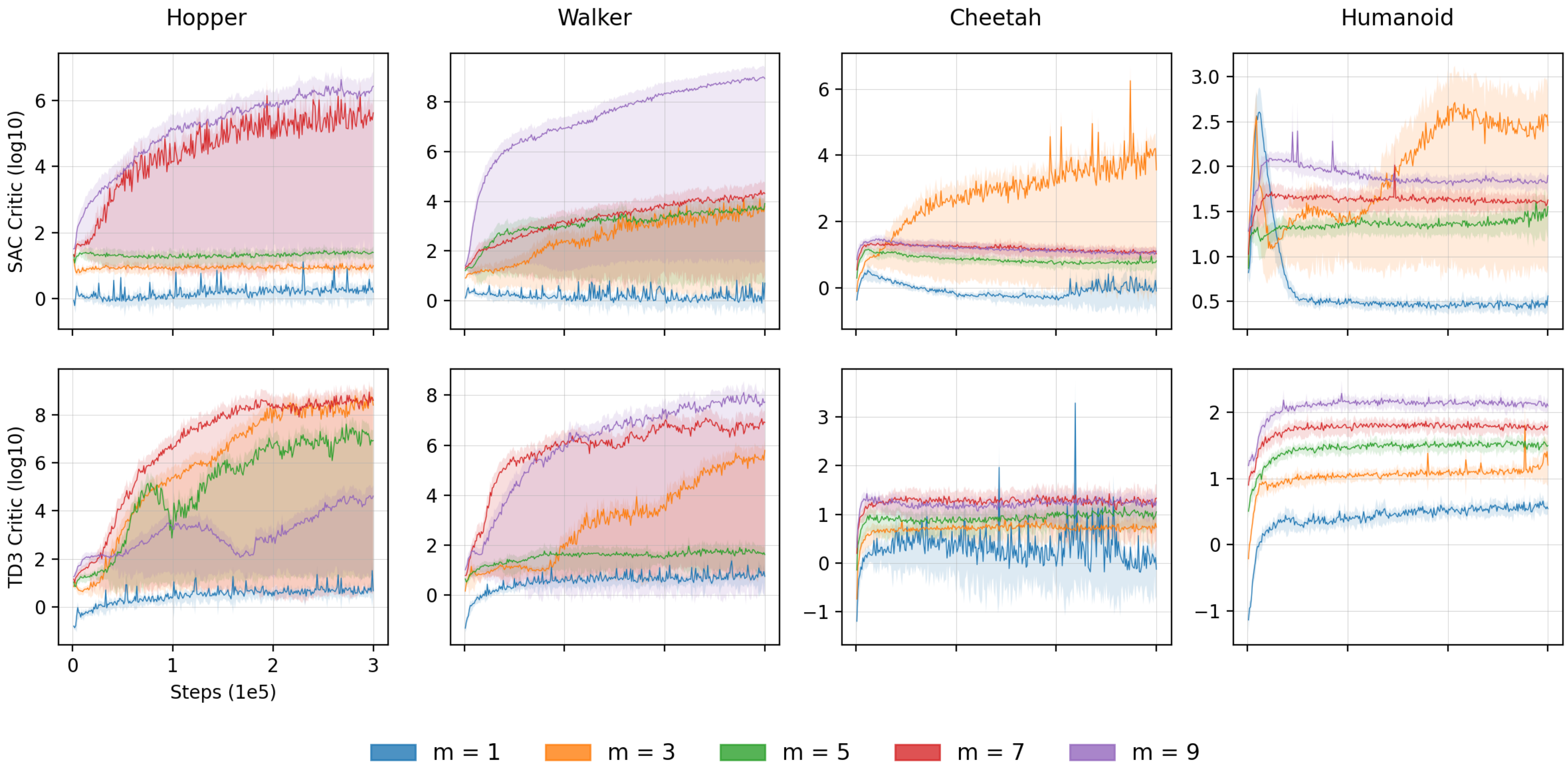

Figure 20: Empirical mini-batch (arithmetic) mean critic losses for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

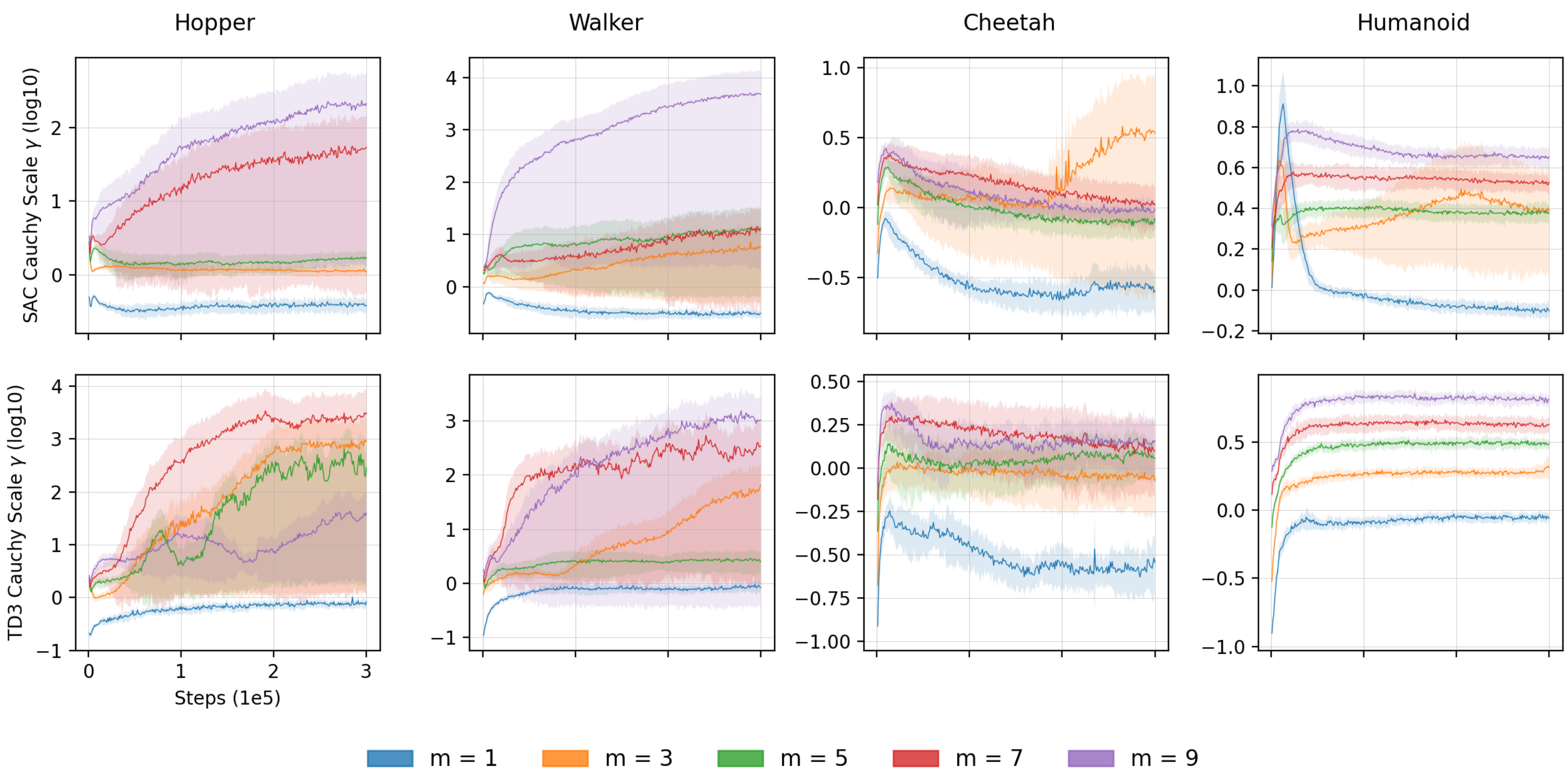

Figure 21: Cauchy scale parameters for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

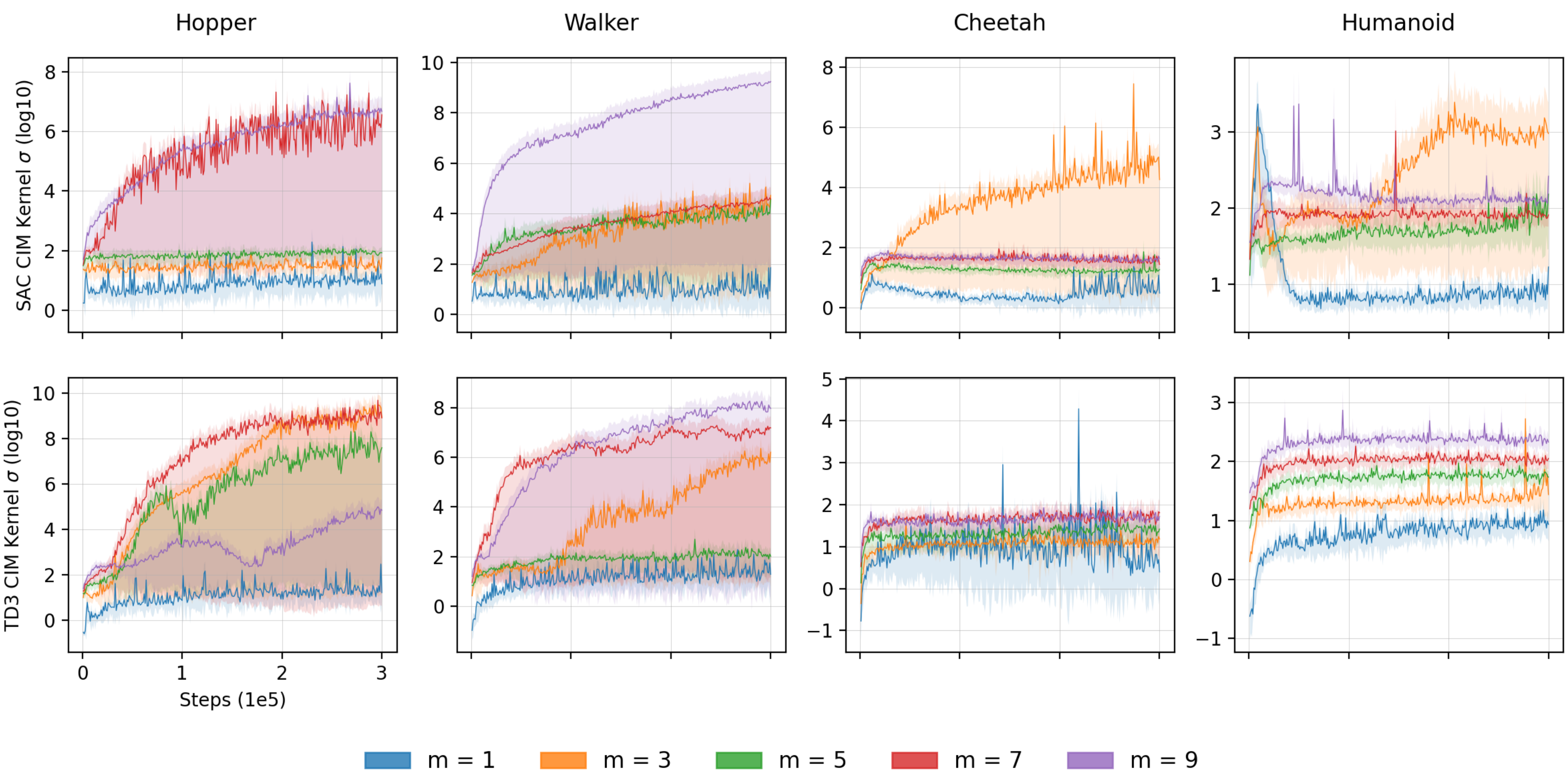

Figure 22: CIM kernel sizes for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

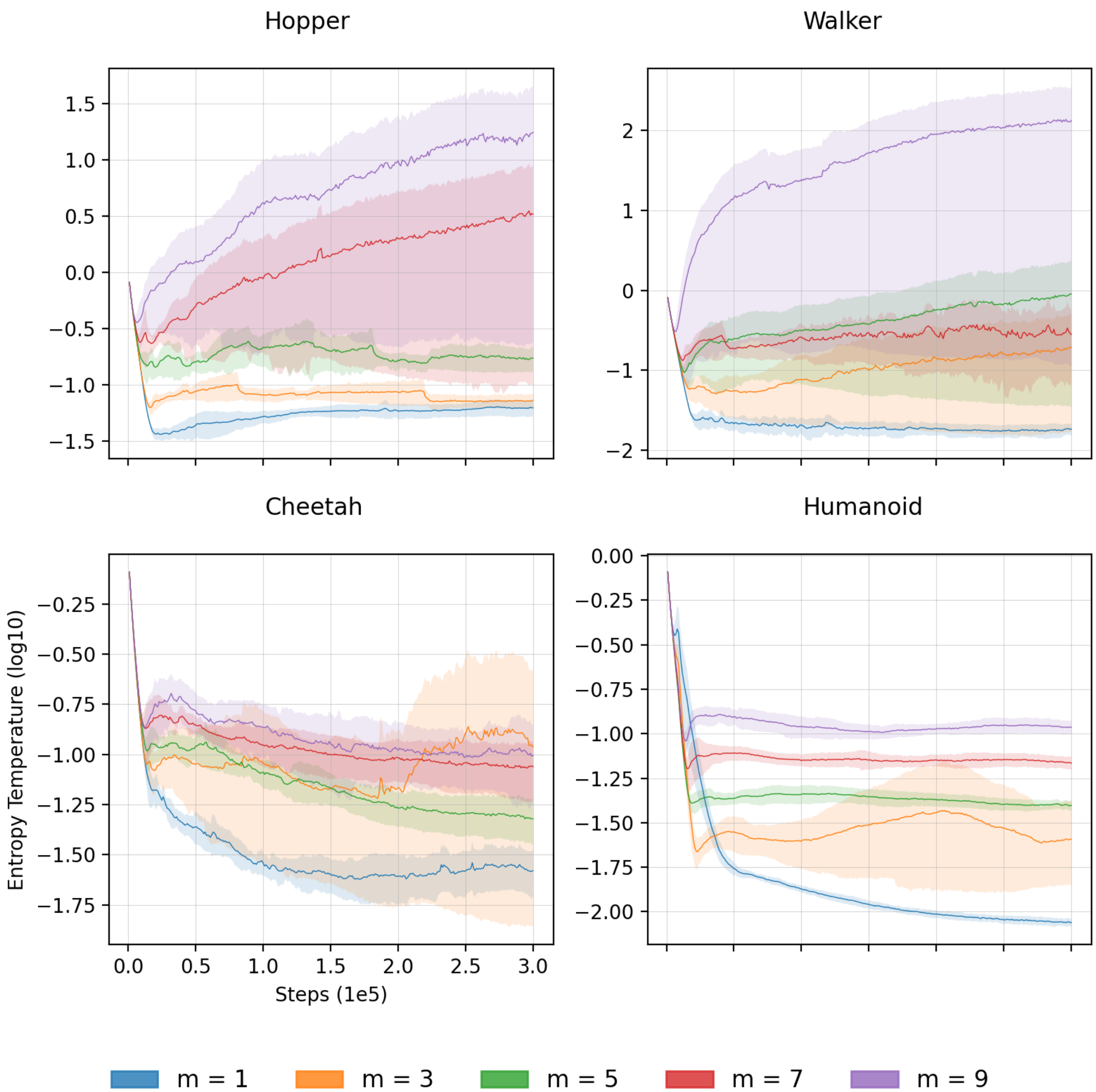

Figure 23: Entropy temperatures for odd multi-step bootstrapped targets across additive environments for SAC algorithm. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

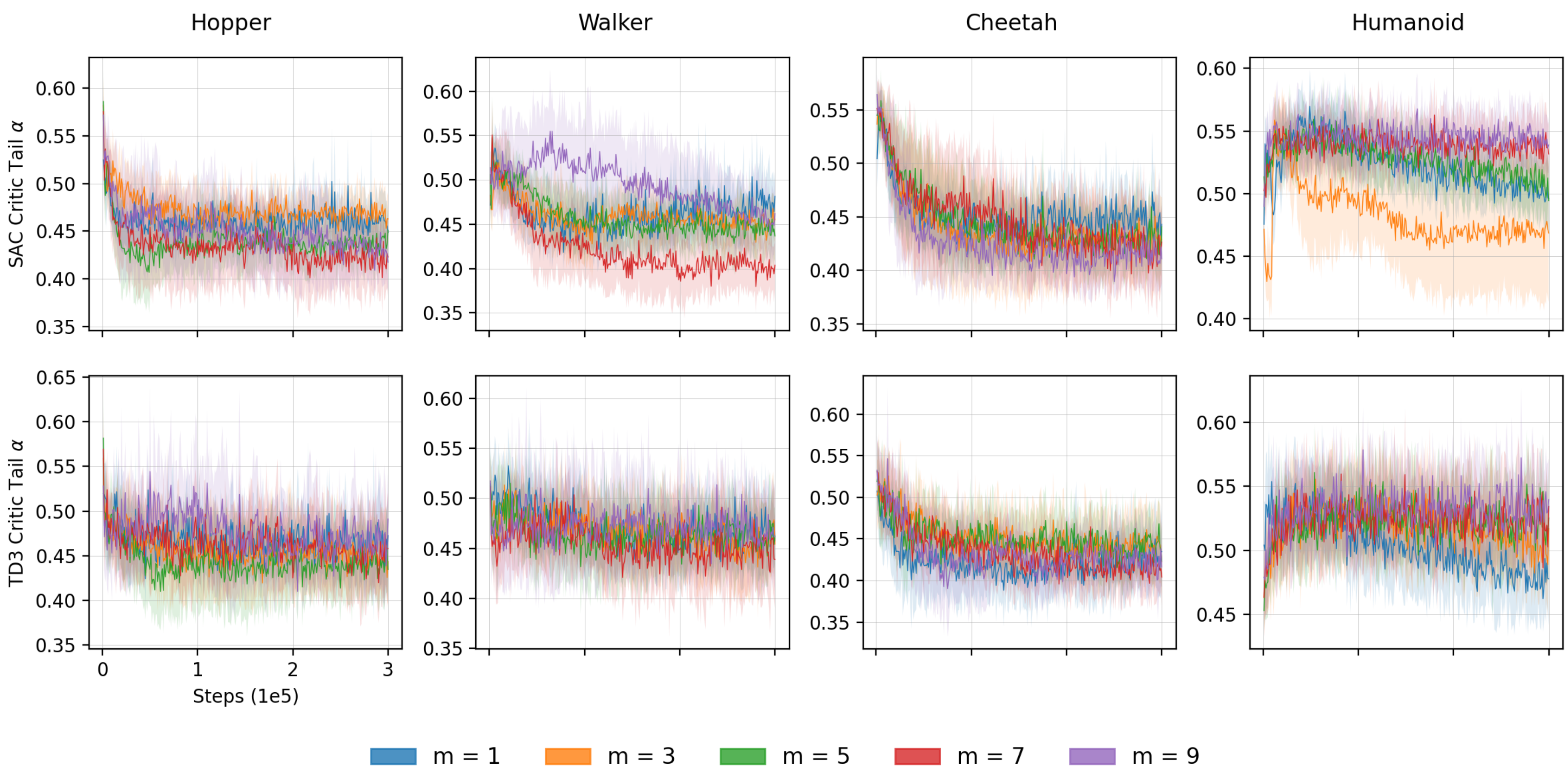

Figure 24: Tail exponents for mini-batch critic losses for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

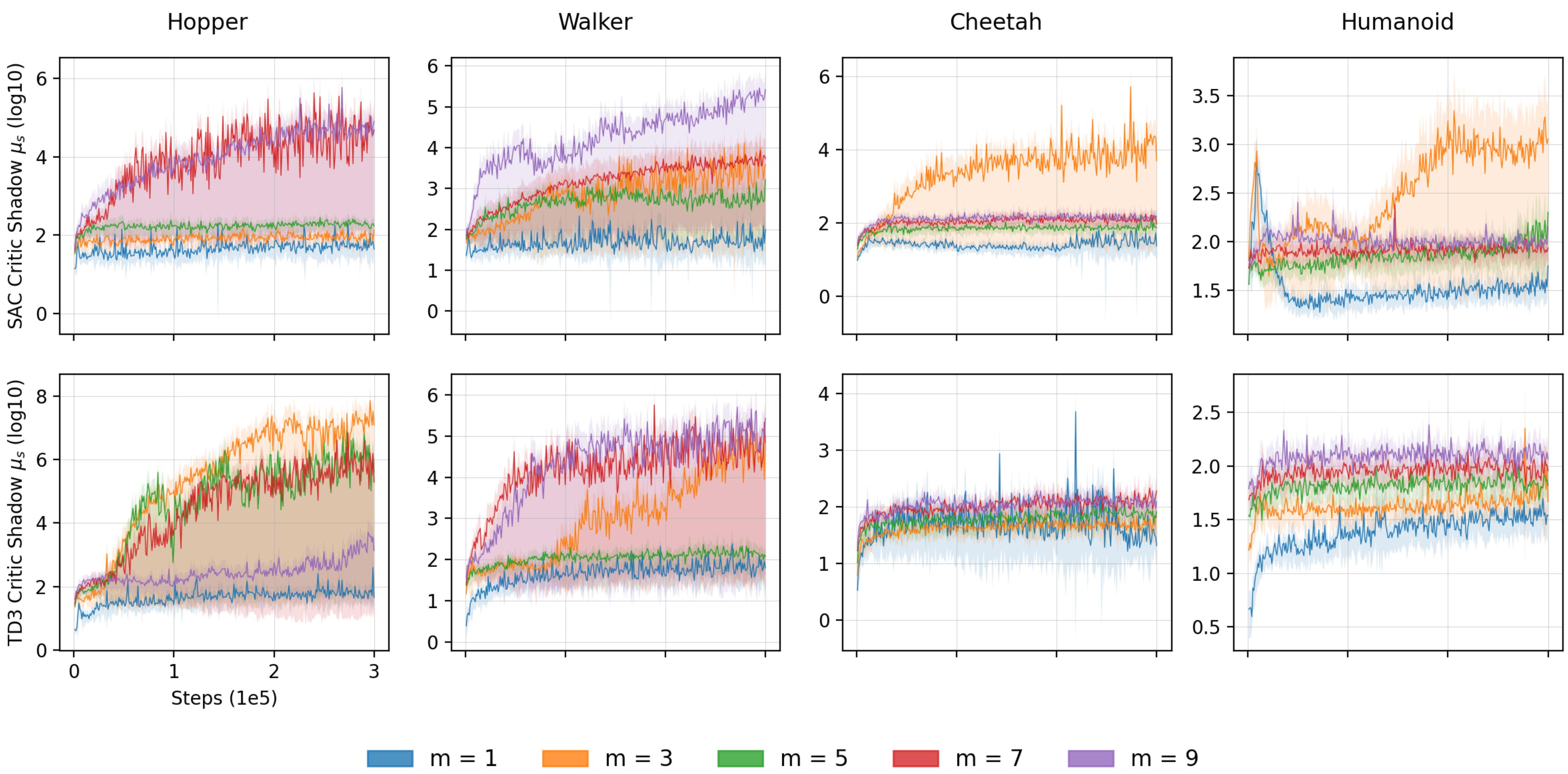

Figure 25: Shadow mean estimated for mini-batch critic losses for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

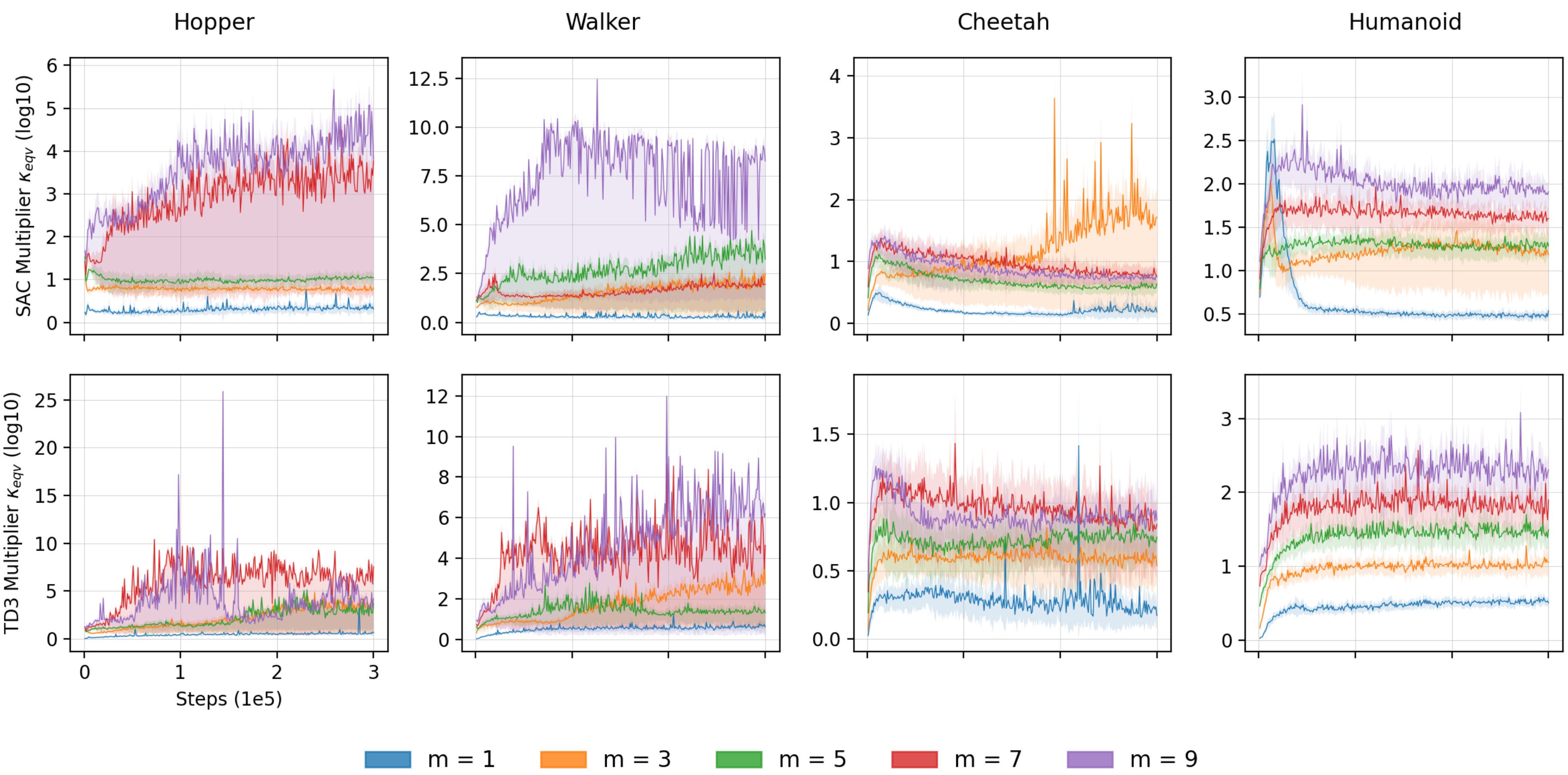

Figure 26: Equivalence multipliers for empirical and shadow means of mini-batch critic losses for odd multi-step bootstrapped targets across additive environments for both SAC and TD3 algorithms. The solid lines indicate the mean value across ten trials and the shaded region represents one MAD about the mean.

11 Multiplicative Experiments

This section contains experimental results pertaining to the investigations contained in Section 1 and extending that approach utilising the machinery of Sections 5-7. We analyse environments where the rewards are strictly multiplicative using MSE critic loss functions, shadow means, and single-step returns. This involves constructing these environments while still being interfaced through the OpenAI gym [63] framework.

First we opt to recreate the simple coin flipping gamble in Section 1.2 with equally probably +50%/-40% returns from [8, 9]. The goal is to create three environments increasing in action complexity exactly mimicking those available to Investors 1-3. Next, we examine the same gamble, but with multiple identical coin flips occurring simultaneously where agent must allocate capital across each of the flips. We then investigate a more complex environment involving rolling a fair die where the Kelly criterion cannot be so simply applied [11]. In these situations, we can attempt empirically find the optimal leverage by simulating a large number of trials and observing the empirical wealth maximising parameters.

Clearly these simple gambles are inadequate to describe the real-world and so we repeat the above situations for financial assets following geometric Brownian motion (GBM) [29] where the behaviour of Investors 1-3 from Section 1.2 will be contrasted with known optimal theoretical results outlined in [5, 15, 16, 104]. To bring even further realism we repeat the above again but with actual historical prices for a market containing multiple assets with the agent performing real-time portfolio management.

In what follows we will limit the discussion to be in reference to wealth maximisation in environments modelled as financial gambles. This in no way implies a limitation of this approach, only that financial trading is objectively straightforward to represent, in line with historical motivations for multiplicative dynamics, and very convenient to explain given its universal importance. In Appendix C we discuss several other applications in a more self-contained manner without experimental validation. The results of this section are a prerequisite to them with examples including robotic control for medical surgery, supply chains, a more detailed prescription for portfolio management, guidance systems, and minimising quantities. Regardless, this development can be applied to any multiplicative process provided the states, actions, and rewards can be accurately parametrised.

11.1 Creating Compounding Environments

Much of the derivations in Sections 5-7 are contingent on adapting existing reinforcement learning environments to multiplicative dynamics as discussed in Section 4. Specifically, the goal was to show that we could convert the absolute rewards an agent receives to relative rates of return and still utilise the existing tools of reinforcement learning. Put simply, the key points were to define the new environment state $\xi_t = s_t \cup \Gamma^+_{t-1}$ so that the state-action pair becomes $(\xi_t, a_t) \leftarrow (s_t, a_t)$, and that for Q-value convergence to still occur in Eq. (140) we require the cumulative reward $\Gamma^+_{t-1} > V_{\min} \geq \gamma \ \forall \ t$ to arrive at Eq. (141) where $\gamma \in [0, 1)$. Therefore, initialising with any $\Gamma_0 > V_{\min} \geq 1$ for all environments is perfectly acceptable.

Practically then, the contents of Sections 5-7 can be ignored as they are only relevant to kinds of people who operate on first-principles, resist using models without proof, and that have a tendency to use works like "convergence" and "uniqueness". *For everyone else, make the cumulative reward Γ^+_t (to be maximised) another input state, and terminate the training episode at time step t if the agent achieves $\Gamma^+_t \leq V_{\min}$.*

Now recall again that the ultimate purpose of the agent is to find a strategy that maximises final wealth, not "risk-adjusted" returns or "diversification" as you cannot 'eat' high Sharpe (or Treynor) ratios [5, 11]. This idea was discussed in detail in Section 1.2 and is encapsulated by the time-average growth $\bar{g}$ rate in Eqs. (3, 10). Maximisation of this quantity is our sole concern as it captures the entire evolution of wealth from the starting point. As we have seen, it is heavily penalising for large losses with each loss forever engraved into history. The impact of time is such that as it increases, the penalty for not increasing wealth is amplified. This provides a standardisation metric for which strategies can be compared as they must not only raise wealth but must do so consistently over time to be competitive. Adapting this to reinforcement learning was the purpose of the reformulated reward in Eq. (117).

Therefore to be consistent with existing nomenclature we define the agents reward $r_t = r(\xi_t, a_t) = r(\Gamma^+_{t-1}, s_t, a_t)$ at time step t to be a function of three variables: initial value $V_0 = \Gamma^+_0$, current time t, and current value $V_t = \Gamma^+_t$. The reward (return) function for multiplicative dynamics is then the geometric mean

$$r_t \equiv 1 + \bar{g}_t = \exp\left[\frac{1}{t}\ln\left|\frac{V_t}{V_0}\right|\right] = \sqrt[t]{\frac{V_t}{V_0}} = \left(\prod_{i=0}^{t-1}\frac{V_{i+1}}{V_i}\right)^{\frac{1}{t}} \tag{213}$$

where $r_t \in (0, \infty)\ \forall\ t$ and the episode terminates at time $t = k$ if $V_k \leq V_{\min}$ which is treated as a complete loss of capital as it also implies $\bar{g}_t < 0$ and so is an undesirable gamble. Note the lower bound is open as for our multiplicative processes it is not possible to reach zero.

Furthermore, the calculation of returns can also be performed internally within the environment so that for all practical purposes the agent receives it in an identical manner the usual reward. Multi-step returns composed of compounding products as in Eq. (158) can then easily be calculated. Hence, all relevant variables are stored in the replay buffer. The critics will again minimise the mean difference between the actual and discounted target Q-value (or Q-return or Q-growth or Q-geomean) functions across the mini-batch, and actors will again maximise the mean Q-value (or Q-return or Q-growth or Q-geomean) functions.

One major pitfall of maximising mean returns is that they may not be the quantity of interest as we have seen Section 1.2 since they have a frequent tendency to be massively skewed by large outliers. Instead, maximising the median would be more theoretically sound [4, 11]. Furthermore, we could institute fractional Kelly betting where we desire to raise the minimum growth of any arbitrary percentile, say the top 95% which would be even more effective [11]. This would involve the actor adjusting network parameters using only, say the bottom 50% for the median, or bottom 5% for the 95th percentile, of Q-value estimates when forming the mean using mini-batch learning. Therefore, we assume that by learning to maximise the mean reward of this bottom percentile for a sufficient amount of time, we are able to train more robust actors to maximise investor wealth of a desired percentile.

We create four environments: coin flip, dice roll, GBM, and financial markets. The first three are tested identically while the fourth is tested more formally. The state space consists of the cumulative reward and the latest price of each asset to make the situation a MDP. The action set at each time step consists of the leverages l_i assigned to each of the N assets $l_1, ..., l_N$ so that $\sum_{i=1}^{N} l_i$ can be considered the total portfolio leverage and is dependent on the price change of each asset (MDP assumption). Along with this we have a global stop-loss λ and retention ratio ϕ. Hence, the cardinality of the spaces are $|\Xi| = N + 1$ and $|\mathcal{A}| \leq N + 2$.

For additional realism, we could add stop-losses and retention ratios for each asset position, but this is outside the scope of our analysis. However, we could speculate that the leverage the agent assigns for any asset in a multi-asset portfolio implicitly signifies both the stop-loss and retention ratio. A very low leverage indicates very high stop-loss and retention ratio, and vice versa.

For these environments, the agent takes a similar form to Investors 1-3 from Section 1.2 with additional freedom. Recall again the mythical Investor 4 in Eq. (9), for an N asset universe of possible portfolio assets with leverages $\vec{l} = (l_1, ..., l_N)$ and prices $\vec{P} = (P_1, ..., P_N)$, we can very generally express the objective as

$$\lambda_1^*, \phi_1^*, \vec{l}_1^*, ..., \lambda_T^*, \phi_T^*, \vec{l}_T^* \in \arg \max_{\lambda_1, \phi_1, \vec{l}_1, ..., \lambda_T, \phi_T, \vec{l}_T} \prod_{t=1}^{T} \frac{V_{t+1}(\lambda_t, \phi_t, \vec{l}_t, \vec{P}_t)}{V_t(\lambda_{t-1}, \phi_{t-1}, \vec{l}_{t-1}, \vec{P}_{t-1})} \tag{214}$$

and so the agents goal is to find the parameters that maximise the geometric mean. This is further simplified by the MDP approximation where we are only concerned with improving the next reward by only analysing the past states. This yields the objective at time t which is to find the actions the maximise growth

$$\lambda_t^*, \phi_t^*, \vec{l}_t^* \in \arg \max_{\lambda_t, \phi_t, \vec{l}_t} \exp\left[\frac{1}{t} \ln \left| \frac{V_{t+1}(\lambda_t, \phi_t, \vec{l}_t, \vec{P}_t)}{V_0} \right|\right] \tag{215}$$

where optimal combination of parameters may not necessarily be unique and where $V_{t+1}(\lambda_t, \phi_t, \vec{l}_t, \vec{P}_t)$ is of course completely unknown. Recall that for this to be valid, the asymptotic reward scaling condition (for absolute rewards) in Eq. (100) must be satisfied for the environment to be deemed truly multiplicative.

For the coin flip, dice roll, and GBM environments, we create three categories of investors again, but now the agent selects the optimal variables at each time step. These will be dubbed Investors A-C to avoid confusion. They consist of the following time-dependent characteristics:

(a) Investor A: Only has control over N asset leverages $\vec{l}_t$ where $|l_{t,i}| \leq \eta$ for $i = 1, ..., N$.

(b) Investor B: Controls both N asset leverages $\vec{l}_t$, and the portfolio stop-loss λ_t.

(c) Investor C: Controls N asset leverages $\vec{l}_t$, the portfolio stop-loss λ_t, and the portfolio retention ratio ϕ_t.

In other words, with Investor A only controlling leverage with a maximum absolute leverage of η, Investor B has the ability to also set a stop-loss, and Investor C further gains the retention ratio. For each coin flip asset, the maximum possible absolute leverage for Investor A is $\eta = 1$, Investors B and C are limited such that one step cannot result in the complete loss of the actively gambled portion of the total portfolio seen in Eq. (7) so that $\eta = 2.5$. The exact same approach is used for the dice roll assets where for Investor A $\eta = 1$ again, and Investors B and C has $\eta = 2$. For GBM, the absolute leverage limit is forcefully set to be $\eta = 6$ for all investors. For multiple asset portfolios as each asset is permitted this maximum leverage $\eta = 6$, the agent must then find an optimum balance.

Regarding technical implementation details for our environments, the following several modifications are required for both computational efficiency and ensuring learning stability:

(i) The states $\xi \in \Xi : \mathcal{S} \times \mathcal{R}$ are all limited by a maximum state value $V_{\max}$. Hence we have the limit on the cumulative reward $V_{\min} < V_t \leq V_{\max}$ and for each state (asset price) $0 < s \leq V_{\max}$. The bound $s > 0$ is due to the fact for our multiplicative processes it is impossible to reach zero. The prices of each of the assets (states) $P_1, ..., P_N$ are all initialised at a common P_0 for convenience. Selection of V_0 as the initial portfolio value and the minimum value $V_{\min} = \psi V_0 < V_0$ where $\psi \in \left[\frac{\gamma}{V_0}, 1\right)$. Choice of this parameter is dependent on the user as it signifies level of risk-aversion functioning as a hard external stop-loss the agent will be trained to always respect. Finally, all state variables are normalised $\xi \leftarrow \xi / V_{\max}$ so that $\xi \in (0, 1]$.

(ii) The output actions $a \in \mathcal{A}$ from policy $\pi : \Xi \to \mathcal{A}$ are also normalised such that $a \in [-1, 1]$ so that no modification of (zero-mean) Gaussian noise injection using the TD3 algorithm is required. Furthermore, recall both the stop-loss and retention ratio $\lambda, \phi \in [0, 1)$, and that we have set the maximum leverage $|l| < \eta$ to prevent in a complete loss of gambled portion of the portfolio. Hence we are required to modify the action bound such that $a \in [-\epsilon_1, \epsilon_1]$ where $\epsilon_1 \lesssim 1$.

Both λ and ϕ are constructed from this output action using the parametrisation $\lambda, \phi \leftarrow (a+1)/2$. The leverages $l_1, ..., l_N$ for each of the N assets are obtained from outputs $a_1, ..., a_N$ and we scale each of them by the maximum absolute leverage so that $l_i \leftarrow \eta a_i$. This implies the total portfolio leverage is capped at a maximum of ηN.

(iii) The change in portfolio valuations $V_{t-1} \to V_t$ are then calculated by the one-step return $V_t = V_{t-1}(1 + R_t)$ where $R_t = \sum_{i=1}^{N} l_{i,t} R_{i,t}$ and each $R_{i,t} = P_{i,t}/P_{i,t-1} - 1$ is the return from the change in price P_i of each asset. For the coin flipping gamble, each R_i is directly simulated from a Bernoulli distribution. Similarly, for the dice roll we use random sampling to directly obtain the return payoff.

The time-average returns $r : \Xi \times \mathcal{R} \to (0, \infty)$ from Eq. (213) used for actual agent learning are then truncated so that $r \in [\epsilon_2, \infty)$ where $\epsilon_2 > 0$ is a small number for computational efficiency as there is little point in continuing to train after a catastrophic loss. Similarly the one-step return $R \in [-1, \infty)$ is also truncated $R \in [\epsilon_3, \infty)$ where $\epsilon_3 \gtrsim -1$ to speed up learning.

Additionally, for multi-asset portfolios, when calculating the time-average growth $r_t = 1 + \bar{g}_t$ we need to obtain $\ln\left|\frac{V_{t-1}(1+R_t)}{V_0}\right|$ which may be undefined since it is possible for $R_t < -1$ as $\sum_{i=1}^{N} l_{i,t} R_{i,t}$ is technically unbounded for a single step. Therefore, we institute another constraint $V_t \leftarrow \max(V_t, V_{\min})$ to prevent this issue.

For GBM, each $P_{i,t}$ is lognormally distributed with fixed generating parameters μ and σ and are continuous processes. Each relative price change $P_{i,t}/P_{i,t-1} = e^{R_{i,t}}$ is sampled from a normal distribution, returns are similarly aggregated $R_t = \sum_{i=1}^{N} l_{i,t} R_{i,t}$, the portfolio value is exponentially compounded $V_t = V_{t-1} e^{R_t}$, and the constraint $V_t \leftarrow \max(V_t, V_{\min})$ is applied. Note that unlike the discrete compounding, we have the situation $V_t > 0 \; \forall \; t$. The exact same approach is taken for actual markets where each $P_{i,t}$ is known from historical data.

An interesting comparison would be to investigate the behaviour of GBM if utilising discrete portfolio compounding as with coin flip and dice roll. Gauging the effect of continuous compounding would enable us to draw a comparison between its effect on the optimal leverage. We know that continuous compounding with large losses or gains R_c should have a greater effect on change in valuations compared to discrete R_d. This can be clearly seen through the Maclaurin series expansion $e^{R_c} = \sum_{k=0}^{\infty} \frac{x^k}{k!} = 1 + R_c + \mathcal{O}(R_c^2)$ which can only be approximated by $e^{R_c} \approx 1 + R_d$ for $|R_c| \ll 1$. Given the agents' goal is to maximise wealth through avoiding steep losses, we anticipate that over the long-run once the agent has learned optimal actions $\bar{g_c} \geq \bar{g_d}$. This is because continuous compounding will lead to larger valuations due to positive terms of $\mathcal{O}(R_c^2)$.

Lastly, we repeat again that this ability to easily calculate the next valuation as $V_t = V_{t-1}(1+R_t)$ or $V_t = V_{t-1} e^{R_t}$ to obtain r_t is the only reason we have chosen to limit our experiments to financial trading. For any multiplicative process, if you are able to accurately parametrise the return $R_t = R_t(s_t, a_t)$ in terms of the state-action tuple (s_t, a_t), this methodology can be universally applied.

(iv) When utilising a global stop-loss λ and retention ratio ϕ, the minimum value $V_{\min}$ is modified accordingly to force the agent to learn under these additional constraints.

Hyperparameters	**Value**	**Description**
$V_{\max}$	10^{18}	Maximum value for portfolio V_t and all asset prices $P_{i,t}$
P_0	10^3	Common initial price P_i of each asset
V_0	10^4	Initial portfolio value Γ_0^+
ψ	0.01	Minimum portfolio value $V_{\min} = \psi V_0$ for convergence and as a hard stop-loss
η		Maximum absolute leverage applied to any asset within the portfolio
ϵ_1	0.99	Maximum absolute action a_t bound required for action spaceparametrisation
ϵ_2	10^{-6}	Minimum reward $r_t = 1 + \bar{g}_t = \exp\left[\frac{1}{t}\ln\left\lvert\frac{V_t}{V_0}\right\rvert\right]$
ϵ_3	-0.99	Minimum one-step return $R_t = \sum_{i=1}^{N} l_{i,t} R_{i,t}$
ϵ_4	10^{-6}	Minimum simultaneous leverages l_i allocated across all assets
θ	100%	Bottom percentile of actor value functions to be maximised

Table 2: Default hyperparameters for multiplicative experiments. Choice of η is dependent on the environment. These are the parameters we have empirically found to be suitable used for training the agent, for evaluation episodes we are free to select any initial values. Other combinations are likely to yield superior results.

(v) Episode termination is contingent on any one of five possible events. The first three are when either $V_t = V_{\min}$, $r < \epsilon_1$, or $R < \epsilon_3$. The fourth is if leverages $|l_i| < \epsilon_4 \;\forall\, i$ where $\epsilon_4 > 0$ is a small number implying zero positioning across all potential assets. This is necessary to prevent the episode running forever with no risk-taking. The final condition is if any $\xi > 1$ as this would violate the normalisation bound. This final condition will lead to some contamination of the replay buffer with experiences that should not be considered terminal, however over mini-batch learning it is likely to have minimal influence on the overall process.

(vi) Recall that raising the median (or any other bottom fractional percentile $0 < \theta \leq 1$) returns is our primary goal as the mean can be heavily inflated by outliers. Therefore, for mini-batch learning with batch size M, our primary concern would be raise the actor Q-values or advantage functions of the bottom θM estimates. For SAC and TD3 this involves ascendingly sorting the experiences to be maximised in Eq. (204) and Eq. (182) respectively. To then train the neural network, the arithmetic mean is formed using only the lower $M \leftarrow \theta M$ sampled experiences and the parameters are adjusted via backpropagation to maximise this partial mean.

Focusing only on improving the bottom θth percentile will no doubt reduce the speed of learning, but ideally, we would create more consistent performing agents akin to fractional Kelly betting. Choice of threshold θ is however up to user preference in terms, the more risk-averse, the lower the threshold. However, it is unadvised to use very small θ as training the network using only a handful of values is unlikely to be efficient as, effectively, the mini-batch becomes tiny.

(vii) Given the nature of returns often being very small numbers, for enhanced accuracy we opt to use double precision floating-point format (float64). This level of detail is unnecessary for additive dynamics, however, when multiplying potentially thousands of miniscule numbers, rounding errors *compound*.

Therefore, we then have several additional hyperparameters on top of those outlined in additive experiments of Section 10.1 that must be externally specified for each environment, and many are entirely dependent on user preference. All of these are summarised in Table 2.

For purposes of reducing computational time, we only utilise TD3 presented in Algorithm 2 with the hyperparameters provided in Table 5. SAC functions well but given that it learns roughly 30% slower than TD3, we will not show these results. Furthermore, for the time being we also strictly utilise the global standard MSE critic loss function, no multi-step bootstrapping, and again calculate the critic shadow mean estimates $\mu_s(L^*, H, \hat{\alpha})$ where L^* and H are the smallest and ten times the largest value contained within each mini-batch respectively.

For all experiments we conduct $M = 10$ randomly initialised trials with $N = 1,000$ evaluation episodes occurring at every $1,000$ training steps. The length of each evaluation episode for coin flip and dice roll will be limited to just one step as their payoff functions are fixed and hence optimal leverage should not vary. For GBM, given the generating variables μ and σ are also fixed, we again utilise single step evaluations which likely going to be subject to high levels of noise, though with $10,000$ evaluations this should be mitigated.

Each environment will consist of an initial $1,000$ warm-up steps for each trial and the total number of training steps will be limited to $40,000$ for both coin flip and dice roll, and $80,000$ for GBM, as these are found sufficient to convey the main message. The exact same software and hardware setups are utilised as described in the previous section [1, 2].

To compare results, recall again in Section 1.2 where we showed that analysing the median investor performance is just as, if not more important than the mean performance. This was again due to the nature of random variables occasionally resulting in a sequence of very favourable payoffs resulting in monumental performance for a very small number of investors causing massive positive skew in the sample (arithmetic) mean. To counteract this, we segregated the total number of investors into a “adjusted” and “top” sample. The choice of the later was arbitrary where we simply selected the highest 0.01% of performers at each time step.

Moving forward, we will no longer make such distinctions and will simply represent the aggregated results of the evaluation episodes in terms of the mean, a single MAD about the mean, the median, and the 25-75% interval (box plot first and third quartiles) for variables. When it comes to analysing the reward $r_t = 1 + \bar{g}_t$, the focus should on both the mean and median, especially, the difference between them, and highlight the first and third quartiles that encompass 50% of the sample.

Note this focus on the median, first and third quartiles is quite arbitrary. While displaying the 5-95% interval would be more a more concrete measure, since agent learning is an incredibly volatile procedure, the range between these values is often large enough to make all other metrics indistinguishable.

There are also several known issues with the implementation such as:

(i) It is crucial keep in mind that when training the agent using TD3, the agent will inject Gaussain noise to what it considers the optimal policy action at every training and evaluation step. This is done to promote exploration within the replay buffer to force the agent to learn more robustly and is a key advantageous feature of the algorithm. In this case the optimal leverage l^* we observe will be composed of two components, the actual leverage selected l^*_{act} selected by the agent, and the noise scaled by maximum action value $\eta \cdot \mathcal{N}(0, \sigma)$. Using the hyperparameters from Table 5, we will have $l^* = l^*_{\text{act}} + \eta \cdot \mathcal{N}(0, 0.1)$. Hence, when using TD3 it will be statistically impossible to exactly converge to known theoretical optimal values.

(ii) Very frequently agents get stuck in local minima by defaulting to the maximum leverage $|l| \to \eta$ for all assets. Normalisation alleviates the frequency, but it still often occurs. Only way to solve this currently is to restart training. Therefore, if the agent gravitates to the maximum bounds $|l| \to \eta$ for an extended period of time, there is a reduced chance of successful learning. Furthermore, as $\eta \to \infty$, the tendency to approach $|l| \to \eta$ greatly increases. We manually remove training trials that succumb to this plague. Possible solutions may include change learning rates, make the neural networks deeper, and or force PyTorch to train with Double Tensors (at severe cost to GPU backpropagation speed).

There is however another interpretation for this tendency that is not exactly tied to neural networks. For a gamble with $\mathbb{E}[R] \neq 0$, perhaps while at maximum leverage the agent has received a sequence of very favourable payoffs corresponding to huge increases in wealth. Hence, the agent believes that selecting the highest possible leverage $l \to \text{sign}\,(\mathbb{E}[R]) \cdot \eta$ is advantageous over the long-run. Of course, this is incorrect, but we can at least forgive the agent for believing this is valid as contemporary decision theory makes the exact same mistake.

(iii) Using default SAC and TD3 hyperparameters are likely sub-optimal for multiplicative processes. These parameters we tuned and selected to be optimal for continuous robotic control environments discussed in the previous chapter. Hence, continuing to use them for these situations is likely not ideal, regardless, these will not (currently) be modified for purposes of standardisation. Therefore, the results of this section can be considered very crude, untuned, lower bounds of what the agent is truly capable if we were to optimise the algorithms on a case-by-case basis.

11.2 A Permanent Solution

Before we begin, let us first recapitulate the motivation for this analysis. Long ago we posed the question that suppose you were given \$100 and then offered a game based on flipping a fair coin, if heads you make 50%, if tails you lose 40%, and the game is played for $3,000$ rounds. Would you play the game? If no, you get to keep the \$100. If yes, how would you select the optimal leverage?

As discussed in Section 1.2, contemporary decision theory taught throughout the surface of Earth would suggest calculating the 'expected' value $\mathbb{E}[R] = 5\%$, which would dictate that it is a favourable gamble, that it should be played over the long-run, and that you should bet the entire amount at each step, that is, a fixed leverage $l = 1$ and call it a day. The findings however revealed that the optimal fixed leverage should be $l^* = 25\%$ obtained using the Kelly criterion (1956) [4]. Results such as these can also be generalised to gambles with more than binary payouts such as those involving multiple payoffs and or multiple assets, such as numerous simultaneous coins and or dices, but only if, provided again the returns and probabilities fixed [186].

Alas, the real-world is not directly amenable to the Kelly criterion as neither returns nor probabilities are fixed and so its feasibility beyond toy games is greatly diminished. Thankfully, results from GBM, a key model for representing self-reproducing entities which may be considered as the definition of life, show that for a lognormally distributed asset with drift μ and volatility σ, the optimal leverage and growth are $l^* = \mu/\sigma^2$ and $g^* = \mu^2/2\sigma^2$ respectively, provided parameters are of course again fixed [5, 15, 16, 104]. Unfortunately, real assets are also not exactly lognormally distributed as drift and volatility exhibit both time-dependence and are function of countless other variables. Furthermore, in Sections 1 and 3 we have highlighted the complete inadequacy of using variance as a measure of volatility compared to MAD [14].

Therefore, as discussed in Section 1.5 we are left at an impasse, the contemporary paradigm built around Markowitz Portfolio Theory (1952) [109–111] does not appear to maximise wealth (which is the whole point), yet it is the unanimous global mandate. On the other hand, 21st century proponents [5–11] of the ideas of Kelly, which were initially conceived by Bernoulli (1738) [3], have laboriously proved this misconception and emphasised maximising the geometric mean. As an aside, according to [11], even Markovitz began promoting the geometric mean by 1959 and became strong proponent of it by 1976, however, by then it was tragically too late as the mathematical allure of "modern" portfolio theory had already conquered campuses.

The question is then can we utilise model-free reinforcement learning powered by deep neural networks to directly and autonomously maximise the geometric mean for any multiplicative process? Can a piece of software (couple hundred lines of code) determine the optimal leverage of $l^* = 25\%$ for the +50%/-40% gamble through simulation?

Furthermore, we make the MDP assumption and so unlike humans, the agent only has access to the previous asset prices to decide what percentage to bet next, not the entire history of prices. This would be especially challenging for humans as it is unlikely that most people would be able to recognise the pattern easily if they were forced to do so by only observing singular prices changes (not even the returns R). This inability to factor in past outcomes forces the agent to learn across the possible distribution prices without no explicit knowledge of what drives their movement.

The model-free aspect is also essential as the agent initially knows absolutely nothing, it has \$100 (not even knowing what this means), has lever to control how much of this it bets (with no idea what is actually being controlled), then the coin is flipped (with no knowledge of the underlying gamble to the agent) and so the agents' fate is sealed. The remaining funds are then calculated and the agent receives a reward r of the form in Eq. (213). The sequence then repeats. Keep in mind for this gamble virtually all humans would know it would be non-nonsensical to bet against it, that is $l < 0$ (go short), but the agent has no clue and will be open to all possibilities. The singular purpose of the agent is then to intelligently find the best leverage to maximise this reward r regardless of what all this signifies.

If this is possible, could it work for GBM by only observing changes in price with no knowledge of the underlying two variables μ and σ responsible for the distribution of prices? Could it work in the real-world where prices are observed but the underlying processes are also unknown?

One potential critique of this approach is in our parametrisation of the reward being exactly in the form of Eq. (213) and so naturally, if anything, we will maximise this quantity. We make zero attempt and could not care less about maximising "risk-adjusted" returns as scientifically quantifying these does not appear to yield any objective or accurate approaches [5, 11, 14]. In contrast, measuring success in terms of final ending wealth is a clear (non-parametric) goal regardless of the path that gets us there. Another criticism would be our inclusion maximum absolute leverage $\eta \ll \infty$ since we have technically given the agent a 'hint' of what range of values the (likely non-unique) optimal leverages must be constrained within. This is solely done to speed up the training processes by excluding nonsensical values through truncation.

11.3 Coin Flip Revisited

The fixed payoffs for a leveraged position for the coin flip from Eq. (2) are

$$R_{t+1} = \begin{cases} +50\% \cdot l_t, & p_u = \frac{1}{2} \\ -40\% \cdot l_t, & p_d = \frac{1}{2} \end{cases} \tag{216}$$

which yield $\mathbb{E}[R_{t+1}] = 5.0\% \cdot l_t$ and $\sigma_{t+1} = \text{MAD}_{t+1} = 45.0\% \cdot |l_t|$. For the slightly more complex situations of two identical assets following the same payoff we have

$$R_{t+1} = \begin{cases} +50\% \cdot l_{1,t} + 50\% \cdot l_{2,t}, & p_{uu} = \frac{1}{4} \\ +50\% \cdot l_{1,t} - 40\% \cdot l_{2,t}, & p_{ud} = \frac{1}{4} \\ -40\% \cdot l_{1,t} + 50\% \cdot l_{2,t}, & p_{du} = \frac{1}{4} \\ -40\% \cdot l_{1,t} - 40\% \cdot l_{2,t}, & p_{dd} = \frac{1}{4} \end{cases} \tag{217}$$

so that $\mathbb{E}[R_{t+1}] = 5.0\% \cdot (l_{1,t} + l_{2,t})$ and the volatility expressions are less clean to express. Similarly, for ten identical assets we have $\mathbb{E}[R_{t+1}] = 5.0\% \cdot \sum_{i=1}^{10} l_{i,t}$.

For all three counts of identical asset portfolios, the key question is what are the optimal combinations of leverages for Investor A. As the assets are identical it is appropriate to measure final outcome in terms of the average leverage across all assets as we should theoretically have $l_i^* = l_j^* \ \forall \ i \neq j$ provided a sufficiently large sample size is used. To bring further realism using Investors B and C we add the global stop-loss λ (B and C) and retention ratio ϕ (C).

The aggregated results for all three asset counts directly comparing the actual actions taken by the agents are shown in Fig. 27 and details regarding the training variables are shown in Fig. 28. Note again that the first $1,000$ steps are composed of randomised actions and hence exhibit large volatility.

We focus the bulk of our explanation on the $N = 1$ case in the first column of Fig. 27 as the other two cases can then be independently analysed. The exact values of the time-average growth $\bar{g}$ are not that essential for this simple gamble as we know that $l = 25\%$ maximises valuations, only their relative placement across the three investors is important. We explicitly do not plot the MAD about the mean (or the median) as its very large magnitude obscures the important details. We observe the mean and median growth rates are all positive and the three investors coincide.

The motivating question regarding what the optimal leverage for Investor A is answered by its convergence to a noisy bound in the vicinity of $l = 25\%$. It does not select $l = 1$ as contemporary decision theory would suggest being the optimal 'expected' wealth maximising leverage. This point is worth repeating, an entirely open-source piece of software, with zero prerequisite knowledge, access only to the previous price, has perhaps provided the first ever reinforcement learning-based vindication of the Kelly criterion. In line with [5–11], the agent makes a strong case for major shortcomings in over half a century of established financial theory and invalidates the most basic core teachings of countless higher education centres worldwide. The nascent field of artificial intelligence powered by deep learning has now independently arrived at the same conclusion that titans like Bernoulli had almost three centuries ago.

Interestingly, by roughly converging to this bound, that is, the Kelly criterion, the agent is implicitly maximising the median geometric return, not the mean geometric return. This is puzzling as we are training with $\theta = 100\%$, not $\theta = 50\%$. Even when learning with $\theta = 50\%$ or $\theta = 5\%$, the agent comes to the same conclusion within reasonable error, albeit taking a far longer amount of time. The reason for this occurrence is not yet obvious, but it is a surprisingly positive occurrence.

Investor B and C have access to far more leverage and so naturally the range of values they experiment with is far wider, yet their medians all favour leverages less than unity. The stop-losses reveal that Investor B decides to convert itself to Investor A rather quickly, deciding that setting a stop-loss is not needed while operating at the same leverage as Investor A. This should be considered a resounding success as the agent uses a stop-loss to preserve capital while it is trying a large variety of leverages, but once it discovers the optimal it realises there is no longer a need as you should be betting 25% of your entire capital. Investor C meanwhile maintains very high stop-losses and retention ratios, deciding that betting only small portions of profits is superior.

Additionally recall that when using TD3, for observed optimal leverages are composed of two components such that $l^* = l^*_{\text{act}} + \eta \cdot \mathcal{N}(0, 0.1)$ where for Investor A we have $\eta = 1$ and for Investors B-C we have $\eta = 2.5$. Therefore, exact convergence to $l^* = 0.25$ is not theoretically possible, though we do observed a fairly tight bound forming around its vicinity.

In Fig. 28 for one asset, the tail indices of critic losses are almost all less than unity, implying fat-tails. The shadow critic losses are of identical structure to the critic losses. The equivalence multiplier for the two means increases with the complexity of the investors. This is to be expected since as you increase the complexity of the environment, you expect larger mistakes.

For two and ten identical assets seen competitively in Fig. 27, Investor A once again converges in the vicinity of $l = 25\%$ highlighting the robustness in agent learning even under increasing distortion. As you increase the number of identical assets, we observe the volatility in growths dramatically increases to the point where for ten assets, the lower 25% of the sample is bankrupt, while the upper 25% has compounding growth in excess of 100% per time step!

We also observe that the stop-loss decreases with number of assets indicating Investor B always convert themselves to Investor A, while Investor C seems to follow the same trend. The behaviour of the retention ratio is not quite so explainable. It appears to increase and then decrease with Investor C also converting themselves to Investor A as you raise the number of assets.

In summary, reaching the Kelly criterion is incredibly robust and the ability to train an agent to independently learn of its existence for a very simple gamble is a huge success. Furthermore, the agent correctly recognises that maximising the median is preferable to the mean growth. What would be more remarkable is to see whether this method could generalise to arbitrarily complex environments.

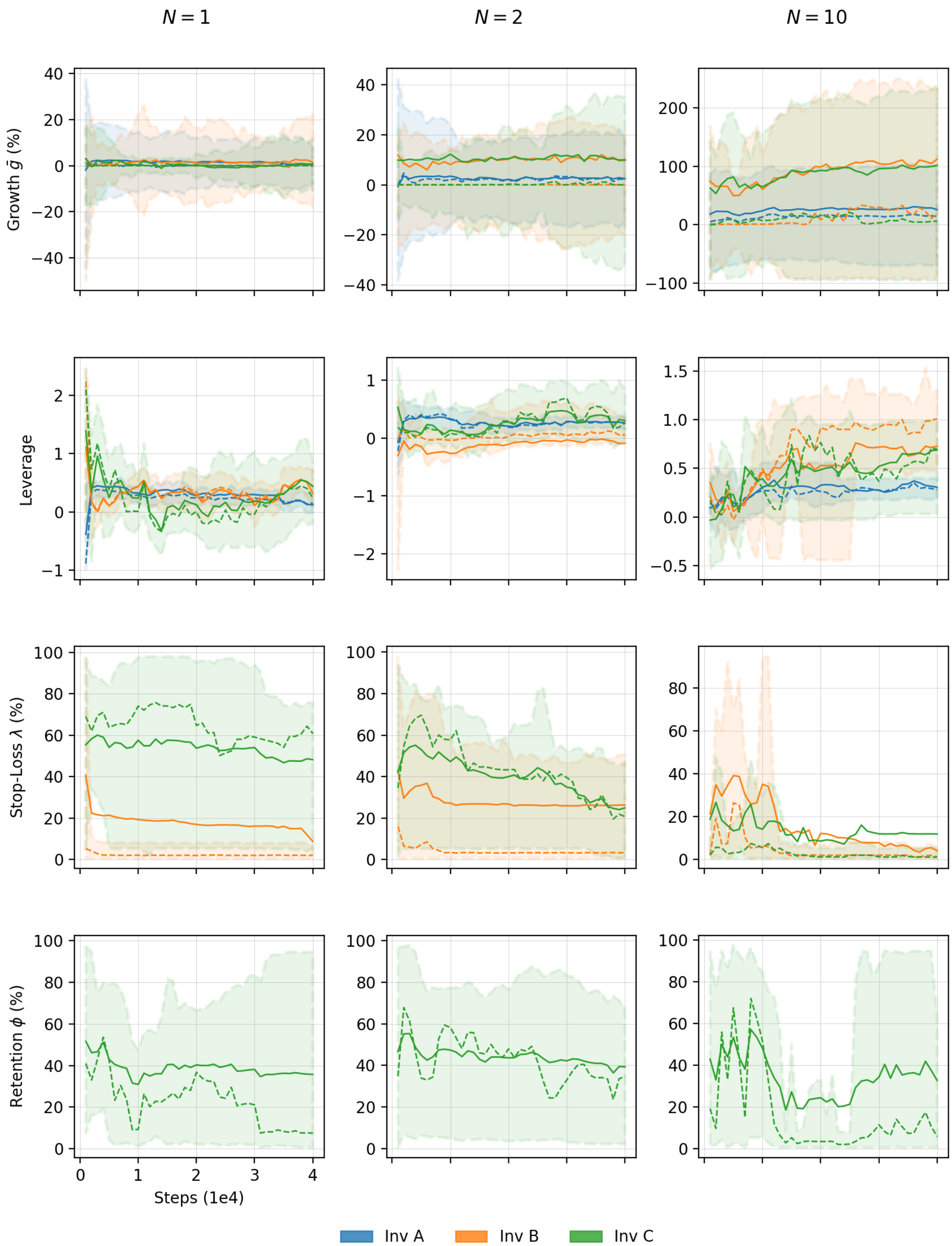

Figure 27: Summary of the coin flip environment agent performance with respect to actions across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions with dashed outline represents the range between the first and third quartiles.

Figure 28: Summary of the coin flip environment agent training across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions without outlines indicate one MAD about the means.

11.4 Dice with Nietzsche's Demon

Before moving to GBM, as an intermediate step we examine the "Dice with Nietzsche's Demon ($N = 1$)" gamble in [11]. This game is essentially a more complex version of the previous as there are three distinct payoffs and so solving Eqs. (15-17) for the optimal leverage for even a single-asset gamble is far more mathematically intensive [186]. Alternatively, what is done in practice is identical to the approach of Section 1.2 where huge amounts of trials are simulated using various strategies and the one that raises the median wealth the highest should be empirically considered superior.

The reference to Nietzsche is due the previous example in [11] being "Dice with Schrödinger's Demon ($N = \infty$)". Their argument is identical to the one presented in Section 1 where "$N = \infty$" refers to the ability of an individual to access the multiverse and obtain the (arithmetic) mean return which we know to be grossly inflated. Schrödinger [187] is called in reference to his famous feline paradox. Nietzsche on the other hand, emphasised the concept of *amor fati* — love your fate, that is, make decisions today that you are capable of living with for the rest of your life as they are forever carved into your history. As discussed in Section 1.1, this is done through maximising a single "$N = 1$" random investors' time-average growth as that is what we should consider our performance to resemble.

The fixed payoffs for a leveraged position in this die roll are

$$R_{t+1} = \begin{cases} +50\% \cdot l_t, & p_1 = \frac{1}{6} \\ +5\% \cdot l_t, & p_2 = \frac{2}{3} \\ -50\% \cdot l_t, & p_3 = \frac{1}{6} \end{cases} \tag{218}$$

which yields $\mathbb{E}[R_{t+1}] = 3.3\% \cdot l_t$, $\sigma_{t+1} = 29.0\% \cdot |l_t|$, and $\text{MAD}_{t+1} = 17.8\% \cdot |l_t|$. Similarly, for extensions to N assets we clearly have $\mathbb{E}[R_{t+1}] = 3.3\% \cdot \sum_{i=1}^{N} l_{i,t}$.

The question is then again what is the optimal leverage for this payoff structure for Investor A? In Appendix A we perform an identical analysis to that of the coin flip in Section 1.2. These findings reveal the median maximising fixed leverage to be $l^* = 40\%$ for Investor 1. Furthermore, empirically through simulation of $10,000$ investors for 300 time steps each, [11] also find that $l^* \lesssim 40\%$ to maximise the median time-average growth rate, and $l^* \lesssim 10\%$ to maximise the 95th percentile. Our results are shown in Figs. 29-30 in the exact same manner as the previous gamble.

This is clearly a more challenging environment. Recall that the agent does not have access to the actual payoffs R_t at any stage, only the changes in the price of a asset that is impacted by the payoff. For a single asset, it will see its value increase or decrease by 50% roughly $\sim 17\%$ of the time each, and for the remainder $\sim 66\%$ of occurrences will involve a 5% increase in the price of the asset. It does not really matter what the asset price actually is given this is a multiplicative process. Furthermore, with only access to the previous price the agent must determine the optimal amount to bet rather than being able to directly observe the history of dice rolls (MDP assumption).

Investor A determines a noisy bound about $l = 45\%$ for a single, $l = 40\%$ for two, and $l = 25\%$ for ten identical assets as shown in Fig. 29. In all cases the optimal $l^* = 40\%$ is found to be within error. This further highlights the robustness of using model-free reinforcement learning to autonomously identify the optimal leverage despite TD3 policy noise $l^* = l^*_{\text{act}} + \eta \cdot \mathcal{N}(0, 0.1)$ with Investor A $\eta = 1$ and Investors B-C $\eta = 2.5$.

As with the previous gamble, as the number of assets increase the volatility of the time-average growth rate is also significantly raised. Investor B again decides to convert themselves to Investor A by completely abandoning the use of a stop-loss. While Investor C continues to retain roughly 50% of its profits at each time step.

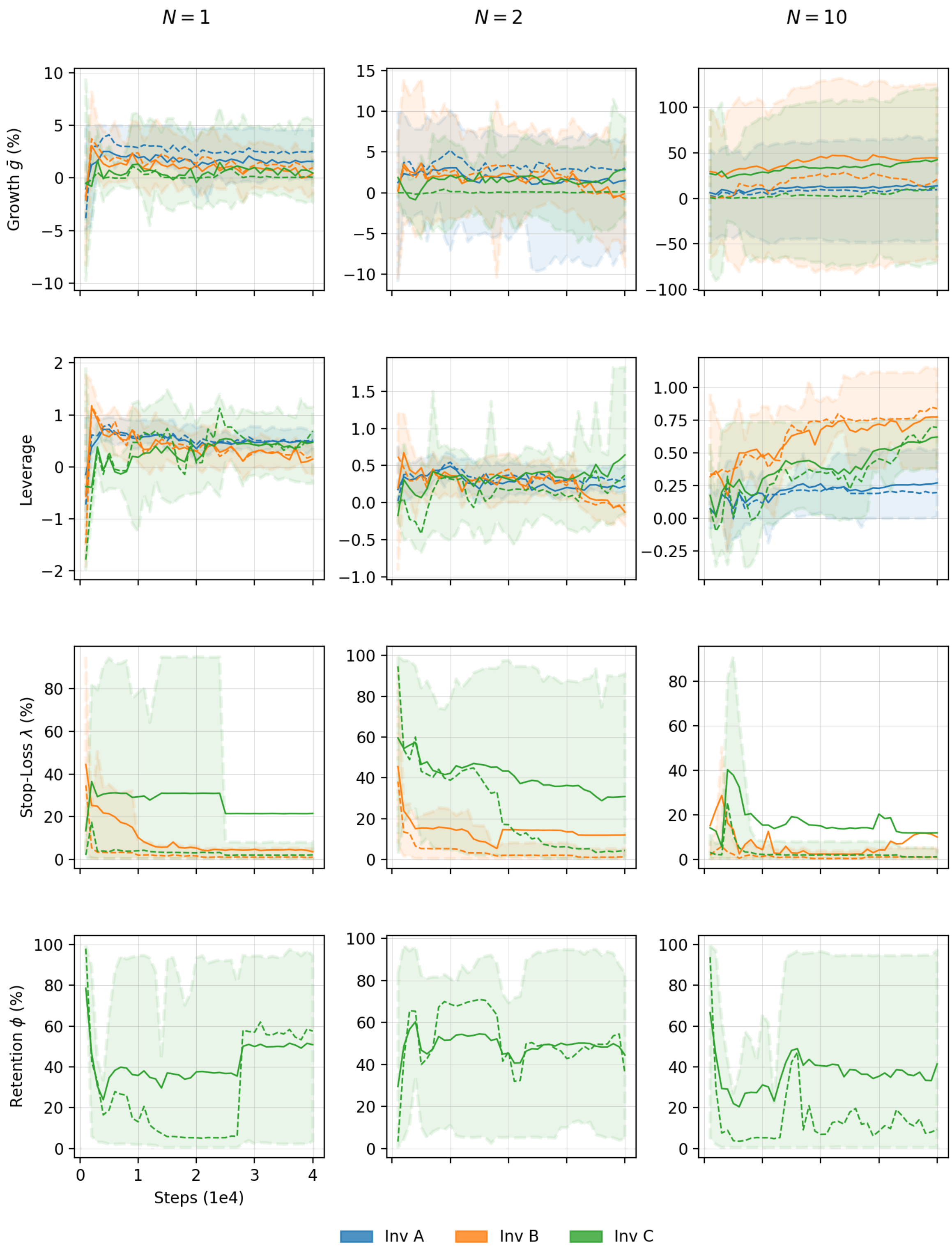

Figure 29: Summary of the dice roll environment agent performance with respect to actions across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions with dashed outline represents the range between the first and third quartiles.

Figure 30: Summary of the dice roll environment agent training across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions without outlines indicate one MAD about the means.

11.5 Geometric Brownian Motion

GBM is a model for entities that undergo self-replication which by definition is a multiplicative process. This occurs from the smallest chemical compound capable of building copies of itself at a fixed (compounding) growth rate to arguably the most complex known social system such as the global economy as capital bases expand and contract relative to their sizes [16]. In our case, a price P_t follows GBM if it evolves over time if it strictly adheres to the stochastic differential equation

$$\mathrm{d}P_t = P_t\left(\mu \mathrm{d}t + \sigma \mathrm{d}W_t\right) \tag{219}$$

where $W_t = \frac{1}{\sqrt{t}}\sum_{i=1}^{t}\xi_i$ with $\xi_i \sim \mathcal{N}(0,1)$ is a Wiener process as a limit of random walk. These are further defined by the increment $\mathrm{d}W_k = W_{t+k} - W_t = \mathcal{N}(0,k)$ and zero Pearson correlation $\mathbb{E}[\mathrm{d}W_{t+k}\mathrm{d}W_t] = \delta(k)\mathrm{d}t$ where $\delta(x)$ is the Kronecker delta function [5, 15, 16, 29]. The relative changes in price $\mathrm{d}P/P$ (a multiplicative process) are then to be independently drawn from a stationary normal distribution at each time step where

$$\ln|P_t| \sim \mathcal{N}\left(\left(\mu - \frac{\sigma^2}{2}\right)t, \sigma^2 t\right) \tag{220}$$

$$P_t = P_0 \exp\left[\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right] \tag{221}$$

with expected price $\mathbb{E}[P_t] = \exp[\mu t]$, median price $\exp[\left(\mu - \sigma^2/2\right)t]$, and time-averaged growth rate for a leveraged position in this asset is $\bar{g} = lr - (l\sigma)^2/2$. Observe that integrating the relative change $\mathrm{d}P/P$ yields the natural logarithm in Eq. (11), which is the basis for this entire project and was first highlighted almost 300 years ago [3].

Any quantity whose relative changes are random variables with finite mean and variance will behave like a GBM after a sufficiently long time [16]. Of course we know from Section 3.6 that there are countless process that do not have finite moments, and on top of this the central limit theorem and convergence of the law of large numbers is only valid in the asymptotic $t \to \infty$ limit [14, 95–97]. In reality, both moments are not only dependent on time but also on a myriad of other unknown variables, and the infinite time limit is simply not true. Additionally, we mention again that quantifying volatility with variance is in general a terrible idea [14].

Regardless, we persevere on with this model with the desire to see whether the agent can learn the optimal values $l^*_{\mathrm{GBM}} = \mu/\sigma^2$ and $g^*_{\mathrm{GBM}} = \mu^2/2\sigma^2$ [5, 15, 16, 104]. The agent must determine this with access to only the previous price P_{t-1} with and the new price P_t is sampled directly from Eq. (221), that is, with no knowledge of either μ or σ.

For our simulations of one to ten assets, the question is then how we should select the drift and variance in a fair manner. While randomly generating them should be considered unbiased, it is rightfully prone to accusations of cherry-picking. Instead, we approximate both variables with the closing values of the S&P500 (SPX) equity market index for the prior 120 years ending December 31st as a proxy for global markets. For this index using logarithmic returns we have annual drift $\mu = 5.16\%$, sample volatility $\sigma = 19.04\%$, and geometric mean $g = 3.10\%$. This choice of time intervals being yearly is so that the percentage changes are of the same order of magnitude as the previous gambles, and the selected year-end date is arbitrary. Regardless, this would imply $l^*_{\mathrm{GBM}} = 1.42$ and so $g^*_{\mathrm{GBM}} = 3.67\%$ under the very crude GBM assumption. Therefore, we have a situation where borrowing funds to take a leveraged position appears to be the best choice.

Before we discuss the results let us first emphasise how impossibly difficult this task would be for any human. Below we show three independent sequences of price evolution for 10 time steps with an initial value of 100:

1. 100.00 → 80.96 → 128.16 → 142.40 → 123.31 → 103.28 → 109.44 → 104.51 → 85.58 → 85.00

2. $100.00 \rightarrow 85.11 \rightarrow 69.05 \rightarrow 71.62 \rightarrow 84.12 \rightarrow 96.34 \rightarrow 82.44 \rightarrow 93.56 \rightarrow 90.35 \rightarrow 120.94$

3. $100.00 \rightarrow 76.64 \rightarrow 80.18 \rightarrow 108.10 \rightarrow 113.83 \rightarrow 111.68 \rightarrow 150.47 \rightarrow 199.16 \rightarrow 265.36 \rightarrow 320.05$

What do these sequences have in common? They are all generated using Eq. (221) with the S&P500 parameters. Is it possible for any human that has ever lived to recognise this pattern without external support? Suppose there was an individual that was confident in identifying the trend without using very explicit tailor-made statistical tools.

Next let us provide them \$100 and offer them a game based on their understanding of the sequence. They are allowed to bet any portion of the given \$100 and their payout will evolve identically to a statistically indistinguishable fourth sequence and the game is played for $3,000$ rounds. We are also even willing to provide them infinite leverage in either direction. Would they play the game? If no, they get to keep the \$100 and continue on with their day. If yes, how would they select the optimal leverage?

The chance they come to conclusion "1.4x leverage please" is quite slim, though we have yet to experimentally validate this claim.

To make their life easier, let's say we provided them millions of sequences of this data and allowed them to use any statistical software they wanted but were able to somehow restrict their access to anything resembling the results contained in [5, 15, 16]. In this case they would be able calculate the mean differences in prices and empirically determine the lognormal drift μ and perhaps even the noise σ. Then we are again at the discussion of Section 1.2 regarding how they would determine optimal leverage. Contemporary decision theory would at the very least suggest betting everything with $l_t = 1 \ \forall \ t$.

Instead of assisting them, now let's dial it up a notch. Imagine only providing the individual one price at a time and ask them to make a choice on what leverage to take for the next time step. For the first sequence with this would be: have 100 and decide l_1, observe 80.96 calculate profit or loss and decide l_2, observe 128.16 calculate profit or loss and decide l_3, and so on till the end. This way the individual has minimal knowledge of the underlying gamble and must learn through trial and error as to what may be the optimal. Importantly, because their capital is stake at each time step and there are only $3,000$ available steps, they cannot simply set zero leverage to gather increasing amounts of data, that is, there is a penalty for not maximising wealth over time. How many people would be capable of deducing $l^* \approx 1.4$ via this approach? Even if we kept providing them \$100 and allowed them to repeatedly play the game, how long would it take for them to get the correct answer?

This is precisely the approach taken by the model-free reinforcement learning agent and the penalty for not increasing wealth over time is perfectly encoded in Eq. (213). The training of the agent consists of repeatedly playing the game and learning from its past failures in order to optimise its current leverage selection. GBM is clearly a far more formidable environment compared to the prior two, yet the real-world is exponentially more complex. Therefore, being able to demonstrate the agent is capable of conquering GBM would at the very least set a lower bound on the power of our formulation.

There is however a subtle issue with this optimal leverage that we encountered in Section 1.2, namely there is a possibility of losing all wealth (reaching $V_{\min}$) when operating with $l^* > 1$ in a single time step for risky gambles. For GBM, the $l^* > 1$ condition is not necessary as the Weiner process is technically unbounded $W_t \in (\infty, \infty)$ and so the price of any asset $P_t = P_{t-1}e^{R_t}$ is contained within $P_t \in (0, \infty)$. However, we have restricted the portfolio value to be $V_t \in [V_{\min}, \infty)$ where $V_{\min} = \psi V_0 \geq \gamma$ is strictly required for Q-value convergence. Using the hyperparmeters in Table 2, our experiments exclusively utilise $V_{\min} = \psi V_0 = 0.01 \cdot 10^4 = 100$.

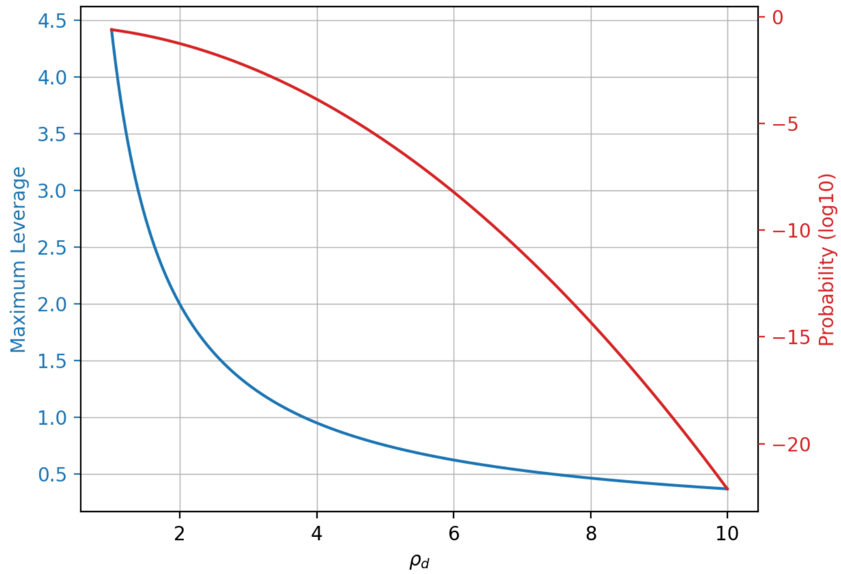

Figure 31: Maximum possible leverage while at $V_{t-1} = 2V_{\min}$ to protect from a ρ_d-sigma down move using the S&P500 GBM approximation. The log10 probabilities of each ρ_d-sigma down move occurring are also displayed.

Therefore, if we were to take a position in a single asset and perfectly replicate its performance, we can use l^*_{GBM} to obtain g^*_{GBM} in the long-run as discussed above. On the other hand, for the model-free agent to function we are unable to train with values in the $(0, 100)$ interval which contains infinite real numbers. The effect of this truncation is that we have a reduced tolerance for loss. This means the optimal leverage taken must be strictly less than the GBM optimal $l^* < l^*_{\text{GBM}}$. The real question is how much lower.

This can be demonstrated by considering a ρ_d-sigma down move for which we have the bankruptcy criterion and maximum corresponding leverage

$$V_{\min} = V_{t-1} \exp\left[l_{\max}\left(\mu - \frac{\sigma^2}{2} - \rho_d\sigma\right)\right] \tag{222}$$

$$l_{\max} = \frac{2}{\sigma(2\rho_d + \sigma) - 2\mu} \ln\left|\frac{V_{t-1}}{V_{\min}}\right| \tag{223}$$

where for a normally distributed variable we have $\rho_d \in (0, \infty)$ and as expected the maximum leverage scales with the current portfolio valuation. Therefore, to completely protect from all down moves $\rho_d \to \infty$ we would technically require $l_{\max} \to 0$, though this is obviously not a wealth-maximising solution.

In Fig. 31 we illustrate this for the S&P500 GBM approximation alongside the probability of each down move and arbitrarily set the ratio $V_{t-1}/V_{\min} = 2$. Observe that operating at constant $l^*_{\text{GBM}} = 1.42$ would roughly only protect the agent for up to a 2.8-sigma down move that has a 1 in 200 chance of occurring. For exponentially rarer downside events, l^*_{GBM} would cause immediate bankruptcy in a single time step if at $V_t = 2V_{\min}$. This concept is astronomically more important when working with fat-tailed distributions such as those discussed in Section 1.3 and 3.6 since the magnitude of tail events have the potential to be catastrophic, that is, black swans.

The crucial distinction is then between the optimal growth rate of an asset following GBM l^*_{GBM} and the portfolio wealth maximising leverage l^*. The asset following GBM is able to approach zero and in the long-run will achieve g^*_{GBM}. Meanwhile, the portfolio must be maximised under the constraint of never reaching $V_{\min}$ in our formulation. Therefore, as the singular purpose of the agent is to increase wealth, it will be interesting to examine what level of downside move tolerance it considers acceptable as its choice of leverage can be interpreted determining the $l^* = l_{\max}$ as a very rough guide.

In Figs. 32-33 the results are shown in an identical fashion to the prior experiments. In the single asset Investor A case we observe the medians $l = 1.18$ and $g = 1.70\%$ with the GBM optimal l^*_{GBM} and g^*_{GBM} within the first and third quartiles. Therefore, the model-free agent is able to sequentially view prices from the GBM distribution and autonomously construct incredibly robust betting strategies with substantial positive skew. Notice how the mean growth is significantly higher than the third quartile highlighting how dangerous it would be to utilise this quantity as indicative of the complete sample, that is, the performance of any random investor.

Similarly, Investors B and C are also clearly capable of learning the dynamics of the gamble with far higher first quartiles at the cost of lower median growths. Investor B again converts itself to Investor A, while Investor C choosing to continue to retain 40% of its profits at each time step. Recall again that for all investors the observed action has significant noise attached to it, formally $l^* = l^*_{\mathrm{act}} + 6 \cdot \mathcal{N}(0, 0.1)$.

The multi-asset cases again show how terribly inflated using mean time-average growth rates are compared to the median. For the ten-asset case, we purposefully retain the enormous magnitudes to further emphasise how the massive performance of a single element of the sample skews the complete picture. The reason for these valuations is due to miniscule but finite probability of sampling very large prices from the lognormal distribution, with the returns then amplified by very high leverage.

Overall, the GBM experiment was a resounding success. The model-free agent was able to learn the underlying dynamics of the system with no explicit knowledge of either drift μ or volatility σ. Through repeated observations of prices under the MDP assumption and very intelligent trial-and-error, the agent maximised wealth for multiple assets following GBM.

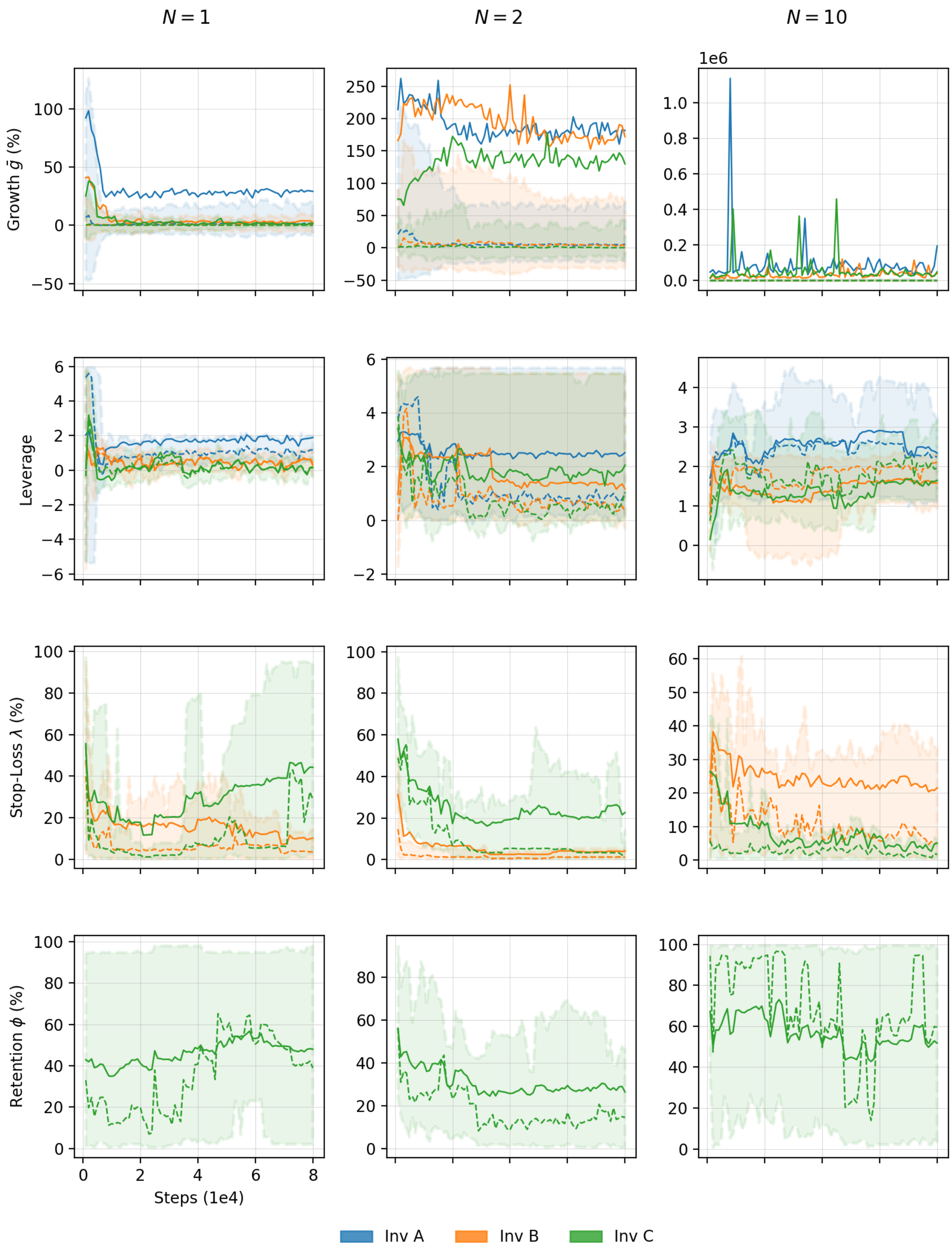

Figure 32: Summary of the GBM environment agent performance with respect to actions across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions with dashed outline represents the range between the first and third quartiles.

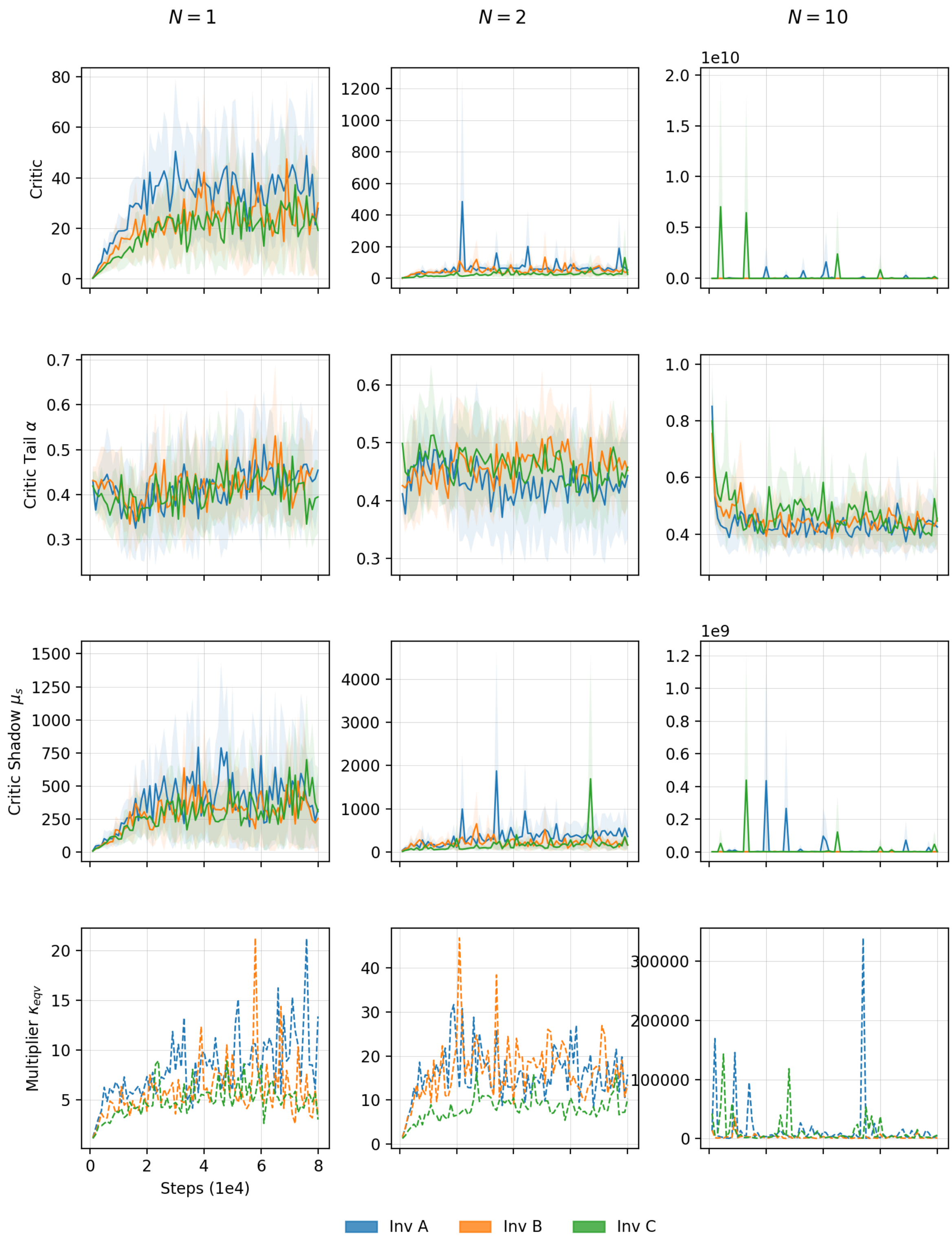

Figure 33: Summary of the GBM environment agent training across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions without outlines indicate one MAD about the means.

11.6 GBM with Discrete Compounding

Our final investigation is regarding the impact of discretely compounding portfolio valuations using GBM as discussed in implementation detail (iii) of Section 11.1. For both the continuous $V_t = V_{t-1}e^{R_t^c}$ and $V_t = V_{t-1}(1 + R_t^d)$ discrete valuations to be identical we require $|R_t^c| \ll 1$. Therefore, we anticipate that over the long-run once the agent has learned optimal actions $\bar{g}_c = 1.70\% \geq \bar{g}_d$. The results are shown in Figs. 34-35.

For the single asset Investor A we have medians $l = 0.65$ and $g = 0.72\%$ confirming our expectations. Reduced leverage is utilised since there is less potential for exponentially compounding gains, hence there is less need to operate with higher exposure.

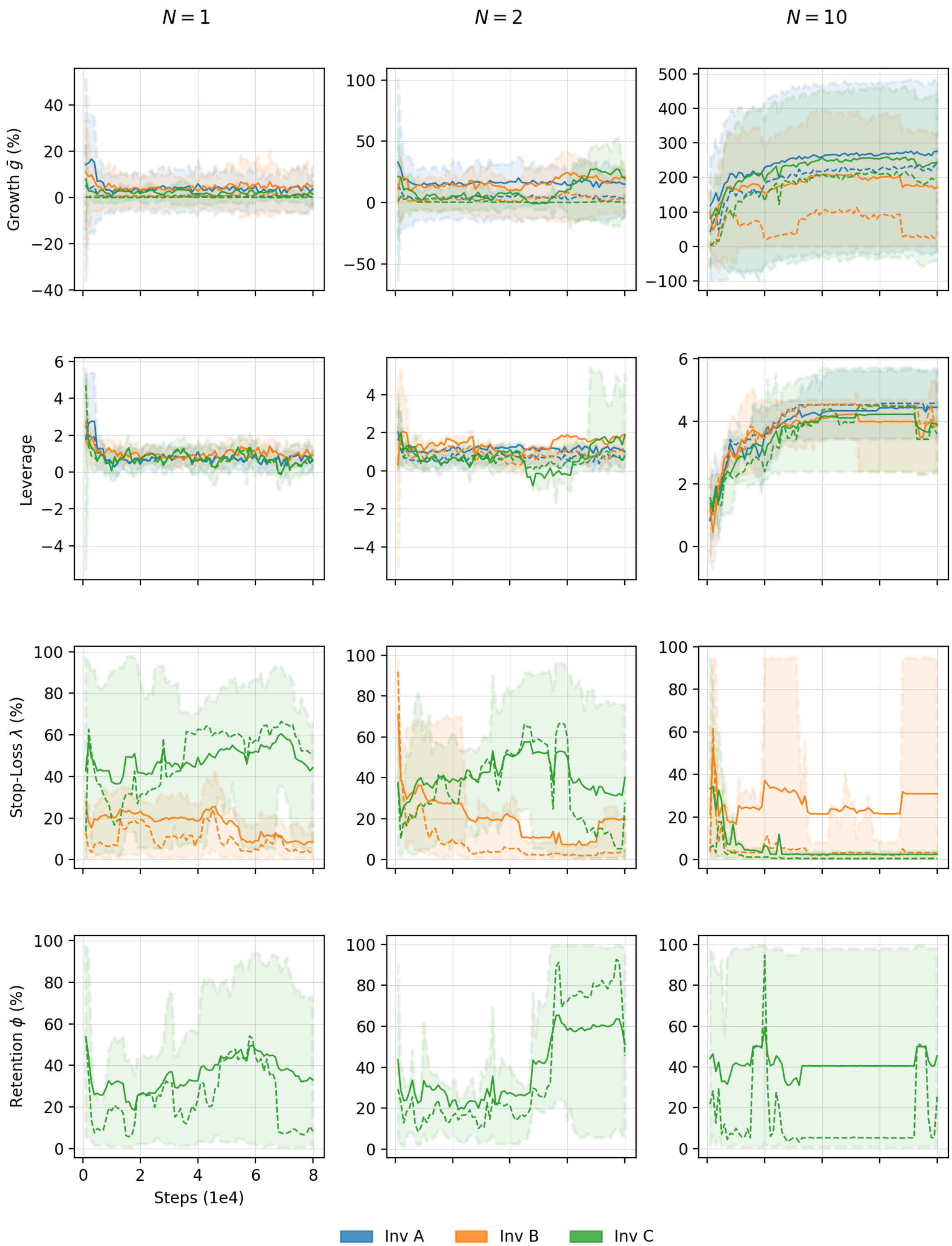

Figure 34: Summary of the GBM with discrete compounding environment agent performance with respect to actions across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions with dashed outline represents the range between the first and third quartiles.

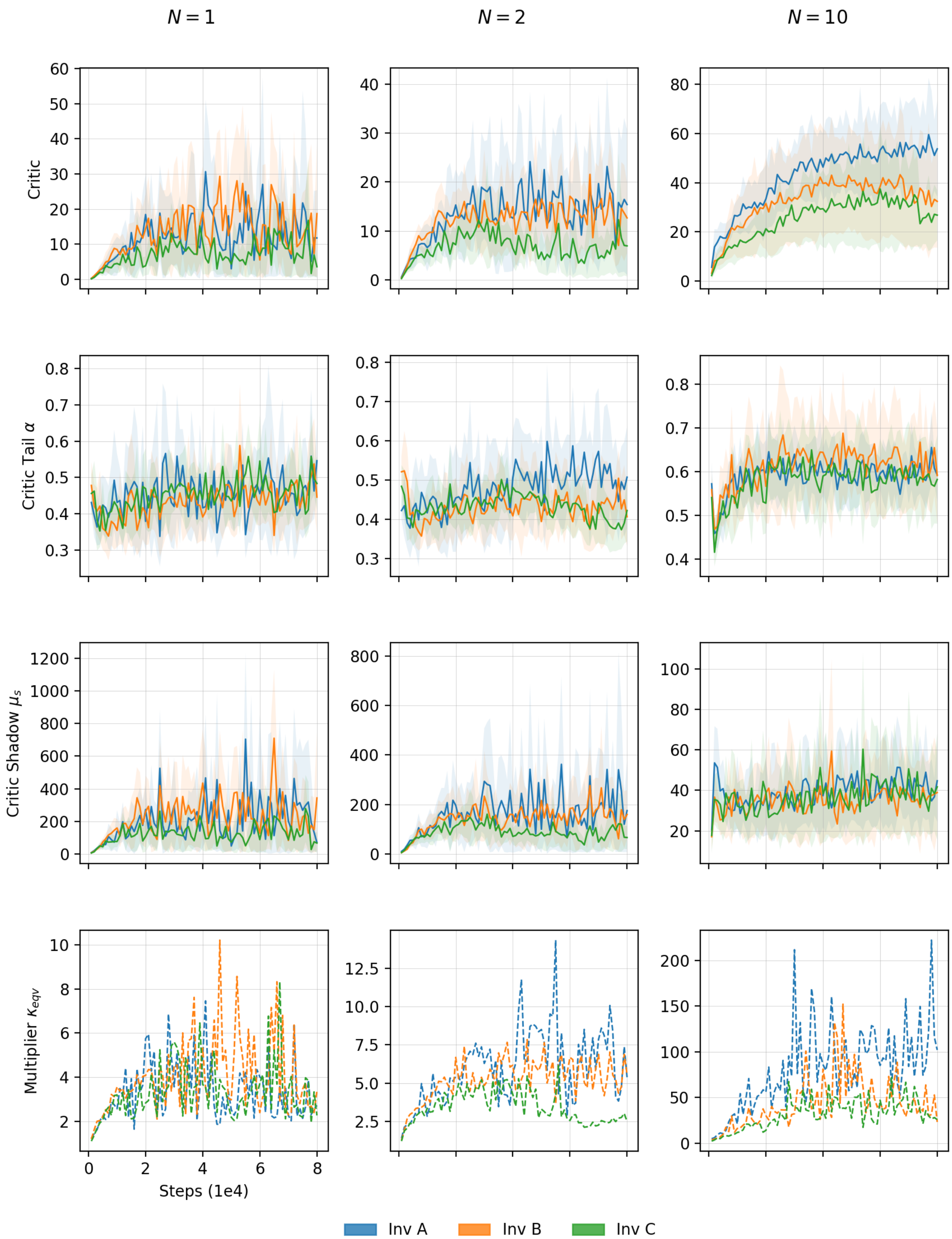

Figure 35: Summary of the GBM with discrete compounding environment agent training across N number of identical assets for Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions without outlines indicate one MAD about the means.

12 Navigating Financial Markets [Preliminary]

This section utilises all previous results but no reference to its contents is made in any sections till completion of experiments.

As discussed previously in Section 11, financial gambles represent a historical motivation for multiplicative dynamics and the ability to then very accurately parametrise the reward signals makes them an ideal proving ground for our ideas, but our formulation is not even the slightest limited to this domain. In this section we extend the previous successes to the real-world by modeling fully autonomous agent decision-making in financial markets.

12.1 Simulating History

For financial markets we use publicly available daily close data for US listed and USD denominated financial assets sourced freely from Stooq [188]. These include the equity indices S&P500 (SPX), Nasdaq 100 (NDX), Dow Jones Industrial Average (DJIA), along with 26 of the 30 earliest included components of the Dow Jones. Additionally, we also include several front month futures contracts for commodities including gold, silver, high grade copper, platinum, palladium, WTI crude oil, RBOB gasoline, lumber, live cattle, coffee, and orange juice. The daily time series of prices spans from the earlier of either the first available date from Stooq or the 10th October 1985 to 19th November 2021. This way we are able to incorporate numerous macroeconomic cycles, that is, recessions and expansions to encapsulate the past. At most, this includes *over 9,000* daily closing prices being equivalent to over 36 years of data using the convention of 252 days per year and 5 days per week. Note additionally that US capital markets dominate global markets and so the prices of these securities in aggregate could be interpreted as a POMDP proxy incorporating information for all assets world-wide.

As alluded to in Section 11.2, a primary constraint in our formulation is that the agent is forced to learn under the MDP assumption which was reasonable in environments where the underlying gamble is unchanged over time, therefore optimal decision making should remain constant regardless of the amount of time steps. Recall that even for the GBM environments, while the prices of assets where lognormally unbounded, the underlying parameters μ and σ remain fixed and so convergence to know optimal leverages was achieved. The real-world however is non-stationary and so the (countless unknown) underlying catalysts for change vary across time in unpredictable ways, in other words, optimal risk-taking is history dependent.

Therefore, on top of our environment representing a extremely complex POMDP, we opt to relax the Markovian assumption ($D = 1$) of only solely using the current prices to determine next action. Instead, we examine the use of $D > 1$ which then leads to a HDP as discussed in Section 3.8. In what follows, we assume that for any action and any two histories, there exists a feature map $\phi(h_t) = \phi(\tilde{h}_t) = \xi_t = s_t \cup \Gamma^+_{t-1}$ for the environment to be a QDP where the agent utilises D past states to determine the next action. This approximation is likely invalid which means convergence to unique optimal Q-values will not occur, however, the best need not be the enemy of the good, and so the autonomous construction of (sub-optimal) profitable trading strategies would still be considered a success.

Another principle challenge when working with time series data involves its non-stationary and non-i.i.d. nature, namely the presence of heteroscedasticity and serial correlation [189–199]. These features theoretically invalidate the use of simple randomised training-test set splits all the way to randomised k-fold cross-validations (CV) [200] as breaking the temporal order of the time series could result in significant reduction in information. Furthermore, the benefits of utilising the more advanced repeated (randomised) k-fold CV are somewhat dubious given its immense computational requirements [201–203].

There are many alternative approaches such as out-of-sample (OOS) procedures where the series is split into a training and subsequent testing window that may feature a gap between windows. This procedure can be extended using repeated holdout OOS (HOOS) [204] where the time series is randomly split (with or without gap) and subset of the total data on both sides of this split are utilised with numerous repetitions of this process. For example, for a series consisting of 10^3 observations using 70% of the sample with 60%/10% splits (no gap), a random point within $[1, 300]$ is selected, and the next 600 and following 100 points are used for training and testing respectively [198]. Modified variants of k-fold CV also exists such as blocked CV [189] where the data is not randomly shuffled, rather, the series is split into k contiguous blocks ensuring that the training set always precedes evaluation with $k-1$ training iterations in order to accurately back test a model's future predictive power. This approach can be extended to hv-blocked CV [190] wherein some number of adjacent observations between the training and test folds are also removed (also known as 'purged' [197] or 'time-series' [199] CV).

For stationary time series, literature finds the use of the usual randomised CV to be more accurate than OOS, underperform blocked CV, and for directional time series forecasting be even further inferior to hv-blocked CV [191, 192, 195]. Equivalence between the use OOS and hv-blocked CV on the estimation of large (stationary) time-ordered Twitter datasets has also been demonstrated along with the conclusive finding that the regular randomised CV should never be used for realistic time-ordered data [194].

For the real-world, [198] use 77 non-stationary univariate numerical time series from finance, physics, economy, energy, and meteorology to find lowest average predictive error in increasing order to be by HOOS, block CV, hv-block CV, and with standard CV being by far the worst. They also utilise Bayesian signed-rank tests to find HOOS to be superior, and that its performance remains relatively constant with increasing sample size. A partial explanation for this success relative to OOS is that non-stationary variables are comprised of temporal nuances relating to the future that may not have revealed themselves in the past and so the use of multiple repetitions are less prone to the biases of single splits [193, 198, 204]. Compared to CV methods, HOOS enables lengthier training and tests that better capture hidden structures within the time series [198, 204]. This finding is crucial as financial markets are an obvious poster child for non-stationary data since the underlying probability distributions governing price sampling radically change over time. Therefore, for our experiments we will opt to utilise HOOS as [198] are confident that their univariate findings should extend for other more general types of time series.

Building on this, one benefit of reinforcement learning by definition is that it is a non-i.i.d. process since the training and evaluation of agent performance occurs sequentially and so it is naturally well-suited for time series. However, there are two prevalent issues: 1. The coordination between training and evaluation sets, and 2. The presence of sufficient amounts of unique historical data resulting in robust learning. The first issue makes the use of any type CV not practical, while implementing HOOS is seamless requiring virtually no modification. This because we can force evaluation episodes to occur at the conclusion of each training interval while ensuring they take place in the relative future, we will expand on this later. The second problem is far more severe as we are limited to roughly $9,000$ historical observations and so overfitting becomes a substantial concern. For example, the experiments in Sections 10 and 11 consists of 10 uniquely initialised trials from $4 \cdot 10^4$ up to $3 \cdot 10^5$ training steps which is critical for simulating robust out-of-sample learning. A potential solution to the lack of data could involve bootstrapping, however this again has the potential to (silently) destroy the non-stationary structure while also being computationally expensive for repeated large sampling necessary for reinforcement learning. Some well-known examples include the moving block jackknife [205] and stationary bootstrap [206].

Instead, we propose the use of a block shuffling method to generate near-infinite positively correlated but unique time series. The method involves randomly sampling a smaller sequential time series of $M < N$ elements, splitting M into equal M/d blocks with $d \leq M$, and then randomly permutating the components of each d block with larger d leading to more 'random' shuffling. The purpose is to generate a fairly random (non-parametric) bootstrap for known historical data while preserving the overall long-term trends and structure. Effectively we are assuming that all prices within each d interval are indistinguishable but through this method we are confident that the agent's performance across a noisy history is tested while assured that the simulated prices are sensible over the long-run.

Overall, the total number of unique (positively correlated) sequences for $M \mod d = r$ can be calculated with

$$u = (N - M) \cdot (d!)^{(M-r)/d} + (r! - \delta(r) - \delta(r-1)) \tag{224}$$

where $\delta(x)$ is the usual Dirac delta function. An order of magnitude analysis can then be conducted when approximating $r_k = 0$ which leads to

$$\begin{aligned} \log_{10}|u| &= \log_{10}|N - M| + \frac{M}{d}\log_{10}|d!| && (225)\\ &\approx \log_{10}|N - M| + \frac{M}{d}\left[d\log_{10}|d| - d\log_{10}|e| + \frac{1}{2}\log_{10}|2\pi d| + \Theta\left(\log_{10}|d|\right)\right] \\ &\approx \log_{10}|N - M| + M\log_{10}\left|\frac{d}{e}\right| && (226) \end{aligned}$$

where the Stirling (or de Moivre) approximation is valid for $d \to \infty$. Notice the trade-off where increasing d causing the logarithm to asymptotically diverge while being inversely suppressed, with the latter effect dominating as $d \to M$ leading to the relation $u \propto d^M$ with crude bounds $u_k \sim [1, M^M]$.

On the topic of coordinating training and evaluation again, in interest of objectivity and fairness, each of the repeat evaluation episodes should be sampled (and shuffled) from occurring in time after the concluding point of each training time series of length M_T. Furthermore, their occurrences should also be initiated within some reasonable gap $G_E \geq 0$ after training time series in order to generate robust learning. To ensure diversity across evaluations, we set G_E to be a uniform random variable $G_E \sim U(G_{\min}, G_{\max})$ with $G_E \in \mathbb{Z}^+$ sampled at the beginning of each evaluation episode. The agent will then be tested on intervals sampled from points $M_T + G_E$ form the complete sequence N for a total of M_E (shuffled) observations. Therefore, for continuous agent training, we provide the agent a unique randomly shuffled series with $d_T \leq M_T < N$, and similarly for evaluation with $d_E \leq M_E < N$ and random window spacing $G_E \in [G_{\min}, G_{\max}]$.

For our experiments, using 252 day years and 5 day weeks, we utilise $M_T = 10^3$ with $d_T = 10$, and $M_E = 252$ with $d_E = 5$ and $G_E \in [5, 20]$. Therefore, the agent is repeatedly trained with a randomly sampled 4 year time series with every 10 day interval being randomised and similarly evaluated on random 1 year slices with 5 day prices shuffled beginning 1-4 weeks after the end of training. The end of training is defined as three possible scenarios being at completion, termination due to agent failure, or intermediately within the training series. Through this method, cumulative number of training steps C per trial will result in C/M_T unique training time series intervals and ensures M_T will always precedes M_E.

The minimum number of unique (positively correlated) sequences for $r_k = 0$ are then

$$u_k = (N - M_T - G_{\max} - M_E) \cdot (d_k!)^{M_k/d_k} \tag{227}$$

$$\log_{10}|u_k| \approx \log_{10}|N - M_T - G_{\max} - M_E| + M_k \log_{10}\left|\frac{d_k}{e}\right| \tag{228}$$

which for $N = 9,000$ yields $u_T \sim 10^{660}$ possible training sets and for each of these there exists $(d_E!)^{M_E/d_E} \sim 10^{104}$ corresponding unique evaluation sets, totaling $u_E \sim 10^{108}$. Observe that agent inference (lower d) mimics true history far more faithfully. Ideally, we hypothesise, that repeat training over many of tens of thousands of steps we are able overcome the time dependence issue, though training for millions of steps would likely again lead to overfitting which would then again demand more advanced history simulation. Another clear limitation of this approach is that the agent will not be permitted to train on the most recent $G_{\max} + M_E$ observations.

Regarding the number of possible unique sequences, for reference, there are estimated to be 10^{80} atoms in the observable universe [27], the sizes of the search tree required identify the winning strategy for Chess and Go are approximately 10^{124} and 10^{360} respectively, and using the unanimously taught contemporary decision theory for the simple coin flip gamble in Section 1.2, the magnitude of error some of the most advanced 'risk-reward' maximising practitioners would encounter is 10^{153}. This demonstration of course does not negate the overall long-term positive correlation across the time series.

In terms of history dependence, there are myriad of combinations of what specific D past states to select. We choose to select the most recent consecutive daily D states to determine the next action as our focus is an exploratory analysis as opposed to designing a live trading strategy. By utilising the most recent D daily prices, the agent may be able to better observe shorter term trends to capitalise on them, that is, effectively day-trading. Instead, if we were concerned with quarterly returns, combining weekly or monthly states may be superior.

We propose seven environments shown in Table 3 with the 26 DJIA components detailed in Table 4. Note that in all cases we train the agent using the default TD3 algorithm in Algorithm 2 with Investor A defined in Section 11.1, initialise all 10 trials with 10^3 warm-up steps, and strictly utilise discrete portfolio compounding. The number of cumulative training steps per trial C is dependent of the complexity of the environment and is split into C/M_T unique time series intervals. For inference, we utilise 10 evaluation episodes of length M_E occurring every intervals of 10^3 training steps. In the event of learning termination before M_T due to agent failure, the evaluations will continue to always take place in these intervals and occur during partial M_T intervals. This procedure can be considered the reinforcement learning equivalent of HOOS. Investigating the use of global stop losses and retention ratios is outside the scope of this work though the code is provided.

Furthermore, we also limit our examination to $D = 1, 3, 5$ consecutive daily sets of prices noting the linear increase in the cardinality $|\Xi_D| = 1 + D \cdot |\Xi|$ quickly leading to far more challenging environments. Additionally, we modify several parameters from Table 2 where $V_{\max} = 10^6$, prices of each asset P_i are directly sourced from our shuffled data, and we universally set maximum absolute leverage of $\eta = 3$ for all assets.

In this way, the agent is effectively trading contract for differences (CFDs) on the prices (or index values) of all available assets with a cap on the amount of leverage for each asset. Furthermore, with M_E=252 length evaluation episodes, the maximum possible time-average growth rate is $\bar{g} = \exp\left[\frac{1}{M_E}\ln\left|\frac{V_{\max}}{V_0}\right|\right] - 1 = (V_{\max}/V_0)^{M_E} - 1 = 1.84\%$ per time step translating to an maximum annualised return of $\bar{g}_{\text{pa}} = (1 + \bar{g})^{252} - 1 = 9,000\%$.

	Components	States $\|\Xi\|$	Actions $\|\mathcal{A}\|$	N	Start Date
Equity Indices					
SNP	SPX	2	1	$9,115$	1985-10-01
USEI	SPX, NDX, DJIA,	4	3	$9,115$	1985-10-01
DJI	USEI + 26 DJIA Components	30	29	$7,972$	1990-03-26
Broader Markets					
Minor	USEI + Gold, Silver, WTI	7	6	$9,071$	1985-10-01
Medium	Minor + Cooper, Platinum, Lumber	10	9	$9,058$	1985-10-01
Major	Medium + Palladium, RBOB, Cattle, Coffee, OJ	15	14	$8,971$	1985-10-01
Full	Major + 26 DJIA Components	41	40	$7,839$	1990-03-26

Table 3: Multiplicative market environments with historical closing daily data up to 26th November 2021 sourced from Stooq [188]. The count of data points N refers to when closing values exist for all included assets.

Ticker	Company	Exchange	Industry	Weighting (%)	Date Added
AAPL	Apple Inc.	NASDAQ	Information Technology	2.76	2015-03-19
AMGN	Amgen	NASDAQ	Biopharmaceutical	3.84	2020-08-31
AXP	American Express	NYSE	Financial Services	3.29	1982-08-30
BA	Boeing	NYSE	Aerospace and Defence	4.01	1987-03-12
CAT	Caterpillar Inc.	NYSE	Construction and Mining	3.73	1991-05-06
CRM	Salesforce	NYSE	Information Technology	5.43	2020-08-31
CSCO	Cisco Systems	NASDAQ	Information Technology	1.03	2009-06-08
CVX	Chevron Corporation	NYSE	Petroleum Industry	2.07	2008-02-19
DIS	The Walt Disney Company	NYSE	Broadcasting and Entertainment	3.18	1991-05-06
HD	The Home Depot	NYSE	Home Improvement	6.65	1999-11-01
HON	Honeywell	NASDAQ	Conglomerate	4.12	2020-08-31
IBM	IBM	NYSE	Information Technology	2.64	1979-06-29
INTC	Intel	NASDAQ	Semiconductor Industry	1.03	1999-11-01
JNJ	Johnson & Johnson	NYSE	Pharmaceutical Industry	3.04	1997-03-17
JPM	JPMorgan Chase	NYSE	Financial Services	3.13	1991-05-06
KO	The Coca-Cola Company	NYSE	Soft Drink	1.01	1987-03-12
MCD	McDonald's	NYSE	Food Industry	4.51	1985-10-30
MMM	3M	NYSE	Conglomerate	3.38	1976-08-09
MRK	Merck & Co.	NYSE	Pharmaceutical Industry	1.48	1979-06-29
MSFT	Microsoft	NASDAQ	Information Technology	5.72	1999-11-01
NKE	Nike, Inc.	NYSE	Apparel	2.93	2013-09-20
PG	Procter & Gamble	NYSE	Fast-Moving Consumer Goods	2.61	1932-05-26
UNH	UnitedHealth Group	NYSE	Managed Health Care	7.88	2012-09-24
VZ	Verizon Communications	NYSE	Telecommunication	0.97	2004-04-08
WBA	Walgreens Boots Alliance	NASDAQ	Retailing	0.89	2018-06-26
WMT	Walmart	NYSE	Retailing	2.69	1997-03-17

Table 4: The included 26 of total 30 DJIA components with historical closing daily data starting from 26th March 1990 to 26th November 2021 sourced from Stooq [188]. The DJIA index weights are observed at 19th October 2021.

12.2 Equity Indices

12.3 Broader Markets

13 Cost-Effective Risk Mitigation

Throughout this work we have made many substantial claims regarding the implications of multiplicative dynamics to the real-world that we succinctly repeat. In Section 1.1 we discussed how there can exist distinct path preferences to a final goal even if their expectation values at each step are identical, a fact that goes in stark contrast to the teaching of behavioural finance. In Section 1.2 using the very simple coin flip gamble, we showed how the probability-based expectation maximisation framework does not maximise the final wealth of any random individual over time. Rather, the optimal leverage was determined using the Kelly criterion that emphasises the maximising the geometric mean through the avoidance of steep losses. In Section 1.3, an explanation for the pitfalls of expectations is explained through the conflating of probability and payoffs using Jensen's inequality.

Implications for this are discussed in Section 1.4 wherein we find that virtually all people with formal training in economics or STEM are unaware of this error given the entirety of their education is devoid of time-averages, let alone the concept of non-ergodicity. As a direct application, we discussed how this misconception is prevalent throughout the financial industry (all the way to the top [28]) as the established prescription regarding maximising expected excess returns (alpha) simply does not work, that is, except for a very small minority who due to pure chance receive sequences of very favourable payoffs. In Section 1.5 we highlighted that through the use of these flawed strategies, countless trillions in pension assets [32, 33] are effectively being managed by people acting as if they can travel through the multiverse and pool the average outcome. Despite this, we highlighted that there does not appear to exist any robust functioning framework to generally maximise the time-average growth rate for all conceivable environments.

We then went on a tangent discussing model-free reinforcement learning including several aspects applicable to both additive and multiplicative that culminated in the experimental investigations in Section 10. Meanwhile, in Sections 4-7 we showed that it was possible to reformulate the existing tools of reinforcement learning to be applicable to multiplicative dynamics provided a couple conditions were satisfied.

In Section 11 we got back on track and empirically validated that it is possible to for model-free agents to autonomously maximise wealth for several environments of increasing complexity. This was done by the agent deciding to take actions that raise time-average growth rate as opposed to maximising expectations. By explicitly showing that this approach is technically feasible, we presented an independent method to increase the applicability of reinforcement learning to now encompass all multiplicative environments.

Now that we are very familiar with contents of Sections 1 and 11, in this brief final section we present one last idea inspired entirely by [11, 18–24]. The purpose of this second and final laser-guided surgical exposé is to further emphasise how ineffective the use of expectation values are when applied to multiplicative process. Many may still shun the outcomes of the coin flip questionnaire of Section 1.4 down to poor design and or the fact that it is an extremely trivial game with seemingly no tangible connections to the real-world. The contents of this section on the other hand are not going to be nearly as easy to dismiss as they will reveal a forbidden secret. The model-free agent will fully autonomously self-learn a strategy that indisputably increases wealth while simultaneously reducing the amount of risk it takes, something contemporary decision theory would universally agree is impossible.

13.1 Insurance Safe Haven

The key message of [11, 18–24] is the importance cost-effective risk mitigation through the use of "safe havens". By allocating of small portion of your total uninsured (U) capital into such a safe have (SH), you are able to construct an insured (I) portfolio. The purpose of this insurance policy is then to enable any random investor to avoid the impact of steep losses and consequently preserve a high time-average growth rate $\bar{g}$, which is synonymous with maximising wealth. A safe haven must be both risk-mitigating, in that it has the ability to offset large percentage declines, but it also must be cost-effective such that its out-right price does not decrease the growth rate. Hence, we have the very strict requirement where over time the insured portfolio must be universally superior to the base uninsured portfolio. This is generally expressed in terms of the compounding return as $\bar{g}^{\mathrm{I}} > \bar{g}^{\mathrm{U}}$.

To show the impact of such a safe haven, consider again the previous dice rolling payoff in Eq. (234) which is now treated as an uninsured portfolio

$$R^{\mathrm{U}}_{t+1} = \begin{cases} +50\%, & p_1 = \frac{1}{6} \\ +5\%, & p_2 = \frac{2}{3} \\ -50\%, & p_3 = \frac{1}{6} \end{cases} \tag{229}$$

with $\mathbb{E}[R^{\mathrm{U}}_{t+1}] = 3.3\%$, $\sigma^{\mathrm{U}}_{t+1} = 29.0\%$, and $\mathrm{MAD}^{\mathrm{U}}_{t+1} = 17.8\%$. Now suppose there exists a complimentary insurance policy with payoffs

$$R^{\mathrm{SH}}_{t+1} = \begin{cases} -100\%, & p_1 = \frac{1}{6} \\ -100\%, & p_2 = \frac{2}{3} \\ +500\%, & p_3 = \frac{1}{6} \end{cases} \tag{230}$$

which by very explicit design yields $\mathbb{E}[R^{\mathrm{SH}}_{t+1}] = 0\%$, $\sigma^{\mathrm{SH}}_{t+1} = 224\%\cdot$, and $\mathrm{MAD}^{\mathrm{SH}}_{t+1} = 167\%\cdot$. The nature of this policy is that during the steep -50% decline in R^{U}_{t+1} the insurance produces an explosive gain, however, 83% of the time the safe haven leads to a complete loss of the capital allocated to the policy. Over a large number of trials, the safe haven is therefore expected to produce a 0% return on average.

The insured portfolio R^{I}_{t+1} is then composed of some combination of both R^{U}_{t+1} and R^{SH}_{t+1} with the return payoff structure

$$R^{\mathrm{I}}_{t+1} = l_t \cdot R^{\mathrm{U}}_{t+1} + (1 - l_t) \cdot R^{\mathrm{SH}}_{t+1} = \begin{cases} 150\% \cdot l_t - 100\%, & p_1 = \frac{1}{6} \\ 105\% \cdot l_t - 100\%, & p_2 = \frac{2}{3} \\ 500\% - 550\% \cdot l_t, & p_3 = \frac{1}{6} \end{cases} \tag{231}$$

and due to linearity, the expectation value is

$$\mathbb{E}[R^{\mathrm{I}}_{t+1}] = l_t \cdot \mathbb{E}[R^{\mathrm{U}}_{t+1}] + (1 - l_t) \cdot \mathbb{E}[R^{\mathrm{SH}}_{t+1}] = l_t \cdot \mathbb{E}[R^{\mathrm{U}}_{t+1}] = 3.3\% \cdot l_t \tag{232}$$

and so it appears we have arrived right back where we started. For universe consisting of only this single gamble and a maximum cap on absolute leverage $|l_t| < \frac{12}{11} \leq \eta$, contemporary decision theory would unequivocal suggest $l^*_t \to \frac{12}{11}\ \forall\ t$ as the maximum to not lose all wealth from a single time step while operating with no stop-loss, see Eq. (7). However, previously we have seen that this is not advised as the optimal leverage for dice roll was shown to be $l^*_t \lesssim 40\%\ \forall\ t$ by the Kelly criterion in [11] and Appendix app:opt, and also validated using reinforcement learning through our experiments Section 11.4. The question is then what is the effect of the insurance policy on the time-average growth?

Recall from Section 1.2 the supposed optimal sequence of investment management decisions in which any gamble should only be undertaken if $\mathbb{E}[R] \neq 0$, that is, there is a possibility of generating wealth through either long or short positioning. Therefore, since the insurance policy has zero expectation, this should dictate that it should not be either independently undertaken or coupled with the uninsured gamble usual dice roll gamble. Next, also recall from the same discussion the concept of the volatility tax ν in Eq. (12) that formulated as the difference between the time-average growth rate and expected return $\nu = \bar{g} - \mathbb{E}[R]$ [11, 18–24]. This quantity encodes the level of non-ergodicity in the system, the larger the difference between what any random investor actually achieves and what we expect them to achieve is a direct measure of the damaging effect steep losses have on wealth.

Suppose next the maximum leverage is capped at $\eta = 1$ and so the difference between the insured (I) and uninsured (U) portfolio time-average growth rates is

$$\begin{aligned}\bar{g}^{\mathrm{I}} - \bar{g}^{\mathrm{U}} &= \Big[l^{\mathrm{I}} \cdot \mathbb{E}[R^{\mathrm{U}}] + (1 - l^{\mathrm{I}}) \cdot \mathbb{E}[R^{\mathrm{SH}}] + \nu^{\mathrm{I}}\Big] - \Big[l^{\mathrm{U}} \cdot \mathbb{E}[R^{\mathrm{U}}] + \nu^{\mathrm{U}}\Big] \\ &= (l^{\mathrm{I}} - l^{\mathrm{U}}) \cdot \mathbb{E}[R^{\mathrm{U}}] + (\nu^{\mathrm{I}} - \nu^{\mathrm{U}}) \qquad (233)\end{aligned}$$

Can a zero expectation insurance policy lead to superior performance $\bar{g}^{\mathrm{I}} > \bar{g}^{\mathrm{U}}$?

The only variables in our control here are the leverages since the volatility taxes are explicitly defined as differences. Hence, would modifying the difference $(l^{\mathrm{I}} - l^{\mathrm{U}})$ raise the median wealth of a random investor? We know from Eq. (13) that for identical gambles increasing leverage also raises the volatility tax given that losses have an asymmetrically larger effect on the geometric return than equal percentage gains. However, keep in mind these gambles are not identical, the insured portfolio has an $(1 - l^{\mathrm{I}})$ allocation to the safe have.

Contemporary decision theory would again unequivocally demand $l^{\mathrm{I}} = l^{\mathrm{U}} = 1$ given that on average the safe haven has no worth. Existing theory is also entirely formulated under the assumption that both $\mathbb{E}[R^{\mathrm{I}}] = \bar{g}^{\mathrm{I}}$ and $\mathbb{E}[R^{\mathrm{U}}] = \bar{g}^{\mathrm{U}}$ over time, which then implies $\nu^{\mathrm{I}} = \nu^{\mathrm{U}} = 0$, and so we must have the equivalence $\bar{g}^{\mathrm{I}} = \bar{g}^{\mathrm{U}}$, that is, insurance is overall a zero-sum game and does not have any impact on long-term wealth.

Before we delve into the known results discussed in [11], we first examine what the model-free reinforcement learning agent considers to optimal. The uninsured agent is the same single asset Investor A from Section 11.4 that we train again to provide further evidence of successful learning, while the insured agent is trained under the strict constraint where $l + l^{\mathrm{SH}} = 1$ where $l \geq 0$, that is, the any portion of the current of wealth not actively staked in the outcome of the next die roll is allocated to the safe haven.

The very interesting results are shown in Fig. 36. For the uninsured case, the agent once again autonomously discoverers the optimal leverage of 40% from the Kelly criterion. In stark contrast, we shockingly observe the agent managing the insured portfolio decides to operate at roughly 98% allocated to the gamble and therefore 2% is placed into the insurance safe haven. What is most interesting visually is that the median wealth of the insured portfolio is noticeably higher. Empirically, at the conclusion of training we find the medians $\bar{g}^{\mathrm{I}} = 3.57\%$, $\bar{g}^{\mathrm{U}} = 2.15\%$, $l^{\mathrm{I}} = 97.8\%$, and $l^{\mathrm{U}} = 42.9\%$. The volatility tax benefit for the insured portfolio is then $\nu^{\mathrm{I}} = \nu^{\mathrm{U}} + 0.41\%$ per time step. As an aside, the means are $\bar{g}^{\mathrm{I}} = 3.20\%$, $\bar{g}^{\mathrm{U}} = 0.94\%$, $l^{\mathrm{I}} = 94.9\%$, and $l^{\mathrm{U}} = 24.4\%$.

Furthermore, it is important to keep in mind that agent evaluation at any given time step for each of the $10,000$ evaluation episodes utilises parameters that are not necessarily identical. If we were to continue training, since the dynamics of the gamble are fixed, the agents' selection of leverages would narrow to its perceived optimal values as evidence of successful learning. Note again that TD3 injects Gaussian noise to all observed agent actions where

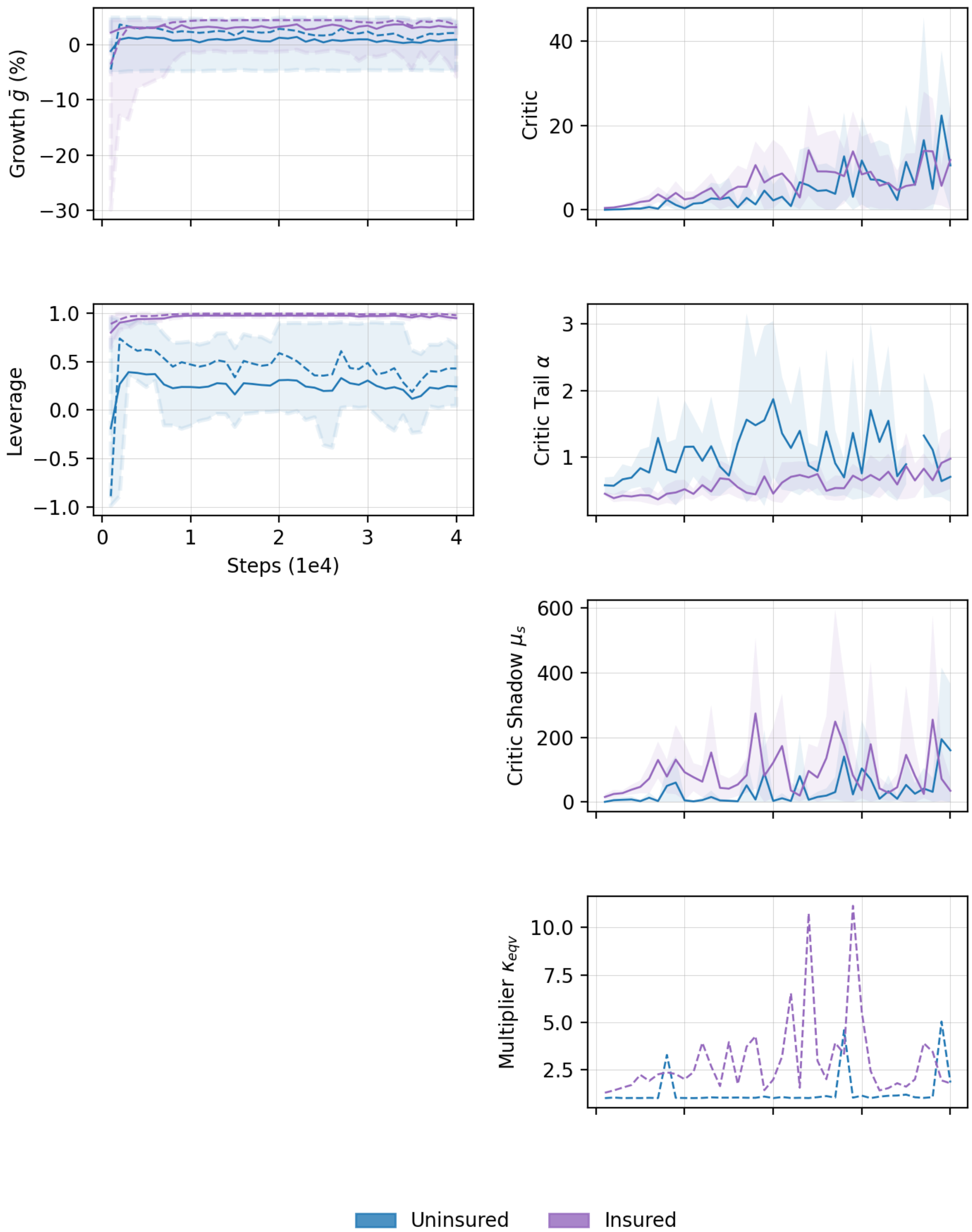

Figure 36: The dice roll environment with insurance safe haven replicating the results of [11] using TD3. The solid and dashed lines indicate means and medians respectively, the shaded regions with dashed outline represents the range between the first and third quartile, and the shaded regions without outlines indicate one MAD about the means.

$l^* = l^*_{\text{act}} + \mathcal{N}(0, 0.1)$ so that exact optimal will never be exactly reached.

Therefore, what we find is that by introducing a zero-expectation insurance policy that results in a complete loss of the allocated $(1 - l^{\text{I}})$ capital 83% of time, we are able to raise the median growth of a random investor by 1.42% per time step. This is entirely attributed to the removal of steep -50% losses. Insurance is clearly not a zero-sum game [102]. The agents learning for the insurance portfolio also has considerable volatility in growths which likely due to the agent experimenting with too large or little allocations to the safe haven which results in steep losses of capital. The key then appears to be to have an appropriate amount of insurance, too much, and the policy is no longer cost-effective given the entire allocation is lost 83% of the time, while too little, and the risk of steep losses are not successfully mitigated.

Returning back to the key point, contemporary decision theory would suggest $l^{\text{C}} = 1$ with $\bar{g}^{\text{C}} = l^{\text{C}} \cdot \mathbb{E}[R] = 3.3\%$. However, the empirical finding using median valuations from both Figure 39(b) and [11] is that $\bar{g}^{\text{C}} = -1.5\%$ which implies a complete loss of capital over time. The Kelly criterion suggests $l^{\text{K}} \sim 40\%$ and yields $\bar{g}^{\text{K}} = 2.2\%$. The use of the insurance safe haven with $l^{\text{I}} \sim 98\%$ appears to provide $\bar{g}^{\text{I}} = 3.6\%$.

To highlight the significance of this let us now compound these median values over 300 time steps with a starting value of $V_0 = 1$: $\mathbb{E}[V^{\text{C}}_{300}] = 18,713$, $V^{\text{C}}_{300} = 0.01$, $V^{\text{K}}_{300} = 591$, and $V^{\text{I}}_{300} = 36,854$. The 2% allocation to the safe haven, that on average is expected to always yield 0% return, is able to raise wealth from 0.01 to over 36, 854. In this way we can interpret insurance as behaving identically to a risk-free fixed annuity coupon. What would be the rate of return on this annuity?

To go from $0.01 \to 36,854$ in 300 steps would require a risk-free fixed return of 5.1% per time step! Through the cost-effective mitigation of steep losses, we are able to construct a insured portfolio capable of providing growth larger than the expected return of the uninsured portfolio, but in this case, does so with absolute certainty. Let that sink in, the model-free agent with zero prerequisite knowledge has found a strategy resulting in larger return per step compared to the optimal 3.3% contemporary decision theory would suggest (but is virtually never achieved by any random investor). Furthermore, this is all done with only mere 2% allocation to insurance that acts as a risk-free annuity. Therefore, by voluntarily forfeiting 2% of our capital 83% of the time, we can unlock massive returns due to the power of compound interest.

The model-free agent has thus fully autonomously self-learned a strategy that increases wealth by reducing the amount of risk it takes — efficient frontier be damned.

In the past 100 or so pages we have also repeatedly placed the Kelly criterion on a pedestal, both emphasising and proving it to exponentially superior to the contemporary prescription. However, to go from $591 \to 36,854$ in 300 steps would imply an additional risk-free return of 1.4% per time step. Therefore, even the much-loved Kelly criterion is unable to remain competitive compared to the use of an insurance safe haven. This is because insurance enables us to operate at far greater leverage reaping the benefits of higher 'expected' returns while simultaneously offsetting the steep loss with the benefits compounding over the long-run.

Despite this puzzling result, the next question is what does [11] reveal to be the optimal solution? Through simulation in an identical manner to Section 1.2 and Appendix A they find the optimal allocation to be $l^{\text{I}} = 91\%$ that simultaneously maximises both the median and 95th percentile growth rates of 2.1% and 1.0% respectively. They also find the allocation to be the singular global maximum where any deviation from 9% allocated to insurance decreases the growth rate. Furthermore, they reveal that to break-even with the Kelly criterion the cost of insurance would need to be $\mathbb{E}[R^{\text{SH}}_{t+1}] = -14\%$, and that it still remains beneficial compared to the industry standard approach for $\mathbb{E}[R^{\mathrm{SH}}_{t+1}] \geq -18\%$. This highlights how paying a premium for highly convex insurance is undeniably effective.

In interest of completeness, in Figs. 37-38 we provide the results for agent training mimicking Investors A-C from the previous section. The uninsured investors are once again the same single asset investors from Section 11.4 that are trained again. For the insured case, each investor is given two price observations, the impact of the uninsured dice roll, and the coupled insurance safe haven payoff. Using these two assets the agents learn the optimal combination of leverages for each with the restriction the dice roll limit again $\eta = 2$ and the limit on the safe haven is set to be $\eta = 1$. In this way we effectively have a three-asset portfolio: cash, dice gamble, and insurance.

For all three investors, the insured portfolio is found to be superior though with considerably greater growth rate volatility. We anticipate that if we were to train for a lengthier period of time the difference would be more visually noticeable. Investor A as expected behaves identically to the insured portfolio discussed above from [11]. Investor B again converts itself to Investor A by abandoning the use of a stop-loss. Though continues operates at twice the leverages than Investor A with the impact observable through the more volatile growth. Investor C is the most interesting, it maintains very high retention ratios and relatively huge allocations to the safe haven while at the same leverages as Investor B. Overall, it appears that the insurance safe haven is not only universally superior to the expectation maximisation approach, but also the Kelly criterion.

The reason the insurance safe haven approach in Eq. (230) is so effective is due to the 500% payout in the down state. This extremely convex payoff allows us to only allocate a very small portion of the overall portfolio to the safe haven, enabling us to benefit with as much leverage as possible from the favourable dice roll gamble. Given the size of the payoff we are will willing to tolerate $\mathbb{E}[R^{\mathrm{SH}}_{t+1}] \geq -18\%$ to still be better off than the contemporary approach, that is, not only lose the $(1 - l^{\mathrm{I}})$ allocation, but accept a reduced insurance payoff. Were it to be more expensive, it would no longer be cost-effective and acts as drain on our returns. At the same time, if the payoff was less 'explosive' it would not mitigate risk and we would require a greater allocation towards the safe haven to mitigate the same level of risk, but this would reduce the allocation to the gamble, lowering returns over time.

The applications of this counter-intuitive methodology are vast provided one is able to construct safe havens with such high levels of convexity. Its applications to finance and portfolio management are thoroughly discussed in [11] and a more concise description is presented in [18–20]. What they find is that by allocating very small 2% allocation to a highly convex insurance portfolio and the remaining 98% placed broadly into a passive market index is universally superior to all other strategies. The common 60/40 split into stocks and bonds is proven to be ineffective as bonds are a poor safe haven, likened to a real-world equivalent of the Kelly criterion. Much more interestingly they find a whole host of alpha-generating strategies, in the most favourable of situations, to also be subpar as they are not cost effective given their payoffs are not 'explosive' enough.

The utility of the modern 'hedge' fund industry in aggregate is best explained in [24] where they write "[h]edge funds would appear to have lost whatever ability they may have once had to provide risk mitigation value ... Hedge funds just don't effectively hedge anything; worse yet, perhaps they have even lost sight of that very objective in the first place. They are without a purpose." This is once again due to the fact that they either do not provide an effective hedge during crashes or because they simply do not yield high enough returns throughout normal times.

The implications to trillions of pension assets [32, 33] are then crystal clear. Given the sheer amount of capital at play, it is incredibly difficult to generate returns through strategies that would work much more effectively on smaller accounts. Thus, pension managers are forced to operate massive scales investing in only assets that are not extremely speculative, mostly consisting of large equity index components and bonds. This leaves pensions highly

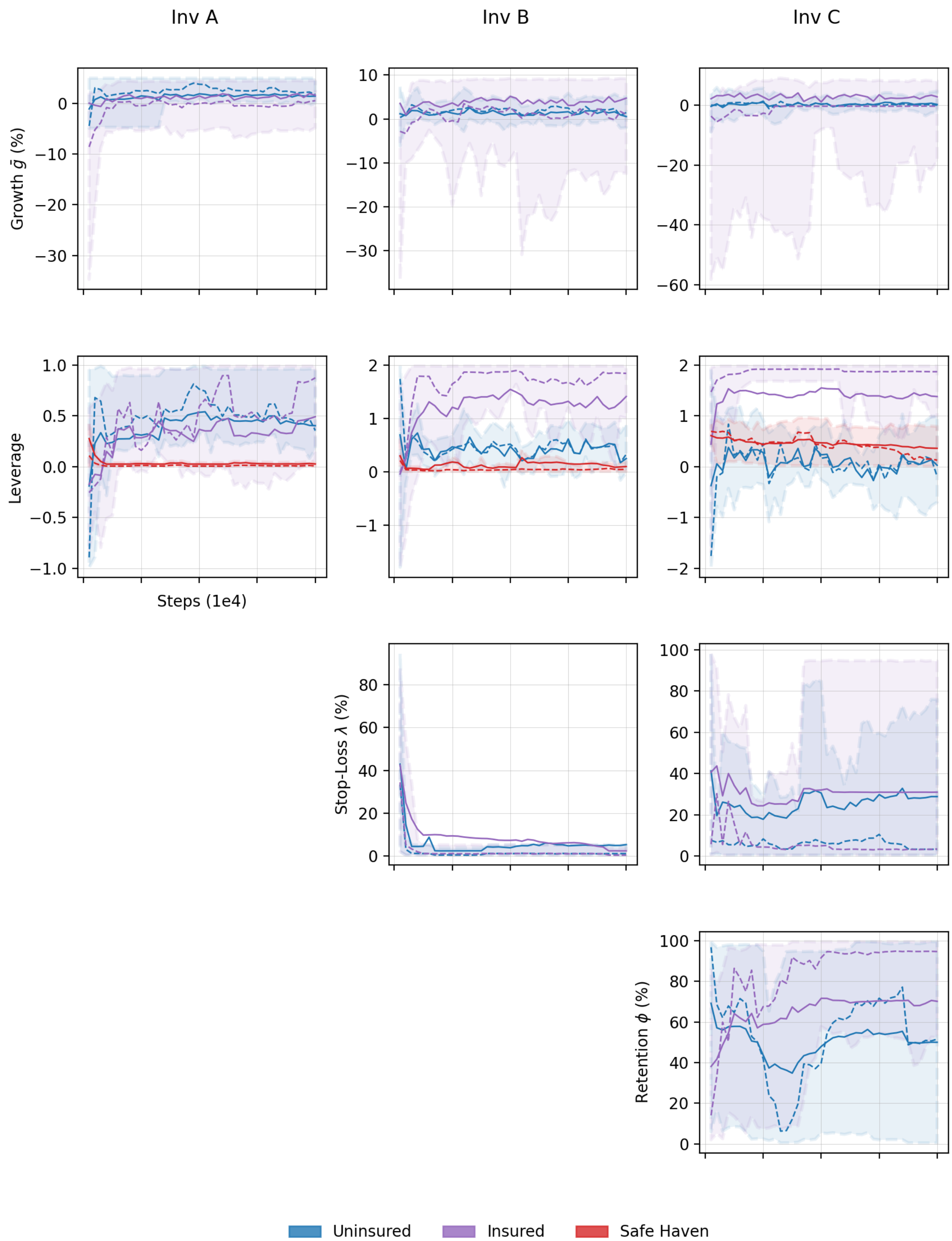

Figure 37: Summary of the dice roll with safe haven environment agent performance with respect to actions across Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions with dashed outline represents the range between the first and third quartiles.

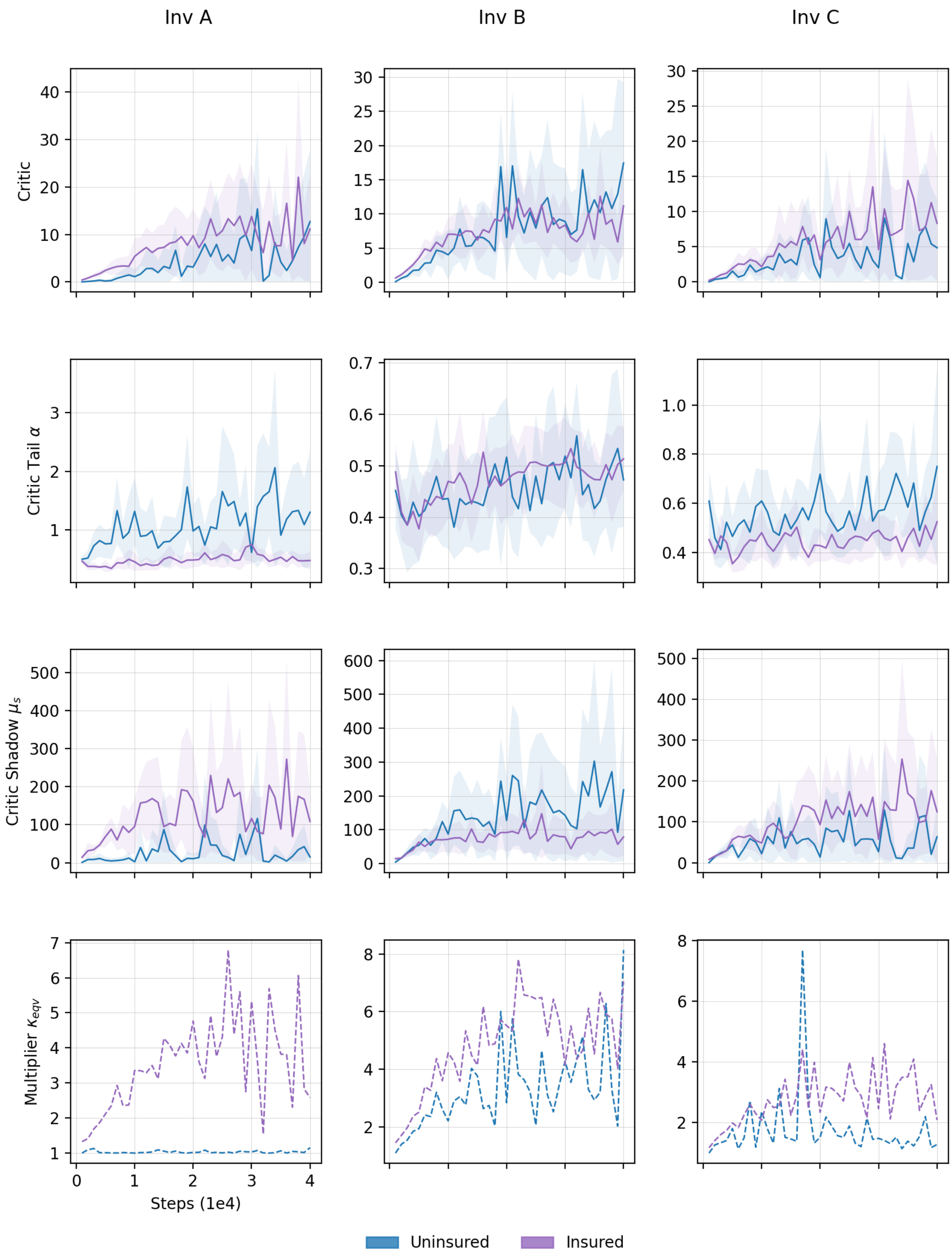

Figure 38: Summary of the dice roll environment with safe haven agent training across Investors A-C using TD3. The solid and dashed lines indicate means and medians respectively, and the shaded regions without outlines indicate one MAD about the means.

vulnerable to cyclical effects, namely periods of crisis where the market valuation of their assets plummet but their pension liabilities remain ever present. The goal is then to be able to not only preserve, but to consistently maximise wealth over time regardless of what happens in the world, which we have shown throughout this work is done by avoiding steep losses. The most cost-effective method of mitigating this risk while maintaining large directional while entirely passive positioning in the market appears to involve the use of the insurance safe haven [11, 18–24]. Never again does any faith need to be placed in the abilities of active portfolio managers to consistently beat the market in manner in which they are able to objectively prove their performance is due to skill as opposed to luck and are also able to make a conclusive case that their future performance will be just as successful. Furthermore, there would also be a significant cost-savings from not paying the associated high fees to these managers.

Therefore, when it comes to growing wealth what is common knowledge amongst all individuals is that small gains and losses are not really that consequential over the long run, what we have now learned is that even large gains are not important, the only thing that matters is avoiding steep losses. This is particularly insightful as it implies, we need not undertake extremely cumbersome research to identify and allocate scarce capital to high-yielding investments before competitors. All that is required is a 'blind' allocation to the market that self-selects the best assets coupled with comprehensive highly-convex insurance policy. We need never concern ourselves with the drama of what is going on in the the world around us, we need only strictly adhere to the optimal allocation and over time it will outperform all alternatives. This combination is incredibly robust out-of-sample in terms of being robust to any crash without prior knowledge, and requires minimal overall rebalancing. Its capability to consistently increase wealth, that is, the time-average growth rate appears to be unparalleled.

Recall again one of the primary motivations for this work was the development of fully autonomous management of strategies that maximise the time-average growth rate. Hence it is conceivable that provided a large universe of assets, the model-free reinforcement learning agent may be able to again determine the optimal combinations of positions in assets in order to maximise wealth through the construction of payoffs that are universally superior, regardless of how profoundly they contradict the mainstream.

And so we have finally come full circle. In Section 1.4, borrowing from [14], we explained how being highly risk-averse while walking in the forest alone is the correct strategy even though the probability of encountering a bear is minute. If you are unable to pool and average outcomes over the multiverse, your singular concern is your payoff function, that is, where can you conceivably end up on the payoff diagram of all possible outcomes. The probabilities of these outcomes are not relevant (provided they are non-zero and not unity), the only thing you must do is prudently take actions to mitigate the cost of the worst possible outcomes.

In the exact same manner, investment management is entirely a multiplicative process where final wealth is the only score card, and the time-average growth rate is the mechanism used to standardise the comparison between different strategies. As any investor is only concerned with the outcome of their portfolio over time, not its expected value weighted across all possible outcomes, the probability distribution of returns is again not that important! All that must be done is to create a payoff structure that is favourable no matter what happens, because for the "$N = 1$" case, it becomes increasingly impossible to escape from a series of steep losses. This is a major issue as all human investors for the time being operate with a finite, though ever-increasing, time horizon, and unlike immortal reinforcement learning agents, terminating an unsuccessful episode and starting again from scratch is not so straightforward.

14 Discussion

This work achieved novel innovations in four distinct areas: we empirically proved that smoothing large mistakes is detrimental to agent learning, revealed that mini-batch critic losses appear not to have empirical means, showed that multi-step returns are less functional and but stable in continuous action spaces, and reformulated reinforcement learning to maximise returns under the constraint of avoiding steep losses.

Our confirmation that agents need to make mistakes to learn may not be news to the world but determining that it also needs to occur in reinforcement learning needed to be explicitly confirmed. We find both reducing the impact of losses through smoothing and magnifying them enormously is unhelpful. However, the fact that learning is still possible under extreme amplification indicates that the relative ordering of critic losses within the mini-batch is more important than their magnitude. This contrasts with humans where a balance must be achieved since greatly inflating mistakes is unhelpful. Overall, using the default MSE function should be considered a reasonable starting point, but the choice of between MSE, Huber, MAE, and HSC functions become a hyperparamter to tune as there is serious potential of leaving 'free' performance on the table.

Through estimating the tail exponent of critic losses, we obtain the original finding that the mini-batch may not have a well-defined true population mean. Our findings are very general in terms of Zipf plots with estimate only likely to decrease with further refinement making our argument even stronger. Perhaps a thorough investigation using alternatives such as the method of moments or MLE may yield a different result. These more advanced methods though require very significant tuning at each learning iteration and it is not clear whether a computational efficient or standardised routine exists. Regardless, we will continue to claim that using empirical means is inappropriate as it most likely underestimates the true mean, to what degree remains an open question. Currently we are unable to directly train the agent using our heuristically obtained shadow mean until deep learning frameworks enable the backpropagation of upper incomplete gamma functions.

Extending multi-step returns to continuous action spaces for TD3 and SAC allows us to confirm that the actor is unable to learn to the same capacity as with single-step target bootstrapping. Experimental results suggest multi-step targets are less effective given global policy maximisation across the action space does not occur as it is not computationally feasible unlike the discrete case. This is issue is then further exacerbated through the geometric dampening of target Q-values. The next logical extension is to examine whether the mysterious coupling with the experience replay buffer size as with discrete action spaces still exists. Determining this requires immense computational resources but must be done as a resolution to this puzzle is necessary since it forms a massive hole in our contemporary understanding of reinforcement learning.

By reformulating reinforcement learning for both discrete and continuous action spaces to be applicable to compounding environments, we have set the stage for when the community (eventually) recognises the validity of multiplicative dynamics. By proving all standard convergence criteria for maximising compounding returns, we have extended reinforcement learning to be applicable to virtually every single real-world process, most of which are exclusively multiplicative and governed by fat tails. These developments are constructed in a manner such that they are all highly compatible with existing and so very minor modifications are needed to incorporate our ideas.

The validity of this methodology was verified through the agents' convergence to known optimal values for simple environments involving coin flips and dice rolls. Under the constraint of learning with a minimum threshold, the agent was able to navigate geometric Brownian motion successfully by avoiding steep losses. Importantly, all these results were accomplished under the MDP assumption where the agent is only provided the most recent states and no knowledge of the underlying gamble dynamics. Extending to multiple states following identical gambles was also shown to be feasible, albeit with greater learning volatility.

The dogma of risk-reward maximisation using expectation values was challenged and proven to be both catastrophically ineffective and downright dangerous strategy for any random investor. Using the insurance safe haven as an example, we demonstrated that an agent is able to autonomously learn a strategy that defies contemporary decision theory. Specifically, the inclusion of a highly convex zero-expectation security into the portfolio at the cost of reduced holdings in favourable positive-expectation securities can dramatically increase wealth, or more accurately, cost-effectively mitigates the effect of steep losses. Implications of these findings are so far reaching that the entire basis of modern financial theory, going back over half-century, including all of risk-management and valuation modelling, objectively appear to be incorrect. While the correct approach was highlighted almost three centuries ago, the more recent 21st century alternatives are yet to gain mainstream acceptance.

Our construction of multiplicative environments is also very modular such that anyone can create their own, and with minimal code reworking, gain access to utilising state-of-the-art algorithms to observe how the agent maximises the time-average growth rate for any environment. Furthermore, users will be able to investigate the effect of different critic loss functions and multi-step returns in their environments. This is extension of model-free reinforcement learning to multiplicative dynamics is crucial if we are to deliver tangible solutions to real-world problems that are being incorrectly analysed with additive dynamics, allowing us to directly bypass the universally agreed upon contemporary approach. Through our approach, we are able to circumnavigate the entire mainstream establishment to offer general empirical solutions to any multiplicative processes without needing to solve formidable stochastic partial differential equations or devise methods to simulate and analyse highly complex systems.

Furthermore, all multiplicative experiments were conducted using TD3 but are also repeatable using SAC. Note that utilising SAC is a significantly slower process given both that stochastic sampling naturally results in more diverse action selection and its relatively less algorithm execution speed, though SAC remains advantageous given the minimal hyperparamteter tuning required. TD3 on the other hand through its zero-mean noise injection is still always selecting the perceived optimal action. Regardless, it is very important to keep in mind that for all these experiments we utilised the default TD3 and SAC hyperparameters from Table 5. Therefore, training with SAC is likely to result in a negative feedback loop depending on the environment if less desired actions are repeated selected. This is extremely significant as these hyperparameters were selected to provide optimal results for additive robotic locomotive control environments, not multiplicative situations. The results we achieved should thus be considered a lower bound of what is truly possible with model-free agents maximising the time-average growth rate. The tuning of neural networks, learning rates, noise injection, discount factors, gradient iterations, and target network updating should yield faster learning to known optimal values. In particular, the amount of Gaussian noise injection into agent actions should be reduced as it is impossible to converge to known optimal values under the presence of this external volatility.

Ultimately, our findings assist in advancing the development reinforcement learning agents for all environments. Their applicability to real-world systems are countless as small improvements in leaning speed and accuracy are crucial when taking into account safety and performance of agents out in the wild. In performing novel investigations into these four areas, we extended our understanding on how to best develop meaningful artificial intelligence in the future. In Appendix D we present a brief discussion regarding the conceptual limitations of this approach.

15 Conclusion

The primary purpose of this work was to create a fully autonomous, self-learning framework for maximising the time-average growth rate. Theoretically, we have developed a general model-free return maximisation framework that is amenable to any well-defined problem provided it satisfies the conditions used in standard reinforcement learning. Experimentally, we have also validated the success of this framework for numerous realistic environments with varying level of difficulty. While on the path to this outcome, we additionally confirmed agents, like humans, must be exposed to relatively large mistakes in order to learn, and unlike humans, are able to continue learning even if these large mistakes are amplified to become astronomically larger. We revealed that empirical means are not empirical when it comes aggregating critic losses, in fact, the true mean likely does not formally exist, implying no convergence in the infinite sampling limit via the strong law of large numbers. Thus, the use of the global standard Monte Carlo approach to construct empirical losses clearly underestimates the true population mean but we are unable to train the agent with a functioning alternative. Bootstrapping of targets is found to be less effective in continuous action spaces compared to discrete case due to the infinitely larger action space leading to lack of global policy maximisation, an issue that is also exacerbated by geometric target Q-value dampening. Overall, reinforcement learning is vibrant field revolutionising the world, yet there are numerous fundamentals built on either shaky foundations, or demand more thorough understanding, before the immortal agents it generates inevitably claim the world.

References

[1] Grewal, J. S. *Revisiting Fundamentals of Model-Free Reinforcement Learning.* `https://github.com/rgrewa1/nonergodic-rl`. 2021.

[2] Sydney Informatics Hub and University of Sydney. *Artemis High Performance Computing (HPC) Cluster.* `https://www.sydney.edu.au/research/facilities/sydney-informatics-hub.html`. 2021.

[3] Bernoulli, D. "Exposition of a New Theory on the Measurement of Risk". *Econometrica* 22, 1 (Jan. 1954), p. 23.

[4] Kelly, J. L. "A new interpretation of information rate". *The Bell System Technical Journal* 35, 4 (1956), pp. 917–926.

[5] Peters, O. "Optimal leverage from non-ergodicity". *Quantitative Finance* 11, 11 (2011), pp. 1593–1602.

[6] Peters, O. "The time resolution of the St Petersburg paradox". *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 369, 1956 (Dec. 2011), pp. 4913–4931.

[7] Peters, O. *Menger 1934 revisited.* 2011. arXiv: `1110.1578 [q-fin.RM]`.

[8] Peters, O. and Gell-Mann, M. "Evaluating gambles using dynamics". *Chaos: An Interdisciplinary Journal of Nonlinear Science* 26, 2 (2016), p. 023103. ISSN: 1089-7682.

[9] Peters, O. "The ergodicity problem in economics". *Nature Physics* 15, 12 (Dec. 2019), pp. 1216–1221.

[10] Peters, O. and Adamou, A. *The time interpretation of expected utility theory.* 2021. arXiv: `1801.03680 [q-fin.EC]`.

[11] Spitznagel, M. *Safe Haven: Investing for Financial Storms.* Hoboken, New Jersey: Wiley, 2021. ISBN: 978-1119401797.

[12] Meder, D. et al. *Ergodicity-breaking reveals time optimal decision making in humans.* 2020. arXiv: `1906.04652 [econ.GN]`.

[13] Peters, O. et al. *What are we weighting for? A mechanistic model for probability weighting.* 2020. arXiv: `2005.00056 [econ.TH]`.

[14] Taleb, N. N. *Statistical Consequences of Fat Tails: Real World Preasymptotics, Epistemology, and Applications (Technical Incerto).* STEM Academic Press, 2020. ISBN: 978-1544508054.

[15] Peters, O. and Klein, W. "Ergodicity Breaking in Geometric Brownian Motion". *Phys. Rev. Lett.* 110 (10 2013), p. 100603.

[16] Peters, O. and Adamou, A. *The sum of log-normal variates in geometric Brownian motion.* 2018. arXiv: `1802.02939 [cond-mat.stat-mech]`.

[17] Landau, L. D. and Lifshitz, E. M. *Statistical Physics, Part 1: Volume 5 (Course of Theoretical Physics Series).* Third Edition. Butterworth-Heinemann, 1980. ISBN: 978-0750633727.

[18] Spitznagel, M. *Safe Haven Investing - Part One: Not All Risk Mitigation is Created Equal.* Universa Investments L.P., 2017.

[19] Spitznagel, M. *Safe Haven Investing - Part Two: Not All Risk is Created Equal*. Universa Investments L.P., 2017.

[20] Spitznagel, M. *Safe Haven Investing - Part Three: Those Wonderful Tenbaggers*. Universa Investments L.P., 2017.

[21] Spitznagel, M. *Safe Haven Investing - Part Four: The Volatility Tax*. Universa Investments L.P., 2018.

[22] Spitznagel, M. *Safe Haven Investing: Amor Fati (The Love of One's Fate)*. Universa Investments L.P., 2019.

[23] Spitznagel, M. *Safe Haven Investing: Why Do People Still Invest in Hedge Funds?* Universa Investments L.P., 2020.

[24] Spitznagel, M. *Interim Decennial Letter*. Universa Investments L.P., 2020.

[25] Goetzmann, W. et al. "Portfolio Performance Manipulation and Manipulation-proof Performance Measures". *Review of Financial Studies* 20, 5 (May 2007), pp. 1503–1546.

[26] Jensen, J. L. W. V. "Sur les fonctions convexes et les inégalités entre les valeurs moyennes". *Acta Mathematica* 30, 0 (1906), pp. 175–193.

[27] Planck Collaboration et al. "Planck 2018 results - VI. Cosmological parameters". *Astronomy & Astrophysics* 641 (Sept. 2020), A6.

[28] Dalio, R. *Principles: Life and Work*. Simon & Schuster, Sept. 2017. ISBN: 978-1501124020.

[29] Hull, J. *Options, Futures, and Other Derivatives*. Tenth Edition. Upper Saddle River, NJ: Pearson Prentice Hall, 2018. ISBN: 978-0131977051.

[30] Doctor, J. N., Wakker, P. P., and Wang, T. V. "Economists' views on the ergodicity problem". *Nature Physics* 16, 12 (Dec. 2020), pp. 1168–1168.

[31] Peters, O. "Reply to: Economists' views on the ergodicity problem". *Nature Physics* 16, 12 (Dec. 2020), pp. 1169–1169.

[32] Despalins, R., Antolin, P., and Payet, S. *Pension Markets in Focus 2020*. OECD, 2020.

[33] Despalins, R., Antolin, P., and Payet, S. *Pension Funds in Figures 2021*. OECD, 2021.

[34] Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. Second Edition. The MIT Press, 2018. ISBN: 978-0262039246.

[35] Bertsekas, D. *Dynamic Programming and Optimal Control, Vol. II: Approximate Dynamic Programming*. Fourth Edition. Belmont, Mass: Athena Scientific, 2012. ISBN: 978-1886529441.

[36] Bertsekas, D. *Dynamic Programming and Optimal Control, Vol. I*. Fourth Edition. Belmont, Mass: Athena Scientific, 2017. ISBN: 978-1886529397.

[37] Bertsekas, D. *Reinforcement Learning and Optimal Control*. Belmont, Mass: Athena Scientific, 2019. ISBN: 978-1886529434.

[38] Wiering, M. and van Otterlo, M., eds. *Reinforcement Learning*. Springer Berlin Heidelberg, 2012. ISBN: 978-3642276446.

[39] Kochenderfer, M. J. *Decision Making Under Uncertainty*. The MIT Press, 2015. ISBN: 978-0262331708.

[40] Lapan, M. *Deep Reinforcement Learning Hands-On*. Second Edition. Sebastopol, California: Packt Publishing, 2020. ISBN: 978-1838826994.

[41] Ravichandiran, S. *Deep Reinforcement Learning with Python*. Second Edition. Sebastopol, California: Packt Publishing, 2020. ISBN: 978-1839210686.

[42] Mnih, V. et al. *Playing Atari with Deep Reinforcement Learning*. 2013. arXiv: `1312.5602 [cs.LG]`.

[43] Mnih, V. et al. "Human-level control through deep reinforcement learning". *Nature* 518, 7540 (Feb. 2015), pp. 529–533.

[44] Hasselt, H. v., Guez, A., and Silver, D. "Deep Reinforcement Learning with Double Q-Learning". *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI'16. Phoenix, Arizona: AAAI Press, 2016, pp. 2094–2100.

[45] Wang, Z. et al. "Dueling Network Architectures for Deep Reinforcement Learning". *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Balcan, M. F. and Weinberger, K. Q. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 2016, pp. 1995–2003.

[46] Bellemare, M. G., Dabney, W., and Munos, R. "A Distributional Perspective on Reinforcement Learning". *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Precup, D. and Teh, Y. W. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 449–458.

[47] Fortunato, M. et al. *Noisy Networks for Exploration*. 2019. arXiv: `1706.10295 [cs.LG]`.

[48] Hessel, M. et al. *Rainbow: Combining Improvements in Deep Reinforcement Learning*. 2017. arXiv: `1710.02298 [cs.AI]`.

[49] Dabney, W. et al. *Distributional Reinforcement Learning with Quantile Regression*. 2017. arXiv: `1710.10044 [cs.AI]`.

[50] Castro, P. S. et al. *Dopamine: A Research Framework for Deep Reinforcement Learning*. 2018. arXiv: `1812.06110 [cs.LG]`.

[51] Hafner, D. et al. *Mastering Atari with Discrete World Models*. 2021. arXiv: `2010.02193 [cs.LG]`.

[52] Silver, D. et al. "Mastering the game of Go with deep neural networks and tree search". *Nature* 529, 7587 (Jan. 2016), pp. 484–489.

[53] Silver, D. et al. "Mastering the game of Go without human knowledge". *Nature* 550, 7676 (Oct. 2017), pp. 354–359.

[54] Silver, D. et al. *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*. 2017. arXiv: `1712.01815 [cs.AI]`.

[55] Silver, D. et al. "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play". *Science* 362, 6419 (Dec. 2018), pp. 1140–1144.

[56] Schrittwieser, J. et al. "Mastering Atari, Go, chess and shogi by planning with a learned model". *Nature* 588, 7839 (Dec. 2020), pp. 604–609.

[57] Tomašev, N. et al. *Assessing Game Balance with AlphaZero: Exploring Alternative Rule Sets in Chess*. 2020. arXiv: `2009.04374 [cs.AI]`.

[58] Vinyals, O. et al. “Grandmaster level in StarCraft II using multi-agent reinforcement learning”. *Nature* 575, 7782 (Oct. 2019), pp. 350–354.

[59] Senior, A. W. et al. “Improved protein structure prediction using potentials from deep learning”. *Nature* 577, 7792 (Jan. 2020), pp. 706–710.

[60] Jumper, J. et al. “Highly accurate protein structure prediction with AlphaFold”. *Nature* (July 2021).

[61] Tunyasuvunakool, K. et al. “Highly accurate protein structure prediction for the human proteome”. *Nature* (July 2021).

[62] Todorov, E., Erez, T., and Tassa, Y. “MuJoCo: A physics engine for model-based control”. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2012.

[63] Brockman, G. et al. *OpenAI Gym*. 2016. arXiv: `1606.01540 [cs.LG]`.

[64] Kenneally, G., De, A., and Koditschek, D. E. “Design Principles for a Family of Direct-Drive Legged Robots”. *IEEE Robotics and Automation Letters* 1, 2 (July 2016), pp. 900–907.

[65] Tan, J. et al. *Sim-to-Real: Learning Agile Locomotion For Quadruped Robots*. 2018. arXiv: `1804.10332 [cs.RO]`.

[66] Peng, X. B. et al. “DeepMimic”. *ACM Transactions on Graphics* 37, 4 (Aug. 2018), pp. 1–14.

[67] Hafner, R. et al. *Towards General and Autonomous Learning of Core Skills: A Case Study in Locomotion*. 2020. arXiv: `2008.12228 [cs.RO]`.

[68] Springenberg, J. T. et al. *Local Search for Policy Iteration in Continuous Control*. 2020. arXiv: `2010.05545 [cs.LG]`.

[69] Team, O. E. L. et al. *Open-Ended Learning Leads to Generally Capable Agents*. 2021. arXiv: `2107.12808 [cs.LG]`.

[70] Coumans, E. and Bai, Y. *PyBullet, a Python module for physics simulation for games, robotics and machine learning*. 2016–2021.

[71] Sutton, R. S. “Learning to predict by the methods of temporal differences”. *Machine Learning* 3, 1 (Aug. 1988), pp. 9–44.

[72] Watkins, C. J. C. H. and Dayan, P. “Q-learning”. *Machine Learning* 8, 3-4 (May 1992), pp. 279–292.

[73] Hasselt, H. van. “Double Q-learning”. *Advances in Neural Information Processing Systems*. Ed. by Lafferty, J. et al. Vol. 23. Curran Associates, Inc., 2010.

[74] Sutton, R. S. et al. “Policy Gradient Methods for Reinforcement Learning with Function Approximation”. *Proceedings of the 12th International Conference on Neural Information Processing Systems*. NIPS’99. Denver, CO: MIT Press, 1999, pp. 1057–1063.

[75] Silver, D. et al. “Deterministic Policy Gradient Algorithms”. *Proceedings of the 31st International Conference on Machine Learning*. Ed. by Xing, E. P. and Jebara, T. Vol. 32. Proceedings of Machine Learning Research 1. Bejing, China: PMLR, 2014, pp. 387–395.

[76] Lillicrap, T. P. et al. *Continuous control with deep reinforcement learning*. 2019. arXiv: `1509.02971 [cs.LG]`.

[77] Popov, I. et al. *Data-efficient Deep Reinforcement Learning for Dexterous Manipulation*. 2017. arXiv: `1704.03073 [cs.LG]`.

[78] Lowe, R. et al. *Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments*. 2020. arXiv: `1706.02275 [cs.LG]`.

[79] Barth-Maron, G. et al. *Distributed Distributional Deterministic Policy Gradients*. 2018. arXiv: `1804.08617 [cs.LG]`.

[80] Fujimoto, S., Hoof, H. van, and Meger, D. *Addressing Function Approximation Error in Actor-Critic Methods*. 2018. arXiv: `1802.09477 [cs.AI]`.

[81] Ziebart, B. D. "Modeling purposeful adaptive behavior with the principle of maximum causal entropy". PhD thesis. Carnegie Mellon University, 2010.

[82] Fox, R., Pakman, A., and Tishby, N. "Taming the Noise in Reinforcement Learning via Soft Updates". *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*. UAI'16. Jersey City, New Jersey, USA: AUAI Press, 2016, pp. 202–211. ISBN: 978-0996643115.

[83] Haarnoja, T. et al. "Reinforcement Learning with Deep Energy-Based Policies". *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML'17. Sydney, NSW, Australia: JMLR.org, 2017, pp. 1352–1361.

[84] Haarnoja, T. et al. *Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor*. 2018. arXiv: `1801.01290 [cs.LG]`.

[85] Haarnoja, T. et al. *Composable Deep Reinforcement Learning for Robotic Manipulation*. 2018. arXiv: `1803.06773 [cs.LG]`.

[86] Haarnoja, T. et al. *Soft Actor-Critic Algorithms and Applications*. 2019. arXiv: `1812.05905 [cs.LG]`.

[87] Haarnoja, T. et al. *Learning to Walk via Deep Reinforcement Learning*. 2019. arXiv: `1812.11103 [cs.LG]`.

[88] Christodoulou, P. *Soft Actor-Critic for Discrete Action Settings*. 2019. arXiv: `1910.07207 [cs.LG]`.

[89] Schulman, J. et al. *Trust Region Policy Optimization*. 2017. arXiv: `1502.05477 [cs.LG]`.

[90] Schulman, J. et al. *Proximal Policy Optimization Algorithms*. 2017. arXiv: `1707.06347 [cs.LG]`.

[91] Li, Z. et al. *Reinforcement Learning for Robust Parameterized Locomotion Control of Bipedal Robots*. 2021. arXiv: `2103.14295 [cs.RO]`.

[92] Mania, H., Guy, A., and Recht, B. *Simple random search provides a competitive approach to reinforcement learning*. 2018. arXiv: `1803.07055 [cs.LG]`.

[93] Szepesvári, C. *Algorithms for Reinforcement Learning*. 2009. ISBN: 978-1608454921.

[94] Guan, N. et al. "Truncated Cauchy Non-Negative Matrix Factorization". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 1 (Jan. 2019), pp. 246–259.

[95] Cirillo, P. and Taleb, N. N. "Expected shortfall estimation for apparently infinite-mean models of operational risk". *Quantitative Finance* 16, 10 (Apr. 2016), pp. 1485–1494.

[96] Cirillo, P. and Taleb, N. N. "Tail risk of contagious diseases". *Nature Physics* 16, 6 (May 2020), pp. 606–613.

[97] Taleb, N. N., Bar-Yam, Y., and Cirillo, P. "On single point forecasts for fat-tailed variables". *International Journal of Forecasting* (Oct. 2020).

[98] Lin, L.-J. “Self-improving reactive agents based on reinforcement learning, planning and teaching”. *Machine Learning* 8, 3-4 (May 1992), pp. 293–321.

[99] Schaul, T. et al. *Prioritized Experience Replay.* 2016. arXiv: `1511.05952 [cs.LG]`.

[100] Zhang, S. and Sutton, R. S. *A Deeper Look at Experience Replay.* 2018. arXiv: `1712.01275 [cs.LG]`.

[101] Fedus, W. et al. “Revisiting Fundamentals of Experience Replay”. *Proceedings of the 37th International Conference on Machine Learning.* Ed. by III, H. D. and Singh, A. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 3061–3071.

[102] Peters, O. and Adamou, A. *Insurance makes wealth grow faster.* 2017. arXiv: `1507.04655 [q-fin.RM]`.

[103] Peters, O. and Adamou, A. *An evolutionary advantage of cooperation.* 2018. arXiv: `1506.03414 [nlin.AO]`.

[104] Peters, O. and Adamou, A. *Leverage efficiency.* 2020. arXiv: `1101.4548 [q-fin.GN]`.

[105] Berman, Y., Peters, O., and Adamou, A. “Wealth Inequality and the Ergodic Hypothesis: Evidence from the United States”. *SSRN Electronic Journal* (2020).

[106] Adamou, A., Berman, Y., and Peters, O. *The Two Growth Rates of the Economy.* 2020. arXiv: `2009.10451 [econ.GN]`.

[107] Adamou, A. et al. “Microfoundations of Discounting”. *Decision Analysis* (Aug. 2021).

[108] Cover, T. M. and Thomas, J. A. *Elements of information theory.* Hoboken, N.J: Wiley-Interscience, 2006. ISBN: 978-0471241959.

[109] Markowitz, H. M. “Portfolio selection”. *The Journal of Finance* 7, 1 (Mar. 1952), pp. 77–91.

[110] Markowitz, H. M. “Investment for the long run: New evidence for an old rule”. *The Journal of Finance* 31, 5 (Dec. 1976), pp. 1273–1286.

[111] Markowitz, H. M. *Portfolio selection : efficient diversification of investments.* Cambridge, Mass: B. Blackwell, 1991. ISBN: 978-1557861085.

[112] L’Her, J.-F., Masmoudi, T., and Krishnamoorthy, R. K. “Net Buybacks and the Seven Dwarfs”. *Financial Analysts Journal* 74, 4 (Sept. 2018), pp. 57–85.

[113] Gell-Mann, M. and Hartle, J. B. “Decoherent histories quantum mechanics with one real fine-grained history”. *Physical Review A* 85, 6 (2012). ISSN: 1094-1622.

[114] Gell-Mann, M. and Hartle, J. B. “Adaptive coarse graining, environment, strong decoherence, and quasiclassical realms”. *Physical Review A* 89, 5 (2014).

[115] Hutter, M. “Feature Reinforcement Learning: Part I. Unstructured MDPs”. *Journal of Artificial General Intelligence* 1, 1 (Jan. 2009).

[116] Nguyen, P., Sunehag, P., and Hutter, M. “Feature Reinforcement Learning in Practice”. *Recent Advances in Reinforcement Learning.* Ed. by Sanner, S. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 66–77. ISBN: 978-3642299469.

[117] Hutter, M. “Extreme state aggregation beyond Markov decision processes”. *Theoretical Computer Science* 650 (Oct. 2016), pp. 73–91.

[118] Daswani, M., Sunehag, P., and Hutter, M. “Q-learning for history-based reinforcement learning”. *Proceedings of the 5th Asian Conference on Machine Learning*. Ed. by Ong, C. S. and Ho, T. B. Vol. 29. Proceedings of Machine Learning Research. Australian National University, Canberra, Australia: PMLR, 2013, pp. 213–228.

[119] Majeed, S. J. and Hutter, M. “On Q-learning Convergence for Non-Markov Decision Processes”. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, July 2018, pp. 2546–2552.

[120] Pendrith, M. D. and McGarity, M. “An Analysis of non-Markov Automata Games: Implications for Reinforcement Learning”. 1997.

[121] Pendrith, M. D. and McGarity, M. “An Analysis of Direct Reinforcement Learning in Non-Markovian Domains”. *Proceedings of the Fifteenth International Conference on Machine Learning*. ICML ’98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 421–429. ISBN: 1558605568.

[122] Perkins, T. J. and Pendrith, M. D. “On the Existence of Fixed Points for Q-Learning and Sarsa in Partially Observable Domains”. *Proceedings of the Nineteenth International Conference on Machine Learning*. ICML ’02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 490–497. ISBN: 1558608737.

[123] Li, L., Walsh, T., and Littman, M. “Towards a Unified Theory of State Abstraction for MDPs”. *ISAIM*. 2006.

[124] Leike, J. “Nonparametric General Reinforcement Learning”. PhD thesis. Australian National University, 2016.

[125] Azizzadenesheli, K. “Reinforcement Learning in Structured and Partially Observable Environments”. PhD thesis. University of California, Irvine, 2019.

[126] Chandak, Y. et al. “Optimizing for the Future in Non-Stationary MDPs”. *Proceedings of the 37th International Conference on Machine Learning*. Ed. by III, H. D. and Singh, A. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 1414–1425.

[127] Chandak, Y. et al. “Towards Safe Policy Improvement for Non-Stationary MDPs”. *Advances in Neural Information Processing Systems*. Ed. by Larochelle, H. et al. Vol. 33. Curran Associates, Inc., 2020, pp. 9156–9168.

[128] Thrun, S. and Schwartz, A. “Issues in Using Function Approximation for Reinforcement Learning”. *Proceedings of the 1993 Connectionist Models Summer School*. Ed. by Mozer, M. et al. Erlbaum Associates, 1993.

[129] Bellman, R. E. “On the Theory of Dynamic Programming”. *Proceedings of the National Academy of Sciences* 38, 8 (Aug. 1952), pp. 716–719.

[130] Bellman, R. E. *Dynamic Programming*. Jan. 2003. ISBN: 978-0486428093.

[131] Fazekas, I. and Klesov, O. “A General Approach to the Strong Law of Large Numbers”. *Theory of Probability & Its Applications* 45, 3 (Jan. 2001), pp. 436–449.

[132] Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2004. ISBN: 978-0521833783.

[133] Samson, C. et al. “A variational model for image classification and restoration”. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 5 (May 2000), pp. 460–472.

[134] Hamza, A. and Brady, D. “Reconstruction of reflectance spectra using robust nonnegative matrix factorization”. *IEEE Transactions on Signal Processing* 54, 9 (Sept. 2006), pp. 3637–3642.

[135] Liu, W., Pokharel, P. P., and Principe, J. C. “Correntropy: Properties and Applications in Non-Gaussian Signal Processing”. *IEEE Transactions on Signal Processing* 55, 11 (Nov. 2007), pp. 5286–5298.

[136] Nagy, F. “Parameter Estimation of the Cauchy Distribution in Information Theory Approach”. *Journal of Universal Computer Science* 12, 9 (Sept. 28, 2006), pp. 1332–1344.

[137] Pokharel, R. and Principe, J. C. “Kernel classifier with Correntropy loss”. *The 2012 International Joint Conference on Neural Networks (IJCNN)*. IEEE, June 2012.

[138] Du, L., Li, X., and Shen, Y.-D. “Robust Nonnegative Matrix Factorization via Half-Quadratic Minimization”. *2012 IEEE 12th International Conference on Data Mining*. IEEE, Dec. 2012.

[139] Huber, P. J. *Robust Statistics*. Mar. 1981. ISBN: 978-0471418054.

[140] Haan, L. de and Ferreira, A. *Extreme Value Theory: An Introduction*. Springer, 2006. ISBN: 978-0387239460.

[141] Maronna, R. A., Martin, D. R., and Yohai, V. J. *Robust Statistics*. May 2006. ISBN: 978-0470010921.

[142] Falk, M., Husler, J., and Reiss, R.-D. *Laws of Small Numbers: Extremes and Rare Events*. Third Edition. Oct. 2010. ISBN: 978-3034800082.

[143] Embrechts, P., Kluppelberg, C., and Mikosch, T. *Modelling Extremal Events*. Jan. 2013. ISBN: 978-3540609315.

[144] Abramowitz, M. and Stegun, I. A. *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*. Dover Publications, 2012.

[145] Jameson, G. J. O. “The incomplete gamma functions”. *The Mathematical Gazette* 100, 548 (June 2016), pp. 298–306.

[146] Hill, B. M. “A Simple General Approach to Inference About the Tail of a Distribution”. *The Annals of Statistics* 3, 5 (Sept. 1975).

[147] Dekkers, A. L. M., Einmahl, J. H. J., and Haan, L. D. “A Moment Estimator for the Index of an Extreme-Value Distribution”. *The Annals of Statistics* 17, 4 (Dec. 1989).

[148] Christopeit, N. “Estimating parameters of an extreme value distribution by the method of moments”. *Journal of Statistical Planning and Inference* 41, 2 (Sept. 1994), pp. 173–186.

[149] Beirlant, J., Vynckier, P., and Teugels, J. L. “Excess Functions and Estimation of the Extreme-Value Index”. *Bernoulli* 2, 4 (Dec. 1996), p. 293.

[150] Beirlant, J., Vynckier, P., and Teugels, J. L. “Tail Index Estimation, Pareto Quantile Plots, and Regression Diagnostics”. *Journal of the American Statistical Association* 91, 436 (Dec. 1996), p. 1659.

[151] Beirlant, J. et al. “Tail Index Estimation and an Exponential Regression Model”. *Extremes* 2, 2 (1999), pp. 177–200.

[152] Beirlant, J., Dierckx, G., and Guillou, A. “Estimation of the extreme-value index and generalized quantile plots”. *Bernoulli* 11, 6 (Dec. 2005).

[153] Asis, K. D. et al. “Multi-Step Reinforcement Learning: A Unifying Algorithm”. *Thirty-Second AAAI Conference on Artificial Intelligence*. AAAI, 2018.

[154] Meng, L., Gorbet, R., and Kulic, D. “The Effect of Multi-step Methods on Overestimation in Deep Reinforcement Learning”. *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, Jan. 2021.

[155] Jaakkola, T., Jordan, M. I., and Singh, S. P. “On the Convergence of Stochastic Iterative Dynamic Programming Algorithms”. *Neural Computation* 6, 6 (Nov. 1994), pp. 1185–1201.

[156] Ribeiro, C. H. C. and Szepesvári, C. “Q-learning Combined with Spreading: Convergence and Results”. *Proceedings of ISRF-IEE International Conference: Intelligent and Cognitive Systems, Neural Networks Symposium*. Tehran, Iran, 1996, pp. 32–36.

[157] Singh, S. et al. “Convergence Results for Single-Step On-Policy Reinforcement-Learning Algorithms”. *Machine Learning* 38, 3 (2000), pp. 287–308.

[158] Duan, Y. et al. “Benchmarking Deep Reinforcement Learning for Continuous Control”. *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Balcan, M. F. and Weinberger, K. Q. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 2016, pp. 1329–1338.

[159] Henderson, P. et al. *Deep Reinforcement Learning that Matters*. 2019. arXiv: `1709.06560 [cs.LG]`.

[160] Birkhoff, G. D. “Proof of the Ergodic Theorem”. *Proceedings of the National Academy of Sciences* 17, 12 (Dec. 1931), pp. 656–660.

[161] Birkhoff, G. D. “What is the Ergodic Theorem?” *The American Mathematical Monthly* 49, 4 (Apr. 1942), pp. 222–226.

[162] Robbins, H. and Monro, S. “A Stochastic Approximation Method”. *The Annals of Mathematical Statistics* 22, 3 (Sept. 1951), pp. 400–407.

[163] Blum, J. R. “Approximation Methods which Converge with Probability one”. *The Annals of Mathematical Statistics* 25, 2 (June 1954), pp. 382–386.

[164] Popoviciu, T. “Sur certaines inegalites qui caracterisent les fonctions convexes”. *Sectia I a Mat* 11 (1965), pp. 155–164.

[165] Sharma, R., Gupta, M., and Kapoor, G. “Some better bounds on the variance with applications”. *Journal of Mathematical Inequalities* 3 (2010), pp. 355–363.

[166] Bhatia, R. and Davis, C. “A Better Bound on the Variance”. *The American Mathematical Monthly* 107, 4 (Apr. 2000), pp. 353–357.

[167] Raffin, A., Kober, J., and Stulp, F. *Smooth Exploration for Robotic Reinforcement Learning*. 2021. arXiv: `2005.05719 [cs.LG]`.

[168] Bengio, Y., Courville, A., and Vincent, P. “Representation Learning: A Review and New Perspectives”. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8 (Aug. 2013), pp. 1798–1828.

[169] Schmidhuber, J. “Deep learning in neural networks: An overview”. *Neural Networks* 61 (Jan. 2015), pp. 85–117.

[170] LeCun, Y., Bengio, Y., and Hinton, G. “Deep learning”. *Nature* 521, 7553 (May 2015), pp. 436–444.

[171] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. Cambridge, Massachusetts: The MIT Press, 2016. ISBN: 978-0262035613.

[172] García-Martín, E. et al. “How to Measure Energy Consumption in Machine Learning Algorithms”. *ECML PKDD 2018 Workshops*. Springer International Publishing, 2019, pp. 243–255.

[173] García-Martín, E. et al. “Estimation of energy consumption in machine learning”. *Journal of Parallel and Distributed Computing* 134 (Dec. 2019), pp. 75–88.

[174] Han, S. et al. “Learning Both Weights and Connections for Efficient Neural Networks”. *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*. Ed. by Cortes, C. et al. Vol. 28. NIPS’15. Montreal, Canada: MIT Press, 2015, pp. 1135–1143.

[175] Rouhani, B. D., Mirhoseini, A., and Koushanfar, F. “DeLight: Adding Energy Dimension To Deep Neural Networks”. *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*. ACM, Aug. 2016.

[176] Yang, T.-J., Chen, Y.-H., and Sze, V. “Designing Energy-Efficient Convolutional Neural Networks Using Energy-Aware Pruning”. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, July 2017.

[177] Cai, E. et al. *NeuralPower: Predict and Deploy Energy-Efficient Convolutional Neural Networks*. 2017. arXiv: `1710.05420 [cs.LG]`.

[178] Rodrigues, C., Riley, G., and Luján, M. “SyNERGY: An energy measurement and prediction framework for Convolutional Neural Networks on Jetson TX1”. English. *PDPTA’18 - The 24th International Conference on Parallel and Distributed Processing Techniques and Applications*. June 2018. ISBN: 1-60132-487-1.

[179] Han, S., Mao, H., and Dally, W. J. *Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding*. 2016. arXiv: `1510.00149 [cs.CV]`.

[180] Iandola, F. N. et al. *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size*. 2016. arXiv: `1602.07360 [cs.CV]`.

[181] Paszke, A. et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. *Advances in Neural Information Processing Systems*. Ed. by Wallach, H. et al. Vol. 32. Curran Associates, Inc., 2019.

[182] Nickolls, J. et al. “Scalable Parallel Programming with CUDA”. *Queue* 6, 2 (Mar. 2008), pp. 40–53.

[183] Tomov, S. et al. “Dense Linear Algebra Solvers for Multicore with GPU Accelerators”. *Proc. of the IEEE IPDPS’10*. Atlanta, GA: IEEE Computer Society, 2010, pp. 1–8.

[184] Tomov, S., Dongarra, J., and Baboulin, M. “Towards dense linear algebra for hybrid GPU accelerated manycore systems”. *Parallel Computing* 36, 5-6 (June 2010), pp. 232–240. ISSN: 0167-8191.

[185] Dongarra, J. et al. “Accelerating Numerical Dense Linear Algebra Calculations with GPUs”. *Numerical Computations with GPUs* (2014), pp. 1–26.

[186] Smoczynski, P. and Tomkins, D. “An explicit solution to the problem of optimizing the allocations of a bettor’s wealth when wagering on horse races”. *Mathematical Scientist* 35, 1 (2010), pp. 10–17.

[187] Schrödinger, E. “An Undulatory Theory of the Mechanics of Atoms and Molecules”. *Physical Review* 28, 6 (Dec. 1926), pp. 1049–1070.

[188] Kulawik, T. *Stooq*. `https://stooq.com/`. 2021.

[189] Snijders, T. A. B. “On Cross-Validation for Predictor Evaluation in Time Series”. *Lecture Notes in Economics and Mathematical Systems*. Springer Berlin Heidelberg, 1988, pp. 56–69.

[190] Racine, J. “Consistent cross-validatory model-selection for dependent data: hv-block cross-validation”. *Journal of Econometrics* 99, 1 (Nov. 2000), pp. 39–61.

[191] Bergmeir, C. and Benítez, J. M. “On the use of cross-validation for time series predictor evaluation”. *Information Sciences* 191 (May 2012), pp. 192–213.

[192] Bergmeir, C., Costantini, M., and Benítez, J. M. “On the usefulness of cross-validation for directional forecast evaluation”. *Computational Statistics & Data Analysis* 76 (Aug. 2014), pp. 132–143.

[193] Cerqueira, V. et al. “A Comparative Study of Performance Estimation Methods for Time Series Forecasting”. *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, Oct. 2017.

[194] Mozetič, I. et al. “How to evaluate sentiment classifiers for Twitter time-ordered data?” *PLOS ONE* 13, 3 (Mar. 2018). Ed. by Emmert-Streib, F., e0194317.

[195] Bergmeir, C., Hyndman, R. J., and Koo, B. “A note on the validity of cross-validation for evaluating autoregressive time series prediction”. *Computational Statistics & Data Analysis* 120 (Apr. 2018), pp. 70–83.

[196] Greene, W. H. *Econometric Analysis*. Eighth Edition. Pearson, 2018. ISBN: 978-0134461366.

[197] de Prado, M. L. *Advances in Financial Machine Learning*. Wiley, 2018. ISBN: 978-1119482086.

[198] Cerqueira, V., Torgo, L., and Mozetič, I. “Evaluating time series forecasting models: an empirical study on performance estimation methods”. *Machine Learning* 109, 11 (Oct. 2020), pp. 1997–2028.

[199] Hyndman, R. J. and Athanasopoulos, G. *Forecasting: Principles and Practice*. Third Edition. OTexts, 2021. ISBN: 978-0987507136.

[200] Geisser, S. “The Predictive Sample Reuse Method with Applications”. *Journal of the American Statistical Association* 70, 350 (June 1975), pp. 320–328.

[201] Varma, S. and Simon, R. “Bias in error estimation when using cross-validation for model selection”. *BMC Bioinformatics* 7, 1 (Feb. 2006).

[202] Powers, D. M. W. and Atyabi, A. “The Problem of Cross-Validation: Averaging and Bias, Repetition and Significance”. *2012 Spring Congress on Engineering and Technology*. IEEE, May 2012.

[203] Vanwinckelen, G. and Blockeel, H. “On estimating model accuracy with repeated cross-validation”. Belgian-Dutch Conference on Machine Learning. 2012, pp. 39–44. ISBN: 978-9461970442.

[204] Tashman, L. J. “Out-of-sample tests of forecasting accuracy: an analysis and review”. *International Journal of Forecasting* 16, 4 (Oct. 2000), pp. 437–450.

[205] Kunsch, H. R. “The Jackknife and the Bootstrap for General Stationary Observations”. *The Annals of Statistics* 17, 3 (Sept. 1989).

[206] Politis, D. N. and Romano, J. P. “The Stationary Bootstrap”. *Journal of the American Statistical Association* 89, 428 (Dec. 1994), pp. 1303–1313.

[207] Kingma, D. P. and Ba, J. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: `1412.6980 [cs.LG]`.

[208] Loshchilov, I. and Hutter, F. *Decoupled Weight Decay Regularization*. 2019. arXiv: `1711.05101 [cs.LG]`.

[209] Fukushima, K. “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position”. *Biological Cybernetics* 36, 4 (Apr. 1980), pp. 193–202.

[210] Rumelhart, D. E., Hinton, G. E., and McClelland, J. L. “A General Framework for Parallel Distributed Processing”. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986, pp. 45–76. ISBN: 026268053X.

[211] Malik, J. and Perona, P. “Preattentive texture discrimination with early vision mechanisms”. *Journal of the Optical Society of America A* 7, 5 (May 1990), p. 923.

[212] Nair, V. and Hinton, G. E. “Rectified Linear Units Improve Restricted Boltzmann Machines”. *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. Haifa, Israel: Omnipress, 2010, pp. 807–814. ISBN: 978-1605589077.

[213] Glorot, X., Bordes, A., and Bengio, Y. “Deep Sparse Rectifier Neural Networks”. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Gordon, G., Dunson, D., and Dudík, M. Vol. 15. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, 2011, pp. 315–323.

[214] Maas, A. L., Hannun, A. Y., and Ng, A. Y. “Rectifier nonlinearities improve neural network acoustic models”. *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*. 2013.

[215] Sun, Y., Wang, X., and Tang, X. *Deeply learned face representations are sparse, selective, and robust*. 2014. arXiv: `1412.1265 [cs.CV]`.

[216] Mitchell, M. *Why AI is Harder Than We Think*. 2021. arXiv: `2104.12871 [cs.AI]`.

A Optimal Leverage for a Simple Die Roll

For the "Dice with Nietzsche's Demon ($N = 1$)" gamble in [11], the fixed payoffs for a leveraged position in it are

$$R_{t+1} = \begin{cases} +50\% \cdot l_t, & p_1 = \frac{1}{6} \\ +5\% \cdot l_t, & p_2 = \frac{2}{3} \\ -50\% \cdot l_t, & p_3 = \frac{1}{6} \end{cases} \tag{234}$$

with $\mathbb{E}[R_{t+1}] = 3.3\% \cdot l_t$, $\sigma_{t+1} = 29.0\% \cdot |l_t|$, and $\text{MAD}_{t+1} = 17.8\% \cdot |l_t|$.

An identical analysis to that conducted in Section 1.2 covering Investors 1-3 is shown in Figs. 39-41. As before, $N = 1,000,000$ random investors starting with $V_0 = \$100$ are simulated for $T = 5,000$ steps and the top 0.01% of performers at each point in time are isolated.

Overall, for Investor 1 we find the median maximising optimal leverage to be $l^* = 40\%$ with lower values ensuring even more favourable outcomes for larger majorities of the sample. This result is consistent with findings of [11]. Therefore we can conclude that following contemporary decision theory using $l^* = 100\%$ would only be beneficial to an incredible small number of outliers and catastrophically detrimental to everyone else.

As with the coin flip gamble, Investor 2 is unable to maintain theoretical maximum leverage for extended time frame. However, the top sample is again still able to achieve enormous gains due entirely to random but favourable sequences of payoffs, though only for a short duration. Depending on what the time steps signify, this would lead to the false belief that this strategy is viable through survivorship bias.

Investor 3 is again able to generate very large median gains by retaining approximately 80% of its profit at each time step with minimal stop-loss. This performance achieved through judicious risk-managing is comparable to the Kelly criterion optimal but requires significant rebalancing at each time step.

For the case of Investor 4 that theoretically derives the optimal leverage using the Kelly criterion, for any non-binary payoff the derivation is not as straightforward as in Eq. (16). Instead the techniques of [186] must be utilised, the specifics of which are outside the scope of this work.

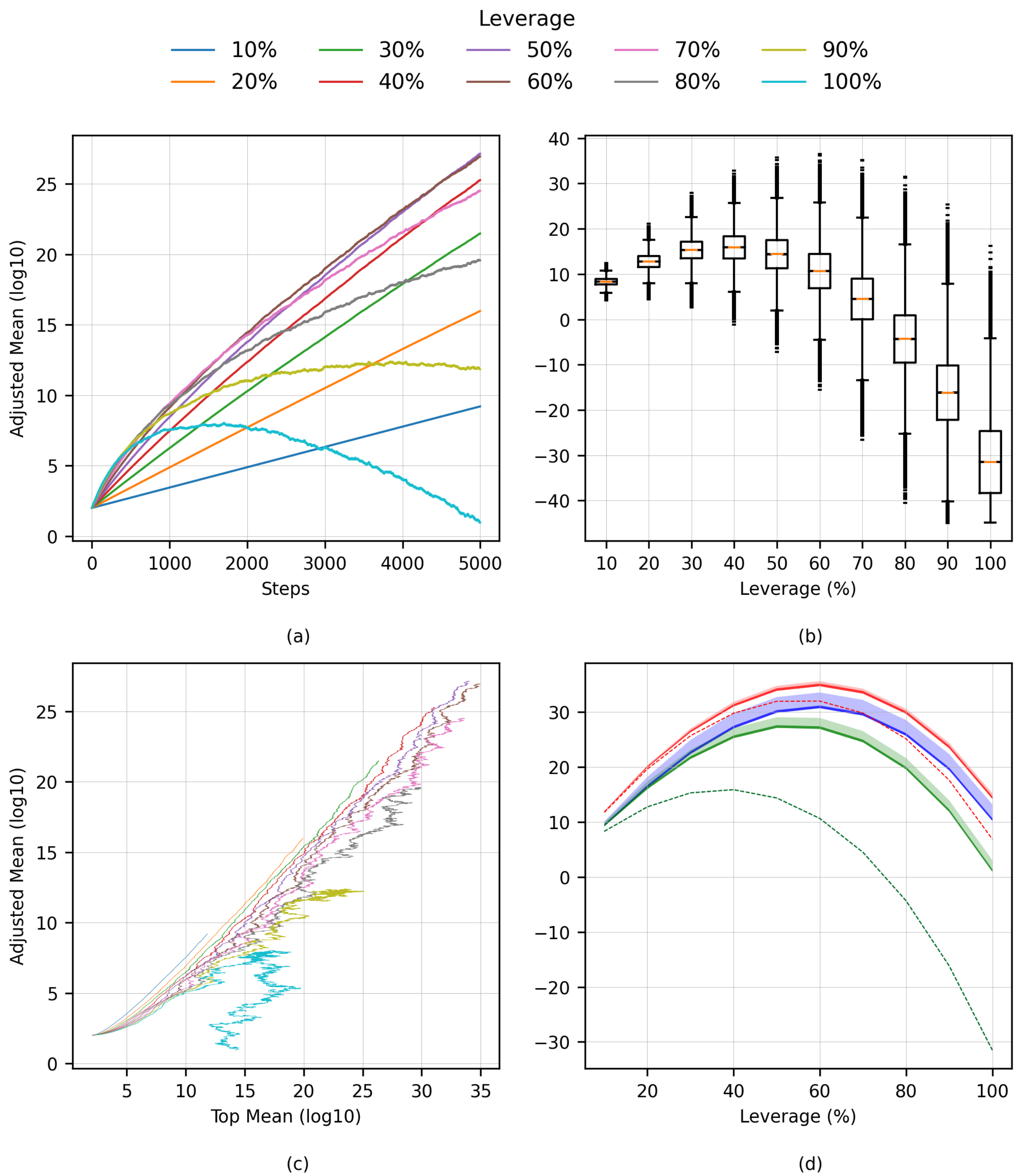

Figure 39: Summary of investor one results: (a) The trajectories of the adjusted (bottom 99.99%) log means for various leverages showing non-ergodicity at display, (b) Box plots of the distribution of adjusted values at maturity of $T = 5,000$ step for various leverages. Note $\sim 10^{-40}$ is not a lower bound, rather it is a numerical accuracy limit, (c) The trajectory of the adjusted mean along with the top (0.01%) log mean for various leverages all initiated at $(2, 2)$. Notice in all cases how astronomically larger the top values are compared to the adjusted values, and (d) Plots the medians (dotted), MAD (dark shading), and STD (light shading) added to the mean for various leverages across all three subgroup complete (blue), top (red), and adjusted (green). Note the medians of adjusted and complete are identical. Observe how STD grossly inflates by several orders of magnitude the true volatility of the gamble due to the small number of high performers in every group.

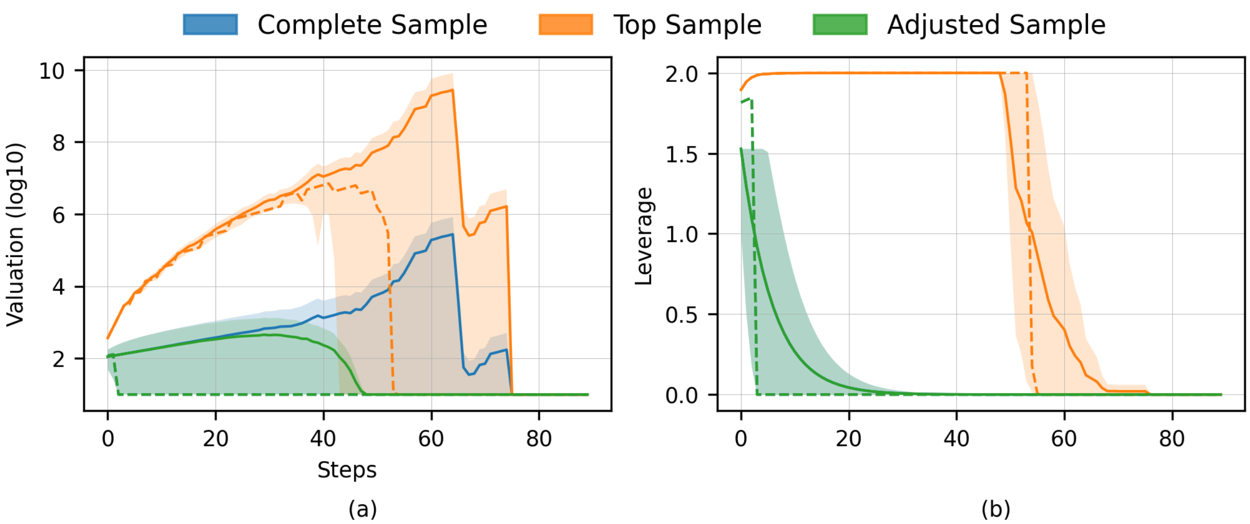

Figure 40: Summary of investor two results: (a) Reveals the mean values for each of the three subgroups highlighting how the top 0.01% are able to misrepresent the complete samples' performance by temporarily greatly skewing it upwards, and (b) The mean leverages for the same groups noting the inevitable crash to zero. The solid lines indicate the (arithmetic) means, dashed lines are the medians, and the shaded bounds represent one MAD about the mean. Note for leverage the adjusted sample is identical to the complete sample.

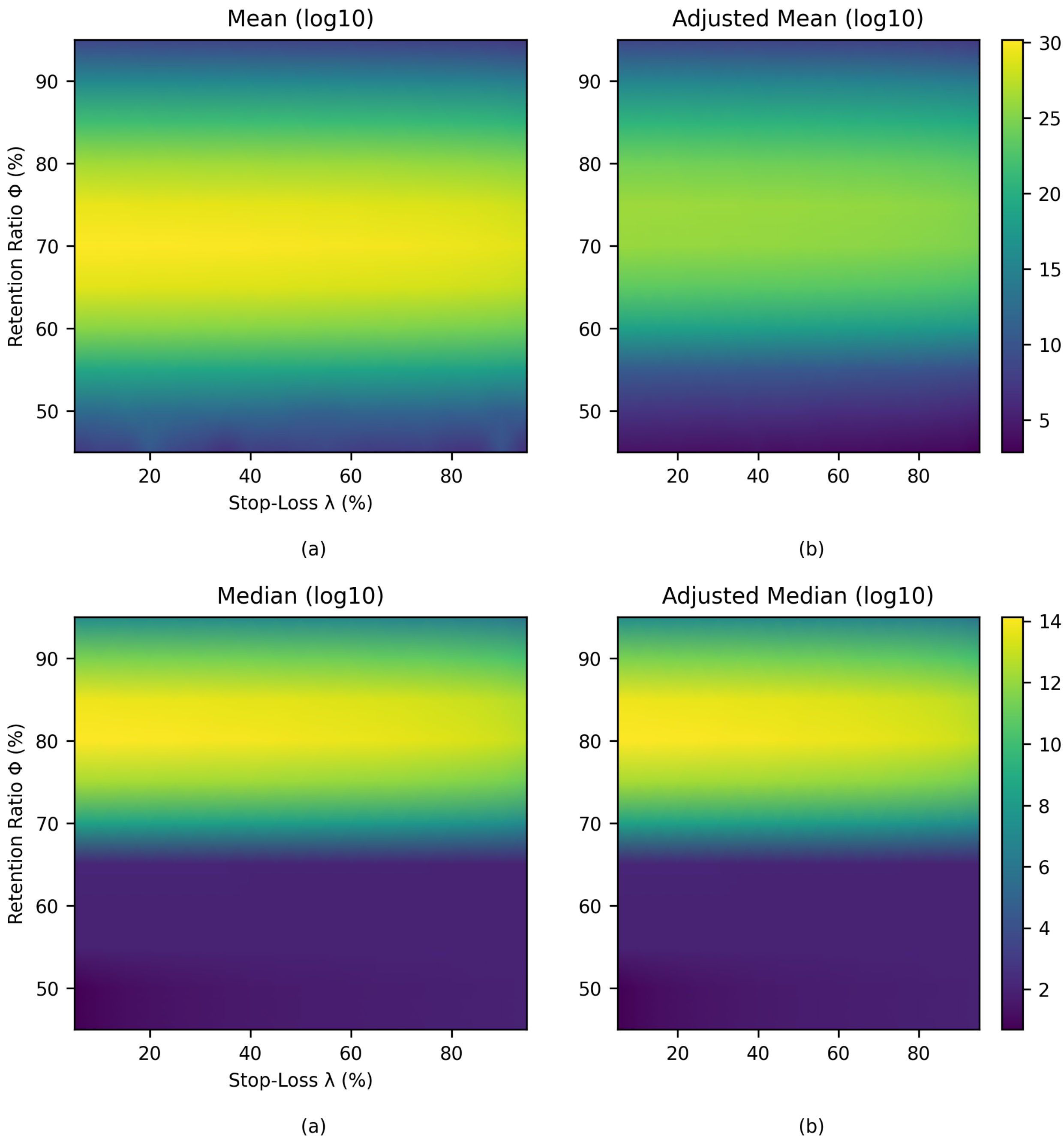

Figure 41: Investor three final valuations as a function of retention ratio and stop-loss at maturity of 5,000 steps summarized in terms of valuations. Top row: mean (a) and adjusted mean (b). Bottom row: median (c) and adjusted median (d). Observing that maintaining a fixed retention ratio of 70% of all winnings at each time step and betting a portion of the remainder appears to generate the highest mean valuations across all stop-loss levels. While the medians reveal that conservative approaches of greater than 80% retention result in a clear wealth gain. Note each row shares a common scale.

B Model-Free Off-Policy Algorithms: TD3 and SAC

Algorithm 1 Twin Delayed Deep Deterministic Policy Gradient (TD3) [80]

procedure TD3($\theta_1, \theta_2, \phi, \tau, \sigma, \bar{\sigma}, c, d, \gamma, \mathcal{S}, \mathcal{D}, m, L$)

  Initialise critic networks $Q_{\theta_1}, Q_{\theta_2}$ with random parameters θ_1, θ_2 ▷ Twin value function approximators

  Initialise actor network π_ϕ with random parameter ϕ ▷ Policy density function approximation

  Initialise target networks $\bar{\theta}_i \leftarrow \theta_i$, $\bar{\phi} \leftarrow \phi$

  Initialise empty replay buffer of fixed size $\mathcal{D} \leftarrow \{\}$

  for each environment step $t = 1$ to T **do**

    Generate exploration noise $\epsilon \sim \mathcal{S}(0, \sigma)$ ▷ Symmetric zero-mean random distribution $\mathcal{S}$

    Select action $a_t \sim \pi_\phi(s_t) + \epsilon$ ▷ Manually inject exploration noise

    Observe reward r_t, next state s_{t+1}, and done flag e_t

    Store $\mathcal{D} \leftarrow \mathcal{D} \cup \{s_t, a_t, r_t, s_{t+1}, e_t\}$ ▷ Add current transition to buffer

    for each gradient step **do**

      Uniformly sample N transitions $\{s_k, a_k, R_k^{(m)}, s_{k+m}, e_{k+m}\}$ from $\mathcal{D}$ ▷ Multi-step mini-batch

      Generate target exploration noise $\bar{\epsilon} \sim \mathcal{S}(0, \bar{\sigma})$ ▷ Target policy smoothing

      Clip the noise $\bar{\epsilon} \leftarrow \text{clip}(\bar{\epsilon}, -c, c)$

      Select next action $\bar{a}_{i+m} \sim \pi_{\bar{\phi}}(s_{k+m}) + \bar{\epsilon}$

      Obtain target Q-values $Q_{\bar{\theta_1}}(s_{k+m}, \bar{a}_{k+m}), Q_{\bar{\theta_2}}(s_{k+m}, \bar{a}_{k+m})$

      Take minimum $Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}) = \min\left(Q_{\bar{\theta_1}}(s_{k+m}, \bar{a}_{k+m}), Q_{\bar{\theta_2}}(s_{k+m}, \bar{a}_{k+m})\right)$ ▷ Double-Q clipping

      Evaluate target critic Q-value $Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}) \leftarrow R_k^{(m)} + \gamma^m Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m})$

      Evaluate twin critic Q-values $Q_{\theta_1}(s_k, a_k), Q_{\theta_2}(s_k, a_K)$

      Construct order-statistic sets ω_i for mini-batch $L(Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}), Q_{\theta_i}(s_k, a_k))$ ▷ Loss function L

      Using ω_i obtain tail exponent estimate $\hat{\alpha}_i$ ▷ Zipf plot gradient

      Construct empirical critic objectives $J(\theta_i) = \mathbb{E}_{U(\mathcal{D})}[\omega_i]$

      Construct shadow critic objectives $J_s(\theta_i) = \mu_s(L_i^*, H_i, \hat{\alpha}_i)$

      Backpropagate $\theta_{1,2} = \arg\min_{\theta_{1,2}}(J(\theta_1) + J(\theta_2))$ ▷ Update both critcs simultaneously

      if t mod d **then** ▷ Delayed actor update interval d

        Construct policy loss $J(\phi) = -\mathbb{E}_{U(\mathcal{D})}[Q_{\theta_1}(s, a)]$

        Backpropagate $\phi = \arg\min_\phi J(\phi)$

        Update $\bar{\theta}_i \leftarrow \tau\theta_i + (1 - \tau)\bar{\theta}_i$ ▷ Delayed Polyak updating of target critics

        Update $\bar{\phi} \leftarrow \tau\phi + (1 - \tau)\bar{\phi}$ ▷ Delayed Polyak update of target actor

      end if

    end for

  end for

Algorithm 2 Soft Actor-Critic (SAC) [86]

procedure SAC($\theta_1, \theta_2, \phi, \tau, d, \gamma, \mathcal{S}, \mathcal{D}, m, L$)
  Initialise soft critic networks $Q_{\theta_1}, Q_{\theta_2}$ with random parameters θ_1, θ_2 ▷ Twin value function approximators
  Initialise actor network π_ϕ with random parameter ϕ ▷ Policy density function approximation
  Initialise target networks $\bar{\theta}_i \leftarrow \theta_i$
  Initialise empty replay buffer of fixed size $\mathcal{D} \leftarrow \{\}$
  for each environment step $t = 1$ to T **do**
    Select action $a_t \sim \pi_\phi(\cdot|s_t)$ ▷ Symmetric zero-mean random policy distribution $\mathcal{S}$
    Observe reward r_t, next state s_{t+1}, and done flag e_t
    Store $\mathcal{D} \leftarrow \mathcal{D} \cup \{s_t, a_t, r_t, s_{t+1}, e_t\}$ ▷ Add current transition to buffer
    for each gradient step **do**
      Uniformly sample N transitions $\{s_k, a_k, R_k^{(m)}, s_{k+m}, e_{k+m}\}$ from $\mathcal{D}$ ▷ Multi-step mini-batch
      Select next actions $\bar{a}_{k+m} \sim \pi_\phi(\cdot|s_{k+m})$
      Evaluate target soft Q-values $Q_{\bar{\theta_1}}(s_{k+m}, \bar{a}_{k+m}), Q_{\bar{\theta_2}}(s_{k+m}, \bar{a}_{k+m})$
      Take minimum $Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}) = \min\left(Q_{\bar{\theta_1}}(s_{k+m}, \bar{a}_{k+m}), Q_{\bar{\theta_2}}(s_{k+m}, \bar{a}_{k+m})\right)$ ▷ Double-Q clipping
      Construct target soft values $V_{\bar{\theta}}(s_{k+m}) = Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}) - \alpha \log \pi_\phi(\bar{a}_{k+m}|s_{k+m})$
      Evaluate target critic Q-value $Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}) \leftarrow R^{(m)} + \gamma^m V_{\bar{\theta}}(s_{k+m})$
      Evaluate twin critic Q-values $Q_{\theta_1}(s_k, a_k), Q_{\theta_2}(s_k, a_k)$
      Take minimum $Q_\theta(s_k, a_k) = \min\left(Q_{\theta_1}(s_k, a_k), Q_{\theta_2}(s_k, a_k)\right)$ ▷ Double-Q clipping
      Construct order-statistic sets ω_i for mini-batch $L(Q_{\bar{\theta}}(s_{k+m}, \bar{a}_{k+m}), Q_{\theta_i}(s_k, a_k))$ ▷ Loss function L
      Using ω_i obtain tail exponent estimate $\hat{\alpha}_i$ ▷ Zipf plot gradient
      Construct empirical critic objectives $J(\theta_i) = \mathbb{E}_{U(\mathcal{D})}[\omega_i]$
      Construct shadow critic objectives $J_s(\theta_i) = \mu_s(L_i^*, H_i, \hat{\alpha}_i)$
      Backpropagate $\theta_{1,2} = \arg \min\limits_{\theta_{1,2}} (J(\theta_1) + J(\theta_2))$ ▷ Update both critics simultaneously
      if t mod d **then** ▷ Delayed actor update interval d
        Construct policy loss $J(\phi) = \mathbb{E}_{U(\mathcal{D})}[\alpha \log \pi_\phi(a_k|s_k) - Q_\theta(s_k, a_k)]$
        Backpropagate $\phi = \arg \min\limits_{\phi} J(\phi)$
        Construct temperature loss $J(\alpha) = \mathbb{E}_{U(\mathcal{D})}\left[-\alpha\left(\log \pi_\phi(a_k|s_k) + \overline{H}\right)\right]$
        Dual gradient descent $\alpha = \arg \min\limits_{\alpha} J(\alpha)$
        Update $\bar{\theta}_i \leftarrow \tau\theta_i + (1-\tau)\bar{\theta}_i$ ▷ Delayed Polyak updating of target critics
      end if
    end for
  end for

Parameters	**TD3**	**SAC**
Optimiser	Adam[207, 208]	Adam
Learning Rates (θ, ϕ, α)	$1 \cdot 10^{-3}$	$3 \cdot 10^{-4}$
Layer 1 Hidden Nodes (θ, ϕ)	400	256
Layer 2 Hidden Nodes (θ, ϕ)	300	256
Non-linearity (θ, ϕ)	ReLU[209–215]	ReLU
Mini-Batch Size (N)	100	256
Gradient Iterations per Time Step	1	1
Policy Gaussian Noise (σ)	0.1	
Target Policy Gaussian Noise $(\bar{\sigma})$	0.2	
Target Policy Noise Clip Rate (c)	0.5	
Minimum Expected Entropy Target $(\overline{H})$		$-\lvert\mathcal{A}\rvert$
Target Update Smoothing Rate (τ)	$5 \cdot 10^{-3}$	$5 \cdot 10^{-3}$
Actor Network Update Interval (d)	2	1
Target Critic Network Update Interval	2	1
Discount Factor (γ)	0.99	0.99
Replay Buffer Size $(\mathcal{D})$	$1 \cdot 10^{6}$	$1 \cdot 10^{6}$

Table 5: Common hyperparameters for TD3 [80] and SAC [86] including deep linear neural network architectures.

C Assorted Multiplicative Applications

Following immediately on from the results of Section 11, we present a discussion of the significance of multiplicative dynamics, more applications, and how state-of-the-art reinforcement learning algorithms can be utilised to train agents to operate in these environments. The term agent can be considered synonymous with the commonly regarded notion of 'artificial intelligence', that is, and entity that takes input parameters known as states, outputs behaviour known as actions, in order to maximise a reward signal. A typical schematic of the system is shown in Fig. 42.

Two distinguishing features of our selected agent algorithms (TD3 and SAC) are that they are both model-free and off-policy. Model-free algorithms involve minimal assumptions regarding the environments and therefore are a general approach to any problem. The agent initially knows nothing about the world in which it is operating, what it can control, and how the rewards are tied its actions. Off-policy means that when the agent is performing a task, it recalls many previous attempts it performed the task and then bases its next current action by determining what the optimal past actions might have been measure in terms of maximising the reward.

C.1 An Analogy

The contents of this work cover the following areas: 1.Investigating the use of critic loss functions other than MSE, 2. Estimating tail exponents and inferring shadow means, 3. Extending multi-step returns to continuous action spaces with TD3 and SAC, and 4. Designing agents to operate in multiplicative environments.

Agent learning can then be described to be analogous to how a toddler learns to walk. We do not provide them a (complete or partial) instruction set on controlling their tiny musculoskeletal system, rather, they learn through experience consisting of a fine balance between failure and praise. When it comes to quantifying the scale of failure, a common relatively highly penalising (MSE) method is used in all existing models. Recent advancements in other fields have highlighted the effectiveness of 'softer' approaches that lessen the significance of large mistakes as discussed in Section 3.5.

Regarding shadow means, if were to evaluate toddler walking performance as the average of their past observed performance, we are very likely underestimating their true long-term capabilities which we know will be governed by (what at the time will be seen as) rare positively-skewed tail events, that is, periods of surprisingly lengthy successful walks. Therefore, perhaps it would be wiser (and fairer) to non-uniformly skew their average performance towards these scarce tail events as they are going to be much more indicative of their future.

The overwhelming majority of existing models would also record the toddlers walking success in terms of every single step they attempt. These logs would then be used to provide updated feedback. Instead, aggregating multiple steps before their evaluating performance might lead to more stable learning as mentioned in Section 3.7.

The multiplicative aspect can be described as ensuring that when the toddler falls, the magnitude of the fall is not large enough to make recovery questionable. Existing reinforcement learning models do not internally provide any such safety feature, see Sections 1, 5-7. Including this feature requires considerable additional resources for constant supervision, but since toddlers are not fungible, it is the only valid approach. Finally, unlike toddlers that operate on a fixed universal time scale, agent training can be simulated and accelerated using supercomputers.

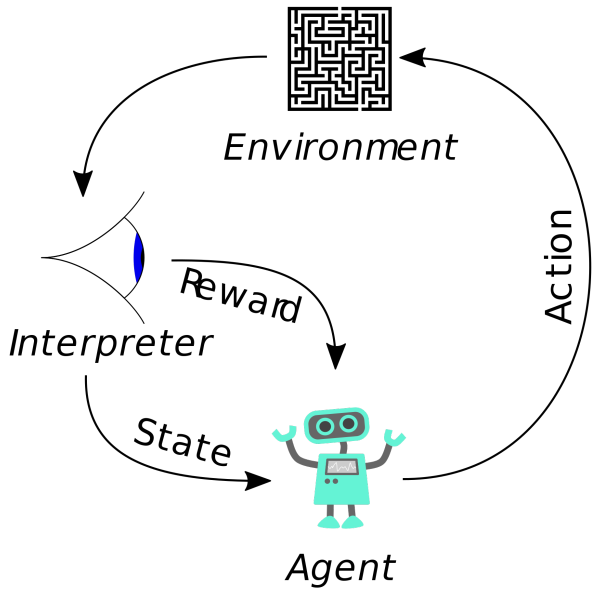

Figure 42: Simplified depiction of the system an agent in reinforcement learning operates within where the interpreter is the mechanism for translating the environments response to a given set of actions.

C.2 Overview

If we were to summaries reinforcement learning with multiplicative dynamics with only one quantity it would be: time. The purpose is to acknowledge mortality in terms of accepting that you only have a single future and then to make optimal decisions minimising the likelihood of ruin over time as recovering from steep losses becomes increasingly difficult. In these cases, the total performance is measured in terms of the time-average growth rate $g = \mathbb{E}[r] + \nu$ in Eq. (12) which is composed of the 'expected' return and the volatility tax. The tax can essentially be thought of as the penalty applied to our expectations by the effect of steep losses.

The value of our approach is largely based around tasks where the amount of risk-taking (leverage) can either be amplified or reduced to magnify or shrink potential rewards. The systems are also non-ergodic essentially meaning that the volatility tax $\nu \neq 0$ and so bouncing back from failure is tougher if not impossible. Domains where $\bar{g} \neq \mathbb{E}[r]$ are generally the mark of strong multiplicative dynamics. These are situations where time order matters and losses have an asymmetrically larger effect on total return as discussed in Section 1. The goal is to find the optimal balance of risk-taking using model-free algorithms that can be generally applied to any environment, so that we do not have to teach any specific concept or theory to the agent.

In contrast, the training of assembly line robots with this approach is inappropriate since they are producing identical fungible products, and hence the existing additive dynamics approach is perfectly fine. As a rule of thumb, if we want to optimise the results for a single random individual or operation, multiplicative dynamics should be used as compounding is key. If the goal is maximising the performance a collective group performing the operations, then additive dynamics is fine since the averaged result converges to the "expected" value with virtually no one in the group experiencing this precise value. The problem with the latter approach is that it has sampling bias since a small number of constituents can dominate the sample average. Therefore, we should generally be far more interested in determining what the average performance over time for an any random constituent of this sample would be as this is more informative.

Below we provide several further examples building directly on the successful results in Section **??** where multiplicative dynamics can be easily applied in real-world systems. The following assumptions are formally required:

(i) The input states form a QDP as discussed in Sections 3.8 and 5.

(ii) The output actions can be parametrised.

(iii) The state-action pairs can be represented in a format capable of being fed to deep neural networks.

(iv) The state-action pairs permit random but sensible simulation.

(v) There exists an accurate reward signal, or once can be designed.

Note these descriptions only serve to illustrate how our approach may be superior. We focus solely on optimal decision-making rather than discussing computer vision and object recognition details.

C.3 Robotic Control for Medical Surgery

Performing non-trivial surgery on someone is a highly non-ergodic process as people's composition, medical history, and needs are unique. For robots to perform this task, first it is a very difficult computer vision problem, assuming this has been solved how would the actions a robot takes during surgery be evaluated? Multiplicative dynamics is necessary as this is certainly a system where mistakes are far more detrimental in relative terms compared to successes. As the expectation for the outcome of any surgery is $\mathbb{E}[r] > 0$, as otherwise it would defeat the purpose, the preference is strictly for surgeries with $\bar{g} \geq 0$ to patient health.

If one serious injury that would never have occurred by a human surgeon occurs ($\nu \ll 0$), it would likely be sufficient to getting the robot pulled from supermarket shelves, R&D costs never recouped, and huge and possibly irreversible reputational damage. An upside would be watching to see how the CEO tries to explain that their Doctor Robot was designed and trained to (incorrectly?) operate under the assumption that mistakes do not matter as long as the majority of surgeries are a success, that is, maximise $\mathbb{E}[r]$ with no consideration for penalising effect of ν. This purely expectation maximising approach is unacceptable from a sales perspective as the scale of a single mistake (tail event) might greatly exceed the benefits of numerous successful of the procedures.

In terms of reinforcement learning, consider each surgery occurring at a time step and the reward form each step being proportional to how successful the surgery was by some measure. Assuming we know all the states, existing algorithms would be trained to take the expectations approach where the total reward would be the sum of each surgery treated independently. This is the wrong way for the agent to perform surgeries. The reward should be compounding over correct procedures, and highly penalising for mistakes. This way, to work back up to the total reward prior to the mistake, the agent will have to achieve more relative success. The final agent selected to perform the surgeries should be the one that is trained to operate successfully while avoiding large mistakes rather than one that blindly maximises expected performance, that is, maximises $\bar{g}$.

Furthermore, recall the impact of a mistake depends very heavily on the time it occurred. For example, assuming the agent parameters remains unchanged over time, suppose the first surgery Doctor Robot gets critically wrong with the same mistake is #13 compared to #99,323. Under existing additive models, the agent will consider these occurrences to be of equal significance. In reality however, we know there is a world of a difference to long-term profits from sales of Doctor Robot between the two. The former is far more likely to lead to complete failure of the project as opposed to the latter.

Mathematically this can be represented as follows: consider a common initial $V_0 > 0$, known $V_{99322} > V_{12} > V_0$, and a substantial negative change $\delta V_{13} = \delta V_{99323} < 0$ so that $V_{13} = V_{12} + \delta V_{13} \to 0$. Hence we have time-averages

$$1 + \bar{g}_{99323} = \exp\left[\frac{1}{99323}\ln\left|\frac{V_{99322} + \delta V_{13}}{V_0}\right|\right] \gg \exp\left[\frac{1}{13}\ln\left|\frac{V_{12} + \delta V_{13}}{V_0}\right|\right] = 1 + \bar{g}_{13} \tag{235}$$

$$\bar{g}_{99323} \gg \bar{g}_{13} \to -100\% \tag{236}$$

due to the concavity of the logarithm where we assume $V_{99322} = V_{12}e^{r\cdot 99309} \gg V_{12}$ due to exponential compounding $r > 0$ of a seemingly successful strategy for $99,309$ intermediate time steps. Equivalently, we can write the assumption in terms of the steep loss as $V_{99322} \gg \frac{99323}{13}\ln\left|\frac{V_{12}+\delta V_{13}}{V_0}\right|$.

This is the exact issue training with multiplicative dynamics is attempting to resolve, the key goal is to encode the importance of time when it comes to optimal decision making. The best agent for this situation is one that takes a more conservative approach to patient health. As far as we are aware, there are very few (if any) attempts within the literature to design algorithms that explicitly consider this asymmetry (non-ergodicity) which is beyond crucial in everyday life.

C.4 Supply Chain Management

Management of postal delivery services very quickly becomes an unbelievable complicated problem when trying to derive optimal distribution chains, see the NP-hard Travelling Salesman Problem. Humans have not been able to solve this and have instead resorted to approximate solutions like most things. Agents using deep neural networks might be able to yield even superior results.

Consider the case of a city (or any geography) under the assumption all postal mail and packages are traceable and there is a given finite number of fungible delivery mechanisms (postmen). The system is a POMDP but we assume QDP holds and so MDP modeling is acceptable. As it stands, existing algorithms using additive dynamics would be fine. To make the situation realistic, introduce the notion of express post where there is guarantee of delivery within a known time frame and there are severe financial penalties for late submissions. The time-evolution of profits from this express service are best modelled using $\bar{g}$.

The goal is then to design a agent that effectively allocates collections of items to postmen, and then specifies each of them a delivery trajectory for each day (episode). The priority of the agent is to maximise reward under the constraint express post parcels are a priority (avoiding $\nu \ll 0$). This is a highly non-ergodic system as the long-run effects of delays can have damaging repercussions to the post offices reputation and future sales since express post is higher priced. As before, there is a huge difference if the first express parcel the agent misses the delivery deadline on is #13, compared to if it was #99, 323. See the previous application.

C.5 Portfolio Management

In Section 1.4 we highlighted how financial markets in the real world are much more complicated than the simple coin flipping gamble. Then in Section 11 we experimentally showed how reinforcement learning can be used to manage multiple simultaneous gambles in order to maximise the time-average growth rate. Here we repeat much of that formulation again with more realistic details.

Consider an arbitrary financial portfolio with initial value V_0 and a known fixed N number of tradable securities. The number of action components $|\mathcal{A}|$ would equal the number of tradeable securities. Each component has maximum and minimum values equal to the leveraged position limits in those securities. For example, continuous action space limits $[-5, 8]$ imply we can take up to a maximum leveraged 8x long and -5x short position in a product. The constraint of total funds available can also easily be satisfied if the position is thought of as proportions of funds to be allocated, that is, $\sum_{i=1}^{N} w_i \leq \eta N$ where $0 < \eta \ll \infty$ and ηN is the maximum total portfolio leverage

The input states Ξ would consist of the portfolio value, all the tradeable securities prices, and extra features. The extra features would include all other variables we consider important such as: additional security prices, index values, NLP data, memes, and economic data. This quickly leads to a POMDP, but since we assume the QDP assumption holds, we can simplify the problem to an MDP. This is assumption is made throughout finance, see for example geometric Brownian motion in derivatives and options pricing [29]. Regardless, the selection of extra features is going to be essential component as including too many superfluous states will add noise leading to slower learning, while using too few will lead to underfitting. Some form of extravagant dimension reduction will be needed to represent global markets.

The assumption fixed N number of tradable securities (agent actions) can also be relaxed. When adding new products, we can use transfer learning to import existing values for currently managed products and train the agent again which should provide a substantial speed boost. Removal of products would also work similar with exclusion of those parameters and then re-training. Large changes input states however would not be so easy and would most likely require significant re-training. On top of this, adversarial training can also be used against humans to speed up learning.

The rewards per time step would be the time-average growth from the change in portfolio value. We learn off-policy from a batch of past portfolio histories with total values greater than the threshold of say $V_{\min}$ set as our portfolio stop-loss. Policy optimisation is unchanged as its sole purpose is to maximise the discounted compounding growth rates of each sample in the batch. This minimum is a necessary feature of non-ergodicity we no longer need to allow for the unrealistic existence of 'temporarily bankrupt' intermediate states as they are no longer be possible when formulating the reward as a growth rate. This facilitates the construction of fully autonomous, self-learning, and wealth maximising algorithms that have both robust signal-to-noise detection and far more importantly, explicitly account for the asymmetric effect that large losses have on total compounding portfolio return.

The agent is self-learning as at no point did we input or make assumptions regarding the actual environment other than being a QDP. No mentions of efficient frontier, mean-variance optimisation, investor utility, correlation, security details, any type of DCF, or whether a particular tradeable security is a position in options, futures or spot. We also never even referred to standard deviation, volatility or even alpha. All the model-free agent knows is that it gets input states (prices and extras) that are all context-less streams of numbers, correspondingly, the agent has a bunch of levers it can pull with each controlling positioning in a tradeable security and then receives an easily calculable reward signal at every time step. The agent then does its best to maximise this reward over time regardless of what it all means. The deep neural networks being universal function approximators automatically construct optimal variables in a 'black-box' fashion.

The algorithm is fully autonomous to a certain extent. We assume we can accurately simulate states, select the correct extra features (no easy task), and are able to perform extensive hyperparameter tuning for the details of networks (some of this can be automated using LSTM/GRU networks). Assuming the best parameters for the agent algorithm have been found, risk-management in a live environment would work very similar to that shown throughout Section 11. Firstly, the agent would be trained to operate under the constrain of a portfolio stop-loss at $V_{\min}$ and utilise some sort of profit retention strategy. Single position limits would be set through action space limits, while sector/asset concentration limits or enforcing delta (or any other Greek) neutral portfolios cannot be done with this very simplistic approach. These shortcomings can however be addressed through the addition of more constraints forcing the agent to take actions at each step that are consistent with these limits. Theoretically, we anticipate the optimal agent should learn appropriate concentration limits independently, for example, it would construct a delta neutral portfolio if it is considered the best possible case.

Another important feature is that agent is trained under the assumption it can make decisions forever with absolutely zero human interference or overwriting. Hence, while the actions taken by the agent might seem nonsensical to humans, they may turn out to be prophetic. Ideally, under no circumstance should any individual, regardless of experience and expertise, ever be permitted to override the agents actions.

An uninteresting, but possible final outcome might simply be holding a market-capitalisation weighted market portfolio with no leverage. This is justified in prior work on optimal leverage efficiency where any deviations from unity are unwise over longer time scales [104]. Furthermore, efficient market hypothesis arguments and contemporary portfolio theory also lend weight to this outcome. A more novel outcome might be a delta neutral, extremely convex, insurance portfolio constructed using complex net long (mainly put) option payoffs relative to the underlying market portfolio. The basic idea is a portfolio that has struck the perfect balance between incinerating capital (through premium or theta decay), and long enough gamma to astronomically profit from drawdowns. The agent would then govern the rolling over of options contracts where the only downside risk would be a known theta.

Ultimately as discussed in Sections 13.1, we expect the final result will be a very unequal combination of the above two cases with a rough 95-98% will be allocated to a standard market portfolio and the remainder placed in the insurance portfolio [11, 18–24]. By focusing on payoffs and not "expectations" we will have synthetically constructed a portfolio that outperforms all similarly risky assets (on the efficient frontier), is robust to any crash without requiring prior knowledge, and requires minimal overall rebalancing. The elegance of this approach is that at no point do we have to find 'alpha', we simply use non-ergodic theory to find the portfolio that maximises compounding growth through avoidance of steep losses, and then we wait for inevitable drawdowns.

C.6 Guidance Systems

A highly topical area of discussion is self-driving cars. This is a computer vision problem to accurately map the world through sensors, and a decision problem for the next action to take. The decision making can be made using multi-task learning or reinforcement learning. In this environment agent training should be conservative and there should be no continuing from a failed state such as driving of a cliff. It does not matter how high the expectation $\mathbb{E}[r]$ for taking any action, all the matter is there does not exist any possibility of $\bar{g} \to -100\%$ from the action. The area of concern is then the performance of a particular agent in the infinite time limit, rather than the final average performance across infinite fixed time trials. Naturally this is again a multiplicative problem, generally with multi-modal solutions and hence the stochastic SAC algorithm is applicable while the deterministic TD3 may limited in its utility. There is a world of a difference in the long-term growth in profits of selling self-driving software if the first accident happens on sale #13, compared to if it was on sale #99, 323. For the latter, the seller may have already reached critical mass and be able to downplay the negativity of the crash as being a 'one-off', whereas the former will be far more difficult to justify.

We can also create state-of-the-art autonomous AI-based guidance systems for projectiles that operate at extremely fast speeds. Combined with efficient state-action segregation using SARAE^2, if sufficient computational resources are provided, the trained agents will be totally dominating in their ability to reach destinations with the act of developing effective countermeasures being an exponentially more challenging problem — catching is more difficult than throwing.

In light of this, we discuss the hypersonic delivery of multiple flower bouquets. An extremely low hanging fruit that is being actively worked on is the designing of AI-based targeting systems, which are far more simpler than self-driving cars since there are no road rules, zero passenger safety requirements, modelling cylinders is easier than cars, and there is a much easier objective to complete. While existing guidance systems are likely adequate, the incoming era of hypersonic devices that can circumnavigate the globe in a matter of minutes poses new challenges. The speeds they travel at are incomprehensible to human senses and may as well be the speed of light. Humans are therefore too slow to provide any real-time feedback, similarly giant human-written decision-making codebases would have to be bug-free and must account for all scenarios. It is therefore inevitable that to remain competitive, all control should and must be relinquished to AI systems. Letting 'black-boxes' autonomously decide on target acquisition, flight trajectory and delivery details would be optimal given the second-to-none expressiveness of deep neural networks. This would also be advantageous in terms of safety and security since deciphering how the network operates is next to impossible and so making changes to the code after completion would be highly non-trivial. While critics may argue this approach poses additional dangers, unless they are able to provide equivalent alternatives, their concerns should be dismissed.

Consider a hypersonic device carrying a dozen identical flower bouquets and we assume the binary case of either all or none of the bouquets are set free. Next, we provide the delivery addresses for each of the bouquets and leave it to agent to figure out all the logistics. This is a two-stage problem where deterministic agent polices such as TD3 may not be effective. Stochastic actor policies such as SAC should be used as this problem involves multi-modal objectives, namely optimising the successful delivery of as many (somewhat independent) bouquets as possible. This is because unlike deterministic polices, stochastic policies allow varying probability mass across the action space.

The two stages would be: 1. The hypersonic devices state-actions across time, and 2. The state-actions of each of the bouquets if launched. For the first stage, the input states would involve the world with actions governing the its flight path, and reward would be some compound measure of distance to targets parametrised such that closer deliveries are exponentially more rewarded. The second state would be the world and each of the 12 bouquets with actions would be each 12 flight paths, effectively the same action set as before, but now replicated by the dozen. The reward would be compounding and coupled amongst the 12 bouquets. The important point is that this is the epitome of non-ergodic as you might only get one chance to deliver bouquets and so the approach taken must strike a balance between maximising successful deliveries and taking a flight paths knowing it has only one chance. In other words, take actions that maximise $\bar{g} > 0$.

This is coupled problem where agent must get stage one correct to even initiate stage two. The success in stage two also depends heavily on the final state of stage one. Taking an iterative approach optimising one stage while holding the other fixed per episode might provide an avenue for good results, that is multi-stage policy control with the sequential operation of mutually-exclusive two actors.

The user must specify 12 fixed delivery addresses under the constrain that all must be simultaneously reachable. This constraint would produce 3D region that the device must reach in order for 12 linear paths to final destinations be possible. Approximate this region as a sphere (exact shape is unlikely to be needed). Simulate the environment with maximum accuracy and train the stage one actor to reach this sphere under random starting conditions. Once a high probability threshold for reaching the sphere is met, pause stage one actor training, begin stage two actor training, waiting till another appropriate threshold is met. At this point we are ready to make the stage two actor more robust. Holding the spherical region fixed, simulate different random 12 delivery points, and continue training till a new threshold is met. Now we can relax the spherical assumption as the stage one actor will have some idea of what actions are needed to maximise reward. Begin training again for the same region with randomly selected 12 points iteratively, alternating between updating the stage one and two actor weights per episode. After the agent gets competent at this, but not too expert (to prevent overfitting), introduce slightly different spherical regions for the stage one actor to reach. Then as the stage one actor becomes more robust to geography, we can start heavily varying the regions till it covers the surface of the Earth.

We can take this further by introducing a stage zero actor. As response time is crucial in the flower delivery business, it is advisable to have in standby, fleets of fully autonomous devices in flight at all times that would ideally have pre-programmed delivery addresses. The stage zero actor network then would govern this standby flight trajectory, encoding refuelling and regional details. This approach will be extremely secure as no party will be able to predict with precision the behaviour of the agents, making countermeasures very difficult.

This is a situation where a general reward signal is incredibly difficult to parameterise. If one can be designed, the potential of this approach is boundless. Examples include some combination of quantified infrastructure damage, population decrease, and reduction of opposing forces.

C.7 Minimising Quantities

Throughout this work we have repeatedly emphasised that the key point of our formulation is autonomously finding and taking optimal actions that lead to the maximisation of multiplicative quantities such wealth through the avoidance of steep losses. At the same time, it is difficult to imagine a quantity that is truly more multiplicative than destruction. One example is the population growth of an infestation that is compounding and reducing it requires measures that lead to taking actions that convert positive growth to negative growth. Another example is orchestrating structural collapse where you need only target very specific small subset of support locations for successful breakdown as numerous demolition experiments have revealed. These seemingly simple tasks have distinctly non-linear payoff functions and so cannot be accurately represented using expectation values, making the contemporary approach for modelling destruction extremely sub-optimal.

Therefore, if we are able to maximise multiplicative values, we are equally capable of minimising quantities. The objective of the model-free agent will then become to self-learn the path that leads to the steepest of losses as quickly as possible. A rapid decline is not only favourable, but greatly desired. Learning this strategy is likely a less difficult task since breaking is often considered easier than building but finding the optimal path to this outcome still remains an open question.

To perform this conversion there are two obvious methods to incorporate value reduction into our general reward in Eq. (213). The first is the use of 'absolute destruction' where the variable to be maximised is measured in terms of the cumulative level of reduction relative to a initial level of destruction so that $V_t^{(1)} \gg V_0^{(1)} \geq \gamma$. The second involves the use of a negative multiplier on overall rewards using a final level and a higher initial level where $V_0^{(2)} \gg V_t^{(2)} \geq \gamma$. These two methods can be represented with the 'anti-rewards'

$$r_t^{(1)} = \exp\left[\frac{1}{t}\ln\left|\frac{V_t^{(1)}}{V_0^{(1)}}\right|\right] \tag{237}$$

$$r_t^{(2)} = -\exp\left[\frac{1}{t}\ln\left|\frac{V_t^{(2)}}{V_0^{(2)}}\right|\right] \tag{238}$$

For the first situation no changes to our prescription is required, while the maximisation of the second type of reward will require modification to the episode termination criteria as we are incentivising asymptotic value reduction.

Overall, with our novel formulation we have laid the mathematical foundations and created publicly available tools opening the door to developing a plethora of fully autonomous, self-learning tools that are capable of causing inconceivable levels of value reduction. Note that we mean inconceivable in the literal sense as we have no prior knowledge as to what actions the agent will determine to be optimal given the use of deep neural networks. All that is required from the end-user is the creation of a well-parametrised environment, after this they are able to gain access and insight into how state-of-the-art algorithms would reduce their desired quantities. Whether the minimised quantities are devastating to society or a public good, the model-free agent cares not and will execute its user-defined objective till the end of time (or program termination) with a precision so magnificently cold that no human could ever replicate, at least not in perpetuity.

The optimal sequence of actions the agent selects may be indistinguishable from the choice of human experts, or perhaps they agent might be able to innovate. For example, given a set of tools, if the goal is destroying a vase, the agent is unlikely to develop a uniquely superior method. In contrast, given a broader set of tools, if the goal is to remove an infestation or nuclear contamination from city, it would be very interesting to gauge what actions the agent considers optimal. The key idea is well-expressed by inverting the ruleset of tower-disassembling game of Jenga. The model-free agents' goal is finding the most efficient path leading to the towers collapse, that is, finding the perfect blocks to remove for the building to topple as quickly as possible. The agents' decisions can then be compared and contrasted with the of choices taken by human experts.

D Taming Hubris

The field of artificial intelligence has forever been hyped to change the world with a summary provided in [216]. They also present four fallacies for human-level AI that we discuss in the context of our findings, namely we provide limitations of our approach. Overall, as we are not aiming to create artificial general intelligence, much of the critiques in this work are less strong.

- Fallacy 1: Narrow intelligence is on a continuum with general intelligence.

  As the goal is not to create human-like general intelligence, rather we aim to create agents with very specific goals when operating in non-ergodic situations consisting of well-defined state-action spaces with the environment accurately parametrised. Hence this fallacy is less poignant, though as the environments become more complex the agents' operations will need to not only become human, but to greatly surpass humans.

- Fallacy 2: Easy things are easy and hard things are hard.

  The non-ergodic tasks we intend for the agent to perform are directly measured against humans and we find that the agent is able to autonomously arrive at the correct answers. The environments are well-defined and consist of entirely numerical data and so agent ability to perform computations exceeds all humans. Coupling this with the expressiveness of deep neural networks, the agent is able to learn how to maximise rewards.

- Fallacy 3: The lure of wishful mnemonics.

  The strongest critique is of our use of deep neural networks wherein shortcut learning is definitely a concern. Regardless, throughout this work we have stated the agent has 'learned' various concepts. Deep learning despite its graphical similarity to the human brain, does not imply learning in the same fashion as humans learns. For example, we have to train the agent from scratch for every environment, whereas humans would be able to transfer skills across all environments. While transfer learning does exist, applying it is not a seamless experience as with humans. Furthermore, we are using benchmarking tools we currently have available, hence any results we achieve should not be assumed to be easily generalisable to reality. However, if we can accurately simulate states and train the agent for extended periods of time, ideally this potential weak point is mitigated.

- Fallacy 4: Intelligence is all in the brain.

  The intelligence encoded in deep neural network weights tend to give the impression of a functioning brain, and since the action space includes all ways an agent can interact with environment states, we can say that our model is multi-sensory when it comes to feedback and response. The major assumption of this approach that we have perfectly parametrised all states, actions, and rewards. If we have not done this, the agent will not be able to explore the entire universe of possible outcomes that may lead to sub-optimal solutions. This fallacy is far more warranted in real-world systems where the agent must function with incomplete information.