Compress and Control

J. Veness, M. Bellemare, M. Hutter, A. Chua, G. Desjardins
Google DeepMind, Australian National University
Marc Bellemare and Alvin Chua at AAAI too.
A meta-algorithm for converting data compression / density estimation algorithms into RL agents.

E.g. Can make Zip play Pong!

Builds on earlier compression based classification / clustering work.

[Frank, Chui, Witten, 2000]
[Cilibrasi, Vitanyi, 2005]
CnC is a meta-algorithm for policy evaluation.

Converts any compressor / state density model into a policy evaluation algorithm.

Can be used for heuristic on-policy control.

Achieves generalization via density estimation; provides an alternative to the usual function approximation route.
Many model-based RL techniques involve learning a model that can imagine the future from the present given the past.
At a high level

- Determines Q-value by compression similarity of $s$ to previously seen states stratified by return.
Problem Setup

- Assume stationary policy $\pi$, $m$-horizon return $Z_t := \sum_{i=t}^{t+m-1} R_i$, a stationary MDP environment $\mu$, and finite $|S|$, $|A|$, $|R|$.

- Further assume $\mu + \pi$ gives rise to an ergodic (IR + AP + PR) Markov Chain.

- Goal: Estimate

$$Q^\pi(s_t, a_{t+1}) := \mathbb{E}[Z_{t+1} \mid S_t = s_t, A_{t+1} = a_{t+1}]$$
Re-express $Q$ in terms of a time independent distribution:

$$Q^\pi(s, a) = \sum_{z \in \mathcal{Z}} \mathbb{P}(Z = z \mid S = s, A = a)$$

Apply Bayes Rule:

$$Q^\pi(s, a) = \sum_{z \in \mathcal{Z}} \frac{\nu(s \mid z, a) \nu(z \mid a)}{\sum_{z' \in \mathcal{Z}} \nu(s \mid z', a) \nu(z' \mid a)}$$
Conditioning on the future return?!?!?

We show how this time independent distribution exists and can be learnt online.

The trick is to construct an augmented, ergodic HMC whose stationary distribution contains all the information we need.
Augmented HMC Construction

- Can show augmentation preserves ergodicity of underlying ergodic process $\{X_t := (A_t, S_t)\}$ given by $\mu + \pi$. 

$$W_{t-1} :=$$
Stationary Distribution

- Long term behaviour of the augmented HMC is governed by a unique stationary distribution $\nu_w$.

- Then we add on the return $Z'$ i.e.

$$(Z', A'_0, S'_0, R'_0, \ldots, A'_m, S'_m, R'_m) \sim \nu$$

- And can marginalize to get: $\nu(s, z, a)$
Value Estimation

\[ \hat{Q}_t^\pi (s, a) := \sum_{z \in Z} z \, w_{t}^{z,a} (s) \]

\[ w_{t}^{z,a} (s) := \frac{\rho_s (s \mid s_{0:n-1}^{z,a}) \, \rho_Z (z \mid z_{1:n}^a)}{\sum_{z' \in Z} \rho_s (s \mid s_{0:n-1}^{z',a}) \, \rho_Z (z' \mid z_{1:n}^a)} \]
Algorithm 1 CNC POLICY EVALUATION

Require: Stationary policy $\pi$, environment $\mathcal{M}$
Require: Finite planning horizon $m \in \mathbb{N}$
Require: Coding distributions $\rho_s$ and $\rho_z$

1: for $i = 1$ to $t$ do
2: Perform $a_i \sim \pi(\cdot \mid s_{i-1})$
3: Observe $(s_i, r_i) \sim \mu(\cdot \mid s_{i-1}, a_i)$
4: if $i \geq m$ then
5: Update $\rho_s$ in bucket $(z_{i-m+1}, a_{i-m+1})$ with $s_{i-m}$
6: Update $\rho_z$ in bucket $a_{i-m+1}$ with $z_{i-m+1}$
7: end if
8: end for
9: return $\hat{Q}_t^\pi$
Consistency of density estimator implies CnC provides consistent value estimates.

Frequency estimates can be used, and converges stochastically at rate $O(n^{0.5})$.

CTW can be used for larger problems, idealized version converges stochastically at rate $O(n^{0.5})$.
Figure 4: Average score over the last 500 episodes for three Atari 2600 games. Error bars indicate one inter-trial one standard error.
Varying model complexity

- A closer look at Pong...
Discussion

- Converts the problem of value estimation into one of probabilistic modelling. When is it worthwhile?

- Generalization occurs to the extent it occurs in the density/compression model.

- Seems to work well with essentially bad models. Learning can be quite data efficient.
Future Work

- Should account for policy drift when doing on-policy control. How?

- Not clear how to do exploration in a principled way for on-policy control.

- Bootstrapping CnC?

- Present work suited for problems where return space is sparse.

- Discretization should be straightforward, but needs demonstration; needed to run on all Atari games.
Questions...