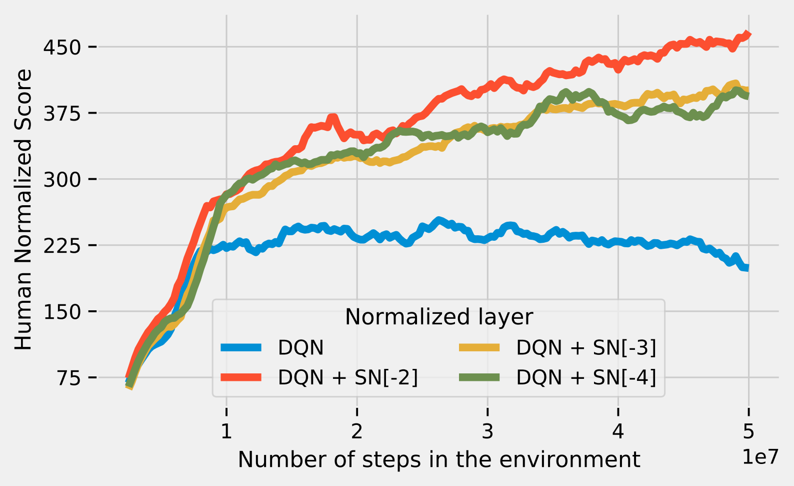
Spectral Normalisation for Deep Reinforcement Learning

An Optimisation Perspective

Florin Gogianu, **Tudor Berariu**, Mihaela Roșca,
Claudia Clopath, Lucian Bușoniu, Răzvan Pașcanu



Mean Human Normalized Score on 54 Atari games.

1. Does smoothness explain the performance?
2. Are there other optimisation-related effects at play?

Spectral Normalization

Linear case: $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$. Then:

$$f \text{ is K-Lipschitz in} \|\cdot\|_2 \;\Leftrightarrow\; \|\boldsymbol{W}\|_2 \leq K,$$

where $\|\boldsymbol{W}\|_2$ is the largest singular value or spectral radius of $\boldsymbol{W}$.

Spectral Normalization

Linear case: $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$. Then:

$$f \text{ is K-Lipschitz in} \|\cdot\|_2 \;\Leftrightarrow\; \|\boldsymbol{W}\|_2 \leq K,$$

where $\|\boldsymbol{W}\|_2$ is the largest singular value or spectral radius of $\boldsymbol{W}$.

Approximate the largest singular value ρ of $\boldsymbol{W}$ and normalize [Miyato et al., 2018]:

$$\begin{aligned} \rho &= \text{one-step-power-iteration}(\boldsymbol{W_t}) \\ \hat{\boldsymbol{W}}_{\boldsymbol{t}} &= \boldsymbol{W_t}/\rho \end{aligned}$$

Lipschitz constant of a Neural Network

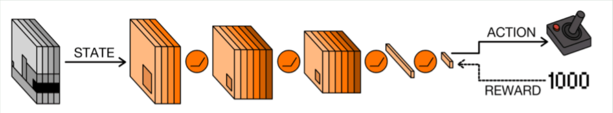


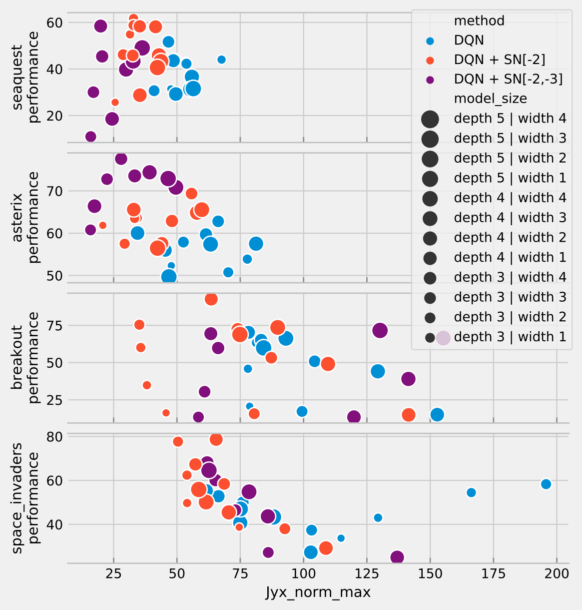
Figure 1: Typical architecture in DRL when learning from pixels, [Arulkumaran et al., 2017]

- Linear layers $\phi_{fc}(\boldsymbol{x}) = \boldsymbol{Wx} + \boldsymbol{b}$ can be K-Lipschitz if we constrain the spectral radius of $\boldsymbol{W}$.
- Convolutional operators are also linear maps.
- ReLU non-liniarities and Max-Pooling are 1-Lipschitz.

For a full neural network:

$$L(f) \leq \prod_{i=1} L(\phi_i)$$

Is it the smoothness?



Smoothness of the neural network (small norm of the Jacobian $\|\boldsymbol{J}_{yx}\|$) is not consistently correlated with performance.

Normalised score on four MinAtar games, ten seeds each.

The optimization perspective

MLP

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{W}_i} = \boldsymbol{J}_i \boldsymbol{\delta}_L \boldsymbol{a}_{i-1}^\intercal$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{b}_i} = \boldsymbol{J}_i \boldsymbol{\delta}_L$$

SN+bias scaling

$$\frac{\partial \hat{\mathcal{L}}}{\partial \boldsymbol{W}_i} = \rho^{-1} \ \boldsymbol{J}_i \hat{\boldsymbol{\delta}}_L \boldsymbol{a}_{i-1}^\intercal$$

$$\frac{\partial \hat{\mathcal{L}}}{\partial \boldsymbol{b}_i} = \rho^{-1} \ \boldsymbol{J}_i \hat{\boldsymbol{\delta}}_L$$

where $\rho^{-1} = \prod_{i \in S} \rho_i^{-1}$. Several equivalent schedulers become apparent:

1. `DivOut`: output divided by ρ^{-1}.
2. `DivGrad`: gradient divided by ρ^{-1}.
3. `MulEps`: Adam's ϵ multiplied by ρ^{-1}.

Schedulers recover most of the effect of SN

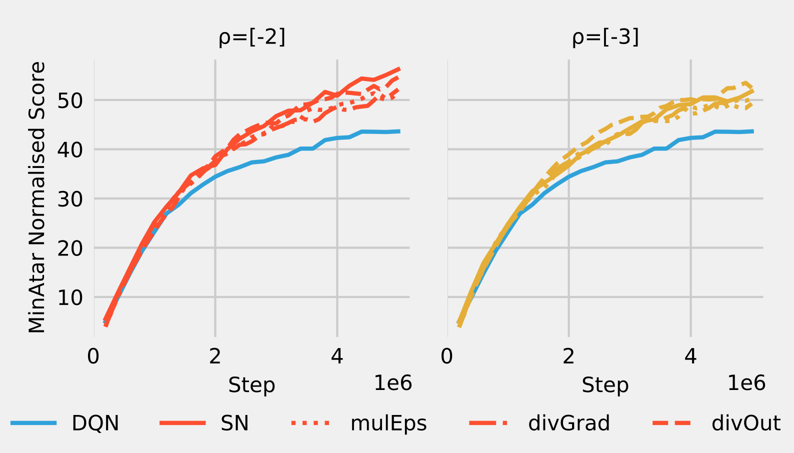


Figure 2: Spectral schedulers recover SN performance. Average normalised scores over MinAtar games and four different models

Atari results

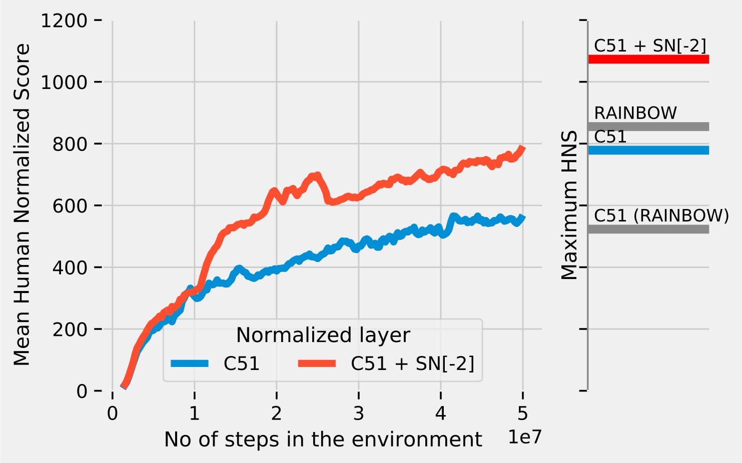


Figure 3: Categorical-DQN Performance. Mean Human Normalized Score 54 Atari games.

We improve on Rainbow, a much more complex agent that combines many DRL advances, while using only its cost function.

There is much to gain by designing better adapting optimisers for Deep Reinforcement Learning.

References i

Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017).
A brief survey of deep reinforcement learning.
CoRR, abs/1708.05866.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018).
Spectral normalization for generative adversarial networks.