Do we use the Right Measure?  
Challenges in Evaluating Reward Learning Algorithms  
–Supplementary Material–

A Proofs

**Theorem 1** (Unbounded Reward Difference). Let \( w^{\text{user}} \) be a user weight, and \( w' \) be an estimate, where the alignment is \( \delta \leq \alpha(w', w^{\text{user}}) < 1 \) for some \( \delta < 1 \). The difference in reward \( R(T^{\text{user}}, w^{\text{user}}) - R(T', w^{\text{user}}) \) is unbounded.

**Proof.** We consider a discrete planning problem with two features. Let there be only two solutions \( T^A \) and \( T^B \) with features \( \phi(T^A) = [-1, -M] \) and \( \phi(T^B) = [-N, 0] \) where \( M > 0 \) and \( N > 1 \). Further, let the user weights be \( w^{\text{user}} = [1, 0] \), and the estimate \( w' = [1, \epsilon] \), where \( \epsilon > 0 \). We notice that \( \alpha(w', w^{\text{user}}) \geq \delta \) as \( \epsilon \to 0 \) for any \( \delta < 1 \), that is, the alignment becomes arbitrarily close to 1 for small \( \epsilon \).

First, we calculate \( w^{\text{user}} \cdot \phi(T^A) = -1 \) and \( w^{\text{user}} \cdot \phi(T^B) = -N \). Since \( N > 1 \) the trajectory \( T^A \) collects the higher reward and hence is the optimal solution for \( w^{\text{user}} \). Further, we have \( w' \cdot \phi(T^A) = -1 - M \epsilon \) and \( w' \cdot \phi(T^B) = -N \). We construct the case where \( T^B \) is the optimal solution for the estimate \( w' \), i.e., the estimated weights result in a different, suboptimal trajectory:

\[
-1 - M \epsilon < -N. \tag{1}
\]

The difference in reward is

\[
R(T^{\text{user}}, w^{\text{user}}) - R(T', w^{\text{user}}) = w^{\text{user}} \cdot \phi(T^A) - w^{\text{user}} \cdot \phi(T^B) = -1 + N. \]

This is unbounded if we can pick an arbitrarily large \( N \) such that (1) is satisfied as \( \epsilon \to 0 \). Choosing \( M = N^2 / \epsilon \) simplifies (1) to \( N^2 > N - 1 \) which is satisfied for any \( N > 1 \). Hence, \( N \) has no upper bound, making the reward difference unbounded.

**Theorem 2** (Unbounded Test Error). Let \( w^{\text{user}} \) be a user weight, and \( w' \) be an estimate. Further, let \( I^{\text{train}} \) be a training instance, where the relative reward \( R_{\text{rel}}^{\text{train}}(w', w^{\text{user}}) \) is taking values in \( [\delta, 1] \) for some \( \delta < 1 \). There exist test instances \( I^{\text{test}} \) where the relative reward \( R_{\text{rel}}^{\text{test}}(w', w^{\text{user}}) \) has no tighter lower bound than 0.

**Proof.** Again we consider a discrete planning problem with two features. We construct a training instance \( I^{\text{train}} \) with only two solutions \( T^A \) and \( T^B \) with features \( \phi(T^A) = [-1, -2] \) and \( \phi(T^B) = [-5, 0] \). Further, let the user weights be \( w^{\text{user}} = [1, 1] \), and the estimate \( w' = [1, 0] \). For both weights,
\( T^A \) achieves the higher reward, i.e., is the respective optimal solution. Thus, we have relative reward 
\( R_{\text{rel}}^{\text{train}}(w', w_{\text{user}}) = 1 \), the estimated weights yields an optimal solution.

Now, consider the test instance \( I_{\text{Test}} \) where we again have only two solutions \( T^C \) and \( T^D \) with 
features \( \phi(T^C) = [-2, 0] \) and \( \phi(T^D) = [-1, -N] \) for some \( N > 1 \). Thus, \( w' \cdot \phi(T^C) = -2 \) 
and \( w' \cdot \phi(T^D) = -1 \); the solution \( T^D \) is optimal for weights \( w' \). Further, \( w_{\text{user}} \cdot \phi(T^C) = -2 \) 
and \( w_{\text{user}} \cdot \phi(T^D) = -1 - N \), implying that \( T^C \) is always the optimal solution for weights \( w_{\text{user}} \).

Hence, the relative reward is
\[
R_{\text{rel}}^{\text{test}}(w', w_{\text{user}}) = \frac{w_{\text{user}} \cdot \phi(T^C)}{w_{\text{user}} \cdot \phi(T^D)} = \frac{2}{1 + N}.
\]

By picking an arbitrarily large \( N \) the relative reward in the test scenario can become arbitrarily small, implying that no lower bound greater than 0 exists.

### B Additional Simulation Results

Figure 1 provides example plots for the numerical experiments with the Server task, showing alignment against relative reward, as well as relative reward in training against testing.

(a) Alignment-reward relationship. (b) Training-Testing Relationship.

Figure 1: Examples for Server task.