

Affordance Learning Math

Soft Affordances

Our aim is to develop an algorithm for learning a distribution over actions given a current state and lifted goal description:

$$P(a|s, LGD) \tag{1}$$

Process Overview

To learn Equation 2, we assume the distribution takes the following form:

$$P(a|s, LGD) = \textit{dirichelte} \tag{2}$$

- 1) **World Generation:** Generate M random worlds, each annotated with a Lifted Goal Description (LGD)
- 2) **Solve Policies:** Form a policy π_i for each of the M worlds using RTDP and null affordances. Some worlds will be too hard to solve without affordances; ignore worlds that cannot be solved.
- 3) **Compute distribution over actions, α :** Use each policy π_i , with the corresponding LGD s and predicates that “light up” (hand crafted) to get counts, α . Use α as the parameter for the Multinomial-Dirichlet distribution
- 4) **Compute action set size parameter, β :** Sample trajectories from each policy π_i , use the size of the set of actions used in each trajectory to get counts β . Use β as the parameters to a Dirichlet distribution.
- 5) Each time we use an affordance, sample its action set size (so we choose the number of actions to select, k), then sample k actions from that affordances distribution to get the pruned action set. Use as normal.

Math

$$Pr(\lambda \mid \alpha, \beta) = Pr(\lambda = \{a_1, \dots, a_k\} \mid n = k, \alpha) \cdot Pr(k \mid \beta) \tag{3}$$

Where the number of actions to select k is distributed as follows:

$$k \sim Dir(\beta) \quad (4)$$

And the probability of select any k actions is given by the dirichlet-multinomial distribution:

$$Pr(\lambda = \{a_1, \dots, a_k\} \mid n = k, \alpha) = \frac{\Gamma(A)}{\Gamma(k + A)} \prod_{i=1}^{|actions|} \frac{\Gamma(\delta(a_i \in \lambda) + \alpha_i)}{\Gamma(a_i)} \quad (5)$$