LiMMIRL-benchmarks

September 8, 2022

Things to check

- EM termination criterion
- Value of `em_nll_tolerance` used in the experiments
- Check why differences between responsibility matrices are large (try convex combinations of parameters with nearly the same nll, and see whether we get the same nll)
- Check behaviors are sufficiently differently (compute the value matrix for all policy-reward pairs); check whether each behavior is sufficiently different from random behavior;

Random MDP Target (large state space, high dimensional feature vectors) $S = 1000$, $A = 10$, $d = 1000$, $K = 5$

Check whether Aaron's code support sparse transition matrix. If yes, we can use much larger state space and action space.

- K behavior modes. The parameter vector $\theta_i \in R^d$ for the i-th behavior is randomly sampled from some distribution.

  Our implementation: each component of θ_i is equally likely to be 1 or -1, encoding preference or aversion. To ensure that the behaviors are sufficiently different, may need to ensure θ_i's are sufficiently different.

- States: $1, \ldots, S$. Each state s is assigned a random feature vector $\phi(s)$.

  Our implementation: $\phi(s)$ is sampled from the mixture of multivariate Gaussian $\sum_i \rho_i N(x; \theta_i, \sigma^2 I)$, where $\rho_i = 1/K$, and $\sigma = 0.5$.

- Actions: $1, \ldots, A$.
- Transition dynamics: each $P(\cdot|s, a)$ is randomly sampled from the unit simplex.
- Reward functions: $R_i(s, a) = \phi(s)^\top \theta_i$.
- Discount factor: 0.99, 0.98, 0.95

Trekking

- Grid world, with K destinations $g_1, \ldots, g_K$ of different characteristics. The feature vectors for these K destinations are randomly sampled, but make sure that they're sufficiently different.

  Each behavior aims to reach a destination. Choose the parameter vectors to be similar to the feature vectors of the goals.

  Feature vectors for non-destination states are determined as weighted combination of the destinations, where the weights are inversely proportional to the distances.

- Random transition matrix.

- $R_i(s, a) = \phi(s)^\top \theta_i$.