Dynamically optimal treatment allocation using Reinforcement Learning

Karun Adusumilli (UPenn), Friedrich Geiecke (LSE) & Claudio Schilter (U. Zurich)

January 1, 2020
Dynamic Treatment Allocation

- The treatment assignment problem:
  - How do we assign individuals to treatment using observational data?

- Decision problem of maximizing population welfare
  - Large literature on this in the ‘static’ setting
  - Exploits similarity with classification

- This paper:
  - Individuals arrive sequentially (e.g when unemployed)
  - Planner has to assign individuals to treatment (e.g job training):
  - Various planner constraints: Finite budget/capacity, borrowing, queues...
  - Turns out similar to optimal control/Reinforcement Learning
Dynamics vs Statics: Two examples

- **Borrowing constraints**
  - Assume rate of arrival of individuals and flow of funds is constant
  - ‘Static’ rule (e.g. Kitagawa-Tetnov ‘18): only depends on covariates
  - However: Under a static rule budget follows a random walk!
  - Eventually shatters any borrowing constraints
  - Optimal rule: Change with budget ≡ optimal control of budget path

- **Finite budget**
  - Planner starts with pot of money that is not replenished
  - Training depletes budget and future benefits are discounted
  - Existing methods not applicable even if we just want a ‘static rule’
  - They need specification % of population to be treated
  - But this is endogenous to policy!
Other examples

▶ Finite budget and time
  ▶ Planner is given pot of money to be used up within a year

▶ Finite capacity
  ▶ E.g fixed number of caseworkers for home visits etc
  ▶ If capacity is full, people turned away (or waitlisted)
  ▶ People finish treatment at known rates which frees up capacity

▶ Queues
  ▶ Why? Time for treatment is longer than arrival rates
  ▶ Waiting is costly and not treating someone shortens wait times
  ▶ Current length of queue is a state variable

▶ Related: Multiple queues
  ▶ Some cases are more time-sensitive
  ▶ Can use two queues: shorter queue for riskier patients
Preliminary remarks

- We focus on ‘offline’ learning
  - Use historical/RCT data to estimate policy
  - In infinite horizon, our algorithm can be used fully online
  - However we do not have any claim on optimality
  - Note: bandit algorithms are not applicable!
- Key assumption: Individuals do not respond strategically to policy
  - Arrival rates are exogenous and unaffected by policy
  - However results apply if we have a model of policy response
What we do: Overview

- Estimation of optimal policy rule in pre-specified class
  - Ethical/computational/legal reasons (Kitagawa-Tetenov, 2018)

- Basic elements of our theory
  - For each policy, write down a PDE for expected value fn (a la HJB)
  - Using data, write down sample version of PDE for each policy
  - Maximize over sample PDE solutions to estimate optimal policy
  - Bound difference in solutions using PDE techniques
    ⇒ Regret bounds
Overview (contd.)

▶ Computation

▶ Approximate PDE with (semi-discrete) dynamic program

▶ Solve using Reinforcement Learning (RL): Actor-Critic algorithm

▶ Solves for maximum within pre-specified policy classes

▶ Computationally fast due to parallelization

▶ Some results

▶ $\sqrt{\frac{v}{n}}$ rates for regret where $v$ is complexity of policy class
Setup

- State variable: $s \equiv (x, z, t)$
  - $x$ individual covariates
  - $z$ budget/institutional constraint
  - $t$ time

- Arrivals: Poisson point process with parameter $\lambda(t)N$
  - Set $\lambda(t_0) = 1$ as normalization
  - $N$ is scale parameter that will be taken to $\infty$

- Distribution of covariates: $F$
  - Assumed fixed for this talk
  - In paper: allowed to change with $t$
Setup (contd.)

- Actions: $a = 1$ (Train) or $a = 0$ (Do not train)

- Choosing $a$ results in utility $Y(a)/N$ for social planner
  - Utility scaled to a ‘per-person’ number

- Rewards: expected utility given covariate $x$
  \[
  r(x, a) = E[Y(a)|x]
  \]
  - Look at additive welfare criteria so normalize $r(x, 0) = 0$
Setup (contd.)

- Law of motion for $z$:

$$z' - z = \frac{G_a(s)}{N}, \; a \in \{0, 1\}$$

- Interpreting $G_a(s)$: Flow rate of budget wrt mass $m$ of individuals
- Here, $m$ is defined by giving each individual $1/N$ weight
- If planner chooses $a$ for mass $\delta m$ of individuals, $z$ changes by

$$\delta z \approx G_a(s) \delta m$$

- Example: Denote

  - $\sigma(z, t)$: Rate of inflow of funds wrt time
  - $c(x, z, t)$: Cost of treatment per person
  - $b$: Interest rate for borrowing/saving

$$G_a(s) = \lambda(t)^{-1} \{\sigma(z, t) + bz\} - c(x, z, t)\mathbb{I}(a = 1)$$
Policy class

- Policy function: $\pi(.|s) : s \rightarrow [0, 1]$
  - Taken to be probabilistic

- We consider policy class $\{\pi_\theta : \theta \in \Theta\}$
  - Can include various constraints on policies
  - For theoretical results: $\theta$ can be anything

- In practice we use soft-max class

$$\pi^{(\sigma)}_\theta(1|x, z) = \frac{\exp(\theta^T f(x, z)/\sigma)}{1 + \exp(\theta^T f(x, z)/\sigma)}$$

- $\sigma$ is ‘temperature’: can be fixed or subsumed into $\theta$
- E.g: $\sigma \rightarrow 0$ gives linear-eligibility scores (Kitagawa & Tetenov, ‘18)
Value functions

- Integrated value function: $h_\theta(z, t)$
  - Expected welfare for social planner at $z, t$ before observing $x$

- Define
  \[
  \bar{r}_\theta(z, t) := E_{x \sim F} [r(x, 1) \pi_\theta (1|x, z, t)],
  \]
  and
  \[
  \bar{G}_\theta(z, t) := E_{x \sim F} [G_1(s) \pi_\theta (1|s) + G_0(s) \pi_\theta (0|s)|z, t]
  \]

- $\bar{r}_\theta(z, t)$: expected flow (wrt mass of people) utility at state $(z, t)$
- $\bar{G}_\theta(z, t)$: expected flow change to $z$ at state $(z, t)$
PDE for the integrated value function

\[ \beta h_\theta(z, t) - \lambda(t) \tilde{r}_\theta(z, t) - \lambda(t) \tilde{G}_\theta(z, t) \partial_z h_\theta(z, t) - \partial_t h_\theta(z, t) = 0 \]

- Obtained in the limit \( N \to \infty \)
  - In fact \( N = 1 \) also gives same PDE in infinite horizon setup

- PDE encapsulates ‘no arbitrage’
  - Think of \( \beta \) as natural rate of interest and \( h_\theta(z, t) \) as valuation

- We need to specify boundary condition

- In general differentiable solution does not exist!
  - Work with viscosity solutions (Crandall & Lions 83)
Boundary conditions

- **Dirichlet:**
  - Finite time horizon, finite budget or both
    
    \[ h_\theta(z, t) = 0 \text{ on } \Gamma; \quad \Gamma \equiv \{(z, t) : z = 0 \text{ or } t = T\} \]

- **Periodic:**
  - Infinite horizon setting with \( t \) periodic with period \( T_p \)
    
    \[ h_\theta(z, t) = h_\theta(z, t + T_p) \quad \forall (z, t) \in \mathbb{R} \times [t_0, \infty) \]

- **Generalized Neumann (Finite\|Infinite horizon versions):**
  - Basic idea: behavior at boundary is different from interior
  - Useful to model borrowing constraints
    
    \[
    \beta h_\theta(z, t) - \sigma(z, t) \partial_z h_\theta(z, t) - \partial_t h_\theta(z, t) = 0, \quad \text{on } \{z\} \times [t_0, T] \\
    h_\theta(z, T) = 0, \quad \text{on } (z, \infty) \times \{T\} \quad \text{OR} \\
    h_\theta(z, t) = h_\theta(z, t + T_p), \quad \forall (z, t) \in \mathcal{U}
    \]
Social planner objective

\[ \beta h_\theta(z, t) - \lambda(t)\bar{r}_\theta(z, t) - \lambda(t)\bar{G}_\theta(z, t)\partial_z h_\theta(z, t) - \partial_t h_\theta(z, t) = 0 \]

- Class of PDEs: one for each policy
- We will think of \( \lambda(\cdot) \) as a ‘forecast’ and condition on it
- Policy objective given \( \lambda(\cdot) \):
  \[ \theta^* = \arg\max_{\theta \in \Theta} W(\theta); \quad W(\theta) := h_\theta(z_0, t_0) \]
  - \( z_0, t_0 \): Initial budget and time
- More generally: planner has distribution over forecasts \( \lambda(t) \)
  - Then: \( W(\theta) = \int h_\theta(z_0, t_0; \lambda) dP(\lambda) \)
The sample counterparts

- Denote $F_n$ empirical distribution of RCT data
  - Assume $F_n \to F$

- Estimate $r(x, a)$ using RCT data with a doubly robust estimate

- Define

  $$\hat{r}_\theta(z, t) = E_{x \sim F_n} [\hat{r}(x, 1) \pi_\theta (1|x, z, t)] ,$$

  and

  $$\hat{G}_\theta(z, t) := E_{x \sim F_n} [G_1(x, z, t) \pi_\theta (1|x, z, t) + G_0(x, z, t) \pi_\theta (0|x, z, t)]$$
Computation: Estimating the value function

- We can use sample counterparts and obtain sample PDE:

\[
\beta \hat{h}_\theta(z, t) - \lambda(t) \hat{G}_\theta(z, t) \partial_z \hat{h}_\theta(z, t) - \partial_t \hat{h}_\theta(z, t) - \lambda(t) \hat{r}_\theta(z, t) = 0
\]

- But solving this directly is too difficult

- Solution: approximate with a dynamic program instead

\[
\tilde{h}_\theta(z, t) = \frac{\hat{r}_\theta(z, t)}{b_n} + E_{n, \theta} \left[ e^{-\beta(t' - t)} \tilde{h}_\theta(z', t') \bigg| z, t \right]
\]

- Here: \( z' = z - b_n^{-1} G_a(s), \ b_n(t' - t) \sim \exp(\lambda(t)) \)

- \( 1/b_n \): discrete change to mass of individuals (basically same as \( 1/N \))

- Determines numerical error: same idea as step size in PDE solvers
Reinforcement Learning

▶ We create simulations of dynamic environment, called Episodes
  ▶ Using estimated rewards $\hat{r}$ and sampling individuals from $F_n$

▶ Just the environment for Reinforcement Learning
  ▶ Take action from current policy, observe $\hat{r}$, move to next state
  ▶ Based on reward, update policy

▶ We use Actor-Critic algorithm
  ▶ Stochastic Gradient Descent (SGD) updates along $\nabla_{\theta} \tilde{h}_\theta(z_0, t_0)$
  ▶ Gradient requires an estimate of $h_\theta(z, t)$ for current $\theta$
  ▶ Parametrize $\tilde{h}_\theta(z, t) = \nu^T \phi(z, t)$ and use another SGD to update $\nu$
  ▶ Key idea: update $\theta, \nu$ simultaneously!
  ▶ Two timescale trick uses faster learning rate for $\nu$
Statistical and numerical properties

Probabilistic bounds on regret

Suppose that $\hat{r}$ is a doubly robust estimate. Then under some regularity conditions

$$W(\theta^*) - W(\hat{\theta}) \leq C\sqrt{\frac{v}{n}} + K\sqrt{\frac{1}{b_n}}$$

uniformly over $(\lambda(\cdot), F)$

Remarks:

- $v$ is VC dimension of
  $$G_a = \{ \pi_\theta(a|\cdot,z,t)G_a(\cdot,z,t) : (z,t) \in \bar{U}, \theta \in \Theta \}$$
- Second term is numerical error from approximation
- Proof uses results from the theory of viscosity solutions
- For infinite horizon need $\beta$ to be sufficiently large
Application: JTPA study

- RCT data on training for unemployed adults
  - $n \approx 9000$, done over 2 years
  - Outcomes: 30 month earnings - cost of treatment ($774$)

- Finite budget and time: Can only treat 1600 people within a year
  - Discount factor $\beta = -\log 0.9$ or 0.9 over course of year

- Estimation of arrival rates:
  - Cluster data into 4 groups (k-means)
  - Estimate $\lambda(t)$ using Poisson regression for each cluster

- Policy class ($x : 1, \text{age, education, prev. earnings}$)
  $$\pi(a = 1|s) \sim \text{Logit}(x, x \cdot z)$$
Normalized relative to random policy (also roughly same as treating everyone)
Policy maps
Conclusion

- Actor-Critic algorithm for learning constrained optimal policy

- Some other extensions that we include in paper
  - Heterogenous non-compliance using IVs
  - Continuing to learn after coming online

- Ongoing work
  - Online learning
  - Dynamic treatment regimes
The Actor-Critic algorithm

Policy Gradient Theorem

\[ \nabla_\theta \tilde{h}_\theta (z_0, t_0) = E_{n, \theta} \left[ e^{-\beta(t-t_0)} \left\{ \hat{r}_n(x, a) + \beta \hat{h}_\theta (z', t') - \hat{h}_\theta (z, t) \right\} \nabla_\theta \ln \pi (a|s; \theta) \right] \]
The Actor-Critic algorithm

Policy Gradient Theorem

$$\nabla_\theta \tilde{h}_\theta(z_0, t_0) = E_{n,\theta} \left[ e^{-\beta(t-t_0)} \left\{ \hat{r}_n(x, a) + \beta \hat{h}_\theta(z', t') - \hat{h}_\theta(z, t) \right\} \nabla_\theta \ln \pi(a|s; \theta) \right]$$

Functional Approximation:

$$\nabla_\theta \tilde{h}_\theta(z_0, t_0) \approx E_{n,\theta} \left[ e^{-\beta(t-t_0)} \left\{ \hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t} \right\} \nabla_\theta \ln \pi(a|s; \theta) \right]$$
The Actor-Critic algorithm

Policy Gradient Theorem

\[ \nabla_\theta \tilde{h}_\theta(z_0, t_0) = E_{n, \theta} \left[ e^{-\beta(t-t_0)} \left\{ \hat{r}_n(x, a) + \beta \hat{h}_\theta(z', t') - \hat{h}_\theta(z, t) \right\} \nabla_\theta \ln \pi(a|s; \theta) \right] \]

Functional Approximation:

\[ \nabla_\theta \tilde{h}_\theta(z_0, t_0) \approx E_{n, \theta} \left[ e^{-\beta(t-t_0)} \left\{ \hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t} \right\} \nabla_\theta \ln \pi(a|s; \theta) \right] \]

Temporal-Difference (TD) Learning

\[ \nu^*_\theta = \arg \min_{\nu} E_{n, \theta} \left[ \left\| \tilde{h}_\theta(z, t) - \nu^T \phi_{z, t} \right\|^2 \right] := \hat{Q}(\nu|\theta) \]
Stochastic Gradient Updates

\[ \nabla_\theta \tilde{h}_\theta(z_0, t_0) \approx E_{n, \theta} \left[ e^{-\beta(t-t_0)} \{ \hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t} \} \nabla_\theta \ln \pi(a|s; \theta) \right] \]

\[ \nabla_\nu \hat{Q}(\nu|\theta) \approx E_{n, \theta} \left[ (\hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t}) \phi_{z, t} \right] \]

- Convert both to SGD updates (AC algorithm)

  \[ \theta \leftarrow \theta + \alpha_\theta e^{-\beta(t-t_0)} (\hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t}) \nabla_\theta \ln \pi(a|s; \theta) \]

  \[ \nu \leftarrow \nu + \alpha_\nu (\hat{r}_n(x, a) + \beta \nu^T \phi_{z', t'} - \nu^T \phi_{z, t}) \phi_{z, t} \]

- Updates are ‘online’
  - Take \( a \sim \pi_\theta \) and continually update while interacting with env.

- Updates to \( \theta, \nu \) done simultaneously at two timescales: \( \alpha_\nu \gg \alpha_\theta \)
  - No need to wait for \( \nu_\theta \) to converge
Convergence of Actor-Critic

Convergence of Actor-Critic algorithm

Suppose the learning rates satisfy \( \sum_k \alpha^{(k)} \to \infty \), \( \sum_k \alpha^{2(k)} < \infty \), and \( \alpha^{(k)} \theta / \alpha^{(k)} \nu \to 0 \). Then under some regularity conditions

\[
\theta^{(k)} \to \theta_c, \quad \nu^{(k)} \to \nu_c,
\]

where convergence is local. Furthermore given \( \epsilon > 0 \) there exists \( M \) s.t

\[
\| \hat{\theta} - \theta_c \| \leq \epsilon \quad \text{whenever } \text{dim}(\nu) \geq M.
\]

Remarks:

- \( k \) is order of updates
- There is no statistical tradeoff for choosing \( \text{dim}(\nu) \), ideally \( \nu = \infty \)
Application 2: Finite budget

- Finite budget: Can only treat 1600 people
  - Discount factor $\beta = -\log 0.9$ or 0.9 over course of year
  - Note: there is no time constraint anymore

- Policy class $(x: 1, \text{age, education, prev. earnings})$

$$\pi(a = 1 | s) \sim \text{Logit}(x, x \cdot \cos(2\pi t), x \cdot z)$$
Doubly Robust (preliminary)

- Episodes approximately trained in each of 23 parallel processes
- Average cumulative episode reward achieved

Reward trajectory

- # people considered: 145K ≈ 23 years
Policy maps (DR)

**Age Coefficient**

**Education Coefficient**

**Previous Earnings Coefficient**