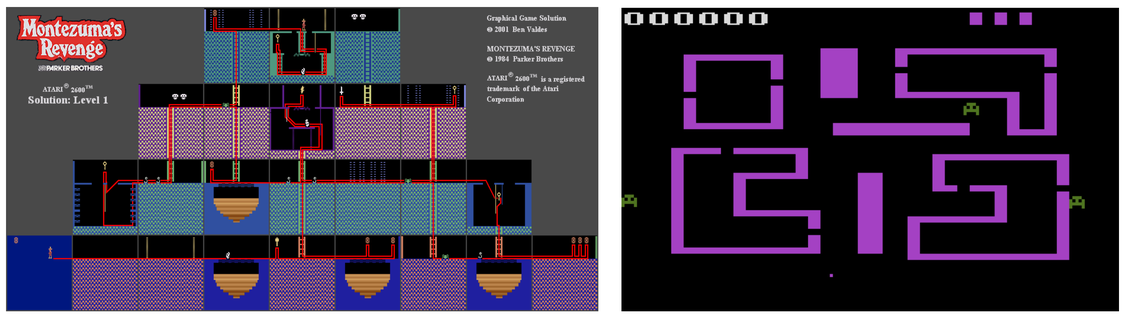
Count-based Exploration in Feature Space for Reinforcement Learning

{ jarrydmartinx and surajx }@gmail.com
{tom.everitt and marcus.hutter }@anu.edu.au

Problem



Efficient exploration in large domains: Many state-of-the-art RL algorithms still use inefficient exploration methods like ϵ-greedy [1]. There are efficient tabular *count-based* exploration algorithms for small MDPs, which drive the agent to reduce its uncertainty by visiting states that have low visit-counts [2]. However, these algorithms are ineffective in high-dimensional MDPs, because most states are never visited during training and the visit-count is zero almost everywhere.

Sparse Rewards: Efficient exploration is still more crucial if rewards are not dense enough to guide the agent through the state space. We require a generalised novelty measure for states (a pseudocount) to compute intrinsic rewards for exploration.

The ϕ-EB Algorithm

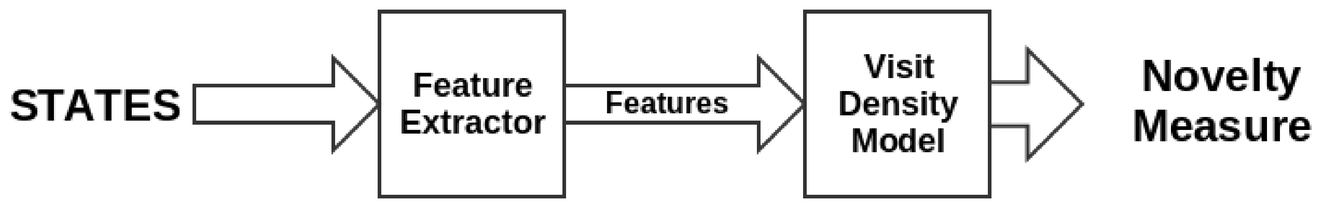
Require: β, t_{end}
  while $t < t_{\text{end}}$ **do**
    Observe r_t and features $\phi(s)$ for the current state s
    Compute joint feature probability $\rho_t(\phi) := \prod_i^M \rho_t^i(\phi_i)$
    for i in {1,...,M} **do**
      Update each probability ρ_{t+1}^i with observed feature ϕ_i
    end for
    Recompute joint probability $\rho_{t+1}(\phi) := \prod_i^M \rho_{t+1}^i(\phi_i)$
    Compute the ϕ-pseudocount $\hat{N}_t^\phi(s) := \dfrac{\rho_t(\phi)(1-\rho_{t+1}(\phi))}{\rho_{t+1}(\phi)-\rho_t(\phi)}$
    Compute the exploration bonus $\mathcal{R}_t^\phi(s,a) := \dfrac{\beta}{\sqrt{\hat{N}_t^\phi(s)}}$
    Add the bonus to the reward $r_t^+ := r_t + \mathcal{R}_t^\phi(s,a)$
    Pass $\phi(s)$, r_t^+ to RL algorithm to update θ_t
  end while
  return $\theta_{t_{\text{end}}}$

Contributions

We present a new method for computing novelty-based intrinsic rewards for exploration in large MDPs. Our method:

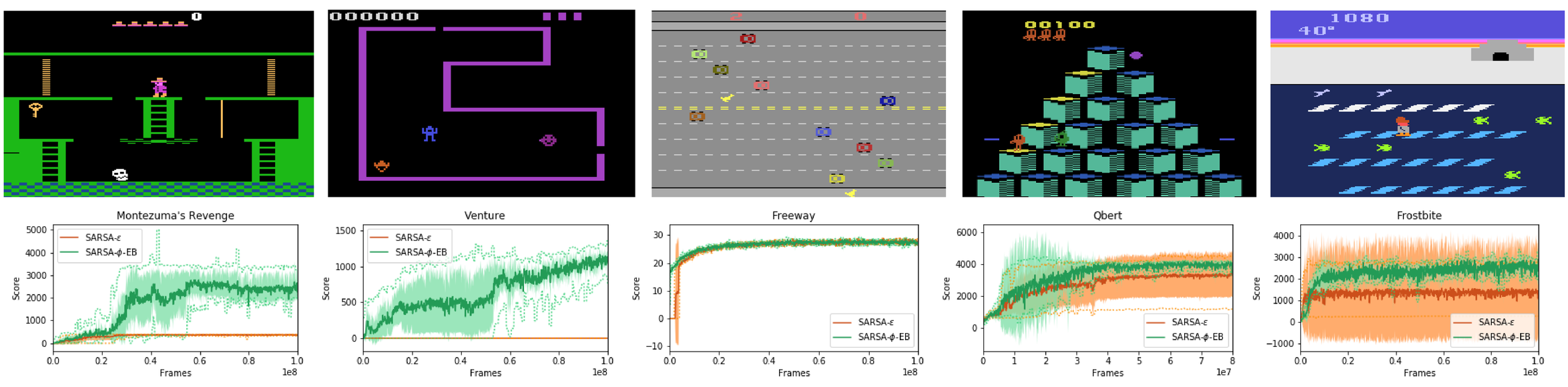
- Is simpler to implement and less computationally expensive than competitive proposals.
- Is compatible with any RL method that uses linear value-function approximation.
- Does not require an exploration specific feature representation, unlike previous proposals [1].
- Achieves near state-of-the-art results on the Arcade Learning Environment on sparse reward games.

Method: Measure Novelty in Feature Space



1. We construct a visit-density model in order to measure the novelty of a state. We exploit the feature map that is used for value function approximation, and construct a density model over the transformed *feature space*. The model assigns high probability (low novelty) to state feature vectors that share features with visited states.
2. A pseudocount (novelty measure) is then computed from these probabilities, using the method of [1]. States with frequently observed features are assigned higher counts. These counts serve as an approximate measure of the uncertainty associated with a state.
3. Exploration bonuses are then computed from these counts in order to encourage the agent to visit regions of the state-space with less familiar features.

Results



Plots compare learning curves for SARSA with ϵ-greedy and ϕ-EB exploration. Note that both algorithms use the same feature set and RL algorithm, and differ only in their exploration policies. ϕ-EB with $\beta = 0.05$ performs better on all tested games except Freeway.

On Venture, ϕ-EB's score is the second highest ever reported, and the third highest on Montezuma's Revenge. No other algorithm has achieved good scores on *both* these sparse reward games. On dense reward games (Frostbite, Qbert), nonlinear methods perform better.

References

[1] Bellemare et al. Unifying count-based exploration and intrinsic motivation. *CoRR*, abs/1606.01868, 2016.

[2] Strehl et al. An analysis of model-based interval estimation for Markov decision processes. *JCSS*, (8):1309–1331, 2008.