C-learning: Estimating Optimal Dynamic Treatment Regimes from a Classification Perspective

Min Zhang
Department of Biostatistics
University of Michigan, Ann Arbor

Joint work with Baqun Zhang
Shanghai University of Finance and Economics, China
Why Optimal Dynamic Treatment Regimes?

**Goal:** the *right treatment* for the *right patient* at the *right time*  
(Personalized medicine / Precision medicine)

**In contrast to:** “one-size-fits all” and “once-and-for-all” approach
Why Optimal Dynamic Treatment Regimes?

Patient heterogeneity:

- Demographic characteristics
- Physiological characteristics
- Medical history, concomitant conditions
- Genetic/genomic characteristics
Why Optimal Dynamic Treatment Regimes?

Patient heterogeneity:

- Demographic characteristics
- Physiological characteristics
- Medical history, concomitant conditions
- Genetic/genomic characteristics

Clinical practice: Treatment of chronic diseases/disorders is an ongoing process and clinicians manage a patient’s illness over the course of a patient’s disease

- Clinicians make (a series of) treatment decision(s) over the course
- At key decision points
- Multiple treatment options at each
- Accrued information on the patient
Dynamic Treatment Regimes

Formalizing precision medicine:

- A set of decision rules for a sequence of decision points at which decisions on treatment are made
  - Decision point $k = 1, \ldots, K$
  - Treatment options at $k$th stage: $a_k \in \mathcal{A}_k = \{0, 1\}$
  - Treatment history up to $k$: $\bar{a}_k = (a_1, \ldots, a_k)$

- At each point, the next step of treatment is determined by the decision rule according to information (variables) on the patient up to that point
  - Covariate information between decision $k - 1$ and $k$: $x_k$
  - Covariate history up to $k$: $\bar{x}_k = (x_1, \ldots, x_k)$
  - Covariate and treatment history available before decision $k$: $l_k = (\bar{x}_k, \bar{a}_{k-1})$
  - Dynamic treatment regime: $g = (g_1, \ldots, g_K)$, where $g_k(l_k) \in \mathcal{A}_k$
Dynamic Treatment Regimes

Stage 1:

\[ g_1(\text{age, PR}) \in \text{Two treatment options}\{ c_1, c_2 \}, \text{ coded as } \{0, 1\} \]

Decision rule of linear form:

\[ g_1(\text{age, PR}) = I\{\text{age} > 60 - 8.7 \log(\text{PR})\} \]

Decision rule of a tree form:

\[ g_1(\text{age, PR}) = I(\text{age} < 50 \text{ and PR} < 10) \]

Stage 2:

\[ g_2(\text{age, PR, treatment at decision 1, responder status...}) \]

\[ \in \text{Four options } \{m_1, m_2, s_1, s_2\} \]
**Optimal dynamic treatment regime**

**Potential outcomes framework:**

- Potential outcome associated with any regime \( g = (g_1, \ldots, g_K) \in \mathcal{G} \):
  \[
  Y^*(g)
  \]
  - The outcome that would result if the subject followed \( g \)

**Optimal dynamic treatment regime:**

- \( g^{opt} = (g_1^{opt}, \ldots, g_K^{opt}) \in \mathcal{G} \)
  \[
  E\{Y^*(g^{opt})\} \geq E\{Y^*(g)\} \quad \text{for all} \quad g = (g_1, \ldots, g_K) \in \mathcal{G}
  \]
  - The one that would yield maximum expected outcome if were followed by all patients in the population
Notation

**Observed data:**

\[(\bar{A}_{K_i}, \bar{X}_{K_i}, Y_i), i = 1, \ldots, n\]

- Observed treatment at \(k\)th stage, \(A_k\); Observed *treatment history* up to decision \(k\), \(\bar{A}_k = (A_1, \ldots, A_k)\)

- \(X_k\) is the covariate information observed between decision \(k - 1\) and \(k\); the observed *covariate history* up to \(k\), \(\bar{X}_k = (X_1, \ldots, X_k)\)

- Covariate and treatment history available before decision \(k\), \(L_k = (\bar{X}_k, \bar{A}_{k-1})\)

**Assumptions**

- Consistency: \(Y = Y^*(\bar{A}_K)\) and \(X_k = X^*_k(\bar{A}_{k-1})\)

- A patient’s covariates and outcome are not affected by treatments received by other patients

- No unmeasured confounders assumption
Notation

Single decision point setting

- **Observed data:** \((X_i, A_i, Y_i), i = 1, \ldots, n\)
  - \(X\) covariates
  - \(A\) treatment
  - \(Y\) outcome
- **Treatment options:** \(a = 0, 1\)
- **Decision rule:** \(g(X) = 0, 1\)
- **Potential outcomes:** \(Y^*(g) = Y^*(1)g(X) + Y^*(0)\{1 - g(X)\}\)
- **Optimal treatment regime:** \(g^{opt}(X) = \arg \max_{g \in G} E\{Y^*(g)\}\)
  - the one yielding the largest expected potential outcomes among all regimes
Existing Methods

\[ g^{opt}(X) = I\{\mu(1, X) > \mu(0, X)\} = I\{C(X) > 0\} \]

where \( \mu(a, X) = E(Y|A = a, X) \), \( C(X) = \mu(1, X) - \mu(0, X) \)

**Outcome regression-based methods**

- Parametric regression model for \( \mu(A, X) = E(Y|A, X) \), say \( \mu(A, X; \beta) \)
  - \( \hat{g}^{opt}(X) = I\{\mu(1, X, \hat{\beta}) > \mu(0, X, \hat{\beta})\} \)
  - Extension to multiple decision point setting leads to **Q-learning** *(backward induction)*

- Semiparametric regression model for \( \mu(A, X) = h_1(X) + AC(X; \psi) \)
  - \( h_1(X) \) is unspecified

  - \( \hat{g}^{opt}(X) = I\{C(X; \hat{\psi}) > 0\} \)
  - Extension to multiple decision point setting leads to **A-learning** *(backward induction)*
Existing methods

Backward induction

- **Q-function:** \( Q_K(\bar{x}_K, \bar{a}_{K-1}, a_K) = E(Y \mid \bar{X}_K = \bar{x}_K, \bar{A}_K = \bar{a}_K) \)

- **\( g_{opt}^K(\bar{x}_K, \bar{a}_{K-1}) = \arg \max_{a_K \in \Phi_K(\bar{x}_K, \bar{a}_{K-1})} Q_K(\bar{x}_K, \bar{a}_{K-1}, a_K) \)**

- Recursively define
  - **Value function** as \( V_k(\bar{x}_k, \bar{a}_{k-1}) = \max_{a_k \in A_k} Q_k(\bar{x}_k, \bar{a}_{k-1}, a_k) \) for \( k = K, \ldots, 2 \), with \( \bar{a}_0 \) being null,
  - **Q-functions** as \( Q_k(\bar{x}_k, \bar{a}_{k-1}, a_k) = E\{V_{k+1}(\bar{x}_k, X_{k+1}, \bar{a}_k) \mid \bar{X}_k = \bar{x}_k, \bar{A}_k = \bar{a}_k\} \) for \( k = K - 1, \ldots, 1 \)

- The optimal decision rule at the \( k \)-th point satisfies \( g_{opt}^k(\bar{x}_k, \bar{a}_{k-1}) = \arg \max_{a_k \in A_k} Q_k(\bar{x}_k, \bar{a}_{k-1}, a_k) \)
Existing Methods

\[ g^{opt}(X) = \arg \max_{g \in G} E\{Y^*(g)\} \]

Direct optimization methods

  - For each regime \( g \in G_\eta \), estimate \( E\{Y^*(g)\} \) using augmented inverse probability weighted estimator (AIPWE)
    \[
    AIPWE(\eta) = n^{-1} \sum_{i=1}^{n} \left\{ \frac{C_{\eta,i} Y_i}{\pi_c(X_i; \hat{\gamma})} - \frac{C_{\eta,i} - \pi_c(X_i; \eta, \hat{\gamma})}{\pi_c(X_i; \hat{\gamma})} m(X_i; \eta, \hat{\beta}) \right\},
    \]
  - \( C_{\eta,i} = I\{A_i = g(X_i, \eta)\} \);
    \[
    \pi_c(X_i, \gamma) = A_i \pi(X_i, \gamma_i) + (1-A_i)\{1 - \pi(X_i, \gamma_i)\}
    \]
  - Double robust property of AIPWE
  - Requires to prespecify \( G_\eta \)
  - Extension to multiple decision point setting: Zhang, et al (2013, Biometrika) (Monotone coasening/missing)
Existing Methods

- **Outcome weighted learning** (OWL; Zhao et al, 2012, JASA)
  - Estimate $E\{Y^*(g)\}$ using inverse probability weighted estimator (IPWE)

$$IPWE = n^{-1} \sum_{i=1}^{n} \left\{ \frac{I\{A_i = g(X_i)\}Y_i}{A_i\pi(X_i, \gamma_i) + (1 - A_i)\{1 - \pi(X_i, \gamma_i)\}} \right\},$$

- Equivalent to minimizing $\sum_{i=1}^{n} \left\{ \frac{I\{A_i \neq g(X_i)\}Y_i}{A_i\pi(X_i, \gamma_i) + (1 - A_i)\{1 - \pi(X_i, \gamma_i)\}} \right\}$

- $I\{A_i \neq g(X_i)\}$ is viewed as a **zero-one loss in classification**, blue part as weight when $Y$ is positive

- No use of outcome regression model (not taking advantage of patient characteristics)

- Extension to multiple decision point setting: BOWL, SOWL (Zhao, et al, 2015, JASA) *(Monotone coarsening/missing)*
**Theorem 1:** Let $g^* = (g^*_1, \ldots, g^*_K)$, be a treatment regime that satisfies

$$g^*_k(L_k) = \arg \min_{g_k \in G_k} E[|C_k(L_k)|I\{Z_k \neq g_k(L_k)\}]$$

where $Z_k = I\{C_k(L_k) > 0\}, k = K, \ldots, 1$, then $g^*$ is the optimal dynamic treatment regime.

- $Q_K(\bar{x}_K, \bar{a}_K) = E(Y | \bar{X}_K = \bar{x}_K, \bar{A}_K = \bar{a}_K)$
- $V_k(\bar{x}_k, \bar{a}_{k-1}) = \max_{a_k \in A_k} Q_k(\bar{x}_k, \bar{a}_{k-1}, a_k)$
- $Q_k(L_k, a_k) = E\{V_{k+1}(L_{k+1}) | L_k, a_k\}$
- $C_k(L_k) = Q_k(L_k, 1) - Q_k(L_k, 0)$
C-learning

\[ g_{k}^{\text{opt}}(L_k) = \arg \min_{g_k \in G_k} E[|C_k(L_k)| I\{Z_k \neq g_k(L_k)\}], \quad Z_k = I\{C_k(L_k) > 0\} \]

**Intuition:** Consider the last stage,

- Directly optimizing estimate of \( E\{Y^*(\bar{A}_{K-1}, g_K)\} \) over a class of regimes
- Retain only part of \( E\{Y^*(\bar{A}_{K-1}, g_K)\} \) relevant for decision making
- \( g_{K}^{\text{opt}}(X) = \arg \max_{g_K \in G_K} E\{Y^*(\bar{A}_{K-1}, g_K)\} \)
  \[ = \arg \max_{g_K \in G_K} [E\{g_K(L_K)C_K(L_K)\} + Q_K(L_K, 0)] \]
- Decompose \( C_K(L_K) \) into magnitude \( |C_K(L_K)| \) and sign \( I\{C_K(L_K) > 0\} \)
- \( g(L_K)C_K(L_K) = Z_K|C_K(L_K)| - |C_K(L_K)|I\{Z_K \neq g(L_K)\} \)
- \( g_{K}^{\text{opt}}(X) = \arg \min_{g_K \in G_K} E[|C_K(L_K)| I\{Z_K \neq g(L_K)\}] \)
- Classification perspective leads to powerful and flexible learning algorithms
C-learning

Proposition 1: The value functions satisfy the following condition:

\[ E[V_{k+1}(L_{k+1}) + Q_k(L_k, 1) - Q_k(L_k, 0)] \{g_{k}^{opt}(L_k) - A_k\} | L_k] = V_k(L_k), \]

\( k = K, \ldots, 1, V_{K+1} \equiv Y, \) where \( g_{k}^{opt} \) is the optimal decision rule at stage \( k \).
C-learning

At stage $K$:

- $g_{K}^{opt}(L_K) = \arg\min_{g_K \in G} E[|C_K(L_K)| \{Z_K \neq g_K(L_K)\}]$

- Considering data $(Y_i, L_{Ki}, A_{Ki})$, estimate $C_K(L_{Ki})$ by the AIPWE estimate

$$
\hat{C}_K(L_{Ki}) = \frac{A_{Ki}}{\hat{\pi}_K(L_{Ki})} Y_i - \frac{A_{Ki} - \hat{\pi}_K(L_{Ki})}{\hat{\pi}_K(L_{Ki})} \hat{Q}_K(L_{Ki}, 1)
$$

- Weighted classification:

$$
\hat{g}_{C,K}^{opt} = \arg\min_{g_K \in G} \sum_{i=1}^{n} [\hat{W}_{Ki} \{\hat{Z}_{Ki} \neq g_K(L_{Ki})\}]
$$

  - Class label: $\hat{Z}_{Ki} = I\{\hat{C}_K(L_{Ki}) > 0\}$
  - Weight: $\hat{W}_{Ki} = |\hat{C}_K(L_{Ki})|$

using by existing optimization/classification techniques
C-learning

After obtaining $\hat{g}^{opt}_{C,K}$, the C-learning moves backward till the first stage

At stage $k$: $k = K - 1, \ldots, 1$,

- $g^{opt}_k(L_k) = \arg\min_{g_k \in G_k} E[|C_k(L_k)|\{Z_k \neq g_k(L_k)\}]$

- $C_k(L_k) = Q_k(L_k, 1) - Q_k(L_k, 0)$, and $Q_k(L_k, a_k) = E\{V_{k+1}(L_{k+1})|L_k, a_k\}$

- Denoting $\tilde{V}_{(K+1)i} = Y_i$, estimate $V_k(L_{ki})$ recursively by
  $\tilde{V}_{ki} \equiv \tilde{V}_k(L_{ki}) = \tilde{V}_{(k+1)i} + \{\hat{Q}_k(L_{ki}, 1) - \hat{Q}_k(L_{ki}, 0)\}\{\hat{g}^{opt}_{C,k}(L_{ki}) - A_{ki}\}$,

- Treating $(\tilde{V}_{k+1,i}, L_{ki}, A_{ki})$ as “data”, estimate $C_k(L_{ki})$ by AIPWE estimate $\hat{C}_k\{L_{ki}\}$

- $\hat{g}^{opt}_{C,k} = \arg\min_{g_k \in G_k} \sum_{i=1}^{n}[\hat{W}_{ki}\{\hat{Z}_{ki} \neq g_k(L_{ki})\}]$, where
  - Class label: $\hat{Z}_{ki} = I\{\hat{C}_k(L_{ki}) > 0\}$
  - Weight: $\hat{W}_{ki} = |\hat{C}_k(L_{ki})|$
C-learning

Algorithm:

1. At stage $K$, based on data $(Y_i, L_{Ki}, A_{Ki}), i = 1, \ldots, n$,
   1.1 Build model for $P(A_K = 1|L_K)$ to estimate propensity score, $\hat{\pi}_K(L_{Ki})$
   1.2 Build model for $Q_K(L_K, A_K) = E(Y|L_K, A_K)$ and obtain estimate of $Q_K(L_K, a_K), a_K = 0, 1$, for each subject, $\hat{Q}_K(L_{Ki}, a_K)$
   1.3 Estimate the contrast function $C_K(L_{Ki})$ for each subject by the AIPWE $\hat{C}_K(L_{Ki})$
   1.4 Estimate $g_{K}^{opt}$ by minimizing a weighted misclassification error
      * classification data set $(\hat{Z}_i, \hat{W}_i, L_i)$
      * $\hat{Z}_i$ is class label: $I(\hat{C}_K(L_{Ki}) > 0)$
      * $\hat{W}_i$ is weight: $|\hat{C}_K(L_{Ki})|$
      * $L_i$ covariates and treatment history
   1.5 Estimate $\hat{V}_{K,i}$

2. Repeat 1.1-1.4 for stage $k = K - 1, \ldots, 1$ sequentially, based on “data” $(\hat{V}_{k+1,i}, L_{ki}, A_{ki}), i = 1, \ldots, n$, to obtain estimate of $g_{k}^{opt}$
C-learning and other methods

C-learning and Zhang et al. (2013)

- Similarity:
  - Direct optimization
  - AIPWE

- Difference:
  - Weighted classification perspective
  - Sequential optimization vs. simultaneous optimization across stages
C-learning and BOWL:

- AIPWE vs. IPWE

- Classification perspective
  - BOWL: misclassify if $A_i \neq g(X_i)$
    - Undesirable feature: estimated regime tries to keep observed treatment assignments (Zhou et al., 2015, JASA)
    - Dependent on the IPWE of $E\{Y^*(g)\}$
  - C-learning: misclassify if $I\{C(X_i) > 0\} \neq g(X_i)$
    - Theorem 1 is a general result and does not depend on estimator of $E\{Y^*(g)\}$
    - Meaningful and consistent with the goal of optimizing treatment decisions

- Backward sequential optimization
  - BOWL loses sample size geometrically with stages
C-learning: high dimensionality

**Regimes of linear form:** Variable selection

- Target selecting prescriptive variables as opposed to predictive variables
- Weighted misclassification error rate corresponding to \( \{X_{j1}, \ldots, X_{jm}\} \) as

\[
err(X_{j1}, \ldots, X_{jm}) = \min_{\beta} \sum_{i=1}^{n} \hat{W}_i I\{\hat{Z}_i \neq I(\beta_0 + \beta_1 X_{j1} + \ldots + \beta_m X_{jm} > 0)\},
\]

- **Forward Minimal Misclassification Error Rate (ForMMER) Selection** (Zhang and Zhang, 2018)
  - First selected variable: \( X_{j1} = \arg\min_{X_j \in (x_1, \ldots, x_p)} err(X_j) \)
  - \( m \)-th selected variable: \( X_{jm} = \arg\min_{X_j \in \mathcal{F}\setminus\mathcal{S}_{(m-1)}} err(\mathcal{S}_{(m-1)}, X_j) \)
  - \( \mathcal{S}_{(m)} = \{X_{j1}, \ldots, X_{jm}\} \): set of selected variables up to steps \( m \)-th step

**Regimes of the form of a decision tree**

- Existing classification algorithms capable of handling high dimensional set of covariates
- e.g., CART
C-learning: Simulations (scenario I)

Simulation setting I:

- Adopted from Zhao et al. (2015; JASA)

- Treatments $A_1, A_2$ and $A_3$ are generated from $\{1, 0\}$ with equal probability 0.5.

- $X_{1,1}, X_{1,2}, X_{1,3}$ are generated from $N(45, 15^2)$. $X_2$ is generated according to $X_2 \sim N(1.5X_{1,1}, 10^2)$ and $X_3$ is generated according to $X_3 \sim N(0.5X_2, 10^2)$

- Outcome: $Y = \mu(\bar{A}_3, \bar{X}_3) + \epsilon$ for $\epsilon$ standard normal and

  $\mu(\bar{A}_3, \bar{X}_3) = 20 - |0.6X_{1,1} - 40|(A_1 - g_1^{opt})^2 - |0.8X_2 - 60|(A_2 - g_2^{opt})^2 - |1.4X_3 - 40|(A_3 - g_3^{opt})^2$, where

  - $g_1^{opt} = I(X_{1,1} - 30 > 0)$
  - $g_2^{opt} = I(X_2 - 40 > 0)$
  - $g_3^{opt} = I(X_3 - 40 > 0)$

- The optimal treatment decision rule at each stage depends only on a single covariate
### C-learning: Simulations (scenario I)

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOWL</td>
<td>$10.84(1.85)$</td>
<td>$12.13(1.54)$</td>
<td>$13.02(1.36)$</td>
</tr>
<tr>
<td>Q-learning</td>
<td>$12.49(1.83)$</td>
<td>$12.76(1.46)$</td>
<td>$13.05(1.14)$</td>
</tr>
<tr>
<td>Zhang et al. (2013)</td>
<td>$13.25(2.12)$</td>
<td>$15.08(1.46)$</td>
<td>$16.28(1.01)$</td>
</tr>
<tr>
<td>C-learning</td>
<td>$17.27(0.97)$</td>
<td>$18.52(0.74)$</td>
<td>$19.37(0.41)$</td>
</tr>
</tbody>
</table>

- $E\{Y^*(g^{opt})\} = 20$
- Specification of Q-functions the same as in Zhao, et al (2015, JASA)
- AIPWE in Zhang et al. (2013) and C-learning use the same Q-functions for augmentation terms
- In method of Zhang, et al (2013), consider all available covariates at each stage in parameterizing regimes
- In C-learning, the optimization step is carried out by a genetic algorithm (R package Rgenoud)
### C-learning: Simulations (scenario I)

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOWL</td>
<td>10.84(1.85)</td>
<td>12.13(1.54)</td>
<td>13.02(1.36)</td>
</tr>
<tr>
<td>Q-learning</td>
<td>12.49(1.83)</td>
<td>12.76(1.46)</td>
<td>13.05(1.14)</td>
</tr>
<tr>
<td>Zhang et al. (2013)</td>
<td>13.25(2.12)</td>
<td>15.08(1.46)</td>
<td>16.28(1.01)</td>
</tr>
<tr>
<td>C-learning</td>
<td>17.27(0.97)</td>
<td>18.52(0.74)</td>
<td>19.37(0.41)</td>
</tr>
</tbody>
</table>

- Zhang et al. (2013) and C-learning outperform Q-learning, even though the same Q-functions are used: **direct optimization vs. outcome regression based method**
- C-learning outperforms Zhang et al. (2013), even though both are based on the same AIPWE: **sequential vs. simultaneous optimization**
- C-learning outperforms BOWL: **classification perspective; AIPWE vs. IPWE; sequential optimization**
C-learning: Simulations (scenario I)

Figure 1: Classification data set (n=200)

- C-learning vs. BOWL: classification perspective; AIPWE vs. IPWE; sequential optimization
Simulation setting II:

- Increase the dimension of covariates at each stage so that the total number is 50.

- 40 baseline covariates $X_{1,1}, \ldots, X_{1,40}$ are generated from $N(45, 15^2)$. At stage 2, $X_{2,j} \sim N(1.5X_{1,j}, 10^2)$, $j = 1, \ldots, 5$. At stage 3, $X_{3,j} \sim N(0.5X_{2,j}, 10^2)$, $j = 1, \ldots, 5$.

- The outcome was generated as $Y = \mu(\bar{A}_3, \bar{X}_3) + \epsilon$ for $\epsilon$ standard normal and

  $\mu(\bar{A}_3, \bar{X}_3) = 20 - |0.6X_{1,1} - 40|(A_1 - g_{1opt})^2 - |0.8X_{2,1} - 60|(A_2 - g_{2opt})^2 - |1.4X_{3,1} - 40|(A_3 - g_{3opt})^2$, where

  - $g_{1opt} = I(X_{1,1} - X_{1,2} > 0)$
  - $g_{2opt} = I(X_{2,1} - X_{2,2} > 0)$
  - $g_{3opt} = I(X_{3,1} - X_{3,2} > 0)$

- Otherwise similar to scenario 1, except that the optimal decision rule at each stage a little bit more complicated in that it depends on linear combination of two covariates.
## C-learning: Simulations (scenario II)

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOWL</td>
<td>3.38(1.62)</td>
<td>5.93(1.37)</td>
<td>7.79(1.10)</td>
</tr>
<tr>
<td>BOWL$^\dagger$</td>
<td>14.76(1.74)</td>
<td>15.43(1.38)</td>
<td>15.74(1.12)</td>
</tr>
<tr>
<td>Q-learning$^\dagger$</td>
<td>14.01(1.05)</td>
<td>13.94(0.78)</td>
<td>13.78(0.56)</td>
</tr>
<tr>
<td>Zhang et al.(2013)$^\dagger$</td>
<td>17.98(1.42)</td>
<td>18.83(0.87)</td>
<td>19.35(0.45)</td>
</tr>
<tr>
<td>C-learning-Q</td>
<td>17.70(1.75)</td>
<td>19.45(0.61)</td>
<td>19.78(0.22)</td>
</tr>
<tr>
<td>C-learning-RF</td>
<td>16.59(2.14)</td>
<td>19.21(0.80)</td>
<td>19.75(0.14)</td>
</tr>
</tbody>
</table>

- In specification of Q-functions, use only “important covariates”
- Methods$^\dagger$ consider only regimes constructed by “important covariates” (not feasible in practice)
- In C-learning-RF, the augmentation term is fit by random forest
- In C-learning, use data-driven way to choose important covariates from the high dimension of covariates and search the optimal regime among all regimes of the linear form
C-learning: Simulations *(scenario II)*

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOWL</td>
<td>3.38(1.62)</td>
<td>5.93(1.37)</td>
<td>7.79(1.10)</td>
</tr>
<tr>
<td>BOWL†</td>
<td>14.76(1.74)</td>
<td>15.43(1.38)</td>
<td>15.74(1.12)</td>
</tr>
<tr>
<td>Q-learning†</td>
<td>14.01(1.05)</td>
<td>13.94(0.78)</td>
<td>13.78(0.56)</td>
</tr>
<tr>
<td>Zhang et al.(2013)†</td>
<td>17.98(1.42)</td>
<td>18.83(0.87)</td>
<td>19.35(0.45)</td>
</tr>
<tr>
<td>C-learning-Q</td>
<td>17.70(1.75)</td>
<td>19.45(0.61)</td>
<td>19.78(0.22)</td>
</tr>
<tr>
<td>C-learning-RF</td>
<td>16.59(2.14)</td>
<td>19.21(0.80)</td>
<td>19.75(0.14)</td>
</tr>
</tbody>
</table>

- BOWL has considerable worse performance when the dimension of covariates is high.
- C-learning-Q outperforms Q-learning, even though they both use use the same outcome-regression model: direct optimization vs. outcome regression based method.
- C-learning is comparable to Zhang et al. (2013)†, even though Zhang et al. (2013)† considers a much smaller class of regimes: sequential vs. simultaneous optimization.
C-learning: Simulations (scenario III)

Simulation setting III:

- Data generating scenario is the same as scenario II except that
  \[ g_{1}^{opt} = I(X_{1,1} > 40)I(X_{1,2} < 60) \]
  \[ g_{2}^{opt} = I(X_{2,1} > 60)I(X_{2,2} < 90) \]
  \[ g_{3}^{opt} = I(X_{3,1} > 30)I(X_{3,2} < 50) \]

- The optimal decision rule at each stage is of the form of a decision tree
## C-learning: Simulations (scenario III)

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOWL</td>
<td>3.01(1.63)</td>
<td>5.02(1.42)</td>
<td>6.73(1.15)</td>
</tr>
<tr>
<td>BOWL†</td>
<td>12.55(1.28)</td>
<td>12.91(0.95)</td>
<td>13.12(0.72)</td>
</tr>
<tr>
<td>Q-learning†</td>
<td>13.12(0.45)</td>
<td>13.08(0.35)</td>
<td>13.07(0.23)</td>
</tr>
<tr>
<td>Zhang et al.(2013)†</td>
<td>17.02(1.25)</td>
<td>18.02(0.90)</td>
<td>18.71(0.63)</td>
</tr>
<tr>
<td>C-learning-Q</td>
<td>17.44(1.29)</td>
<td>18.91(0.73)</td>
<td>19.52(0.32)</td>
</tr>
<tr>
<td>C-learning-RF</td>
<td>16.94(1.48)</td>
<td>18.92(0.63)</td>
<td>19.61(0.24)</td>
</tr>
</tbody>
</table>

- **Methods†** limit the search among regimes constructed using only “important covariates” (not feasible in practice)
- In C-learning-RF, the augmentation term is fit by random forest
- In C-learning-Q, the augmentation term uses the same model as in Q-learning
- C-learning uses CART to choose important covariates from the high dimensional set of covariates and optimizes regimes across all regimes of the form of a decision tree
### C-learning: Simulations (scenario III)

<table>
<thead>
<tr>
<th>Estimator</th>
<th>n=200</th>
<th>n=400</th>
<th>n=800</th>
</tr>
</thead>
<tbody>
<tr>
<td>E((\hat{g}^{opt}))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOWL</td>
<td>3.01(1.63)</td>
<td>5.02(1.42)</td>
<td>6.73(1.15)</td>
</tr>
<tr>
<td>BOWL(^\dagger)</td>
<td>12.55(1.28)</td>
<td>12.91(0.95)</td>
<td>13.12(0.72)</td>
</tr>
<tr>
<td>Q-learning(^\dagger)</td>
<td>13.12(0.45)</td>
<td>13.08(0.35)</td>
<td>13.07(0.23)</td>
</tr>
<tr>
<td>Zhang et al. (2013)(^\dagger)</td>
<td>17.02(1.25)</td>
<td>18.02(0.90)</td>
<td>18.71(0.63)</td>
</tr>
<tr>
<td>C-learning-Q</td>
<td>17.44(1.29)</td>
<td>18.91(0.73)</td>
<td>19.52(0.32)</td>
</tr>
<tr>
<td>C-learning-RF</td>
<td>16.94(1.48)</td>
<td>18.92(0.63)</td>
<td>19.61(0.24)</td>
</tr>
</tbody>
</table>

- BOWL has considerably worse performance when the dimension of covariates is high.
- C-learning-Q outperforms Q-learning, even though they both use the same outcome-regression model: direct optimization vs. outcome regression based method.
- C-learning is comparable to Zhang et al. (2013)\(^\dagger\), even though Zhang et al. (2013)\(^\dagger\) considers a much smaller class of regimes: sequential vs. simultaneous optimization.
- C-learning-RF is data-driven.

\(^\dagger\) Indicates significantly different performance.
### C-learning: Simulations (variable selection)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\rho$</th>
<th>Size</th>
<th>TP</th>
<th>ER</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>n=200</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAS (Fan, Lu and Song, 2016)</td>
<td>0.2</td>
<td>14.49 (2.07)</td>
<td>0.72 (0.60)</td>
<td>43.9 (5.3)</td>
<td>71.8 (5.6)</td>
</tr>
<tr>
<td>C-learning ForMMER-AIPWE</td>
<td>3.99 (0.95)</td>
<td>1.97 (0.16)</td>
<td><strong>11.8</strong> (5.1)</td>
<td><strong>97.3</strong> (2.5)</td>
<td></td>
</tr>
<tr>
<td>C-learning ForMMER-reg</td>
<td>4.39 (1.04)</td>
<td><strong>1.98</strong> (0.15)</td>
<td>12.5 (6.3)</td>
<td>96.7 (2.9)</td>
<td></td>
</tr>
<tr>
<td><strong>n=800</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAS</td>
<td>0.8</td>
<td>8.55 (2.03)</td>
<td>0.96 (0.35)</td>
<td>21.3 (6.3)</td>
<td>90.9 (5.1)</td>
</tr>
<tr>
<td>C-learning ForMMER-AIPWE</td>
<td>3.57 (1.00)</td>
<td>1.81 (0.39)</td>
<td>7.7 (2.9)</td>
<td>98.7 (0.9)</td>
<td></td>
</tr>
<tr>
<td>C-learning ForMMER-reg</td>
<td>3.84 (1.00)</td>
<td><strong>1.99</strong> (0.11)</td>
<td><strong>4.4</strong> (1.8)</td>
<td><strong>99.6</strong> (0.4)</td>
<td></td>
</tr>
</tbody>
</table>

- Single decision point setting, $K = 1$.
- A total of 500 covariates.
- **Size**: number of selected prescriptive variables; **TP**: number of true positive (important) prescriptive variables; **ER**: error rate of the treatment decision; **VR** (value ratio): ratio of the value of the estimated regime relative to that of the true optimal regime. Numbers in parenthesis are Monte Carlo standard deviations.
Discussion

- The proposed C-learning is a **powerful and flexible** learning method
  - Existing powerful and off-the-shelf **classification/optimization** techniques can be used to optimize the decision rules (eg, CART, the genetic algorithm discussed by Goldberg, 1989)
  - Existing **model building techniques** (parametric and nonparametric, eg, random forest) can be used to best estimate the **Q-function**
  - Accommodate variable selection targeting **selecting prescriptive variables** (variables relevant for treatment decision making) as opposed to predictive variables
  - Each step (estimate Q-function and optimization) targets the **right goal**
    - Q-function: best model outcomes and make predictions
    - Optimization: Optimize decision rules
    - However, as pointed out by Murphy (2005), there is a **mismatch** between **outcome-regression based methods** (eg, Q- and A-learning) and the goal of optimizing decision rules


