Appendices

A. Recalibration

Most predictive models are not calibrated out-of-the-box due to modeling bias and computational approximations. However, given an arbitrary pre-trained forecaster $H : \mathcal{X} \to (\mathcal{Y} \to [0, 1])$ that outputs CDFs F, we may train an auxiliary model $R : [0, 1] \to [0, 1]$ such that the forecasts $R \circ F$ are calibrated in the limit of enough data. This procedure, called recalibration, is simple to implement, computationally inexpensive, and can be applied to any probabilistic regression model in a black-box manner. Furthermore, it does not increase the loss function of the original model if it belongs to a large family of objectives called proper losses (Kull & Flach, 2015; Kuleshov & Ermon, 2017).

Algorithm 2 CALIBRATE: Recalibration of Transition Dynamics

Input: Uncalibrated transition model $\widehat{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ that outputs CDFs $F_{s,a} : \mathcal{S} \to [0, 1]$, and calibration set $\mathcal{D}_{\text{cal}} = \{(s_t, a_t), s_{t+1}\}\}_{t=1}^{N}$
Output: Auxiliary recalibration model $R : [0, 1] \to [0, 1]$.

1. Construct a recalibration dataset

$$\mathcal{D} = \left\{ \left(F_{s_t,a_t}(s_{t+1}), \hat{P}(F_{s_t,a_t}(s_{t+1})) \right) \right\}_{t=1}^{N}$$

   where

$$\hat{P}(p) = \frac{1}{N} \sum_{t=1}^{N} \mathbb{I}[F_{s_t,a_t}(s_{t+1}) \leq p].$$

2. Train a model $R : [0, 1] \to [0, 1]$ (e.g. sigmoid or isotonic regression) on $\mathcal{D}$.

When $\mathcal{S}$ is discrete, a popular choice of R is Platt scaling (Platt et al., 1999); Kuleshov et al. (2018) extends Platt scaling to continuous variables. Either of these methods can be used within our framework. Since this paper focuses on continuous state spaces, we use the method of Kuleshov et al. (2018) described in Algorithm 2, unless otherwise indicated.

B. Calibrated Discrete MDP

We provide a proof in the discrete case that calibrated uncertainties result in correct expectations with respect to the true probability distribution, and thus using calibrated dynamics allow accurate evaluation of policies.

Consider a discrete state MDP (S, A, T, R) and a policy π over this MDP. We are interested in evaluating the goodness of this policy at any state s using the usual value iteration:

$$V_\pi(s) = R(s) + \gamma \mathop{\mathbb{E}}_{a \sim \pi(\cdot|s)} \mathop{\mathbb{E}}_{s' \sim T(\cdot|s,a)} [V(s')]. \tag{5}$$

For the given policy π, there exists a stationary distribution σ_π that would be obtained from running this policy for a long time. We define the value of the entire policy $V(\pi)$ as an expectation with respect to this stationary distribution i.e.

$$V(\pi) = \mathop{\mathbb{E}}_{s \sim \sigma_\pi} [V(s)]. \tag{6}$$

We want to show that replacing the true dynamics T with calibrated dynamics $\widehat{T}$ does not affect our evaluation of the policy π. To have a well-defined notion of calibration, we need to define a joint distribution over the inputs and outputs of a predictive model. The inputs are the current state-action pair (s, a) the outputs are distributions over the next state s'. To define a joint distribution $\mathbb{P}$ over (S, A) and S', we use the stationary distribution σ_π, the policy π, and the transition dynamics T to define the sub-components using the chain rule:

$$\mathbb{P}((s, a), s') = \mathbb{P}(s'|s, a)\, \mathbb{P}(a|s)\, \mathbb{P}(s) \tag{7}$$

$$= T(s'|s, a)\, \pi(a|s)\, \sigma_\pi(s). \tag{8}$$

Note that defining the joint distribution $\mathbb{P}$ this way lets us rewrite $V(\pi)$ more simply as

$$V(\pi) = \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[V(s)] \tag{9}$$

$$= \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[R(s)] + \gamma \mathop{\mathbb{E}}_{s\sim\sigma_\pi} \mathop{\mathbb{E}}_{a\sim\pi(\cdot|s)} \mathop{\mathbb{E}}_{s'\sim T(\cdot|s,a)}[V(s')] \tag{10}$$

$$= \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[R(s)] + \gamma \mathop{\mathbb{E}}_{((s,a),s')\sim\mathbb{P}}[V(s')] \tag{11}$$

$$= \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[R(s)] + \gamma \mathop{\mathbb{E}}_{s'\sim\mathbb{P}(S')}[V(s')] \tag{12}$$

These definitions allow us to state our theorem.

Theorem 2. *Let (S, A, T, R) be a discrete MDP and let π be a stochastic policy over this MDP. Define $\mathbb{P}$ to be a joint distribution over state-action and future state pairs $((s, a), s')$ as outlined in Equation 8. Then the value of policy π under the true dynamics T is equal to the value of the policy under some other dynamics $\widehat{T}$ that are calibrated with respect to $\mathbb{P}$.*

Proof. Since $\widehat{T}$ is calibrated with respect to $\mathbb{P}$, we have $\mathbb{P}(s' = j \mid \widehat{T}(s' = j|s, a) = p)) = p$. Let $\widehat{V}(\pi)$ be the value of policy π under $\widehat{T}$. Then we have

$$\widehat{V}(\pi) = \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[R(s)] + \gamma \mathop{\mathbb{E}}_{(s,a)\sim\mathbb{P}((s,a))} \mathop{\mathbb{E}}_{s'\sim\widehat{T}(s'|s,a)}[V(y)] \tag{13}$$

$$= \mathop{\mathbb{E}}_{s\sim\sigma_\pi}[R(s)] + \gamma \mathop{\mathbb{E}}_{s'\sim\mathbb{P}(S')}[V(y)] \tag{14}$$

$$= V(\pi), \tag{15}$$

where the second line follows immediately from Lemma 2.

■

Lemma 2. *Consider a pair of jointly distributed variables $(X, Y) \sim \mathbb{P}$ over $\mathbb{X}$ and $\mathbb{Y}$ where $\mathbb{X} = \{x_1, \ldots, x_k\}$ and $\mathbb{Y} = \{y_1, \ldots, y_m\}$ are discrete spaces, and let $Q(Y|X)$ be a distribution that is calibrated with respect to $\mathbb{P}$. In other words, $\mathbb{P}(Y = y \mid Q(Y = y \mid X) = p) = p$. Then, for any arbitrary function $g : \mathbb{Y} \to \mathcal{S}$ with which we want to take an expectation, the following equality holds:*

$$\mathop{\mathbb{E}}_{y\sim\mathbb{P}(Y)}[g(y)] \quad = \mathop{\mathbb{E}}_{\substack{x\sim\mathbb{P}(X)\\ y\sim Q(Y|X=x)}}[g(y)]\,. \tag{16}$$

Proof. We can rewrite the expectation on the LHS of Equation 16 using the law of total probability and the chain rule to get:

$$\begin{aligned}
\mathop{\mathbb{E}}_{y\sim\mathbb{P}(Y)}[g(y)] &= \sum_{y\in\mathbb{Y}} g(y)\mathbb{P}(Y = y) \\
&= \sum_{y\in\mathbb{Y}} g(y) \int_0^1 \mathbb{P}(Y = y, Q(Y = y \mid X) = p)\, dp \\
&= \sum_{y\in\mathbb{Y}} g(y) \int_0^1 \mathbb{P}(Y = y \mid Q(Y = y \mid X) = p)\, \mathbb{P}(Q(Y = y \mid X) = p)\, dp.
\end{aligned}$$

Note that in the above derivation we perform the following slight abuse of notation:

$$\{Q(Y = y|X) = p\} = \{X \mid Q(Y = y|X) = p\}.$$

We can apply the calibration assumption to replace the conditional term with p and rewrite $\mathbb{P}(Q(Y = y \mid X) = p)$ as a sum over elements of $\mathbb{X}$. This gives: p

$$\begin{aligned}
\mathop{\mathbb{E}}_{y\sim\mathbb{P}(Y)}[g(y)] &= \sum_{y\in\mathbb{Y}} g(y) \int_0^1 p \cdot \mathbb{P}(Q(Y = y \mid X) = p)\, dp \\
&= \sum_{y\in\mathbb{Y}} g(y) \int_0^1 p \cdot \sum_{x\in\mathbb{X}} \mathbb{I}[Q(Y = y \mid X = x) = p] \cdot \mathbb{P}(X = x)\, dp \\
&= \sum_{y\in\mathbb{Y}} g(y) \sum_{x\in\mathbb{X}} Q(Y = y \mid X = x) \cdot \mathbb{P}(X = x) \\
&= \sum_{x\in\mathbb{X}} \mathbb{P}(X = x) \sum_{y\in\mathbb{Y}} g(y) \cdot Q(Y = y \mid X = x) \\
&= \mathop{\mathbb{E}}_{x\sim\mathbb{P}(X)} \mathop{\mathbb{E}}_{y\sim Q(Y|X=x)} [g(y)]\,.
\end{aligned}$$

■

C. Additional Figures

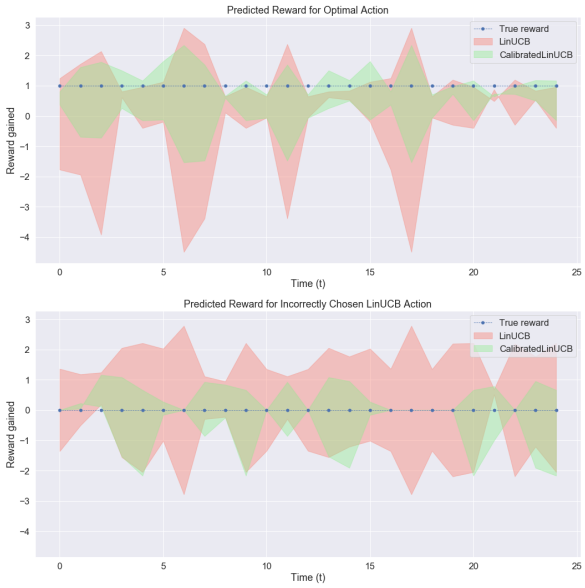

Figure 4. Predicted expected reward for both LinUCB and CalLinUCB algorithms on the covertype dataset. Figures show predictions at random timesteps where CalLinUCB chose the optimal action but LinUCB did not. Top: Predicted reward of both algorithms for the optimal action. Bottom: Predicted reward of both algorithms for the action which the algorithm chose to pick instead of the optimal action at that timestep. We can see LinUCB consistently underestimates reward from optimal action and overestimates reward from other actions. On the other hand, CalLinUCB is more accurate in its uncertainty predictions.

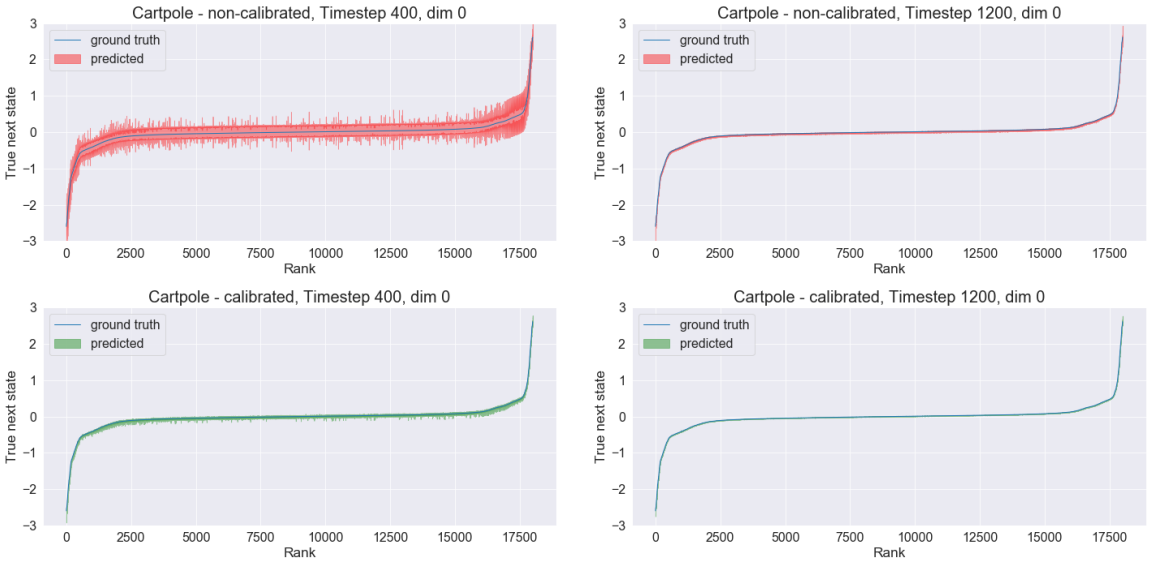

Figure 5. Cartpole future state predictions. The calibrated algorithm has much tighter uncertainties around the true next state in early training iterations. Later into training, their uncertainties are almost equivalent.