Reliable evaluation in reinforcement learning

Olivier Sigaud

Sorbonne Université
http://people.isir.upmc.fr/sigaud

Outline

- Evaluation issues in deep reinforcement learning
- Using appropriate statistical tests
- A set of better metrics to compare two algorithms
- Advices to publish research on RL algorithms
- More general advices

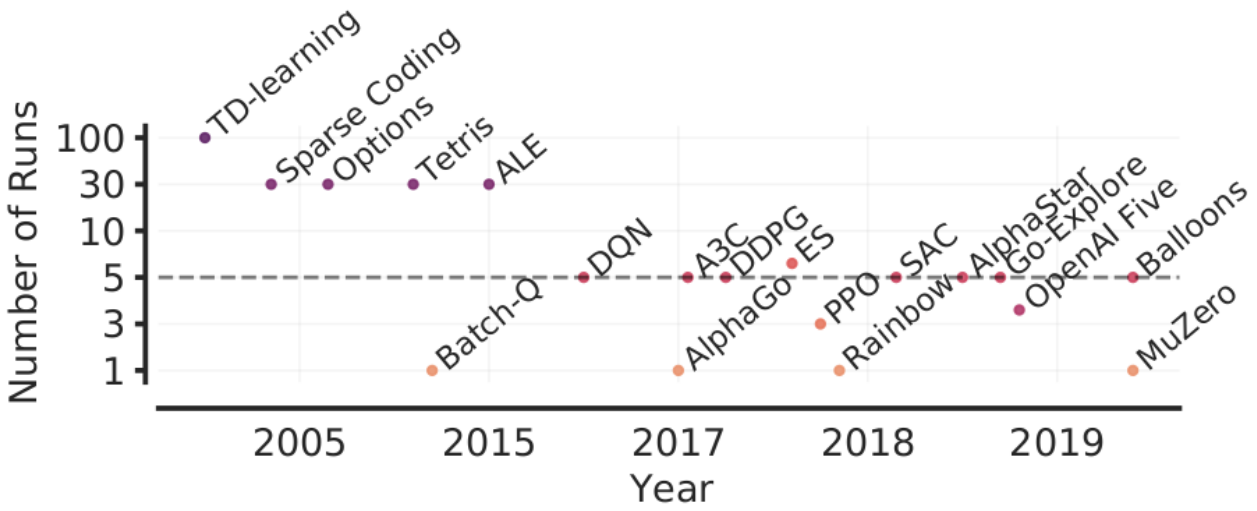
Various research goals

- Exploratory research: reach beyond frontiers, reveal new phenomena
- Theoretical research: prove some properties
- Empirical research: establish some properties from experience
- Empirical research requires a strong empirical methodology

Bouthillier, X., Laurent, C., and Vincent, P. (2019) Unreproducible research is reproducible. In *International Conference on Machine Learning*, pages 725–734. PMLR

Patterson, A., Neumann, S., White, M., and White, A. (2023) Empirical design in reinforcement learning. *arXiv preprint arXiv:2304.01315*

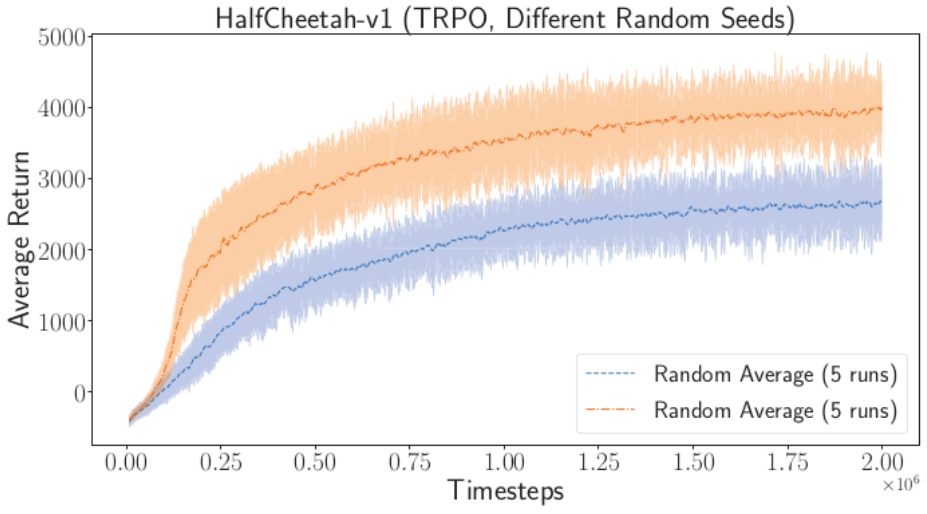
Insufficient number of seeds (common practices)



- With heavier environments, one cannot run enough seeds

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., and Bellemare, M. (2021) Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320

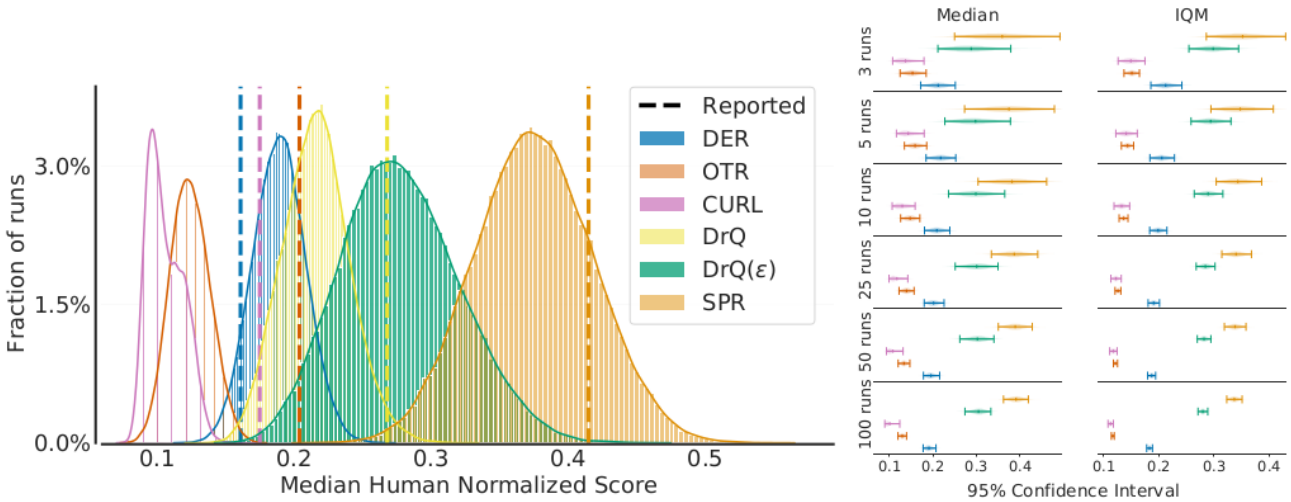
Insufficient number of seeds: the danger



- Without enough seeds, one may wrongly conclude to superiority of a method over another

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. (2018) Deep reinforcement learning that matters. In McIlraith, S. A. and Weinberger, K. Q., editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pp. 3207–3214. AAAI Press

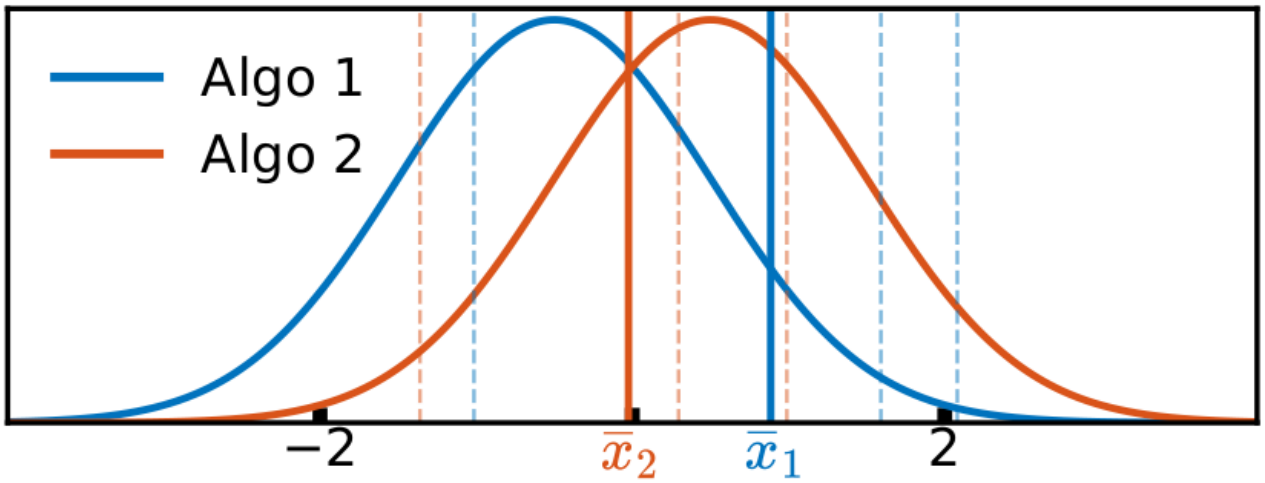
Poor reporting practices



- Authors generally overestimate their method

Statistical tests

Introduction: the problem



- Usually, RL is stochastic (in the policy and/or in the environment)
- Two episodes can give different results
- A superiority in data can be due to chance
- Need to rigorously compare two algorithms
- Statistical tests are meant to provide this rigor

Statistical tests: the framework

- One wants to compare the (true) central performances (mean or median) μ_1, μ_2 of two algorithms
- The null hypothesis $\mathcal{H}_0 : \mu_1 - \mu_2 = 0$ algorithms perform the same
- Alternative hypothesis $\mathcal{H}_a : |\mu_1 - \mu_2| > 0$ one algorithm is better
- Given a set of realizations, we observe $\bar{x}_1, \bar{x}_2$ (empirical central performances)
- With what confidence can we reject the null hypothesis?
- The confidence level cannot be 100%, would require an infinity of samples

Colas, C., Sigaud, O., and Oudeyer, P.-Y. (2019) A hitchhiker's guide to statistical comparisons of reinforcement learning algorithms. *arXiv preprint arXiv:1904.06979*

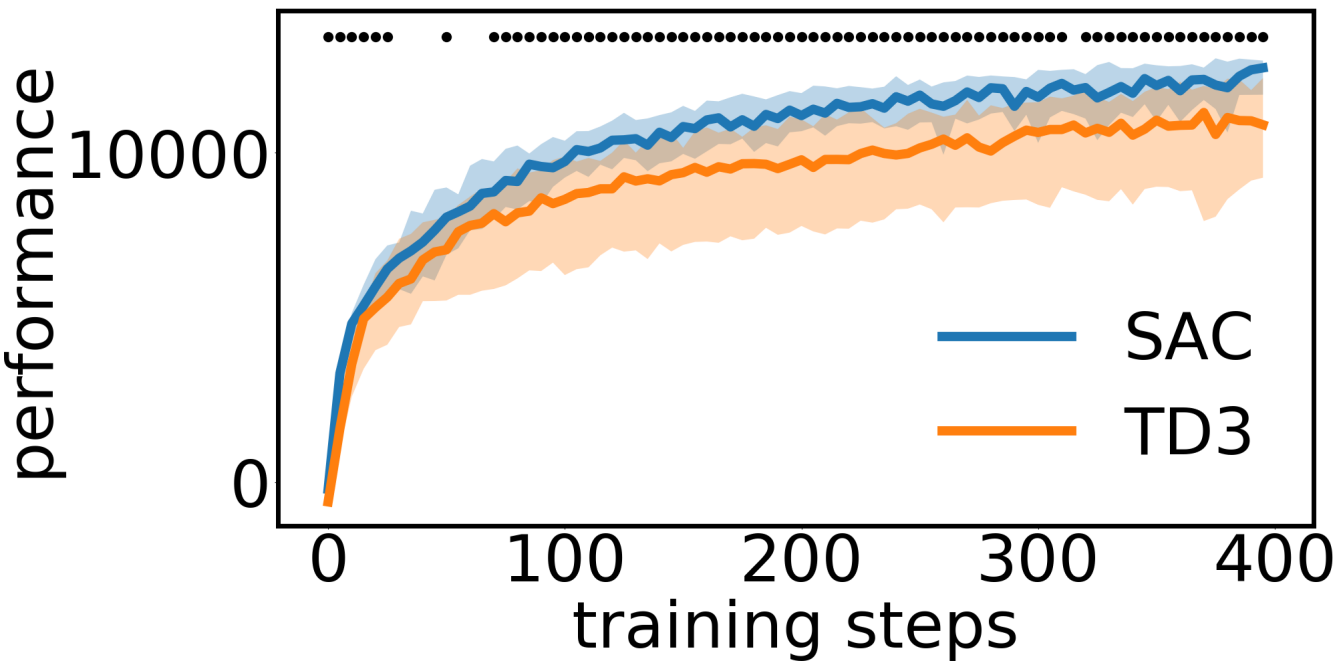
Statistical tests: definitions

- p-value: risk that the test wrongly rejects the null hypothesis
- probability of a "false positive" (difference found, but there is none)
- Usually, make sure $p - value < 0.05$
- We may claim that there is a (non-existing) difference 1 time out of 20...
- Statistical power: depends on sample size (how many data?) and effect size (how much difference?). The larger, the better

Various statistical tests and their assumptions

- (Student's) T-test: variances are equal (false when comparing two RL algorithms)
- Welch's t-test: variances are not equal (fine!)
- Wilcoxon Mann-Whitney (WMW) rank sum test: distributions are continuous, have the same shape and spread (wrong)
- Ranked t-test: close to MWM, with ranking before t-test
- Bootstrap confidence interval test: no assumptions, but requires large sample size (empirical testing)
- Permutation test: expensive
- From [Colas et al., 2019], use Welch's t-test

Tests along training



- One can test differences at each evaluation step along training
- The above two algorithms are different most of the time, but not always

Summary

- Use Welch's t-test
- Use the mean rather than the median
- Whenever possible, use at least 15 seeds
- Give p-value, check statistical power
- Select adapted plots
- When comparing more than two algorithms, add Bonferroni correction
- See `https://github.com/flowersteam/rl_stats`
- Try the notebook

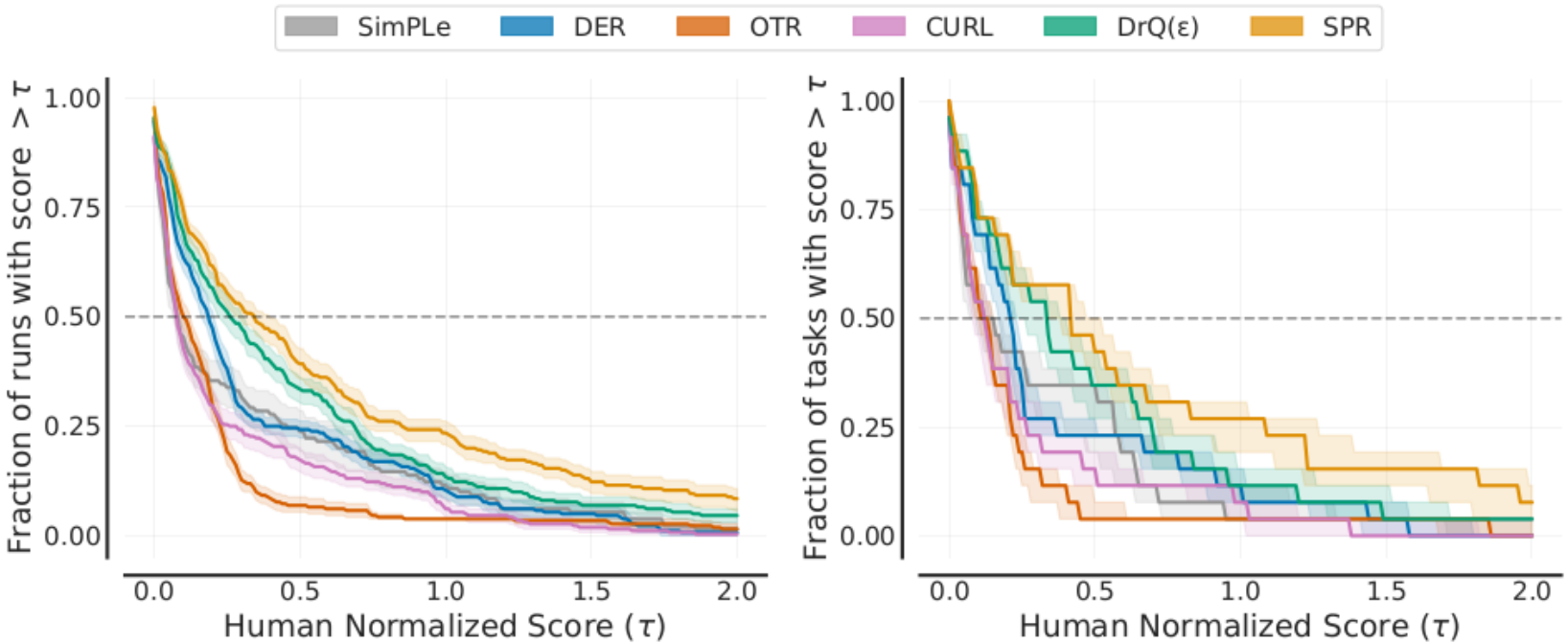
Better metrics

Better practices

Desideratum	Current Evaluation Protocol	Our Recommendation
Uncertainty in aggregate performance	**Point estimates** • Ignore statistical uncertainty • Hinder *results reproducibility*	Interval estimates via **stratified bootstrap confidence intervals**
Variability in performance across tasks and runs	**Tables with mean scores per task** • Overwhelming beyond a few tasks • Standard deviations often omitted • Incomplete picture for multimodal and heavy-tailed distributions	**Performance profiles** (*score distributions*) • Show tail distribution of scores on combined runs across tasks • Allow qualitative comparisons • Easily read any score percentile
Aggregate metrics for summarizing performance across tasks	**Mean** • Often dominated by performance on outlier tasks **Median** • Requires large number of runs to claim improvements • Poor indicator of overall performance: zero scores on nearly half the tasks do not affect it	**Interquartile Mean** (IQM) across all runs • Performance on middle 50% of combined runs • Robust to outlier scores but more statistically efficient than median To show other aspects of performance gains, report average *probability of improvement* and *optimality gap*.

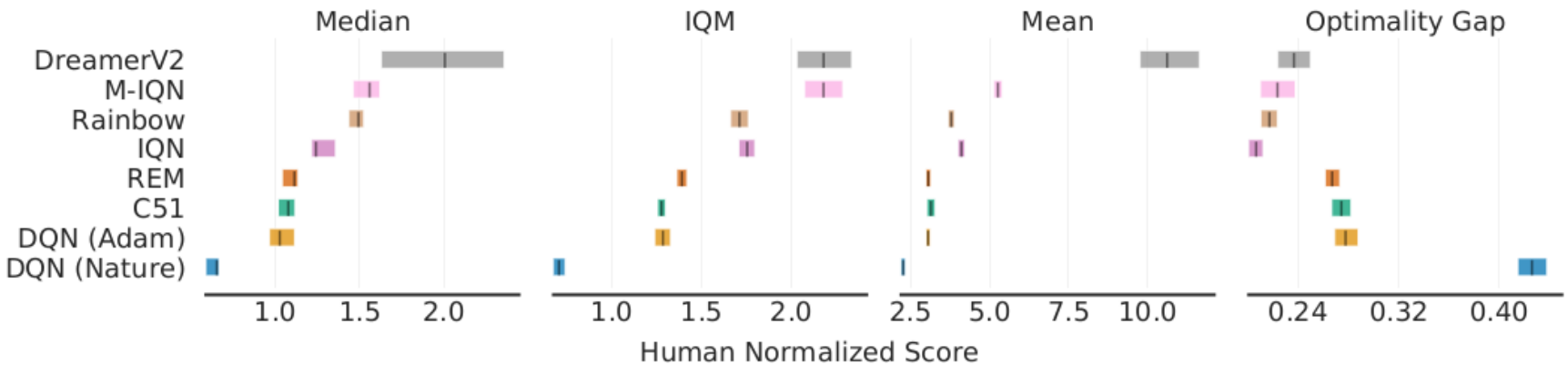
- More details on the next slides

Performance profiles



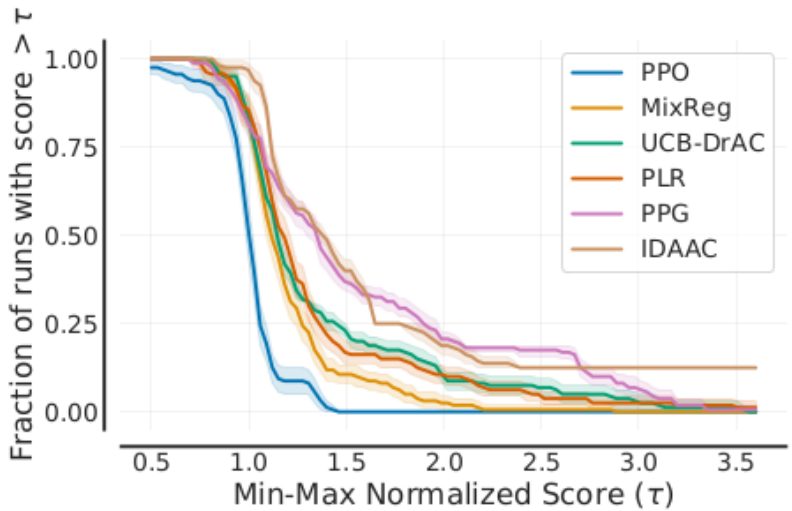
- How to normalize scores is problem dependent

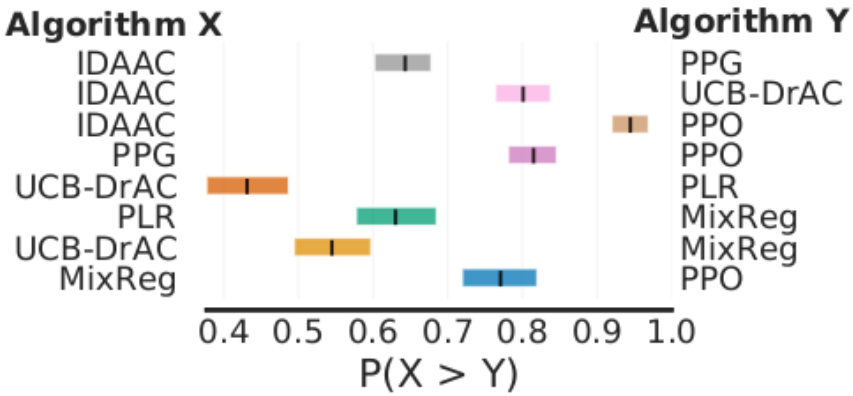
Aggregate metrics



- IQM: InterQuartile Mean
- Performance on middle 50% of combined runs

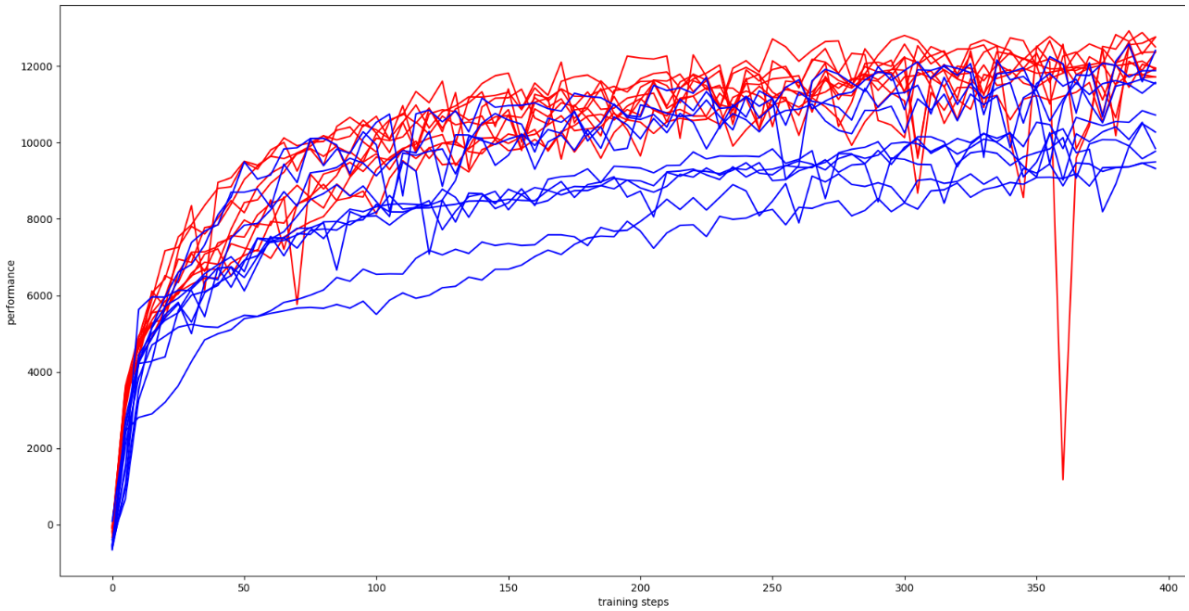
Probability of improvement





- If the whole interval is over 0.5, improvement can be claimed safely

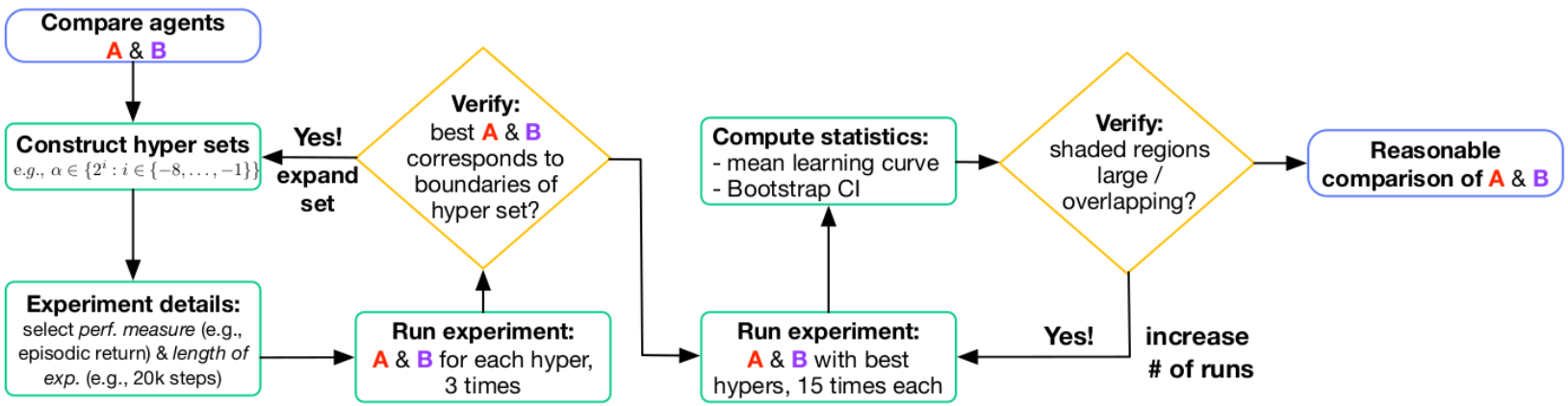
Plotting



- Showing the mean/median is never enough (need info about variance)
- The standard deviation is representative if the spread is Gaussian
- Rather take the $[0.1, 0.9]$ interval of values
- If less than 10 curves, plot them all

General advices

The evaluation organigram



- Don't spend too much budget before being sure that it is worth it

Evaluating a fully specified algorithm

1. Reporting mean or median performance is not enough
2. Do not report standard errors, based on wrong Gaussian assumption
3. Use steps rather than episodes (episodes are of varying length)
4. Choose to study the speed of early learning or the optimal performance (depending on your budget)

Dealing with hyper-parameters

1. Use parameter sensitivity plots to find adequate parameter ranges
2. Avoid reporting performance for the best hyperparameters for your algorithm. You will overestimate the quality of your algorithm
3. To debug your algorithm, you probably already performed hyperparameter tuning. Count this in your tuning budget to compare with others
4. Give more chance to your baselines than to your algorithm. If it still beats them, you can be more confident about improvement

Sensitivity curves

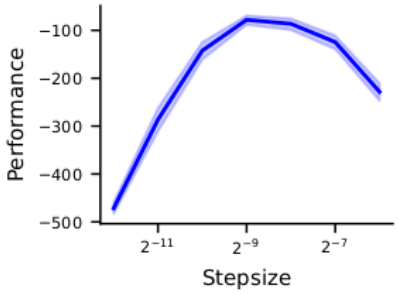


Figure 9: A good sensitivity curve that captures a wide range of the variable of interest and illustrates that performance changes smoothly as the hyperparameter changes.

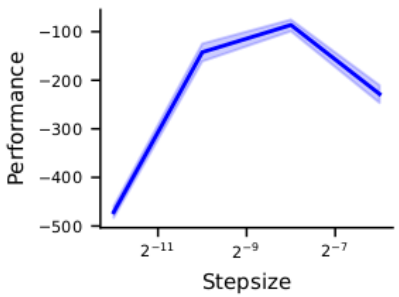


Figure 10: A sensitivity curve where the range of tested values may be too wide, instead of being focused in the region of interest. We lose some information around the peak performance and the algorithm appears quite sensitive. This sensitivity might be an artefact of the plot—testing insufficiently many values—rather than a property of the algorithm.

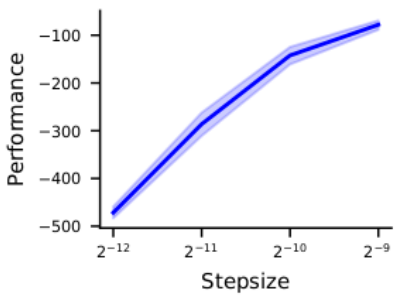


Figure 11: A sensitivity curve where we potentially missed the best performance. The best performing hyperparameter may be outside the range or may be the endpoint of the range, but we cannot tell with the presented information.

Comparing multiple algorithms

1. The seed is NOT a tunable hyperparameter. Random noise must keep random
2. Use baselines to contextualize performance: random, oracle
3. An algorithm rarely outperforms another one in all contexts. Be aware of the limits of your comparisons
4. Use confidence intervals, tolerance intervals, or probability of improvement. Intervals provide a sense of effect size and error rate
5. Comparing more than two algorithms requires even more care than comparing two algorithms (add Bonferroni correction)

Environment selection

1. Issue oriented research: small diagnostic environments reveal the properties of a method. They provide conceptual clarity and critical sanity checks.
2. By contrast, RL behavior in sophisticated environment is generally hard to analyse (the curse of RL benchmarks)
3. Designing environments is hard, do not assume an existing environment is reasonable
4. A minor change in a existing environment can have a major impact on RL performance. Only apply changes when strictly necessary and recheck the behavior of baselines
5. Adding more environments should not get your paper accepted if they do not provide additional insights (unless you are writing a systematic benchmarking paper)
6. Do not run an incomplete experiment due to insufficient resources: calibrate your experiment depending on your resources
7. Do not stick to the environments of the exploratory phase: you may have specialized your algorithm for these environments

Understanding agents with multiple metrics

1. Multidimensional analysis is key for truly understanding algorithm performance
2. You do not need to report all exploratory attempts. Only report those which provide insights
3. Include behavioural metrics, not just performance metrics

Any question?

Send mail to: Olivier.Sigaud@isir.upmc.fr

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., and Bellemare, M. (2021).

Deep reinforcement learning at the edge of the statistical precipice.
Advances in neural information processing systems, 34:29304–29320.

Bouthillier, X., Laurent, C., and Vincent, P. (2019).

Unreproducible research is reproducible.
In *International Conference on Machine Learning*, pages 725–734. PMLR.

Colas, C., Sigaud, O., and Oudeyer, P.-Y. (2019).

A hitchhiker's guide to statistical comparisons of reinforcement learning algorithms.
arXiv preprint arXiv:1904.06979.

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. (2018).

Deep reinforcement learning that matters.
In McIlraith, S. A. and Weinberger, K. Q., editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 3207–3214. AAAI Press.

Patterson, A., Neumann, S., White, M., and White, A. (2023).

Empirical design in reinforcement learning.
arXiv preprint arXiv:2304.01315.