The Benefits of Interaction for Learning Sequential Predictions

Stephane Ross
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
stephaneross@cmu.edu

J. Andrew Bagnell
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
dbagnell@ri.cmu.edu

Sequential prediction problems arise commonly in many areas of robotics and information processing. Examples of such problems include: 1) imitation learning, where a learner must predict a sequence of actions given observations revealed over time to perform a task, based on previously observed demonstrations; 2) structured prediction, which can often be tackled by constructing the structured output from a sequence of interdependent predictions as in SEARN [1]; 3) model-based reinforcement learning (MBRL), where we learn a model of a system from observations in order to predict its behavior under various action sequences for planning. Learning in such sequential problems is challenging as the predictor and data-generation process are inextricably intertwined. This often leads to a significant mismatch between the distribution of observed data during training and test executions (see Figure 1). Naively applying statistical learning methods can yield a predictor that makes good predictions during training but performs badly at the task during test execution.

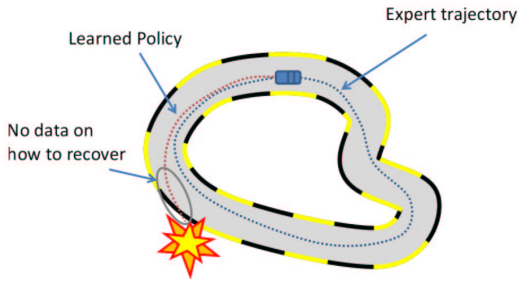

Figure 1: Example driving scenario. Training: an expert demonstrates how to drive. Testing: the learned predictor drives and, due to mistakes, ends up in new situations leading to poor behavior.

We present a simple and general strategy to address this issue where learning occurs over repeated rounds of interaction. By applying no-regret online algorithms over these rounds of interaction, we obtain good predictors for test execution, with improved guarantees over traditional statistical learning methods. We show how this can be applied in imitation learning, structured prediction and MBRL. Experimental results demonstrate the efficacy and wide applicability of this technique in a number of applications. Our results demonstrate that sequential prediction tasks fundamentally require interaction between the process and learner to achieve good prediction performance.

Imitation Learning: In [2], we showed how repeated rounds of interaction can lead to good performance guarantees in imitation learning problems. Our approach called DAgger, for Dataset Aggregation, is simple and consists in training the predictor over several iterations. First, an expert performs the task and we train a predictor to best mimic the expert on this data. At following iterations, the learner and expert interact as follows: the learner performs the task using the last trained predictor and ask the expert to provide the action he would perform at each visited state. The visited states with the expert's actions as target are aggregated with all previously collected data to train the next predictor. Via this interaction, the expert can demonstrate the correct behavior where the learner spends time to improve its performance (e.g. how to recover from previous mistakes). This simple data aggregation strategy can be interpreted as a Follow-the-(Regularized)-Leader algorithm on a particular online learning problem. By exploiting its no-regret property, this approach has good guarantees: if there exists a predictor which makes ϵ average prediction loss on the aggregate dataset, then after enough iterations, DAgger must find a predictor which incurs $T\epsilon$ prediction loss in expectation over sequences of T steps during test executions. Additionally, any other no-regret algorithm (e.g. gradient descent) can be used to generate the sequence of predictors over the iterations to provide the same guarantee. In contrast, the no interaction statistical learning approach that trains only from expert's demonstrations guarantees no better than $T^2\epsilon$ expected loss over T-steps in general [2] (due to mismatch in train/test distributions). We demonstrated the scalability and efficacy of this

approach in two electronic games: learning to drive a car in a 3D racing game, and learning to play Super Mario Bros [2]. Our results show significant improvement over previous approaches.

Structured Prediction: In [2] and [3], we have also showed that predictors for structured prediction problems can be learned with the exact same DAgger algorithm. By adopting a sequence prediction view of structured prediction as in SEARN [1], where the ouput is constructed from a sequence of interdependent predictions, then structured prediction is essentially reduced to an imitation learning problem, where we seek to learn a predictor that mimics an ideal predictor that always makes correct predictions (as defined by the training data). For example, in a handwriting recognition task where we need to decode handwriten words, we might seek to learn a predictor which predicts each character in sequence, using the character's image as input as well as the previously predicted character to give additional contextual information. In this scenario, DAgger first generates a training dataset by using the ideal predictor to make the sequence of predictions on each training word (this is analog to the expert demonstrations in the imitation learning setting). In this dataset, the inputs always contain correct predictions of the previous character. At following iterations, the current predictor is used on the training words to generate a new dataset. During these interactions, the input image and target character are still the same, but the previously predicted character in the input features might be incorrect. As DAgger trains the next predictor on the aggregate dataset, it learns predictors that are more robust to the kind of mistakes they make during the decoding process. The same guarantees hold in this structured prediction setting. Experimental results on an handwriting recognition task [2], as well as vision tasks such as 3D point cloud classifition and 3D surface layout estimation from 2D images [3], demonstate that our approach competes with state-of-the-art structured prediction methods such as SEARN and graphical model based approaches.

Model-Based Reinforcement Learning: Recently, we have extended this approach to MBRL. Here, we seek to learn a model of a system that can predict the next state given a current state and action, which is used for finding an optimal policy. A similar mismatch between train/test distribution arises: to collect state transitions, we must run an "exploration" policy, but at test time, the policy optimized on the learned model may visit states/actions that were never or rarely explored by the exploration policy, where the model poorly captures system behavior and lead to poor control performance. A similar iterative and interactive procedure can be used to address this issue. In this case, DAgger iterates between: 1) collecting state transitions by running the current policy, as well as by running an exploration policy, 2) fitting a new model on the aggregate dataset of all data collected so far, and 3) planning with the new model to obtain our next policy. Alternatively, any no-regret algorithm can be used to update the sequence of models picked over the iterations of the algorithm. When collecting data, we must collect an equal amount of exploration data and data from interactions with the current policy. The exploration data ensures we can predict well the cost of other policies, while data from running the current policy ensures we can predict well the cost of the policy we execute. This approach guarantees good test execution performance compared to any policy whose state-action distribution is close to that of the exploration policy, as long as a good model of the aggregate dataset exists. In contrast, a naive statistical learning approach that only collects exploration data could lead to a policy that performs arbitrarily badly (e.g. if the learned policy goes to states that are never explored). The exploration policy should ideally be a near-optimal policy, i.e. spend much of its time exploring where good policies go, to ensure near-optimal performance. This can often be obtained in practice, e.g. from an expert who can demonstrate the task. Our approach as a number of desirable properties. First, it is agnostic, i.e. it provides good guarantees even if the real system is not in the class of model considered (as long as a model with low training loss exists). In contrast, existing MBRL methods can only provide guarantees when the real system belongs to the model class. Additionally, DAgger's sample complexity scale only with the complexity of the model class, and not the size of the MDP (number of states and actions). Experimental results at learning to perform aerobatic maneuvers with a simulated helicopter indicate our method outperforms the naive statistical learning approach as well as other methods.

References

[1] H. Daumé III, J. Langford, and D. Marcu. Search-based structured prediction. *Machine Learning*, 2009.

[2] S. Ross, G. Gordon, and J. A. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, 2011.

[3] S. Ross, D. Munoz, M. Hebert, and J. A. Bagnell. Learning message-passing inference machines for structured prediction. In *CVPR*, 2011.