Calibrated Model-Based Deep Reinforcement Learning

ICML 2019

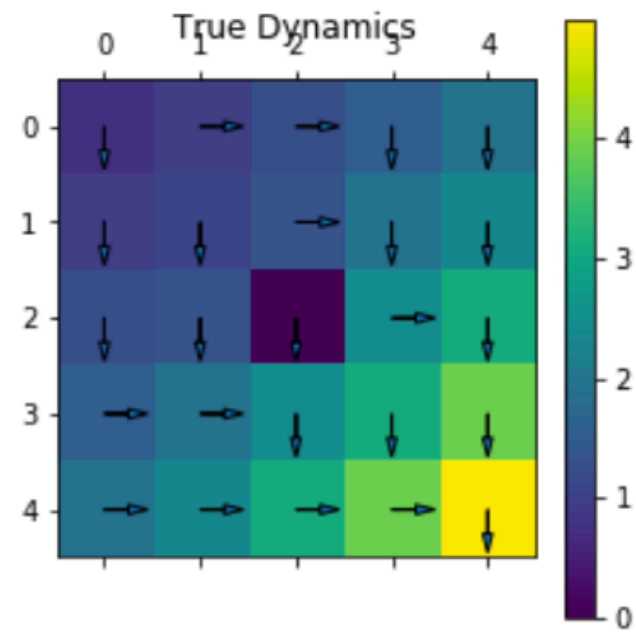
Ali Malik*, Volodymyr Kuleshov*, Jiaming Song,
Danny Nemer, Harlan Seymour, Stefano Ermon

June 13, 2019

*equal contribution

Overview

- Importance of predictive uncertainty
- Which uncertainties matter for MBRL?
- Calibration in MBRL
- Recalibrating MBRL
- Results



Importance of Predictive Uncertainty

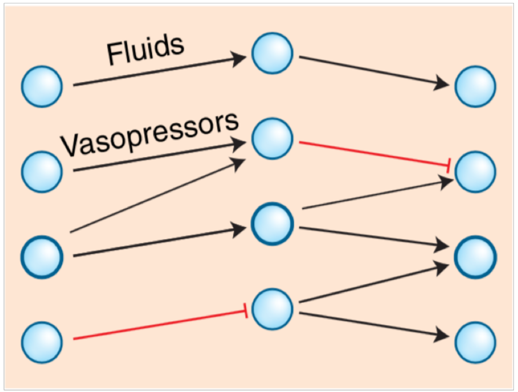
Assessing uncertainty is crucial in modern decision-making systems

RL + Control

Obstacle avoidance, reward planning

Kahn et al. (2018)
Chua et al. (2018)

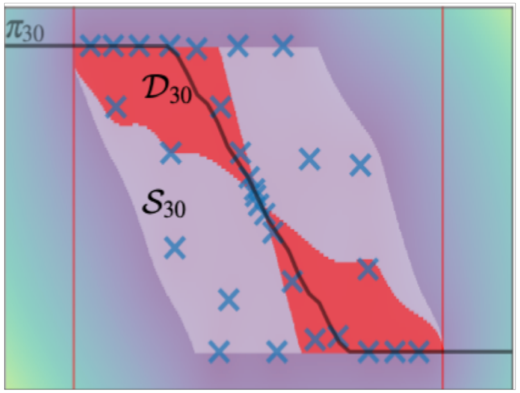
Medicine



Diagnosis, risk prediction, treatment recommendation.

Saria (2018)
Heckerman et al. (1989)

Safety



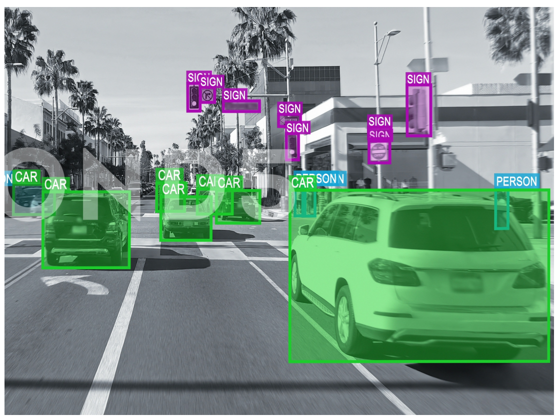
Safe exploration

Berkenkamp et al. (2017)

Importance of Predictive Uncertainty

Assessing uncertainty is crucial in modern decision-making systems

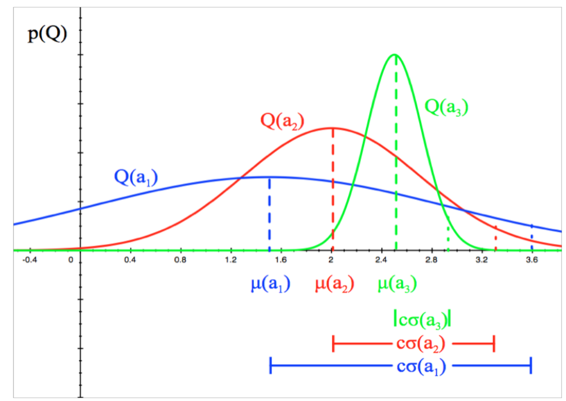
Autonomous Driving



Segmentation, object detection, depth estimation.

Smith & Cheeseman (1986)
McAllister et al. (2017)

Upper Confidence Bounds



Balancing exploration and exploitation

Auer et al. (2002)
Li et al. (2010)

Importance of Predictive Uncertainty

Modelling uncertainty **accurately** is crucial

Key question:

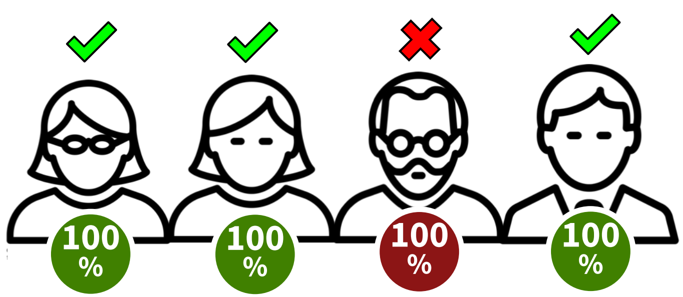
Which uncertainties are important in Model-Based Reinforcement Learning?

What constitutes good probabilistic forecasts?

Literature on **proper scoring rules** suggest two important factors

Sharpness

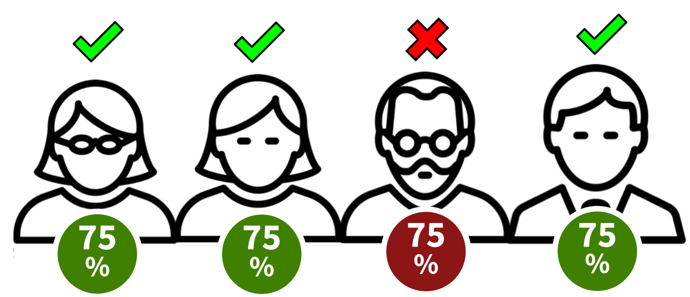
Predictive distributions should be focused i.e have low variance



Sharp

Calibration

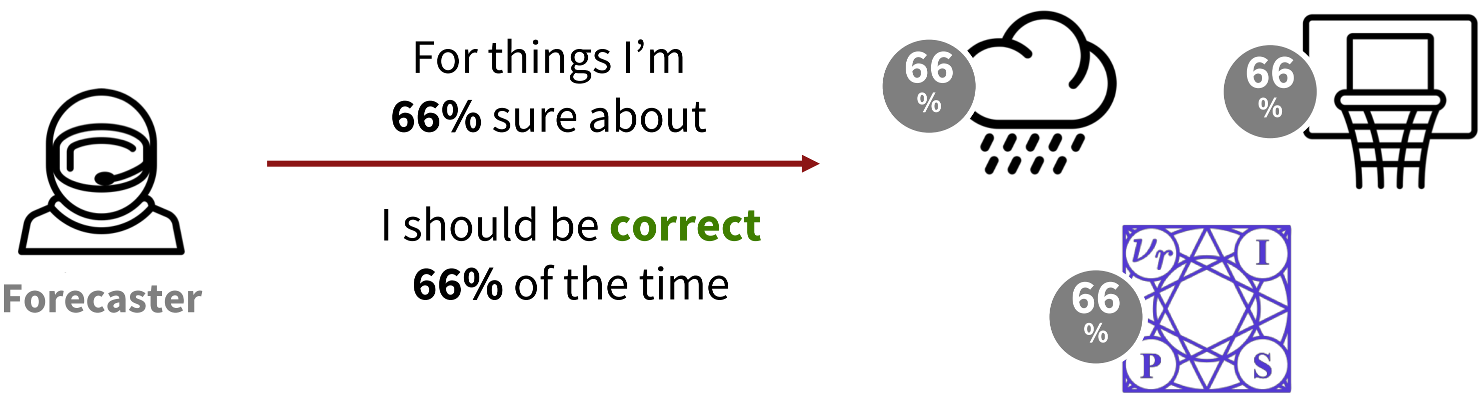
Uncertainty should be empirically accurate i.e. true value should fall in a p% confidence interval p% of the time



Calibrated

Calibration

Calibration measures reliability of probabilistic claims.



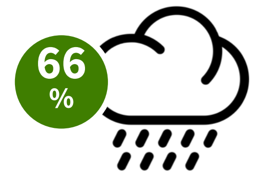
Calibration

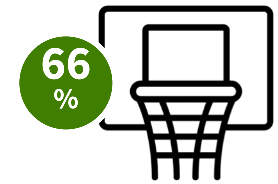
Calibration measures reliability of probabilistic claims.

Forecaster

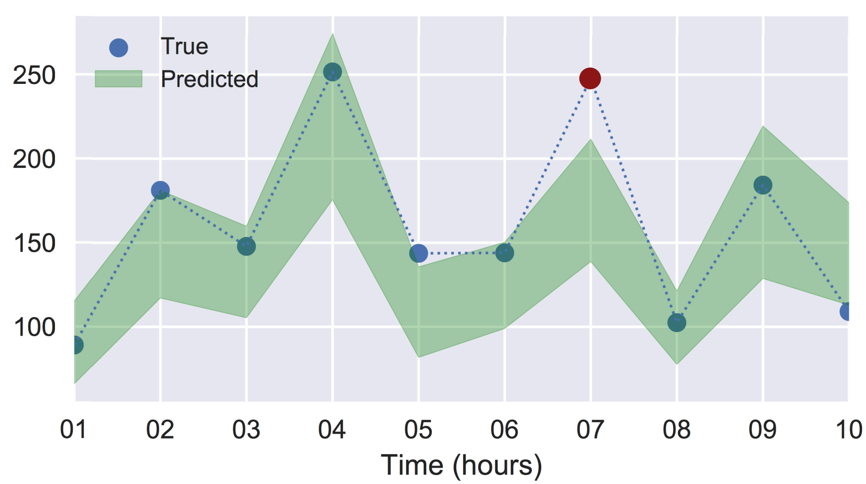
For things I'm
66% sure about

I should be **correct**
66% of the time





For regression:



Predicted probability
for credible interval

=

True probability of Y falling
in the interval

Calibration vs Sharpness

There is an inherent trade-off between calibration and sharpness

What should we prioritise?

Claim:

In model-based reinforcement learning,
uncertainties should be *calibrated*

Importance of Calibration

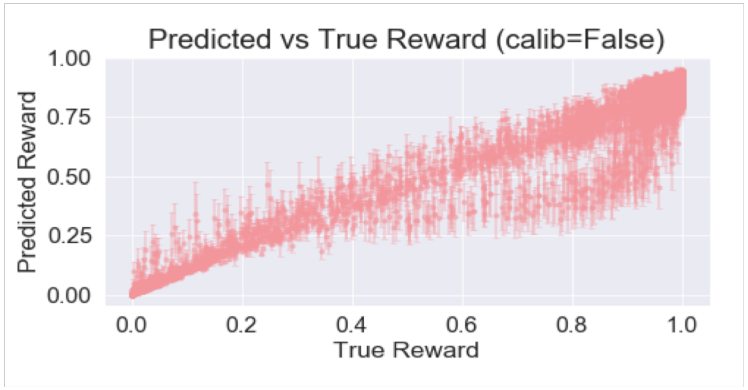
Calibration is really important in model-based reinforcement learning.

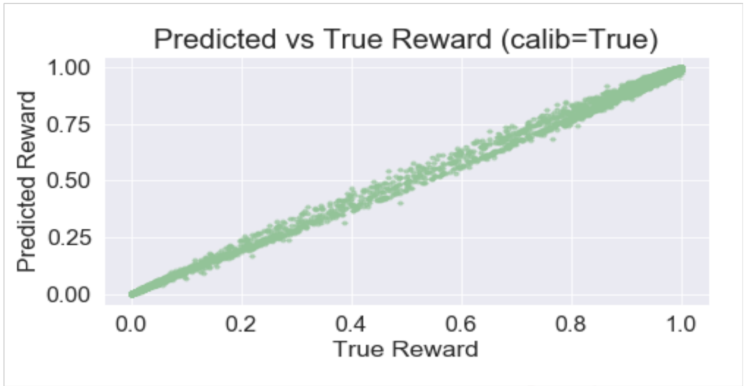
Planning

Calibrated uncertainties lead to better estimates of expectation.

$$V'(s) \leftarrow \mathbb{E}_{a\sim\pi(\cdot|s)}\left[\sum_{s'\in S}\hat{T}(s'|s,a)\left(r(s')+V(s')\right)\right]$$

Theorem: The value of policy π for an MDP under the true dynamics T is equal to the value of the policy under some other dynamics $\hat{T}$ that are calibrated with respect to the MDP.





Importance of Calibration

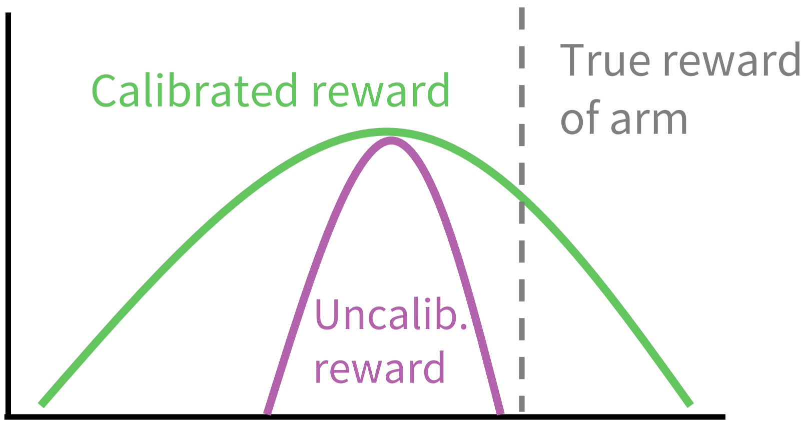
Calibration is really important in model-based reinforcement learning.

Exploration

Many exploration/exploitation algorithms use Upper Confidence Bounds (UCBs) to guide choices:

$$\arg\max_{a \in \mathcal{A}} \left(\mathbf{x}^\top \hat{\theta}_a + \alpha \cdot \sqrt{\mathbf{x}^\top \hat{\Sigma}_a^{-1} \mathbf{x}} \right)$$

Calibration naturally improves UCBs, resulting in better exploration.

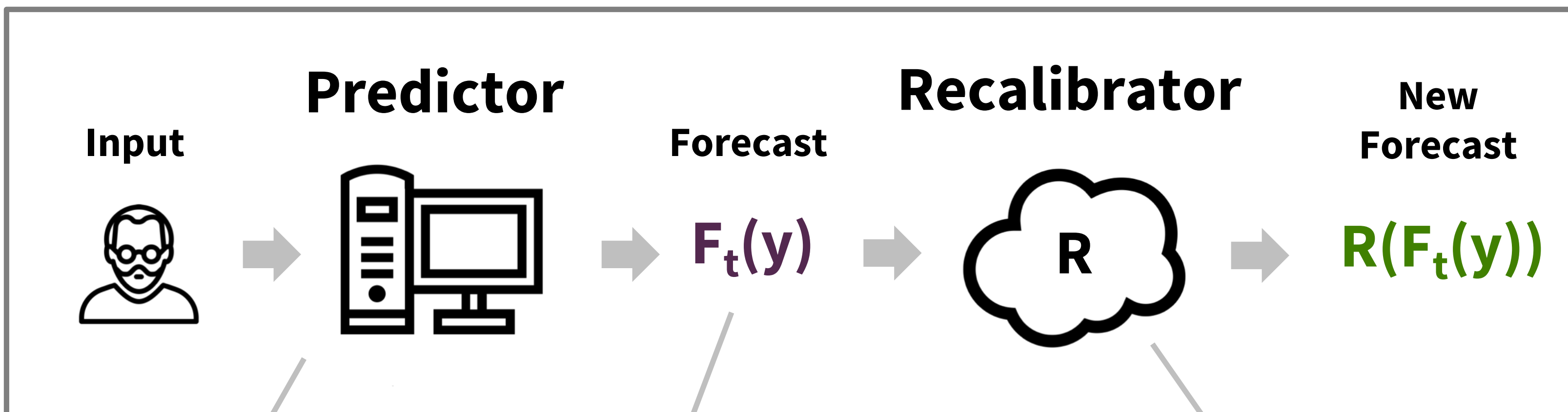


Calibrating Model-Based RL

Uncertainties derived from modern neural networks are often uncalibrated.

We can recalibrate any forecaster using work by Kuleshov et al (2018):

Recalibration



Can be any model (seen as black box)

$H : \mathcal{X} \to (\mathcal{Y} \to [0, 1])$

Uncalibrated CDF

$F : \mathcal{Y} \to [0, 1]$

Transforms probabilities coming out of F

$R : [0, 1] \to [0, 1]$

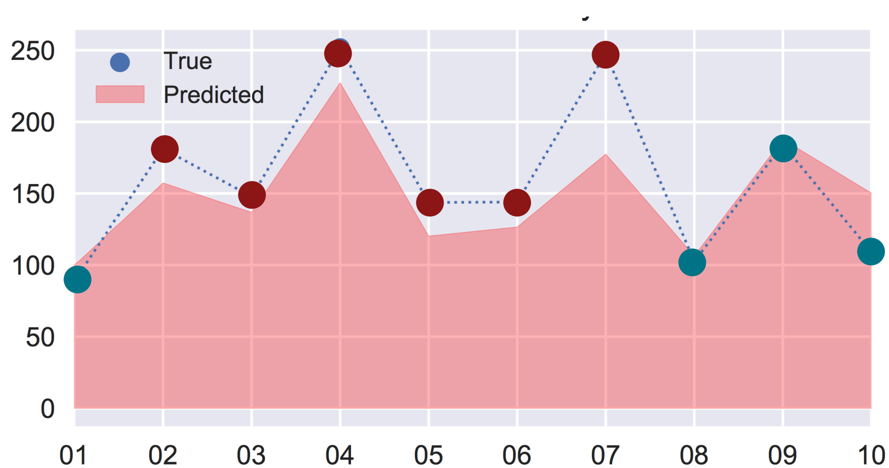
Deriving the Ideal Recalibrator

We learn a mapping between predicted and true (empirical) probabilities.

Calibration

$$p = \mathbb{P}(Y \leq F_X^{-1}(p))$$

Fact: Ideal recalibrator is
R(p) = P(Y ≤ F_X^{-1}(p)).



what model predicts		**what data says**
60% quantile	➡	40% quantile
70% quantile	➡	45% quantile
80% quantile	➡	55% quantile
⋮		⋮
p	➡	$\mathbb{P}(F_X(Y) \leq p)$

Calibrating Model-Based RL

This gives the following algorithm for MBRL:

Calibrated MBRL

Train calibrated transition model $\widehat{T}$ from observations by repeatedly:

1. **`Explore`**: Collect observations using current transition model.
2. **`LearnModel`**: Retrain transition model using new observations.
3. **`LearnCalib`**: Learn recalibrator R on held-out subset of observations.
4. **`Recalibrate`**: Set $\widehat{T} \; = R \; o \; \widehat{T}$

Results: Contextual Bandits

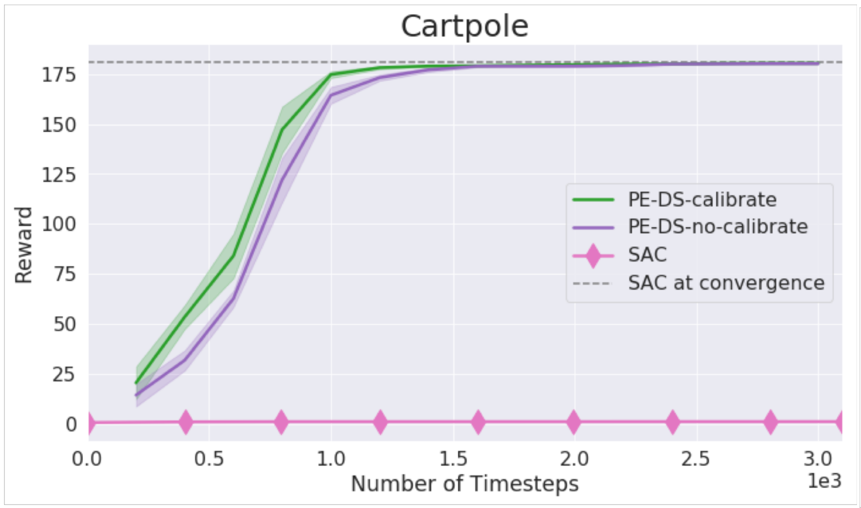
We can apply this scheme to the LinUCB algorithm for contextual bandits:

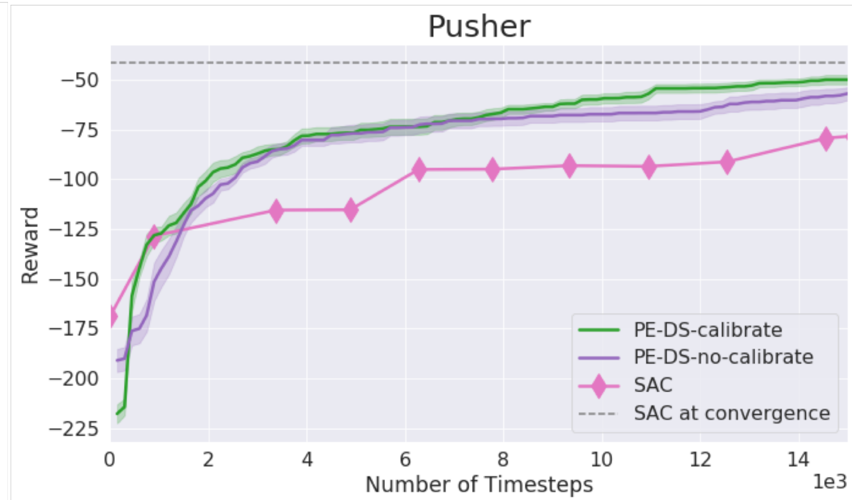
	LinUCB	*CalLinUCB*	*Optimal*
Linear	1209.8 ± 12.1	1210.3 ± 12.1	1231.8
Beta	1176.3 ± 11.9	1174.6 ± 12.0	1202.3
Mushroom	1429.4 ± 154.0	$\mathbf{1676.1 \pm 164.1}$	3122.0
Covertype	558.14 ± 3.5	$\mathbf{677.8 \pm 5.0}$	1200.0
Adult	131.3 ± 1.2	$\mathbf{198.9 \pm 4.7}$	1200.0
Census	207.6 ± 1.7	$\mathbf{603.7 \pm 3.8}$	1200.0

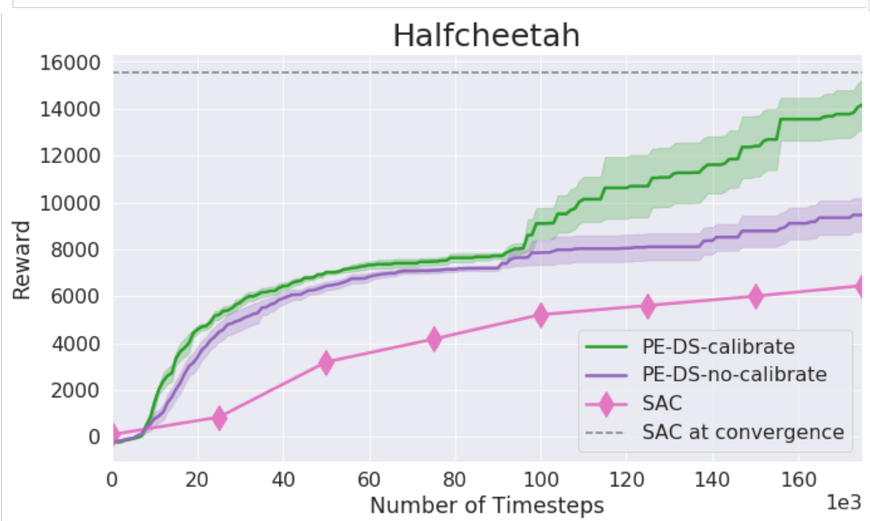
Recalibration consistently improves the exploration/exploitation balance in contextual bandits tasks.

Results: MuJoCo Continuous Control

We calibrate the probabilistic ensemble model from Chua et al. 2018 and show noticeable improvement in sample complexity across different tasks:



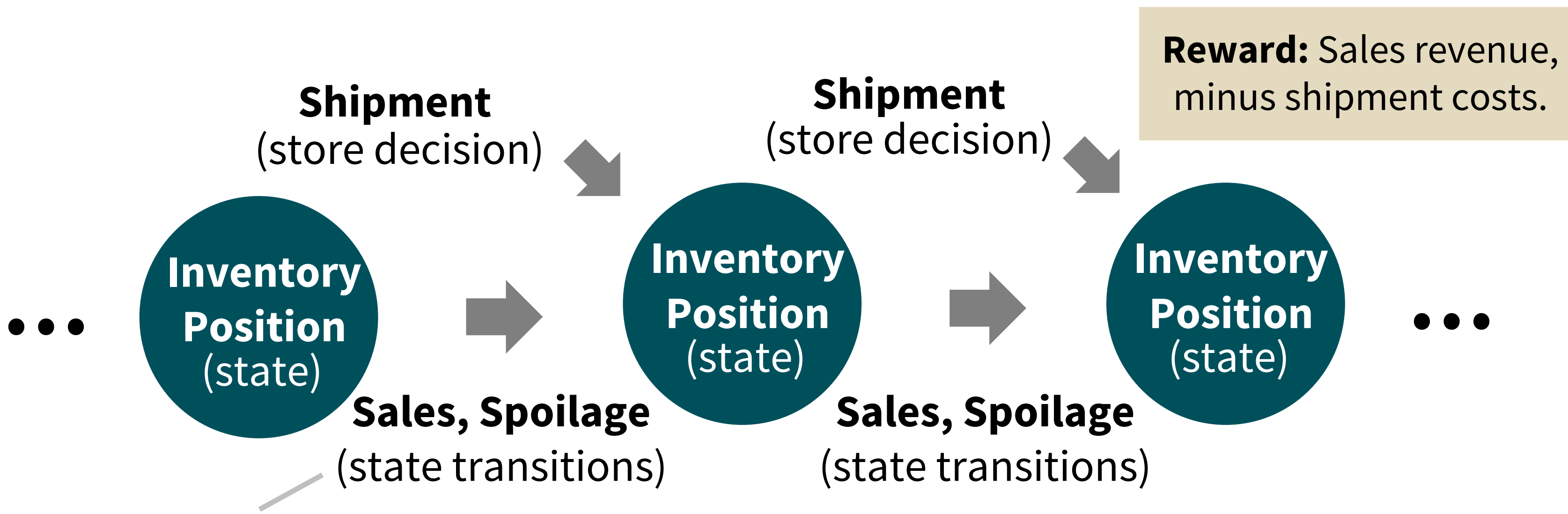




Recalibration improves the sample complexity in continuous control tasks.

Results: Inventory Planning

We also calibrate a Bayesian DenseNet tasked with controlling the inventory of perishable goods in a store



Calibrated	Uncalibrated	Heuristic
-16,793	-20,506	-25,516

Thank you!

Stop by poster #36
for more details