A BRIEF SURVEY OF MODEL BASED METHODS FOR OFFLINE REINFORCEMENT LEARNING.

Aayam Shrestha
Oregon State University
Corvallis, OR 97330, USA
{shrestaa}@oregonstate.edu

ABSTRACT

Model based reinforcement learning involves two distinct steps: model learning and planning. These approaches are promising for offline Reinforcement learning as model learning benefits from convenient and powerful supervised learning methods. Thus, in this survey, we cover model based methods that have been applied in offline reinforcement learning. We study both parametric and non-parametric methods and highlight the strengths and weaknesses of both class of approaches. We further identify standard and conservative sub-classes of each approach based on how they account for out of distribution data points during evaluation. Although less explored, we find non-parametric models promising for offline reinforcement learning as they can leverage the generalization capabilities of deep representations as well as flexibility afforded by optimal planning.

1 INTRODUCTION

Model-Based Reinforcement Learning (MBRL) combines learning and planning under the same framework where we learn a model of the world and in turn use it for planning. They have shown great success over the recent years in both online and offline reinforcement learning (Schrittwieser et al. (2019), Kaiser et al. (2020)) for data efficiency and support for more general planning objectives (for example robustness, different goals) . They are specially promising for offline reinforcement learning as it does not require any online data collection and can benefit from strong supervised learning methods to learn the model of the world. However, there exists a number of way to learn the environment dynamics and each of these models are compatible with specific set of planners such as search based or optimal planners. Moreover, there has not been many works to study the different components of model based RL for offline settings (Levine et al., 2020) or that contrasts between the strengths and weaknesses of these different approaches.

There primarily exists Parametric and Non-Parametric approaches to learn the model of the environment. Parametric models summarize the data with a fixed set of parameters (independent of the number of training examples, e.g., linear function approximation). On the other hand, Non-Parametric models do not make any assumptions regarding the form of the mapping function and are free to learn any functional form, from the training data (Russell & Norvig, 1995). [1] Both non-parametric and parametric methods have their own strengths and weaknesses. Parametric methods are simpler, faster to train and require less data, however, they are also constrained by the choice of functional form, and unlikely to match the underlying mapping function. On the other hand, non-parametric methods are more flexible for a more diverse set of functional forms, and can result in high performant models. However, they require more data, are slower to train, and may overfit to the training data. More specific to modeling the environment, Non-parametric models are fast to adapt with new data-points and more easily allow local changes to the model. Moreover, Non-parametric models have a finite structure that can be solved optimally, which opens doors for integrating deep representation learning and or optimal/symbolic planning.

[1] The literature generally classifies non-parametric methods for value estimation as approximate dynamic programming. However, it is more natural to view these as model based reinforcement learning methods with non-parametric models for our discussion. Hence we will classify these methods under the model based reinforcement learning setting itself.

Both non-parametric and parametric methods have shown to be successful for offline reinforcement learning and cover a range of techniques to tackle the problem of distributional shift;(Fujimoto et al., 2019) which is a central problem for offline RL. In general, they tackle this by building model priors that may not overfit (aka standard approach), or by explicitly penalizing or the uncertainty in the model (aka conservative approach). The model priors can come in the form of added noise or explicit function classes, whereas the measure of uncertainty is extracted from ensembles or distance from the dataset/behavior policy. Different planning methods can then leverage the learned model. A host of planning methods has been explored, including but not limited to model-free RL, heuristic search based planning, model predictive controllers, and more recently, optimal planning. Search based /short-horizon planning can produce good policies even from an inaccurate model at the cost of optimality guarantees ((Hamrick et al., 2020),Schrittwieser et al. (2019)). In contrast, optimal planners are known to exploit the inaccuracies in the model in a way that hurts the performance(cumulative return) of the final policy (Atkeson, 1998). However, some of the recent non-parametric methods such as DAC-MDPs(Shrestha et al., 2020) allow the integration of optimal planning over other methods.

Here, we first discuss non-parametric methods for Model-Based Offline Reinforcement Learning (Section 2) and further investigate representation learning methods along with approaches for bounding space requirements(Section 3). We then cover both standard and conservative parametric model based approaches for offline RL (Section 3) and conclude by outlining the challenges and open problems in the area (Section 4). [2]

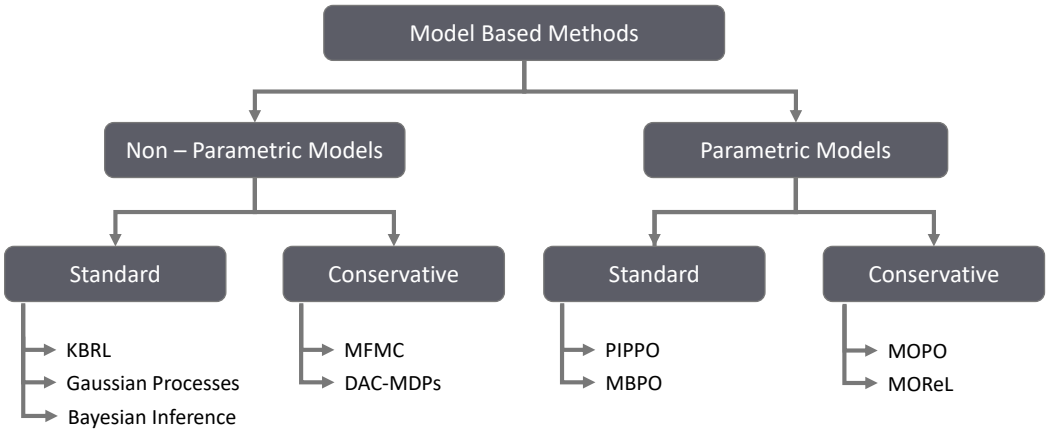

Figure 1: Classification of model based approaches for Offline Reinforcement learning. Model based methods be first classified into two subclasses, (1) which employ either non-parametric methods or (2) parametric methods to learn the model of the environment. These approaches can be further divided into standard and conservative methods based on how they tackle distributional shift.

2 Non-parametric methods for Offline MBRL

The simplest form of models is tabular models with one entry for each state-action pair, which for small and finite state and action spaces. Moreover, its approximation (in batch and in online mode), as well as the control policy, can be readily derived. However, when dealing with continuous or very large discrete state and/or action spaces, the model or the Q-function cannot be represented anymore by a table due to memory and computational requirements. To overcome this generalization problem Gordon & Mitchell (1999) introduced fitted Q iteration framework, which allows it to fit any

[2]We also note that Experience replay(Lin, 1992) may also be used for planning; which can be queried for any state-action pairs that we have observed. We refer readers to Hasselt et al. (2019) for further discussions and comparison with parametric models.

(parametric or non-parametric) approximation architecture to the Q-function. Most non-parametric MBRL approaches extend this framework by using non-parametric approximations such as kernel regression and gaussian processes along with pessimistic approximation.

2.1 STANDARD NON-PARAMETRIC METHODS

Kernel Based Approaches: Ormoneit & Sen (2002) and Ormoneit & Glynn (2002) applies the idea of fitted value iteration (Gordon & Mitchell, 1999) to kernel based reinforcement learning, and reformulates the Q-function determination problem as a sequence of kernel-based regression problems. Hence these approaches can take full advantage of the generalization capabilities of any regression algorithm, contrary to stochastic approximation algorithms (Sutton (2005) ; Tsitsiklis (2004)) that can only use parametric function approximators such as neural networks.

Moreover, Ormoneit & Sen (2002) also shows that the approximate bellman backup operator converges to a unique solution and is asymptotically close to the optimal policy. But it is to be noted that the bias of such estimates are shown to be typically larger in reinforcement learning than in a comparable regression problem due to propagation of errors. (Ormoneit & Glynn, 2002) further shows that KBRL approaches can also be seen as the derivation of a finite Markov decision process whose number of states coincides with the number of sample transitions collected to perform the approximation. They focused on the theoretical results for average-cost MDPs, whereas Ormoneit & Sen (2002) expanded the same for discounted reward MDPs.

Extensions to KBRL: We can also define different variants of Q learning based on local function approximation methods. Szepesvari & Smart (2004) considers Interpolation-based Q-learning with local function approximation methods that are mostly similar to Ormoneit & Sen (2002)(KBRL); however, it is defined for online setting in contrast to offline setting in KBRL. Also, it converges to different limits as compared to KBRL for a fixed dataset however, both algorithms are proven to converge to optimal Q functions for an infinite dataset.

The KBRL approach has also been combined with exploration strategies in an online setting to derive a complete learning algorithm. Jong & Stone (2006), Jong & Stone (2007) and Shah & Xie (2018) all extend KBRL to online settings. More specifically, Jong & Stone (2007) introduced Approximate Models Based on Instances (AMBI), which combines the strengths of KBRL with explicit exploration where they use prioritized sweeping to solve for an RMaX style exploration that eventually converges to the optimal policy.

Other non-parametric approaches: While Ormoneit & Sen (2002) studied the theoretical convergence and consistency properties for kernel-based regressors, Ernst et al. (2005) studied the empirical properties and performances of several tree-based regression algorithms across different applications. Just like kernel-based methods, tree-based methods are also non-parametric and offer great modeling flexibility. From a practical standpoint, tree-based methods are more computationally efficient as they do not require nearest neighbor computations and hence can readily scale to high-dimensional spaces. Moreover, the ensemble methods make them more robust to irrelevant variables, outliers, and noise.

The use of Bayesian models and Gaussian processes have also been explored for non-parametric methods. Ferreira et al. (2018) introduced Bayesian non-parametric representations, such as Infinite-Mixture Models known as Dirichlet Processes, and uses relative entropy policy search for value optimization. Chowdhary et al. (2014) use Gaussian Process models (Kocijan et al., 2004) along with Bayesian optimization to approximate the value function. Gaussian process dynamic programming (GPDP) Deisenroth et al. (2008) is a generalization of dynamic programming/value iteration for continuous state and action spaces which uses probabilistic Gaussian process models. Kveton & Theocharous (2013) models the in-dependencies in the transition and reward models of the environment. In other words, additional training data always improve the quality of the estimates and eventually leads to optimal performance.

There have also been works that extend this framework for different objectives, a detailed study of which is outside the scope of this paper. Notably, Lim & Autef (2019) extends the robust MDP framework Nilim & Ghaoui (2005) for state aggregation to kernel based approximators for robustness objectives. Also, Gu et al. (2020) uses kernel based regressors for multi-agent settings.

2.2 CONSERVATIVE NON-PARAMETRIC METHODS

Model based approaches can use effective planners in order to solve for a globally optimal policy. However, the planners exploit the inaccuracies in the learned model in a way that adversely affects the performance of the learned policy. This is because the learned policy may be evaluated in a space where the model extrapolation is poor. Hence, this can also be viewed as the classic problem of the distributional shift in machine learning. This arises because the learned policy is mostly accurate around the training dataset and may not generalize over all the state space while being evaluated in these uncertain regions where the learned model is inaccurate.

Several approaches have been proposed to mitigate this distributional shift, which amounts to quantifying the uncertainty in the model and taking this into account during planning. One way to mitigate this is by adding a controlled amount of noise in the model learning step, which works as a regularization to prevent over-fitting, as demonstrated by (Atkeson, 1998). Moreover, we can also penalize for the model's uncertainty directly by adding costs for any approximated transitions: the more uncertain it is, the greater the penalty. This uncertainty measure can be approximated by the distance between any query state action pair and seen state-action pair (that is present in the training dataset). This was first demonstrated by MFMC (Fonteneau et al., 2013), which uses the distances to introduce costs on the sampled imaginary transitions. Fonteneau et al. (2013) studied a "trajectory-based" simulation model for offline policy evaluation. Similar in concept to our work, pessimistic costs were used based on transition distances to "piece together" disjoint observations, which allowed for theoretical lower-bounds on value estimates.

This idea of penalizing transitions was further explored by DAC-MDPs(Shrestha et al., 2020), which motivates the distance function by Lipschitz continuity assumptions on the Q function of the underlying MDP. DAC-MDPs also show that the value function from an optimally solved policy for such cost-modified MDPs lower-bounds the optimal value function. Moreover, they combine this with deep representation learning, which is smooth in Q function and shows scalability to large complex domains such as Atari. The representation learning combined with euclidean distance can be viewed as a pseudo-metric learning step as done in Zewdie & Konidaris (2015).

3 PRACTICAL ASPECTS OF NON-PARAMETRIC METHODS

Non-parametric methods provide a very general fast adapting solutions with theoretical properties, however, they also come with their own set of practical issues. These issues range from memory/computational concerns to the implicit assumptions of the underlying function being modelled. Here we explore works that try to alleviate these issues and also highlight connections between simple data graphs and the derived MDPs from non-parametric methods.

3.1 BOUNDING SPACE REQUIREMENTS:

Unfortunately, the good generalization abilities of Non-parametric methods such as KBRL come at a price. Since the model constructed by this algorithm grows with the number of sample transitions, the cost of computing a decision policy quickly becomes prohibitive as more data become available. Such a computational burden severely limits the applicability of KBRL. This may help explain why, despite its nice theoretical guarantees, kernel-based learning has not been widely adopted as a practical reinforcement learning tool.

In other words, the key challenge of conventional non-parametric kernel based methods is that the learner still has to maintain a set of support vectors (SV's) in memory for representing the kernel-based predictive model. The number of SV's can be, however, bounded in a number of ways. *SV Removal:* Cavallanti et al. (2007), Dekel et al. (2008), Wang & Vucetic (2010)(BPA-S) all attempt to bound the number of SV's by removing samples from existing SV pool following some heuristic. (for example, random removal, oldest first, most redundant first). *SV Projection:* Orabona et al. (2008), Wang & Vucetic (2010)(BPA-P, BPA-NN) project the discarded SV's onto the remaining ones. *SV Merging:* Wang & Vucetic (2009) , Wang et al. (2012) attempt to maintain the budget by merging two existing SV's into a new one. *kernel approximation*: Lu et al. (2016) Koppel et al. (2017) employs kernel functional approximation techniques to resolve the budget constraints. We can also take a clustering approach to decrease the number of data points/support vectors with finite-state approximations as explored by Chow & Tsitsiklis (1989) and Rust (1994).

3.2 REPRESENTATION LEARNING IN KERNEL BASED METHODS:

There are now effective and scalable Kernel-based methods (Barreto et al. (2016), Shrestha et al. (2020)) that can be used to approximate the value functions. However, it is important to note that kernel regression requires that the underlying function that is being modeled to be smooth on its domain. Since this is not satisfied by many natural representations such as image-based domains, it becomes crucial to add a representation learning step to kernel based approaches. This representation learning can further leverage the deep representation learning techniques and has been proven to work well in works like DAC-MDPs. There have also been other approaches towards learning better representations such as Zewdie & Konidaris (2015), Zewdie & Kaelbling (2014), which defines a value-consistent pseudometric (VCPM). [3] Here the pseudometric $d_f^*(x, x') = |f(x) - f(x')|$ is the value-consistent pseudometric (VCPM) for f on X. Moreover, they also define DKBRL, an iterative batch RL algorithm interleaving steps of Kernel-Based Reinforcement Learning, and distance metric adjustment. The approach interleaves value estimation and representation adjustment steps to increase the expressive power of a given regression scheme. This creates representations that correlate highly with value, giving kernel regression the power to represent discontinuous functions. While Shrestha et al. (2020) show the potential of learning better representations, they do not fully optimize the representations to be value consistent. Rather they use the penultimate layer for Q iterative approaches or imitation learning networks for the representation.

3.3 CONNECTIONS TO EPISODIC CONTROL AND TRANSITION GRAPHS

Several prior approaches for online RL, sometimes called episodic control, construct different types of explicit transition graphs from data that are used to drive model-free RL (Blundell et al., 2016; Hansen, 2017; Pritzel et al., 2017; Lin et al., 2018; Zhu et al., 2020; Marklund et al., 2020). These transition graphs are very similar to non-parametric MDPs with the difference that it does not consider and state action pair that has not been seen in the dataset. This makes the value estimation of these episodic graphs very compute efficient that can be done by simple KNN look-ups without any bellman backups. Moreover, they use KNN approximations to find the policy for any state that is not contained in the graph. Similar to non-parametric methods, they also use different representations for indexing the graph structure. For example, Blundell et al. (2016) shows that random projections can be quite effective for this purpose, and Pritzel et al. (2017) adapts the representation using a differentiable memory. Works like Zhu et al. (2020) propose a modification to the tabular memory where associative connections are added for value propagation when state representations match precisely. Moreover, works like Pritzel et al. (2017) can be directly seen as a Q iteration approach with KNN regressor and learned non-linear representations.

Finally, we also note that the DAC-MDP formulation proposed by Shrestha et al. (2020) closely resembles the data-graphs from episodic control. If we assume that the representations are held fixed or converged, episodic control can be emulated by a DAC-MDPs with $k_b = 1$ and $C = Q_{max}$. Then, k_l sets the smoothing factor used in episodic control. This, in turn, implies that our theoretical results transfer directly to the episodic control literature. Hence, episodic control can also be seen as the most compute efficient version of DAC-MDPs where the resulting MDP can be solved using episodic backups without any iterations.

4 PARAMETRIC METHODS FOR OFFLINE MBRL

Parametric methods has been relatively more popular for offline model-based reinforcement learning algorithms and simply train the model from the offline data, with minimal modification to its online setting. While model based approaches can be natively applied in offline settings, they also benefit from employing conservative estimates so as to avoid model exploitation that hurts the performance in the real world. Model based methods such as MBPO (Janner et al., 2019) performs relatively better than standard offline RL methods without modification such as soft actor-critic (Haarnoja et al., 2018). Here we briefly explore both standard and conservative model based approaches for offline RL.

[3] A pseudometric is the distance function corresponding to a transformation of the domain into a space where the target function is maximally smooth and thus well-approximated by kernel regression.

4.1 STANDARD PARAMETRIC METHODS

Parametric models have been used to train predictive models and applied for control in complex and high-dimensional domains, including image observations. These predictive models can then be used for both online and offline trajectory optimization. Offline policy optimization includes offline planning or any model free methods, whereas online planning comprises Model Predictive Controllers and its variants. In its simplest form, raw observation models such as action conditional convolutional neural networks (Oh et al., 2015) have been combined with RL by Kaiser et al. (2020). These models can be used with stacked images or a sequence of frames with a recurrent neural network. Moreover, they have been shown to work with data collected by uninformative randomized policy (Finn & Levine (2017), Ebert et al. (2018)), and large, diverse datasets from multiple agents and viewpoints (Dasari et al. (2019)).

One can also assign prior to the learned models such that they do not overfit to the data. Lutter et al. (2020) uses physics based models that are better at extrapolating due to the general validity of their informed structure and does not overfit the model due to inherent noise. Nair et al. (2020) combines sample-efficient dynamic programming with maximum likelihood policy updates that allow them to leverage large amounts of offline data and then quickly perform online fine-tuning of reinforcement learning policies. On the other hand, if expert behavior is available, it can also be directly leveraged to form a prior for the model. Lambert et al. (2020) proposes reweighting the samples such that the samples around the expert behavior is weighted higher; however, this might be hard to do when no expert/optimal data is available/labeled.

But, none of the previously discussed approaches leverage optimal planning. While works like (Corneil et al., 2018) employ an optimal planner, it is important to note that the planner does not use the forward dynamics model; instead, the planner uses a discrete MDP that is extracted from the dataset by simply leveraging the representations. Here, they use autoencoders with Gumbel softmax discretization to learn discrete latent representations. More recently, van der Pol et al. (2020) introduced a contrastive representation-learning and model-learning approach that uses the forward dynamics model when constructing the tabular MDP.(solved using value iteration). Here, they train a deterministic parametric model and use the predicted transitions and nearest-neighbor computations to create a stochastic MDP. Experimental results in both of these works were limited to a small number of domains and small datasets. Along the same line, works like Kurutach et al. (2018) is mostly focused on learning plannable representations than actual planning. Here, they train a GAN to generate a series of waypoints, which is then be leveraged by simple feedback controllers. Also, Yang et al. (2020), and Agarwal et al. (2020) attempt to incorporate long term relationships into their learned latent space and employ search algorithms such as A* and elliptical planners.

Some approaches perform strongly in both online and offline settings. For example, PIPPO (Sun et al., 2019) is an off-policy model-based method introduce for online settings that can also work well for offline learning, whereas Kim (2020) combines offline and online RL by balancing offline Monte Carlo and Online Temporal different learning.

4.2 CONSERVATIVE PARAMETRIC METHODS:

While noise and ensemble in learned models result in more robust performance for offline settings, they do not directly employ conservative estimates in the learned transitions. More recently, pessimistic MDPs have been proposed for offline model-based RL such as MOPO (Yu et al., 2020) and MORel (Kidambi et al., 2020). Both the approaches define surrogate MDPs that penalizes for model uncertainty and approximate the assumed "uncertainty oracle" signal by using an ensemble of learned models. They also derive performance bounds under the assumption of optimal planning. In practice, however, due to the difficulty of planning with learned deep-network models, the implementations rely on model-free RL, which introduces an extra level of approximation.

However, these approaches do not use the models for direct planning and hence cannot leverage the final policy to adapt to an arbitrary goal or a constrained objective. Argenson & Dulac-Arnold (2020) leverages the model for trajectory optimization that can consider different goals/constraints and create zero-shot goal-conditioned policies on a series of environments. Matsushima et al. (2020) introduces Behavior-Regularized Model-Ensemble (BREMEN) that learns an ensemble of dynamics

models and uses the imaginary rollouts to learn a policy. It also regularizes the learned policy via parameter initialization as well as conservative trust-region learning updates.

Works like Ma et al. (2019) and Zhou et al. (2020) propose an uncertainty-aware policy optimization algorithm that takes into account the policy's data coverage. Zhou et al. (2020) employs a policy improvement (PIL) objective that is regularized by policy imitation (IML). On the other hand, Ma et al. (2019) optimizes the policy conservatively to encourage performance improvement with high probability.

5 Benefits, Challenges and Open Problems

Benefits: Albeit less explored, Non-parametric methods offer many benefits such as fast adaptation to new data points as they require less training. Moreover, it allows one to leverage optimal planners even for the continuous domains where the number of states is unbounded. It can further be combined with deep representation learning for effective pseudo-metrics for the corresponding objective we are trying to optimize. While they have been shown to scale, it is still limited for very large dataset sizes and online settings with constantly shifting internal representation. While different non-parametric approaches do exist, kernel based approaches for RL, when combined with deep representation learning and optimal planning, produce strong performances for offline settings.

Parametric methods, on the other hand, are scalable as their size does not scale with the dataset. While there is no straightforward notion of distance of out of distribution inputs, modeling uncertainty of the model with ensembles or limiting the learned model to a specific function class has been shown to work well for handling the distribution shift in offline RL settings. As predictions in parametric methods are very efficient, they can also be readily used as a simulator for search-based planning such as Monte Carlo Tree search or Model predictive controllers.

Challenges and open problems: While deep representation learning has been integrated with KBRL approaches, they are not yet tuned for the exact objectives/underlying assumptions such as Lipschitz continuity. Learning representations to optimize for this smoothness constraint directly in term of distance as done by Zhang et al. (2020) looks promising. It is also evident that the KNN overhead can be significantly reduced by the use of GPU acceleration, the combination of which is still to be explored. Moreover, the benefits that have been previously demonstrated such as exploration and robustness are poised to be explored with modern benchmarks and or scalable approaches employing better representations.

Shifting strong non-parametric offline methods to online settings is another challenge, as it is costly and difficult to bound the space requirements for ever-increasing data buffer. While we draw a corollary between "thinking fast and slow"(Kahneman, 2011) and the solutions for MDPs with different constraints, this area still lacks theoretical work to ground this more formally.

Parametric approaches, while being scalable are not very fast to adapt to different goals and hard to integrate with optimal planners. Many conservative approaches to parametric models rely upon an ensemble on observation space and uncertainty oracles. It is not clear how to use ensembles for latent space models and further reduce the reliance on uncertainty oracles.

In general, for both parametric and non-parametric approaches, it is not clear how to identify the regions that the model extrapolated well and where it did not. It is appealing to be able to find trust regions of the model while limiting extrapolation in others by constraining oneself to the seen dataset in others. An approach that accomplishes the integration of parametric models with trust regions and non-parametric models for fast adaptation would be very exciting. Also, the use of optimal planners such as exact DP and symbolic planners are still less explored for model based RL and look promising for zero-shot tranfer learning to different goals and planning objectives.

Finally, it is worth mentioning that works that specifically deal with generating datasets with specific properties for benchmarking has been explored by like Gulcehre et al. (2020) and Fu et al. (2020). However, there is still room for a large-scale offline RL dataset for challenging tasks such as self-driving and medical diagnosis.

6 SUMMARY

In this work, we explore both parametric and non-parametric methods for offline model based reinforcement learning with a focus on non-parametric methods. We also discuss both conservative and standard variants of both class of approaches. In general, we find a significant amount of work in non-parametric methods in the past and has also been explored recently albeit much less than parametric methods. Non-parametric methods are very flexible, fast to train and adapt, and be integrated with deep representation learning and optimal planning. On the other hand, parametric methods are very scalable and can also be integrated with deep representation learning and approximate planning.

We identify two subclasses: standard and conservative, for each class of approaches. While standard methods do tend to perform well in high data regimes, conservative estimates seem important for low/ diverse data regimes. Moreover, defining uncertainty is more straightforward for non-parametric models as they are already grounded in the dataset. Finally, we discussed the challenges and open problems in this area. We find the scalability of non-parametric approaches with tailored deep representations to be an interesting area for immediate future work.

REFERENCES

A. Agarwal, Sham M. Kakade, A. Krishnamurthy, and W. Sun. Flambe: Structural complexity and representation learning of low rank mdps. *ArXiv*, abs/2006.10814, 2020.

Arthur Argenson and Gabriel Dulac-Arnold. Model-based offline planning. *ArXiv*, abs/2008.05556, 2020.

Christopher G Atkeson. Nonparametric model-based reinforcement learning. In *Advances in neural information processing systems*, pp. 1008–1014, 1998.

A. Barreto, Doina Precup, and Joelle Pineau. Practical kernel-based reinforcement learning. *J. Mach. Learn. Res.*, 17:67:1–67:70, 2016.

Charles Blundell, B. Uria, A. Pritzel, Y. Li, Avraham Ruderman, Joel Z. Leibo, Jack W. Rae, Daan Wierstra, and Demis Hassabis. Model-free episodic control. *ArXiv*, abs/1606.04460, 2016.

Giovanni Cavallanti, Nicolò Cesa-Bianchi, and C. Gentile. Tracking the best hyperplane with a simple budget perceptron. *Machine Learning*, 69:143–167, 2007.

Chef-Seng Chow and J. Tsitsiklis. The complexity of dynamic programming. *J. Complex.*, 5:466–488, 1989.

G. Chowdhary, M. Liu, Robert C. Grande, T. Walsh, J. How, and L. Carin. Off-policy reinforcement learning with gaussian processes. *IEEE/CAA Journal of Automatica Sinica*, 1:227–238, 2014.

Dane S. Corneil, W. Gerstner, and J. Brea. Efficient model-based deep reinforcement learning with variational state tabulation. *ArXiv*, abs/1802.04325, 2018.

Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Surender Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *Third Conference on Robot Learning*, abs/1910.11215, 2019.

M. P. Deisenroth, J. Peters, and C. E. Rasmussen. Approximate dynamic programming with gaussian processes. *2008 American Control Conference*, pp. 4480–4485, 2008.

O. Dekel, S. Shalev-Shwartz, and Y. Singer. The forgetron: A kernel-based perceptron on a budget. *SIAM J. Comput.*, 37:1342–1372, 2008.

Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex X. Lee, and S. Levine. Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. *CORR*, abs/1812.00568, 2018.

D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. *J. Mach. Learn. Res.*, 6:503–556, 2005.

Ana Carolina Borg Ferreira, Jan Peters, U. Konigorski, Tag der Einreichung, and Erklärung zur Bachelor-Thesis. Infinite-mixture policies in reinforcement learning. In *Thesis Technishe universitat darmstadt*, 2018.

Chelsea Finn and S. Levine. Deep visual foresight for planning robot motion. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2786–2793, 2017.

R. Fonteneau, S. Murphy, L. Wehenkel, and D. Ernst. Batch mode reinforcement learning based on the synthesis of artificial trajectories. *Annals of Operations Research*, 208:383–416, 2013.

Justin Fu, Aviral Kumar, Ofir Nachum, G. Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *ArXiv*, abs/2004.07219, 2020.

Scott Fujimoto, Edoardo Conti, Mohammad Ghavamzadeh, and Joelle Pineau. Benchmarking batch deep reinforcement learning algorithms. *ArXiv*, abs/1910.01708, 2019.

G. Gordon and Tom Michael Mitchell. Approximate solutions to markov decision processes. In *Ph.D. Thesis, MIT*, 1999.

H. Gu, X. Guo, Xiaoli Wei, and Renyuan Xu. Mean-field controls with q-learning for cooperative marl: Convergence and complexity analysis. *arXiv: Learning*, 2020.

Caglar Gulcehre, Ziyu Wang, A. Novikov, T. L. Paine, Sergio Gomez Colmenarejo, Konrad Zolna, Rishabh Agarwal, Josh Merel, Daniel J. Mankowitz, Cosmin Paduraru, Gabriel Dulac-Arnold, J. Li, Mohammad Norouzi, Matt Hoffman, Ofir Nachum, G. Tucker, Nicolas Heess, and N. D. Freitas. Rl unplugged: Benchmarks for offline reinforcement learning. *ArXiv*, abs/2006.13888, 2020.

T. Haarnoja, Aurick Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018.

Jessica B. Hamrick, Abram L. Friesen, Feryal M. P. Behbahani, A. Guez, F. Viola, Sims Witherspoon, T. Anthony, Lars Buesing, Petar Velickovic, and T. Weber. On the role of planning in model-based deep reinforcement learning. *ArXiv*, abs/2011.04021, 2020.

Steven Stenberg Hansen. Deep episodic value iteration for model-based meta-reinforcement learning. *ArXiv*, abs/1705.03562, 2017.

H. V. Hasselt, Matteo Hessel, and J. Aslanides. When to use parametric models in reinforcement learning? *ArXiv*, abs/1906.05243, 2019.

Michael Janner, Justin Fu, Marvin Zhang, and S. Levine. When to trust your model: Model-based policy optimization. *NeurIPS 2019*, abs/1906.08253, 2019.

Nicholas K. Jong and P. Stone. Kernel-based models for reinforcement learning. In *Kernel machines for reinforcement learning workshop*, 2006.

Nicholas K. Jong and Peter Stone. Model-based function approximation in reinforcement learning. In *AAMAS '07*, 2007.

D. Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011.

L. Kaiser, M. Babaeizadeh, P. Milos, B. Osinski, R. Campbell, K. Czechowski, D. Erhan, Chelsea Finn, Piotr Kozakowski, S. Levine, Ryan Sepassi, G. Tucker, and H. Michalewski. Model-based reinforcement learning for atari. *ArXiv*, abs/1903.00374, 2020.

R. Kidambi, A. Rajeswaran, Praneeth Netrapalli, and T. Joachims. Morel : Model-based offline reinforcement learning. *ArXiv*, abs/2005.05951, 2020.

Chayoung Kim. Deep reinforcement learning by balancing offline monte carlo and online temporal difference use based on environment experiences. *Symmetry*, 12:1685, 2020.

J. Kocijan, Roderick Murray-Smith, C. E. Rasmussen, and A. Girard. Gaussian process model based predictive control. *Proceedings of the 2004 American Control Conference*, 3:2214–2219 vol.3, 2004.

Alec Koppel, Garrett Warnell, E. Stump, and Alejandro Ribeiro. Parsimonious online learning with kernels via sparse projections in function space. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4671–4675, 2017.

Thanard Kurutach, A. Tamar, Ge Yang, S. Russell, and P. Abbeel. Learning plannable representations with causal infogan. In *NeurIPS*, 2018.

Branislav Kveton and Georgios Theocharous. Structured kernel-based reinforcement learning. In *AAAI*, 2013.

Nathan G. Lambert, Brandon Amos, Omry Yadan, and R. Calandra. Objective mismatch in model-based reinforcement learning. *ArXiv*, abs/2002.04523, 2020.

Sergey Levine, Aviral Kumar, G. Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *ArXiv*, abs/2005.01643, 2020.

Shiau Hong Lim and Arnaud Autef. Kernel-based reinforcement learning in robust markov decision processes. In *ICML*, 2019.

Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8(3-4):293–321, 1992.

Zichuan Lin, Tianqi Zhao, G. Yang, and L. Zhang. Episodic memory deep q-networks. *ArXiv*, abs/1805.07603, 2018.

J. Lu, S. Hoi, Jialei Wang, P. Zhao, and Z. Liu. Large scale online kernel learning. *J. Mach. Learn. Res.*, 17:47:1–47:43, 2016.

Michael Lutter, J. Silberbauer, J. Watson, and Jan Peters. Differentiable physics models for real-world offline model-based reinforcement learning. *ArXiv*, abs/2011.01734, 2020.

Yifei Ma, Yu-Xiang Wang, and Balakrishnan Narayanaswamy. Imitation-regularized offline learning. In *AISTATS*, 2019.

Henrik Marklund, Suraj Nair, and Chelsea Finn. Exact (then approximate) dynamic programming for deep reinforcement learning. In *Bias and Invariances Workshop, ICML*, 2020.

Tatsuya Matsushima, Hiroki Furuta, Y. Matsuo, Ofir Nachum, and Shixiang Gu. Deployment-efficient reinforcement learning via model-based offline optimization. *ArXiv*, abs/2006.03647, 2020.

Ashvin Nair, Murtaza Dalal, Abhishek Gupta, and Sergey Levine. Accelerating online reinforcement learning with offline datasets. *ArXiv*, abs/2006.09359, 2020.

A. Nilim and L. Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Oper. Res.*, 53:780–798, 2005.

Junhyuk Oh, Xiaoxiao Guo, H. Lee, R. L. Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. In *NeurIPS*, 2015.

F. Orabona, Joseph Keshet, and B. Caputo. The projectron: a bounded kernel-based perceptron. In *ICML '08*, 2008.

Dirk Ormoneit and P. Glynn. Kernel-based reinforcement learning in average-cost problems. *IEEE Trans. Autom. Control.*, 47:1624–1636, 2002.

Dirk Ormoneit and Ś. Sen. Kernel-based reinforcement learning. In *Machine Learning*, 2002.

A. Pritzel, B. Uria, S. Srinivasan, Adrià Puigdomènech Badia, Oriol Vinyals, Demis Hassabis, Daan Wierstra, and Charles Blundell. Neural episodic control. In *ICML*, 2017.

S. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Oxford University Press, 1995.

J. Rust. Using randomization to break the curse of dimensionality. *Econometrica*, 65:487–516, 1994.

Julian Schrittwieser, Ioannis Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, Edward Lockhart, Demis Hassabis, T. Graepel, T. Lillicrap, and D. Silver. Mastering atari, go, chess and shogi by planning with a learned model. *ArXiv*, abs/1911.08265, 2019.

D. Shah and Q. Xie. Q-learning with nearest neighbors. *ArXiv*, abs/1802.03900, 2018.

Aayam Shrestha, Stefan Lee, P. Tadepalli, and A. Fern. Deepaveragers: Offline reinforcement learning by solving derived non-parametric mdps. *ArXiv*, abs/2010.08891, 2020.

Yuewen Sun, X. Yuan, Wenzhang Liu, and Changyin Sun. Model-based reinforcement learning via proximal policy optimization. *2019 Chinese Automation Congress (CAC)*, pp. 4736–4740, 2019.

R. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 2005.

Csaba Szepesvari and W. Smart. Interpolation-based q-learning. In *ICML '04*, 2004.

J. Tsitsiklis. Asynchronous stochastic approximation and q-learning. *Machine Learning*, 16:185–202, 2004.

Elise van der Pol, Thomas Kipf, Frans A. Oliehoek, and M. Welling. Plannable approximations to mdp homomorphisms: Equivariance under actions. In *AAMAS*, 2020.

Z. Wang and S. Vucetic. Twin vector machines for online learning on a budget. In *SDM*, 2009.

Z. Wang and S. Vucetic. Online passive-aggressive algorithms on a budget. In *AISTATS*, 2010.

Z. Wang, K. Crammer, and S. Vucetic. Breaking the curse of kernelization: budgeted stochastic gradient descent for large-scale svm training. *J. Mach. Learn. Res.*, 13:3103–3131, 2012.

G. Yang, A. Zhang, Ari S. Morcos, Joelle Pineau, P. Abbeel, and R. Calandra. Plan2vec: Unsupervised representation learning by latent plans. *ArXiv*, abs/2005.03648, 2020.

Tianhe Yu, G. Thomas, Lantao Yu, S. Ermon, J. Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *ArXiv*, abs/2005.13239, 2020.

Dawit Zewdie and L. Kaelbling. Representation discovery in non-parametric reinforcement learning by dawit zewdie. In *CSAIL Technical Reports*, 2014.

Dawit Zewdie and G. Konidaris. Representation discovery for kernel-based reinforcement learning. In *CSAIL Technical Reports*, 2015.

A. Zhang, Rowan McAllister, R. Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *Bias and Invariances Workshop, ICML*, abs/2006.10742, 2020.

Q. Zhou, H. Li, and J. Wang. Deep model-based reinforcement learning via estimated uncertainty and conservative policy optimization. *ArXiv*, abs/1911.12574, 2020.

Guangxiang Zhu, Zichuan Lin, G. Yang, and C. Zhang. Episodic reinforcement learning with associative memory. In *ICLR*, 2020.