Inference for Multistage Decision Policies via Regularized Q-Learning

Bibhas Chakraborty    Susan Murphy

Department of Statistics
University of Michigan

Yahoo! Research
Santa Clara, California
March 10, 2008
Outline

1 Introduction
   - One-stage Decision Problem
   - Multistage Decision Problem

2 Q-Learning
   - Estimation
   - Inference: A Non-regular Problem

3 “Soft-max” Regularization: A Solution
   - Regularized Estimator
   - Distribution of The Estimator

4 Discussion
Two One-stage Decision Problems

- **Example 1: Medical Treatments**
  - $X$ : Patient’s pre-treatment variables
    (e.g., age, gender, blood pressure, ...)
  - $A$ : Treatment
  - $R$ : Patient’s survival time after treatment

- **Example 2: Recommender Systems**
  - $X$ : Customer behavior (e.g., past purchases)
  - $A$ : Product recommendation (e.g., advertisements)
  - $R$ : Profit

**Common Goal:** For a given $X$, we want to make a decision about optimal $A$ such that the expectation of $R$ is maximized.
One-stage Decision Problem Framework

- Training dataset:

\[(X_i, A_i, R_i), \quad i = 1, \ldots, n\]

- \(X\) : Pre-action observation (high-dimensional)
- \(A\) : Action (categorical)
- \(R\) : Reward for taking action \(A\) when the observation \(X\) is available (one-dimensional)

- \(A\)'s are randomized in the training dataset.
- Given this data, we want to estimate a decision policy, a function \(\pi : \mathcal{X} \rightarrow \mathcal{A}\), that outputs an \(A\) for a given \(X\) so that expectation of \(R\) is maximized.
Multistage Decision Problem

- Treatment of chronic diseases, e.g., mental illness, substance abuse, HIV infection ...
- For simplicity, consider just two stages for every patient:

\[
\begin{align*}
X_{1}, A_{1}, R_{1}, & \quad X_{2}, A_{2}, R_{2} \\
_{t=1} & \quad _{i=2}
\end{align*}
\]

- \(X_{t}\) : Pre-treatment variables at the \(t\)-th stage
- \(A_{t}\) : Treatment at the \(t\)-th stage
- \(H_{t}\) : History available at the \(t\)-th stage

\[
H_{1} = X_{1}, H_{2} = \{X_{1}, A_{1}, X_{2}\}
\]

- \(R_{t}\) : Reward following the \(t\)-th stage
Given the training dataset (where $A$’s are randomized)

$$(X_{i1}, A_{i1}, R_{i1}, X_{i2}, A_{i2}, R_{i2}), \quad i = 1, \ldots, n,$$

we want to estimate a policy $\pi = (\pi_1, \pi_2)$ where $\pi_t : \mathcal{H}_t \rightarrow \mathcal{A}_t$, that outputs an $A_t$ for a given $H_t$.

**Goal:** We want to have a policy $\pi$ that maximizes the expectation of $(R_1 + R_2)$. 
Once we have a good policy $\pi$, we can employ the actions determined by it for any future individual with history $(H_1, H_2)$:

$$A_1 = \pi_1(H_1), \quad A_2 = \pi_2(H_2)$$

As policies, here we will consider parametric functions only. For example, suppose there are only two actions: $A_t = \{0, 1\}$. Then a policy can be $\pi_t(H_t) = 1_{\beta^t H_t > 0}$ where $\beta$ is a vector parameter.

This talk is about statistical inference (e.g., hypothesis tests, confidence intervals) on such $\beta$'s.
At the interface of Statistics and Reinforcement Learning ...

- **Direct methods:**
  - Policy Search (Kearns, Mansour, and Ng, 1999)
  - Weighting (Murphy et al., 2001)
  - Policy Mining and Reduction Techniques (Zadrozny, 2003; Langford and Zadrozny, 2004)

- **Likelihood-based methods:**
  - vast literature on POMDPs ...

- **Regression-based methods:**
  - Q-Learning (Watkins, 1989)
  - A-Learning (Murphy, 2003) or Structural Nested Mean Models (Robins, 2004)

We will discuss statistical inference in the context of Q-Learning!
Outline

1. Introduction
   - One-stage Decision Problem
   - Multistage Decision Problem

2. Q-Learning
   - Estimation
   - Inference: A Non-regular Problem

3. “Soft-max” Regularization: A Solution
   - Regularized Estimator
   - Distribution of The Estimator

4. Discussion
Motivation for Q-Learning

- If we knew the true joint distribution of the data, we could use **Dynamic Programming** (Bellman, 1957) for choosing a policy:

  **Stage 2:**

  \[
  Q_2(h_2, a_2) = \mathbb{E}[R_2 | H_2 = h_2, A_2 = a_2], \forall a_2
  \]

  \[
  \pi_2(h_2) \in \arg \max_{a_2} Q_2(h_2, a_2)
  \]

- The above conditional expectation judges the **Quality** of an action, hence called **Q-function**!
Motivation for Q-Learning

- Move backward in time to take care of “delayed effect” of current action

Stage 1:

$$Q_1(h_1, a_1) = \mathbb{E}\left[R_1 + \max_{a_2} Q_2(H_2, a_2) \middle| H_1 = h_1, A_1 = a_1\right], \forall a_1$$

- As we don’t know the true distribution, we need to “learn” (estimate) the Q-functions from training data: Q-Learning
A Simple Version of Q-Learning (following Murphy, 2005)

- For simplicity, assume $A_t \in \{-1, 1\}$.
- Approximate $Q_t(H_t, A_t)$ by $\alpha_t^T H_t + (\beta_t^T H_t) A_t$.

**Stage 2 Regression:** Use least squares with $R_2$ as outcome and $[H_2, H_2 A_2]$ as covariates to obtain $(\hat{\alpha}_2, \hat{\beta}_2)$.

- To address “delayed effect”, construct a “pseudo outcome”:

  $$\hat{R}_1 = R_1 + \max_{a_2} \hat{Q}_2(H_2, a_2) = R_1 + \hat{\alpha}_2^T H_2 + |\hat{\beta}_2^T H_2|$$

**Stage 1 Regression:** Use least squares with $\hat{R}_1$ as outcome and $[H_1, H_1 A_1]$ as covariates to obtain $(\hat{\alpha}_1, \hat{\beta}_1)$. 
Optimal $\pi$ by Q-Learning

Optimal stage-2 action is given by:

$$\hat{\pi}_2(h_2) = \arg\max_{a_2} \hat{Q}_2(h_2, a_2)$$

$$= 2 \cdot 1\{\hat{\beta}_2^T h_2 > 0\} - 1$$

Optimal stage-1 action is given by:

$$\hat{\pi}_1(h_1) = \arg\max_{a_1} \hat{Q}_1(h_1, a_1)$$

$$= 2 \cdot 1\{\hat{\beta}_1^T h_1 > 0\} - 1$$
We want to conduct hypothesis tests regarding parameters in the vector $\beta_1$ (e.g., $H_0 : \beta_1 = 0$), or construct confidence intervals.

Why hypothesis test?

- Reduce the amount of information (e.g., number of $H$-variables on patients) to be collected when implementing $\hat{\pi}$ in future (i.e., variable selection)

- Know when there is insufficient support in the training data to recommend one treatment over another (in such cases, treatment can be chosen by other considerations like cost, familiarity etc.)
Non-regularity

For testing $H_0 : \beta_1 = 0$, we need to know the distribution of $\hat{\beta}_1$.

Note that $\hat{R}_1$ is non-differentiable in the estimators from the stage-2 regression:

$$\hat{R}_1 = R_1 + \max_{a_2} \hat{Q}_2(H_2, a_2) = R_1 + \hat{\alpha}_2^T H_2 + |\hat{\beta}_2^T H_2|$$

This $\hat{R}_1$ is a nuisance parameter that features in $\hat{\beta}_1$.

The asymptotic distribution of $\hat{\beta}_1$ is normal if $P[\beta_2^T H_2 = 0] = 0$ and non-normal if $P[\beta_2^T H_2 = 0] > 0$.

The change between the two asymptotic distributions is abrupt.

Robins (2004) discusses this non-regularity in details.
Whenever true $\beta_2^T H_2 \approx 0$, tests (or, CIs) derived from Taylor series arguments perform poorly (e.g., wrong Type-I error of tests, wrong coverage probability of CI).

Tests obtained by inverting usual bootstrap CIs also perform poorly
- Bootstrap is inconsistent due to non-differentiability (e.g., Shao, 1994)

Let’s illustrate the problem with a toy simulation ...
Simulation Study

- **Generative Model:**
  
  No pre-treatment covariates $X$
  
  \[ A_1, A_2 \in \{-1, 1\} \text{ w.p. } 1/2 \]
  \[ R_1 = 0, \quad R_2|A_1, A_2 \sim N(0, 1) \]

- **Analysis Model:**

  \[ Q_2(H_2, A_2) = \alpha_{20} + \alpha_{21}A_1 + (\beta_{20} + \beta_{21}A_1)A_2; \quad H_2 = (1, A_1)^T \]
  \[ Q_1(H_1, A_1) = \alpha_{10} + \beta_{1}A_1; \quad H_1 = 1 \]

- Consider testing the hypothesis $H_0 : \beta_1 = 0$ using asymptotic normality and bootstrap
Simulation Results

- Non-regular setting: \( P[\beta_2^T H_2 = 0] = P[\beta_2 + \beta_2 A_1 = 0] = 1 \)

- Monte Carlo estimate of Type I error for testing \( H_0 : \beta_1 = 0 \) (simulation size = 1000, nominal Type I error = 0.05)

<table>
<thead>
<tr>
<th>( n )</th>
<th>Asymptotic Normality</th>
<th>Bootstrap</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>0.0250</td>
<td>0.0275</td>
</tr>
<tr>
<td>500</td>
<td>0.0190</td>
<td>0.0235</td>
</tr>
</tbody>
</table>

- This low Type I error translates to low power of the test

We got a problem – need to figure out a valid way to test \( H_0 : \beta_1 = 0 \)
Outline

1. Introduction
   - One-stage Decision Problem
   - Multistage Decision Problem

2. Q-Learning
   - Estimation
   - Inference: A Non-regular Problem

3. “Soft-max” Regularization: A Solution
   - Regularized Estimator
   - Distribution of The Estimator

4. Discussion
“Soft-max” Regularization

- Recall that the stage-1 “pseudo outcome” is:

\[ \hat{R}_1 = R_1 + \hat{\alpha}_2^T H_2 + \max_{a_2} \{ \hat{\beta}_2^T H_2 \} \]

- weighted average over stage-2 actions

- Replace the “hard-max” by a “soft-max”:

\[ \tilde{R}_1 = R_1 + \hat{\alpha}_2^T H_2 + \hat{p} \cdot \hat{\beta}_2^T H_2 + (1 - \hat{p}) \cdot (-\hat{\beta}_2^T H_2) \]

- weighted average over stage-2 actions

- When making a decision at stage-1, assume that at stage-2

\[ a_2 = 1 \]

will be chosen with probability \( \hat{p} \) and

\[ a_2 = -1 \]

with probability \( (1 - \hat{p}) \), instead of the “best” action always.
“Soft-max” Regularization

\[ \tilde{R}_1 = R_1 + \hat{\alpha}_2^T H_2 + \hat{p} \cdot \hat{\beta}_2^T H_2 + (1 - \hat{p}) \cdot (-\hat{\beta}_2^T H_2) \]

- Set \( \hat{p} = F(\lambda \hat{\beta}_2^T H_2) \), where \( \lambda (\geq 0) \) is a tuning parameter and \( F(\cdot) \) is a distribution function with \( F(0) = 1/2 \).

- When \( \lambda \) is large, “soft-max” becomes “hard-max”:

\[ F(\lambda \hat{\beta}_2^T H_2) \cdot \hat{\beta}_2^T H_2 + \left( 1 - F(\lambda \hat{\beta}_2^T H_2) \right) \cdot (-\hat{\beta}_2^T H_2) \approx |\hat{\beta}_2^T H_2| \]
Old wine in a new bottle!

Soft-max is an assimilation of several old ideas ...

- “Soft-max” activation function (logistic distribution function) in neural networks (e.g., Hastie et al., 2001, Ch. 11)

- “Soft-max” choice of actions in $n$-armed bandit problem using Boltzmann distribution (e.g., Sutton and Barto, 1998, Ch. 2)

- Replacing the non-smooth indicator loss function in classification by a smooth surrogate (e.g., exponential loss function in AdaBoost) (e.g., Friedman et al., 2000)

Two things to choose: $F(\cdot)$ and $\lambda$
Choice of $F(\cdot)$

\[ F(u) = \left( -\frac{1}{4}u^3 + \frac{3}{4}u + \frac{1}{2} \right)1_{|u| \leq 1} + 1_{u > 1} \]

This is the distribution function of the Epanechnikov kernel density.
This really means ... 

Regularization just around the point of non-differentiability!

If appropriately “tuned” by $\lambda$, this can take care of the non-regularity.
Stage 2 same as before.

Construct the “regularized” pseudo-outcome:

\[ \tilde{R}_1(\lambda) = R_1 + \hat{\alpha}_2^T H_2 + F(\lambda \hat{\beta}_2^T H_2) \cdot \hat{\beta}_2^T H_2 - \left(1 - F(\lambda \hat{\beta}_2^T H_2)\right) \cdot \hat{\beta}_2^T H_2 \]

Stage 1 Regression:
Use least squares with outcome \( \tilde{R}_1(\lambda) \) and covariates \([H_1, H_1A_1]\) to obtain \( (\hat{\alpha}_1(\lambda), \hat{\beta}_1(\lambda)) \).

Can we use \( \hat{\beta}_1(\lambda) \) to construct a test for \( H_0 : \beta_1 = 0 \)?
- We need to know the distribution of \( \hat{\beta}_1(\lambda) \).
Distribution of $\hat{\beta}_1(\lambda)$

- For every fixed $\lambda \geq 0$, under “mild conditions” on the data, $\hat{\beta}_1(\lambda)$ is approximately normal as $n \to \infty$.

- But for a given training set (i.e., fixed $n$), $\lambda$ must be chosen carefully to enjoy the asymptotic normality
  - If $\lambda$ is too large, we will be back to the non-regular setting!

- This motivates a data-driven choice of $\lambda$, i.e., $\lambda = \lambda_n$. 
If $\lambda_n = o_p(\sqrt{n})$, then under “mild conditions” on the data, $\hat{\beta}_1(\lambda_n)$ is approximately normal as $n \to \infty$.

If $\lambda_n = O_p(\sqrt{n})$, then under “stronger assumptions”, $\hat{\beta}_1(\lambda_n)$ is approximately normal as $n \to \infty$.

So in principle, one can construct a test for $H_0 : \beta_1 = 0$ based on $\hat{\beta}_1(\lambda_n)$. 
The proof of the result is not constructive, so we still have to develop an algorithm to choose $\lambda$ in a data-driven way.

Challenge:
- Depending on the true parameter values, non-regularity causes bias (e.g., Robins, 2004) as well as lightness of tail (illustrated in simulation).
- Difficult to combine these two into a single “objective function”

We are working on a method using bootstrap to find a good $\lambda$. 
Outline

1 Introduction
   - One-stage Decision Problem
   - Multistage Decision Problem

2 Q-Learning
   - Estimation
   - Inference: A Non-regular Problem

3 “Soft-max” Regularization: A Solution
   - Regularized Estimator
   - Distribution of The Estimator

4 Discussion
## Interpretation of $\lambda$

<table>
<thead>
<tr>
<th>Estimator</th>
<th>Estimates stage-1 treatment effect when ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{\beta}_1(0)$</td>
<td>stage-2 treatments are assigned with equal probability, $\lambda = 0$: “defensive” policy</td>
</tr>
<tr>
<td>$\hat{\beta}_1(\infty)$</td>
<td>optimal stage-2 treatment is assigned, $\lambda = \infty$: “optimistic” policy</td>
</tr>
<tr>
<td>$\hat{\beta}_1(\lambda)$</td>
<td>stage-2 treatment $= 1$ is assigned with probability $0 &lt; F(\lambda \beta_2^T H_2) &lt; 1$: “realistic” policy?</td>
</tr>
</tbody>
</table>

So $\lambda$ reflects the “vision” of the policy-maker!
Soft-max estimator can be viewed as a smooth version of James-Stein-type “shrinkage” estimator!
Broadness of the Problem

Non-regularity in other areas of Statistics and Econometrics:

- Inference on *eigenvalues* of a covariance matrix: Beran and Srivastava (1985, 1987)
- And many more ...
Future Work

- Algorithm for finding data-driven $\lambda$ in the current problem

- Analysis of a dataset on smoking cessation to estimate a “good” 2-stage behavioral therapy (treatment policy) to help smokers quit smoking
  - NCI-funded randomized study conducted by researchers at the University of Michigan

- Inference on “value” of a policy: $V(\pi) = E_\pi(R_1 + R_2)$
  - It is important to know if and when two policies are equivalent
  - Test $H_0 : V(\pi_1) = V(\pi_2)$
Contribution

- We developed a regularization scheme to perform hypothesis tests on some non-regular parameter of practical importance.

- “Evidence-based” treatment policies are of great interest in medicine – we tried to address some need of this community.

- The methodology can potentially be used in other multistage decision problems, e.g., market intelligence.

- If we succeed in devising a data-driven $\lambda$, it would be a nice step forward ... (“Hope springs eternal ...”)
Main References


J. Shao (1994). Bootstrap sample size in nonregular cases. 

“Mild” Conditions:
- Data trajectories are i.i.d.
- Design matrix of each regression is of full rank.
- Rewards have finite second moments.
- $\|H\|$’s are a.s. bounded.

“Stronger” Assumption:
(Additionally) $\|H_2\|$ has finite support, say $\mathcal{H}$, and

$$\inf_{h \in \mathcal{H}} \frac{\beta^T h}{\|h\|} > 0$$