IEOR 265 – Lecture 16
The Oracle

1   Naming

In theoretical computer science, oracles are black boxes that take in inputs and give answers. An important class of arguments known as relativizing proofs utilize oracles in order to prove results in complexity theory and computability theory. These proofs proceed by endowing the oracle with certain generic properties and then studying the resulting consequences.

We have named the functions $\mathcal{O}_n$ oracles in reference to those in computer science. Our reason is that we proved robustness and stability properties of LBMPC by only assuming generic properties, such as continuity or boundedness, on the function $\mathcal{O}_n$. These functions are arbitrary, which can include worst case behavior, for the purpose of the theorems in the previous section.

Whereas before, we considered the oracles as abstract objects, here we discuss and study specific forms that the oracle can take. In particular, we can design $\mathcal{O}_n$ to be a statistical tool that identifies better system models. This leads to two natural questions: First, what are examples of statistical methods that can be used to construct an oracle for LBMPC? Secondly, when does the control law of LBMPC converge to the control law of MPC that knows the true model? It will turn out that the second question is complex, and will be discussed in a later lecture.

2   Parametric Oracles

A *parametric oracle* is a continuous function $\mathcal{O}_n(x, u) = \chi(x, u; \lambda_n)$ that is parameterized by a set of coefficients $\lambda_n \in \mathcal{T} \subseteq \mathbb{R}^L$, where $\mathcal{T}$ is a set. This class of learning is often used in adaptive control. In the most general case, the function χ is nonlinear in all its arguments, and it is customary to use a least-squares cost function with input and trajectory data to estimate the parameters

$$\hat{\lambda}_n = \arg\min_{\lambda \in \mathcal{T}} \sum_{j=0}^{n} (Y_j - \chi(x_j, u_j; \lambda))^2, \tag{1}$$

where $Y_i = x_{i+1} - (Ax_i + Bu_i)$. This can be difficult to compute in real-time because it is generally a nonlinear optimization problem.

Example: It is common in biochemical networks to have nonlinear terms in the dynamics such as

$$\mathcal{O}_n(x, u) = \lambda_{n,1} \left(\frac{x_1^{\lambda_{n,2}}}{x_1^{\lambda_{n,2}} + \lambda_{n,3}} \right) \left(\frac{\lambda_{n,4}}{u_1^{\lambda_{n,5}} + \lambda_{n,4}} \right), \tag{2}$$

where $\lambda_n \in \mathcal{T} \subset \mathbb{R}^5$ are the unknown coefficients in this example. Such terms are often called Hill equation type reactions.

2.1 Linear Oracles

An important subclass of parametric oracles are those that are linear in the coefficients: $\mathcal{O}_n(x, u) = \sum_{i=1}^{L} \lambda_{n,i} \chi_i(x, u)$, where $\chi_i \in \mathbb{R}^p$ for $i = 1, \ldots, L$ are a set of (possibly nonlinear) functions. The reason for the importance of this subclass is that the least-squares procedure (1) is convex in this situation, even when the functions χ_i are nonlinear. This greatly simplifies the computation required to solve the least-squares problem (1) that gives the unknown coefficients λ_n.

Example: One special case of linear parametric oracles is when the χ_i are linear functions. Here, the oracle can be written as $\mathcal{O}_m(x, u) = F_{\lambda_m} x + G_{\lambda_m} u$, where $F_{\lambda_m}, G_{\lambda_m}$ are matrices whose entries are parameters. The intuition is that this oracle allows for corrections to the values in the A, B matrices of the nominal model; it was used in conjunction with LBMPC on a quadrotor helicopter testbed that will be discussed in later lectures, in which LBMPC enabled high-performance flight.

3 Nonparametric Oracles

Nonparametric regression refers to techniques that estimate a function $g(x, u)$ of input variables such as x, u, without making *a priori* assumptions about the mathematical form or structure of the function g. This class of techniques is interesting because it allows us to integrate non-traditional forms of adaptation and "learning" into LBMPC. And because LBMPC robustly maintains feasibility and constraint satisfaction as long as Ω can be computed, we can design or choose the nonparametric regression method without having to worry about stability properties. This is a specific instantiation of the separation between robustness and performance in LBMPC.

Example: Neural networks are a classic example of a nonparametric method that has been used in adaptive control, and they can also be used with LBMPC. There are many particular forms of neural networks, and one specific type is a feedforward neural network with a hidden layer of k_n neurons; it is given by

$$\mathcal{O}_n(x, u) = c_0 + \sum_{i=1}^{k_n} c_i \sigma(a_i'[x'\ u']' + b_i), \tag{3}$$

where $a_i \in \mathbb{R}^{p+m}$ and $b_i, c_0, c_i \in \mathbb{R}$ for all $i \in \{1, \ldots, k\}$ are coefficients, and $\sigma(x) = 1/(1+e^{-x}) : \mathbb{R} \to [0, 1]$ is a sigmoid function. Note that this is considered a nonparametric method because it does not generally converge unless $k_n \to \infty$ as $n \to \infty$.

Designing a nonparametric oracle for LBMPC is challenging because the tool should ideally be an estimator that is bounded to ensure robustness of LBMPC and differentiable to allow for its use with numerical optimization algorithms. Local linear estimators are not guaranteed to be bounded, and their extensions that remain bounded are generally non-differentiable. On the other hand, neural networks can be designed to remain bounded and differentiable, but they can have technical difficulties related to the estimation of its coefficients. In future lectures, we will discuss one specific type of nonparametric oracle that works well with LBMPC both theoretically and in simulations.

4   Sequence of Optimization Problems

Suppose we have an optimization problem

$$\min f(x),$$

where $f : \mathbb{R}^p \to \overline{\mathbb{R}}$ is lsc, level-bounded, and proper. In many applications, we may not be interested in directly solving this optimization problem. Instead, we will have a sequence of optimization problems that are meant to approximate this problem, and we instead solve these approximating optimization problems. This setting is not as abstract as it may seem at first glance: For example, this is implicitly what is done when using log-barrier functions, which are common in interior-point methods for numerically solving constrained optimization problems by converting a constrained problem into a sequence of approximating unconstrained problems. We will refer to this sequence of optimization problems as

$$\min f_n(x).$$

This is also the situation in LBMPC, where we have an optimization problem that changes as we get more data about the system being controlled. For LBMPC, the relevant question is ultimately: What is a good oracle to use? This is more difficult than may initially be seen because we would ideally like the oracle to be designed such that the minimizers of LBMPC converge to the minimizers of an MPC that knows the true system model.

The central question with such sequences of approximating optimization problems is whether they provide a good approximation to the solution of the original optimization problem. That is, ideally we would like for $V_n \to V$, where

$$V_n = \min f_n(x)$$
$$V = \min f(x),$$

and we would also like for $\limsup_n \mathcal{X}_n^* \subseteq \mathcal{X}^*$ where

$$\mathcal{X}_n^* = \arg\min f_n(x)$$
$$\mathcal{X}^* = \arg\min f(x).$$

This last statement means that if x is any point such that there exists a subsequence $x_\nu^* \in \mathcal{X}_\nu^*$ such that $x_\nu^* \to x$, then $x \in \mathcal{X}^*$. The reason for this condition is that it may be the case that the limiting optimization problem has more minimizers than each individual optimization in the sequence. Note that when there is only a single minimizer, we must have that $x_n^* \to x^*$.

At its surface, this might seem like an easy exercise; however, certain pathologies can occur, as evidenced by the following example: Consider a sequence of optimization problems

$$\min \ \min\{1 - x, 1, 2n|x + 1/n| - 1\}$$
$$\text{s.t.} \ x \in [-1, 1].$$

The objectives $f_n(x) = \min\{1-x, 1, 2n|x+1/n|-1\}$ converge pointwise to $f(x) = \min\{1-x, 1\}$ (recall that the definition of pointwise convergence is that $\lim_{n\to\infty} f_n(\overline{x}) = f(\overline{x})$ for all $\overline{x} \in \operatorname{dom} f$). Next we turn our attention to the minimizers of the sequence, which are $x_n^* = -1/n$ and converge to zero. However, the minimizer of the optimization problem that we are approximating by the sequence is $x^* = 1$. The minimizers of the sequence do not converge to the minimizer of the limiting optimization problem.

Convergence of local minimizers to the local minimizers of a limiting optimization problem is more difficult because additional pathologies can occur. For instance, it is possible for local minimizers of a sequence of approximating problems to converge to the global maximizer of the limiting problem. Proving such pathologies does not occur requires the use of more sophisticated arguments, such as those using *optimality functions*.

5 Uniform Convergence

This problem of approximating optimization problems has actually already been encountered in this class. Specifically, we constructed a number of M-estimators and proved that the minimizer of the M-estimator converged to the true parameter value. For instance, for ordinary least squares (OLS) we solve

$$\hat{\beta} = \arg\min_u \frac{1}{n} \sum_{i=1}^n (y_i - x_i'u)^2$$

In fact, the OLS problem is really a sample average approximation of the following problem:

$$\beta = \arg\min_u \mathbb{E}((y - x'u)^2).$$

And so when proving the statistical consistency of OLS, we are actually showing that the minimizer to the first problem converges in probability to the minimizer of the second problem.

5.1 Uniform Law of Large Numbers

In statistics, the two major proof techniques for showing such convergence are essentially uniform convergence arguments. For nonlinear least squares problems, a uniform law of large numbers (ULLN) can be proven: This states that if $g(x, y; \beta)$ is continuous in its arguments, then we have

$$\sup_{u\in\Omega} \left| \frac{1}{n} \sum_{i=1}^n g(x_i, y_i; u) - \mathbb{E}(g(x, y; u)) \right| \xrightarrow{p} 0,$$

when Ω is bounded and closed and some technical regularity conditions hold. And so this ULLN implies that

$$\arg\min_{u\in\Omega} \frac{1}{n} \sum_{i=1}^n g(x_i, y_i; u) \xrightarrow{p} \arg\min_{u\in\Omega} \mathbb{E}(g(x, y; u)),$$

when $\arg\min_{u\in\Omega} \mathbb{E}(g(x, y; u))$ consists of a single point.

5.2 Wald's Proof Technique

A related proof technique due to Wald (1949) applies to situations where weaker continuity properties hold. In particular, if $g(x, y; \beta)$ is lower semicontinuous, then the ULLN does not generally hold. However, it can still be shown that

$$\text{distance}(\hat{\beta}, \arg \min_{u \in \Omega} \mathbb{E}(g(x, y; u))) \xrightarrow{p} 0,$$

where $\hat{\beta} = \arg \min_{u \in \Omega} \frac{1}{n} \sum_{i=1}^{n} g(x_i, y_i; u)$. The idea of the Wald approach is to find a finite subcover of the set $\Pi = \{u \in \Omega : \text{distance}(u, \arg \min_{u \in \Omega} \mathbb{E}(g(x, y; u))) > \epsilon\}$ and then apply the weak law of large numbers (WLLN) to the centers of the covering balls to show that *almost surely* we have that

$$\min_{u \in \Pi} \frac{1}{n} \sum_{i=1}^{n} g(x_i, y_i; u) > \arg \min_{u \in \Omega} \mathbb{E}(g(x, y; u)).$$

The result follows since this can be done for any $\epsilon > 0$.

5.3 Rademacher Complexity

The proof techniques underlying Rademacher complexity and the M^*-bound are essentially uniform arguments. For instance, we showed that

$$\mathbb{E}((y - \hat{f}(x))^2) \leq \inf_{f \in \mathcal{F}} \mathbb{E}((y - f(x))^2 + 4\mathsf{R}_n(\mathcal{F}) + \frac{C}{n},$$

where $C > 0$ is a constant and $\mathsf{R}_n(\mathcal{F})$ is the Rademacher complexity of the function class $\mathcal{F}$.

5.4 Stochastic Equicontinuity

Another uniform convergence argument is to show stochastic equicontinuity of the optimization problem, meaning show that for every $\epsilon > 0$ and $\eta > 0$ there exists $\delta > 0$ such that

$$\limsup \mathbb{P}\left(\sup_{u \in \Omega} \sup_{u' \in \mathcal{B}(\theta, \delta)} \left| \frac{1}{n} \sum_{i=1}^{n} g(x_i, y_i; u) - \frac{1}{n} \sum_{i=1}^{n} g(x_i, y_i; u') \right| > \epsilon \right) < \eta.$$

6 Further Details

The counterexample above is from *Variational Analysis* by Rockafellar and Wets.