13. From API to Politex

In the lecture on approximate policy iteration, we proved that for any MDP feature-map pair (M, ϕ) and any $\varepsilon' > 0$ excess suboptimality target, with a total runtime of

$$\text{poly}\left(d, \frac{1}{1-\gamma}, A, \frac{1}{\varepsilon'}\right),$$

least-squares policy iteration with G-optimal design (LSPI-G) can produce a policy π such that the suboptimality gap δ of π satisfies

$$\delta \le \frac{2(1+\sqrt{d})}{(1-\gamma)^2}\varepsilon + \varepsilon', \tag{1}$$

where ε is the worst-case error with which the d-dimensional features can approximate the action-value functions of memoryless policies of the MDP M. In fact, the result continues to hold if we restrict the memoryless policies to those that are ϕ**-measurable** in the sense that the probability assigned by such a policy to taking some action a in some state s depends only on $\phi(s, \cdot)$. Denote the set of such policies by Π_ϕ. Then, for an MDP M and associated feature-map ϕ, let

$$\tilde{\varepsilon}(M, \phi) = \sup_{\pi \in \Pi_\phi} \inf_\theta \|\Phi\theta - q^\pi\|_\infty .$$

Checking the proof, noticing that LSPI produces ϕ-measurable policies only, it follows that provided the first policy it uses is also ϕ-measurable, ε in (1) can be replaced by $\tilde{\varepsilon}(M, \phi)$.

Earlier, we also proved that the amplification of ε by the $\sqrt{d}$-factor is unavoidable by **any** efficient planner. However, this leaves open the question of whether the amplification by a polynomial power of $1/(1-\gamma)$ is necessary, and whether in particular, the quadratic dependence is necessary? Our first result, which is given without proof, shows that in the case of LSPI this amplification is real and the quadratic dependence cannot be improved.

Theorem (LSPI error amplification lower bound): The quadratic dependence in (1) is tight: There exists a constant $c > 0$ such that for every $0 \le \gamma < 1$ and every $\varepsilon > 0$ there exists a featurized MDP (M, ϕ), a policy π of the MDP, a distribution μ over the states such that LSPI

when it is allowed infinitely many rollouts of infinite length produces a sequence of policies $\pi_0 = \pi, \pi_1, \dots$ such that

$$\inf_{k \geq 1} \mu(v^* - v^{\pi_k}) \geq \frac{c\tilde{\varepsilon}(M, \phi)}{(1-\gamma)^2} \, .$$

The result of the theorem holds even when LSPI is used with **state-aggregation**. Intuitively, state-aggregation means that states are groups into a number of groups and states belonging to the same group are treated identically when it comes to representing value functions. This, value-functions based on state-aggregation are constant over any group. When we are concerned with state-value functions, aggregating the states based on a partitioning of the states $\mathcal{S}$ into the groups $\{\mathcal{S}_i\}_{1 \leq i \leq d}$ (i.e., $\mathcal{S}_i \subset \mathcal{S}$ and all the subsets are disjoint from each other), a feature-map that allows to represent these piecewise constant functions is

$$\phi_i(s) = \mathbb{I}(s \in \mathcal{S}_i) \, , \qquad i \in [d] \, ,$$

where $\mathbb{I}$ is the indicator function that takes the value of one when its argument (a logical expression) is true, and is zero otherwise. In other words, $\phi : \mathcal{S} \to \{e_1, \dots, e_d\}$. Any feature map of this form defines a partitioning of the state-space and thus corresponds to the state-aggregation. Note that the piecewise constant functions can also be represented if we rotate all the features by the same rotation. The only important aspect here is that the features of different states are either identical, or orthogonal to each other, making the rows of the feature matrix an **orthonormal** system.

For approximating action-value functions, state-aggregation uses the same partitioning of states regardless of the identity of the actions: In effect, for each action, one uses the feature map from above, but with a private parameter vector. This effectively amounts to stacking $\phi(s)$ A-times, to get one copy of it for each action $a \in \mathcal{A}$. Note that for state-aggregation, there is no $\sqrt{d}$ amplification of the approximation errors: State-aggregation is extrapolation friendly, as will be explained at the end of the lecture.

Returning to the result, an inspection of the actual proof reveals that in this case LSPI leads to a sequence of policies that alternate between the initial policy and π_1. "Convergence" is fast, yet, the guarantee is far from satisfactory. In particular, in the same example, an alternate algorithm, which we will cover next can **reduce the quadratic dependence on the horizon to a linear dependence**.

Politex

Politex comes from **Po**licy **It**eration with **Ex**pert Advice. Assume that one is given a featurized MDP (M, ϕ) with state-action feature-map ϕ and access to a simulator, and a G-optimal design

$\mathcal{C} \subset \mathcal{S} \times \mathcal{A}$ for ϕ.

Politex generates a sequence of policies $\pi_0, \pi_1, \dots$ such that for $k \geq 1$,

$$\pi_k(a|s) \propto \exp\left(\eta \bar{q}_{k-1}(s,a)\right),$$

where

$$\bar{q}_k = \hat{q}_0 + \cdots + \hat{q}_j,$$

with

$$\hat{q}_j = \Pi_\Phi \hat{\theta}_j,$$

where for $j \geq 0$, $\hat{\theta}_j$ is the parameter vector obtained by running the least-squares policy evaluation algorithm based on G-optimal design (LSPE-G) to evaluate policy π_j (see this lecture). In particular, recall that this algorithm rolls out policy π_j from the points of a G-optimal design to produce m independent trajectories of length H each, calculates the average return for each of these design points and then solves the (weighted) least-squares regression problem where the features are used to regress on the obtained values.

Above, $\Pi : \mathbb{R}^{\mathcal{S}\times\mathcal{S}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ truncates its argument to the $[0, 1/(1-\gamma)]$ interval:

$$(\Pi q)(s,a) = \max(\min(q(s,a), 1/(1-\gamma)), 0), \qquad (s,a) \in \mathcal{S} \times \mathcal{A}\,.$$

Note that to calculate $\pi_k(a|s)$, one does need to calculate $E_k(s,a) = \exp\left(\eta \Pi[\phi(s,a)^\top \bar{\theta}_{k-1}]\right)$ and then compute $\pi_k(a|s) = E_k(s,a) / \sum_{a'} E_k(s,a')$.

Unlike in policy iteration, the policy returned by Politex after k iterations is either the "mixture policy"

$$\bar{\pi}_k = \frac{1}{k}\left(\pi_0 + \cdots + \pi_{k-1}\right),$$

or the policy which gives the best value with respect to the start state, or start distribution. For simplicity, let us just consider the case when $\bar{\pi}_k$ is used as the output. The meaning of a mixture policy is simply that one of the k policies is selected uniformly at random and then the selected policy is followed for the rest of time. Homework 3 gives precise definitions and asks you to prove that the value function of $\bar{\pi}_k$ is just the mean of the value functions of the constituent policies:

$$v^{\bar{\pi}_k} = \frac{1}{n}\left(v^{\pi_0} + \cdots + v^{\pi_{k-1}}\right). \tag{2}$$

We now argue that the dependence on the approximation error of the suboptimality gap of $\bar{\pi}_k$ only scales with $1/(1-\gamma)$, unlike the case of approximate policy iteration.

For this, recall that by the value difference identity

$$v^{\pi^*} - v^{\pi_j} = (I - \gamma P_{\pi^*})^{-1} \left[T_{\pi^*} v^{\pi_j} - v^{\pi_j} \right].$$

Summing up, dividing by k, and using (2) gives

$$v^{\pi^*} - v^{\bar{\pi}_k} = \frac{1}{k} (I - \gamma P_{\pi^*})^{-1} \sum_{j=0}^{k-1} T_{\pi^*} v^{\pi_j} - v^{\pi_j}.$$

Now, $T_{\pi^*} v^{\pi_j} = M_{\pi^*}(r + \gamma P v^{\pi_j}) = M_{\pi^*} q^{\pi_j}$. Also, $v^{\pi_j} = M_{\pi_j} q^{\pi_j}$. Let $\hat{q}_j = \Pi \Phi \hat{\theta}_j$. Elementary algebra then gives

$$\begin{aligned} v^{\pi^*} - v^{\bar{\pi}_k} &= \frac{1}{k} (I - \gamma P_{\pi^*})^{-1} \sum_{j=0}^{k-1} M_{\pi^*} q^{\pi_j} - M_{\pi_j} q^{\pi_j} \\ &= \underbrace{\frac{1}{k} (I - \gamma P_{\pi^*})^{-1} \sum_{j=0}^{k-1} M_{\pi^*} \hat{q}_j - M_{\pi_j} \hat{q}_j}_{T_1} + \underbrace{\frac{1}{k} (I - \gamma P_{\pi^*})^{-1} \sum_{j=0}^{k-1} (M_{\pi^*} - M_{\pi_j})(q^{\pi_j} - \hat{q}_j)}_{T_2}. \end{aligned}$$

We see that the approximation errors $\varepsilon_j = q^{\pi_j} - \hat{q}_j$ appear only in term T_2. In particular, taking pointwise absolute values, using the triangle inequality, we get that

$$\|T_2\|_\infty \le \frac{2}{1-\gamma} \max_{0 \le j \le k-1} \|\varepsilon_j\|_\infty,$$

which shows the promised dependence. It remains to show that $\|T_1\|_\infty$ above is also under control. However, this is left to the next lecture.

Notes

State aggregation and extrapolation friendliness

The $\sqrt{d}$ in our results comes from controlling the extrapolation errors of linear prediction. In the case of state-aggregretion, however, this extra $\sqrt{d}$ error amplification is completely avoided: Clearly, if we measure a function with a precision ε and there is at least one measurement per part, then by using the value measured at each part (at an arbitrary state there) over the whole part, the worst-case error is bounded by ε. Weighted least-squares in this context just takes the weighted average of the responses over each part and uses this as the prediction, so it also avoids amplifying approximation errors.

In this case, our analysis of extrapolation errors is clearly conservative. The extrapolation error was controlled in two steps: In our first lemma, for ρ weighted least-squares we reduced this problem to that of controlling $g(\rho) = \max_{z \in \mathcal{Z}} \|\phi(z)\|_{G_\rho^{-1}}$ where G_ρ is the moment matrix for ρ.

In fact, the proof of this lemma is the culprit: By carefully inspecting the proof, we can see that the application of Jensen's inequality introduces an unnecessary term: For the case of state aggregation (orthonormed feature matrix),

$$\sum_{z' \in C} \varrho(z') |\phi(z')^\top G_\varrho^{-1} \phi(z')| = 1$$

as long as the design ρ is such that it chooses any group exactly once. Thus, the case of state-aggregation shows that some feature-maps are more **extrapolation friendly** than others. Also, note that the Kiefer-Wolfowitz theorem, of course, still gives that $\sqrt{d}$ is the smallest value that we can get for g when optimizing for ρ.

It is a fascinating question of how extrapolation errors behave for various feature-maps.

Least-squares value iteration (LSVI)

In homework 2, Question 3 was concerned with least-squares value iteration. The algorithm concerned (call it LSVI-G) uses a random approximation of the Bellman operator, based on a G-optimal design (and action-value functions). The problem was to show a result similar to what holds for LSPI-G holds for LSVI-G, as well. That is, for any MDP feature-map pair (M, ϕ) and any $\varepsilon' > 0$ excess suboptimality target, with a total runtime of

$$\text{poly}\left(d, \frac{1}{1-\gamma}, A, \frac{1}{\varepsilon'}\right),$$

least-squares policy iteration with G-optimal design (LSPI-G) can produce a policy π such that the suboptimality gap δ of π satisfies

$$\delta \le \frac{4(1+\sqrt{d})}{(1-\gamma)^2} \varepsilon_{\text{BOO}} + \varepsilon' . \tag{3}$$

Thus, the dependence on the horizon of the approximation error is similar to the one that was obtained for LSPI. Note that the definition of ε_{BOO} is different from what we have used in analyzing LSPI:

$$\varepsilon_{\text{BOO}} := \sup_\theta \inf_{\theta'} \|\Phi\theta' - T\Pi\Phi\theta\|_\infty .$$

Above, T is the Bellman optimality oerator for action-value functions and Π is defined so that for $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, Πf is also a $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$ function which is obtained from f by truncating for each input (s,a) the value $f(s,a)$ to $[0, 1/(1-\gamma)]$:
$(\Pi(f))(s,a) = \max(\min(f(s,a), 1/(1-\gamma)), 0)$. In ε_{BOO}, "BOO" stands for "Bellman-optimality operator" in reference to the appearance of T in the definition.

In general, the error measures ε used in LSPI and $\varepsilon_{\mathrm{BOO}}$ are incomparable. The latter quantity measures a "one-step error", while ε is concerned with approximating functions defined over an infinite-horizon.

Linear MDPs

Call an **MDP linear** if both the reward function and the next state distributions for each state lie in the span of the features: $r = \Phi\theta_r$ with some $\theta_r \in \mathbb{R}^d$ and P, as an $\mathrm{SA} \times \mathrm{S}$ matrix takes the form $P = \Phi W$ with some $W \in \mathbb{R}^{d \times \mathrm{S}}$. Clearly, this is a notion that captures how well the "dynamics" (including the reward) of the MDP can be "compressed".

When an MDP is linear, $\varepsilon_{\mathrm{BOO}} = 0$. We also have in this case that $\varepsilon = 0$. More generally, defining $\zeta_r = \inf_\theta \|\Phi\theta_r - r\|_\infty$ and $\zeta_P = \inf_W \|\Phi W - P\|_\infty$, it is not hard to see that $\varepsilon_{\mathrm{BOO}} \le \zeta_r + \gamma\zeta_P/(1-\gamma)$ and $\varepsilon \le \zeta_r + \gamma\zeta_P/(1-\gamma)$, which shows that both policy iteration (and its soft versions) and value iteration are "valid" approaches, though, by ignoring the fact that we are comparing upper bounds, this also shows that value iteration may have an edge over policy iteration when the MDP itself is compressible. This should not be too surprising given that value-iteration is "more direct" in aiming to calculate q^*. Yet, they may exist cases when the action-value functions are compressible, while the dynamics is not.

Stationary points of a policy search objective

Let $J(\pi) = \mu v^\pi$. A stationary point of J with respect to some set of memoryless policies Π is any $\pi \in \Pi$ such that

$$\langle \nabla J(\pi), \pi' - \pi \rangle \le 0\,.$$

It is known that if ϕ are state-aggregation features then any stationary point π of J satisfies

$$\mu v^\pi \ge \mu v^* - \frac{4\varepsilon_{\mathrm{apx}}}{1-\gamma}\,,$$

where $\varepsilon_{\mathrm{apx}}$ is defines as the worst-case error of approximation action-value functions of ϕ-measurable policies with the features (the same constant as used in the analysis of approximate policy iteration).

Soft-policy iteration with Averaging

Politex can be seen as a "soft" version of policy iteration with averaging. The softness is controlled by η: When $\eta \to \infty$, Politex uses a greedy policy w.r.t. to an average of all previous Q-functions. Notice that in this case if Politex were to use a greedy policy w.r.t. the last Q-function, then it would reduce exactly to LSPI-G. As we have seen, in LSPI-G the approximation error can get quadratically amplified with the horizon $1/(1-\gamma)$. Thus, one way to avoid this quadratic amplification is to stay soft with averaging. As we shall see in the next lecture, the price of this is

a relatively slower convergence to a target suboptimality excess value. Nevertheless, the promise is that the algorithm will still stay polynomial in all the relevant quantities.

References

Politex was introduced in the paper

- POLITEX: Regret Bounds for Policy Iteration using Expert Prediction. Abbasi-Yadkori, Y.; Bartlett, P.; Bhatia, K.; Lazic, N.; Szepesvári, C.; and Weisz, G. In ICML, pages 3692–3702, May 2019. pdf

However, as this paper also notes, the basic idea goes back to the MDP-E algorithm by Even-Dar et al:

- Even-Dar, E., Kakade, S. M., and Mansour, Y. Online Markov decision processes. Mathematics of Operations Research, 34(3):726–736, 2009.

This algorithm considered a tabular MDP with nonstationary rewards – a completely different setting. Nevertheless, this paper introduces the basic argument presented above. The Politex paper notices that the argument can be extended to the case of function approximation. In particular, it also notes the nature of the function approximator is irrelevant as long as the approximation and estimation errors can be tightly controlled.

The Politex paper presented an analysis for online RL and average reward MDPs. Both add significant complications. The argument shown here is therefore a simpler version. Connecting Politex to LSPE-G in the discounted setting is trivial, but has not been presented before in the literature.

The first paper to use the error decomposition shown here together with function approximation is

- Abbasi-Yadkori, Y., Lazic, N., and Szepesvári, C. Modelfree linear quadratic control via reduction to expert prediction. In AISTATS, 2019.