Policy Evaluation and Optimal Treatment Regime Estimation with Continuous Treatments

Hengrui Cai

Department of Statistics, North Carolina State University

Aug 27th, 2021

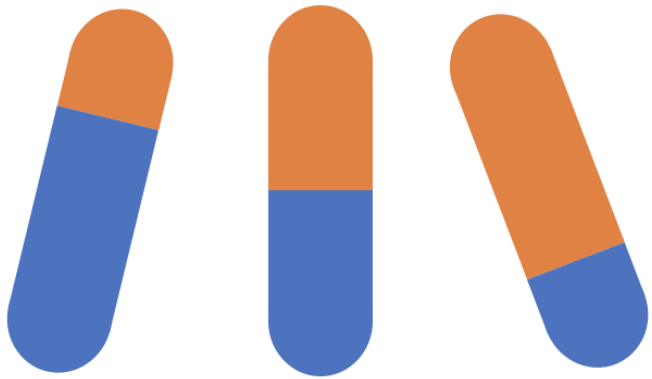
Outline

1. Personalized Decision Making
2. Challenges (1): Policy Optimization in Continuous Treatment Domains
3. Jump Interval-Learning for Individualized Decision Making with Continuous Treatments
4. Challenges (2): Policy Evaluation in Continuous Treatment Domains
5. Deep Jump Learning for Off-Policy Evaluation in Continuous Treatment Settings

Personalized Decision Making

Personalized Decision Making

Developing an individualized treatment rule for patients to optimize expected clinical outcomes of interest [Medicine];

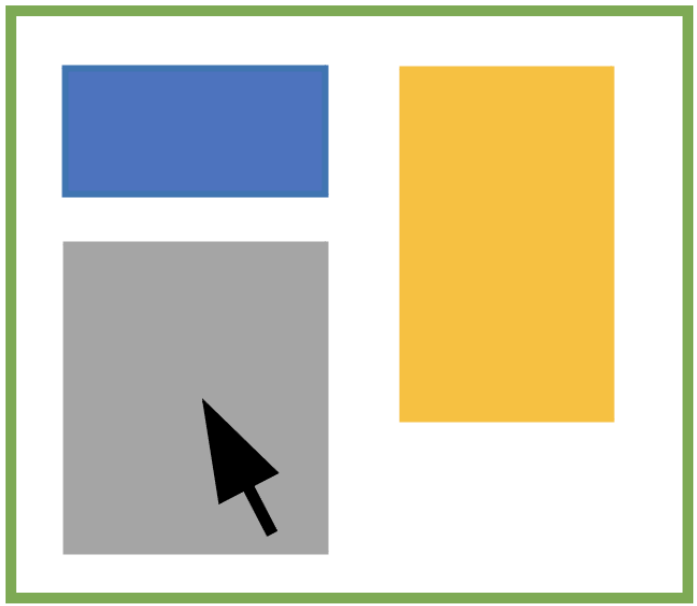


Personalized Decision Making

Offering customized incentives to increase sales and level of engagement [Economics];

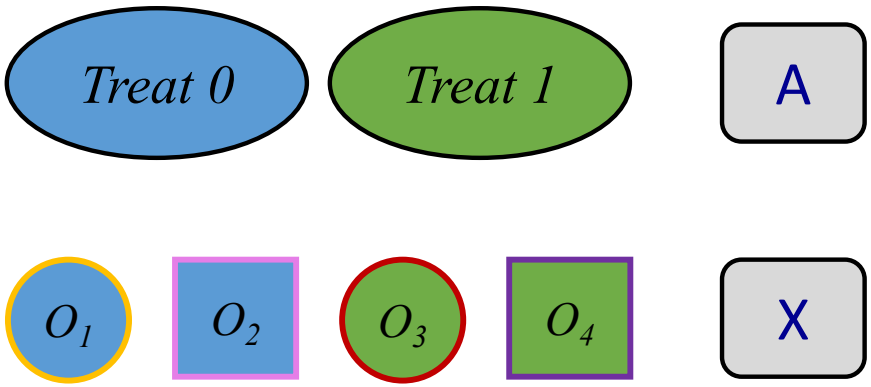
Personalized Decision Making

Designing a personalized advertisement recommendation system to raise the click rates [Marketing].



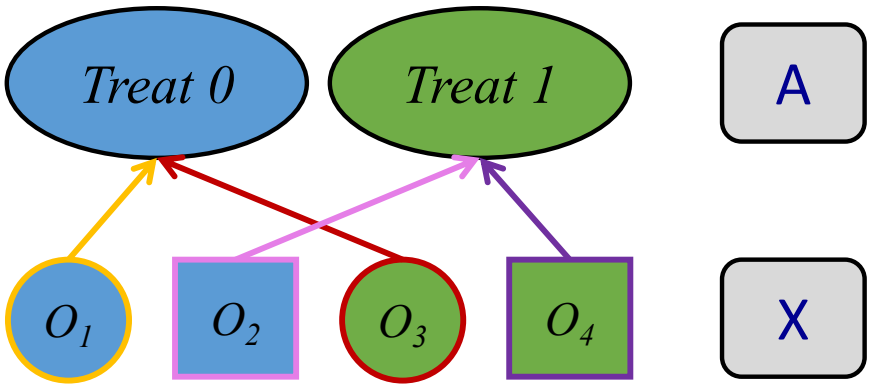
General Setup for Personalized Decision Making

Consider assigning individuals with covariates X to some treatments A.



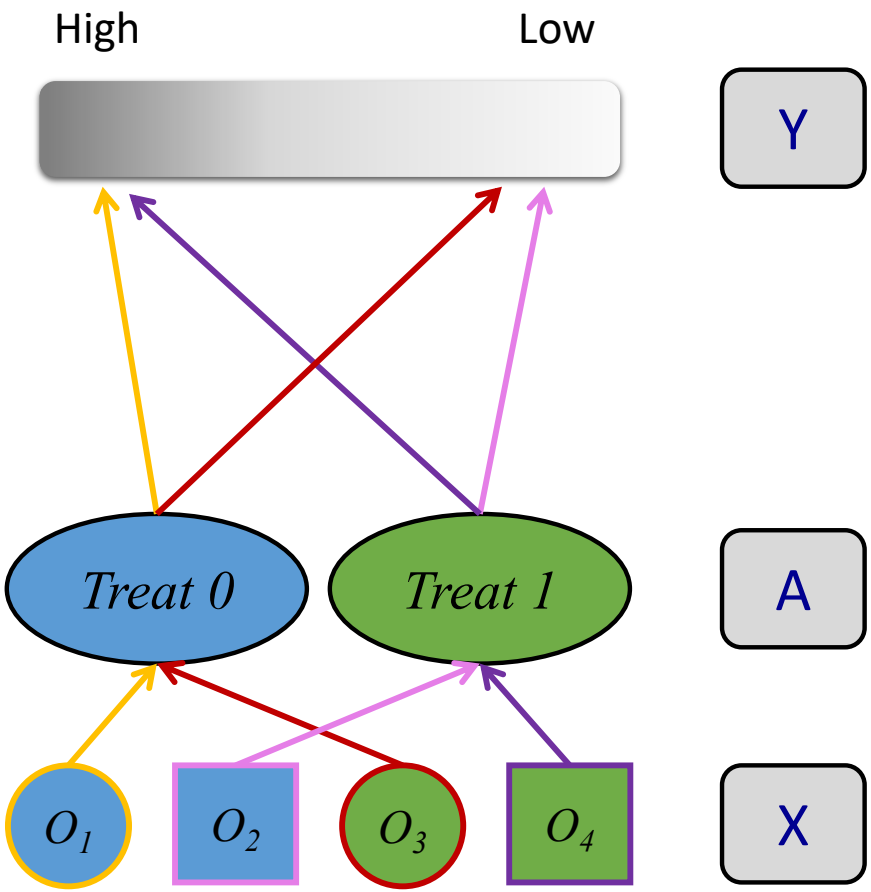
General Setup for Personalized Decision Making

Treatments may be assigned randomly or following some clinical advices.



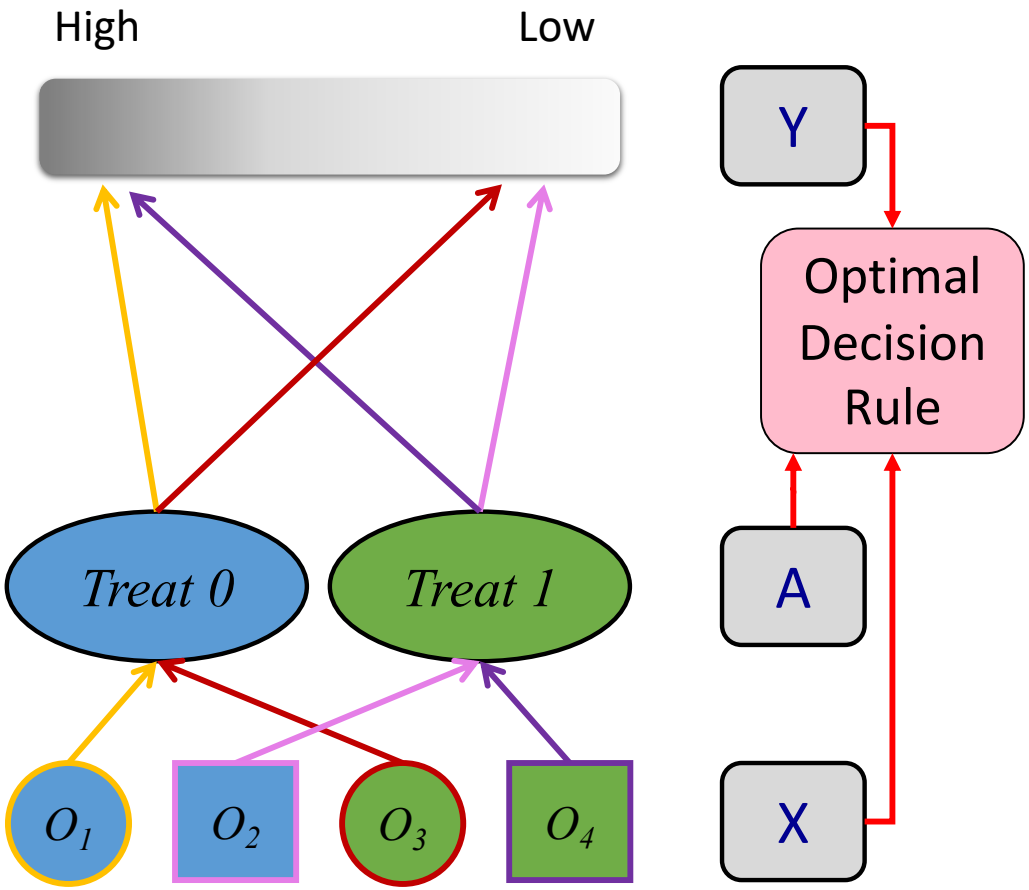
General Setup for Personalized Decision Making

The outcome Y can be observed after A is given. Due to individuals' heterogeneity in Y to different A, there may not exist a unified best decision.



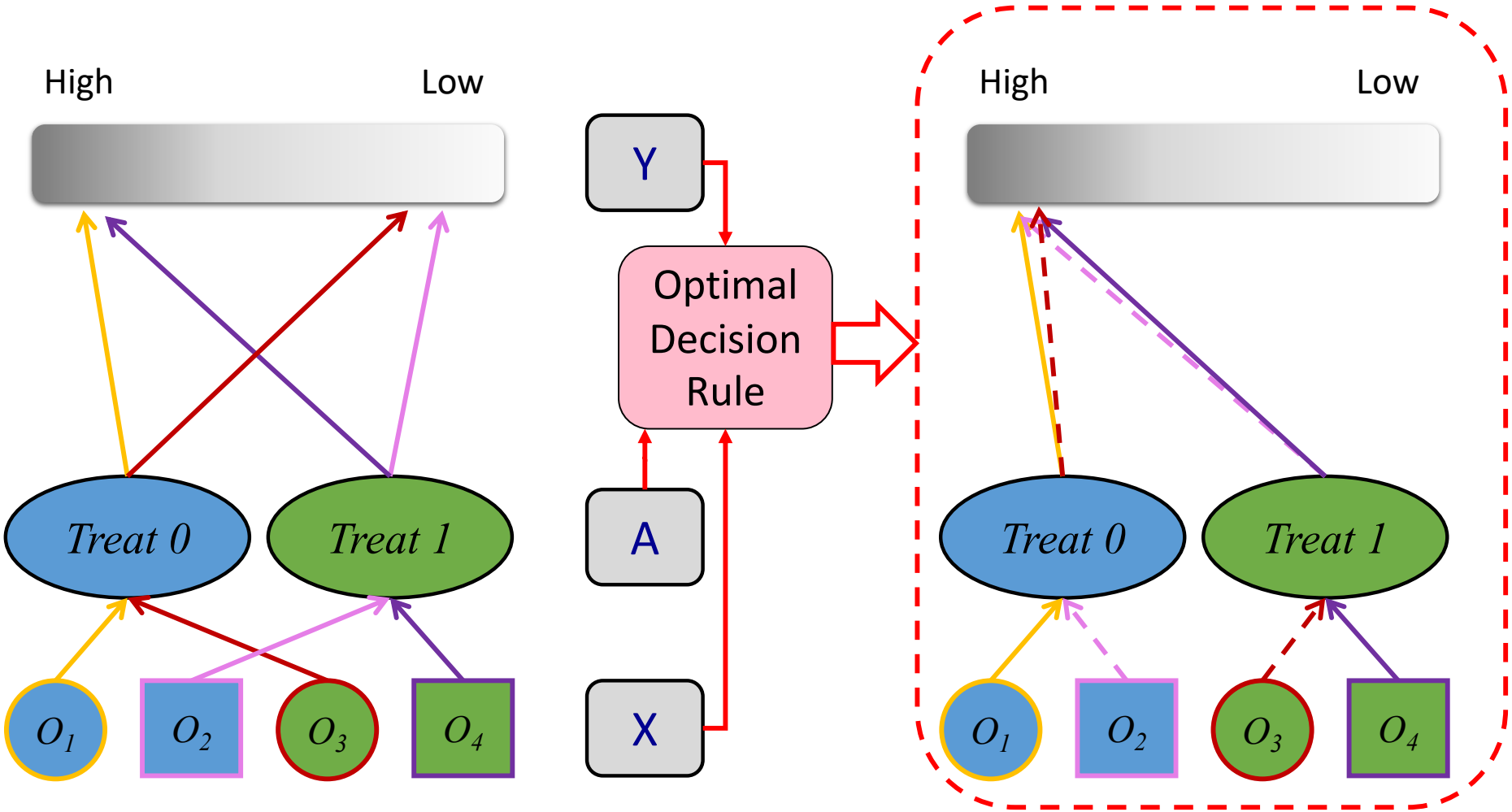
General Setup for Personalized Decision Making

The goal is to learn the optimal decision rule (ODR) that maximizes the mean outcome from either randomized trials or observational studies.



General Setup for Personalized Decision Making

Using ODR, we aim to assign future individuals with the best treatment option according to their covariates.



Challenges (1): Policy Optimization in Continuous Treatment Domains

Challenges (1): Policy Optimization in Continuous Treatment Domains

- Personalized dose finding: derive a dose level or dose range for each patient [Medicine];
- Dynamic pricing: assign each product an optimal price/discount according to their characteristics [Economics].

Challenges (1): Policy Optimization in Continuous Treatment Domains

- Personalized dose finding: derive a dose level or dose range for each patient [Medicine];
- Dynamic pricing: assign each product an optimal price/discount according to their characteristics [Economics].

Motivating Example: Warfarin Dosing Problem

- Warfarin: oral anticoagulant for prevention of thrombosis and thromboembolism;
- Over 30 million prescriptions in US, 2004;
- International Normalized Ratio (INR): measures the time it takes for blood to clot.
- Normal range of INR: 0.8 - 1.2 for a healthy person not using warfarin; targeted range: 2.0 - 3.0 for people on warfarin therapy.
- Dosage: 10mg to 100mg per week (Consortium 2009)
- Higher doses are more effective than lower doses, but may lead to a higher risk of bleeding.
- **Goal**: find the individualized decision rule that gives the optimal dose / dose range to stabilize the INR for patients on warfarin therapy.

Motivating Example: Warfarin Dosing Problem

- Warfarin: oral anticoagulant for prevention of thrombosis and thromboembolism;
- Over 30 million prescriptions in US, 2004;
- International Normalized Ratio (INR): measures the time it takes for blood to clot.
- Normal range of INR: 0.8 - 1.2 for a healthy person not using warfarin; targeted range: 2.0 - 3.0 for people on warfarin therapy.
- Dosage: 10mg to 100mg per week (Consortium 2009)
- Higher doses are more effective than lower doses, but may lead to a higher risk of bleeding.
- **Goal**: find the individualized decision rule that gives the optimal dose / dose range to stabilize the INR for patients on warfarin therapy.

Motivating Example: Warfarin Dosing Problem

- Warfarin: oral anticoagulant for prevention of thrombosis and thromboembolism;
- Over 30 million prescriptions in US, 2004;
- International Normalized Ratio (INR): measures the time it takes for blood to clot.
- Normal range of INR: 0.8 - 1.2 for a healthy person not using warfarin; targeted range: 2.0 - 3.0 for people on warfarin therapy.
- Dosage: 10mg to 100mg per week (Consortium 2009)
- Higher doses are more effective than lower doses, but may lead to a higher risk of bleeding.
- **Goal**: find the individualized decision rule that gives the optimal dose / dose range to stabilize the INR for patients on warfarin therapy.

Motivating Example: Warfarin Dosing Problem

- Warfarin: oral anticoagulant for prevention of thrombosis and thromboembolism;
- Over 30 million prescriptions in US, 2004;
- International Normalized Ratio (INR): measures the time it takes for blood to clot.
- Normal range of INR: 0.8 - 1.2 for a healthy person not using warfarin; targeted range: 2.0 - 3.0 for people on warfarin therapy.
- Dosage: 10mg to 100mg per week (Consortium 2009)
- Higher doses are more effective than lower doses, but may lead to a higher risk of bleeding.
- **Goal**: find the individualized decision rule that gives the optimal dose / dose range to stabilize the INR for patients on warfarin therapy.

Related Works

- Most ODR methods consider finite treatment options:
  - Q-learning (Watkins & Dayan 1992, Chakraborty et al. 2010);
  - A-learning (Murphy 2003, Shi et al. 2018);
  - Direct value search (Zhang et al. 2012, 2013, Zhao et al. 2012, 2015).
- Existing methods for personalized dose finding:
  - Parametric regression methods (Rich et al. 2014): consider a quadratic interactions between dose and covariates.
  - Discretize doses (Laber & Zhao 2015): Cluster patients into subgroups and assign a dosage for each subgroup.
  - Value search methods: O-learning (Chen et al. 2016) and kernel-assisted learning (Zhu et al. 2020).
- A limitation: recommend one single dose level for each individual patient, making it hard to implement in practice.

Related Works

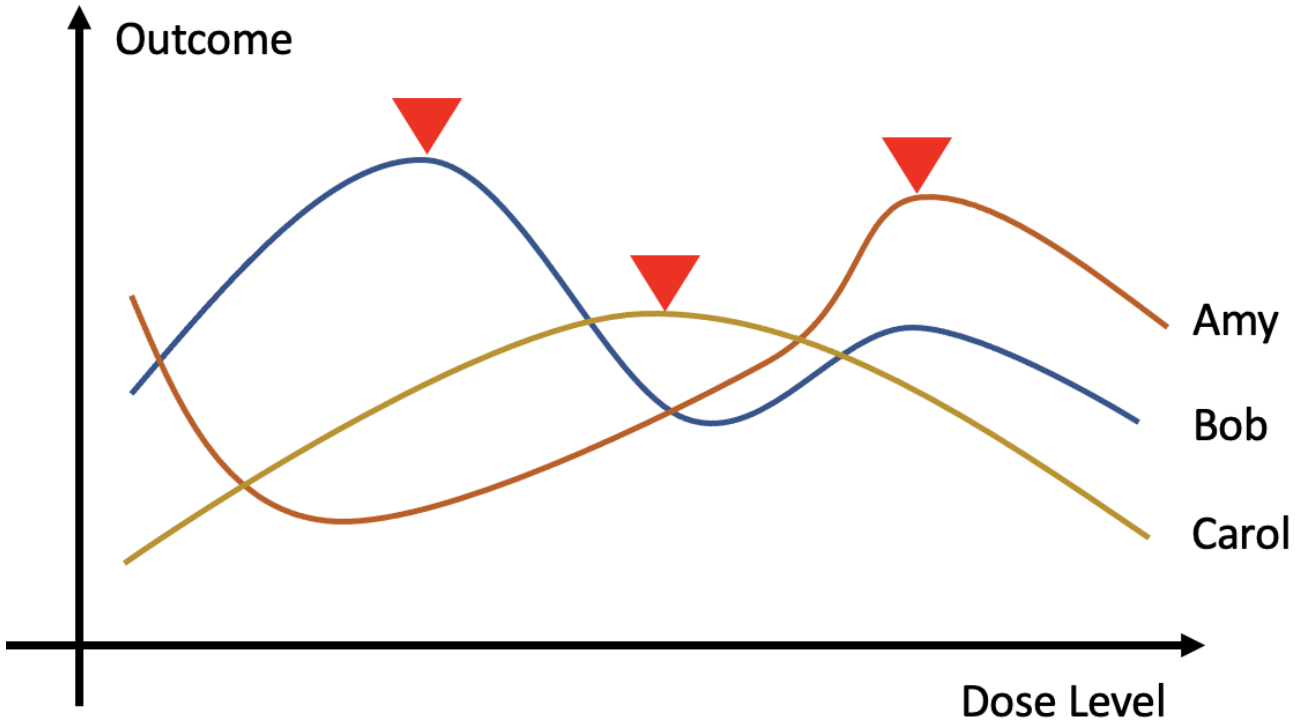
- Most ODR methods consider finite treatment options:
  - Q-learning (Watkins & Dayan 1992, Chakraborty et al. 2010);
  - A-learning (Murphy 2003, Shi et al. 2018);
  - Direct value search (Zhang et al. 2012, 2013, Zhao et al. 2012, 2015).
- Existing methods for personalized dose finding:
  - Parametric regression methods (Rich et al. 2014): consider a quadratic interactions between dose and covariates.
  - Discretize doses (Laber & Zhao 2015): Cluster patients into subgroups and assign a dosage for each subgroup.
  - Value search methods: O-learning (Chen et al. 2016) and kernel-assisted learning (Zhu et al. 2020).
- A limitation: recommend one single dose level for each individual patient, making it hard to implement in practice.

Related Works

- Most ODR methods consider finite treatment options:
  - Q-learning (Watkins & Dayan 1992, Chakraborty et al. 2010);
  - A-learning (Murphy 2003, Shi et al. 2018);
  - Direct value search (Zhang et al. 2012, 2013, Zhao et al. 2012, 2015).
- Existing methods for personalized dose finding:
  - Parametric regression methods (Rich et al. 2014): consider a quadratic interactions between dose and covariates.
  - Discretize doses (Laber & Zhao 2015): Cluster patients into subgroups and assign a dosage for each subgroup.
  - Value search methods: O-learning (Chen et al. 2016) and kernel-assisted learning (Zhu et al. 2020).
- A limitation: recommend one single dose level for each individual patient, making it hard to implement in practice.

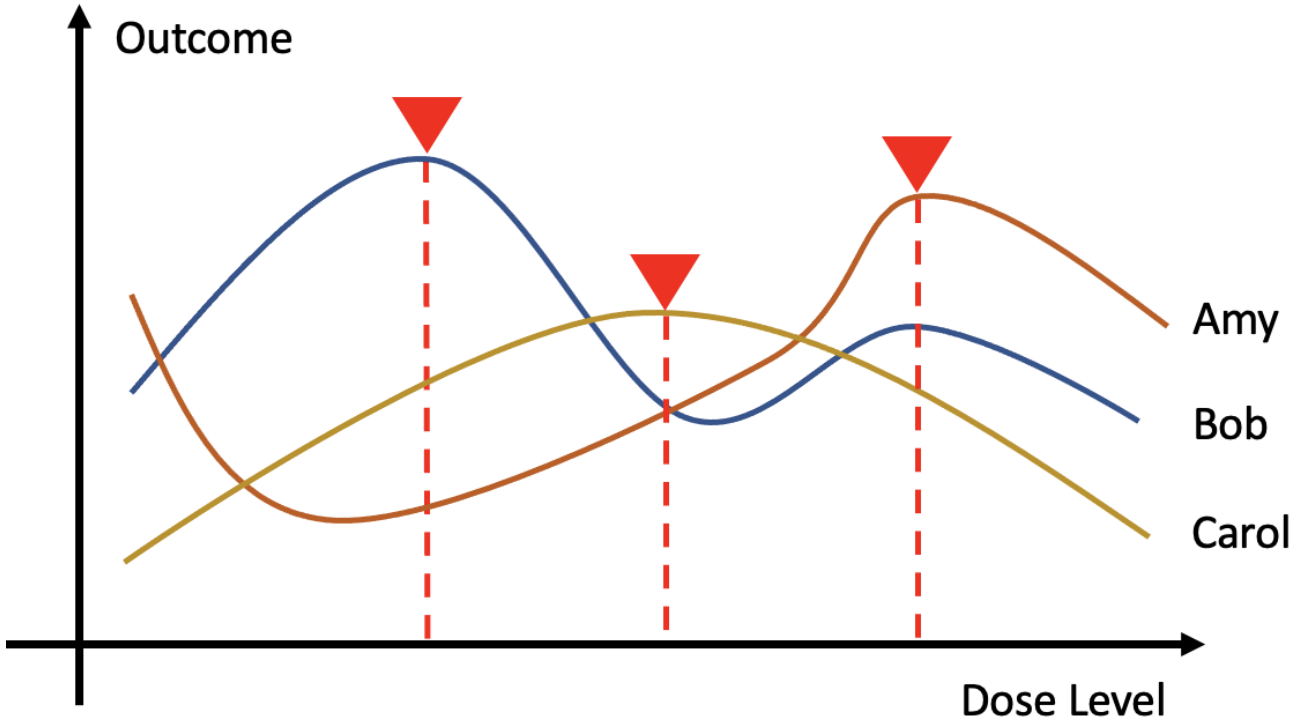
Point Decision Rule v.s. Interval-valued Decision Rule

Individualized **point** decision rule recommends a certain dose level.



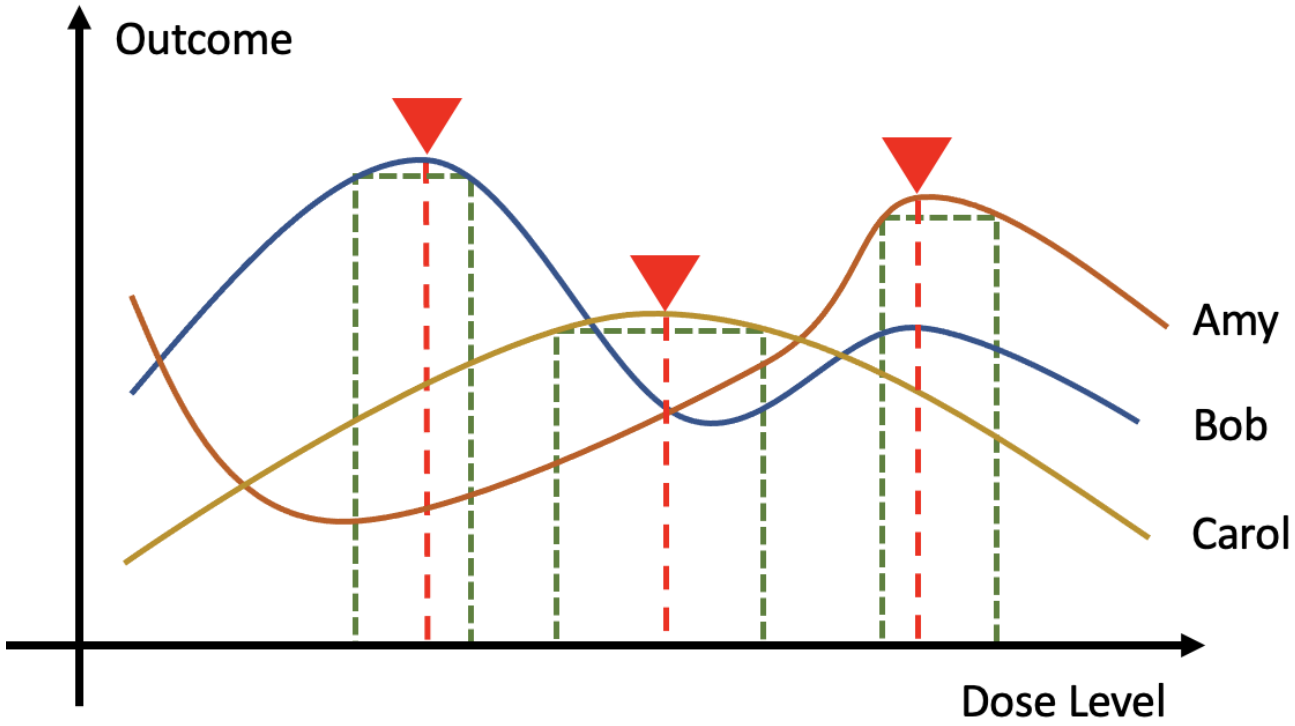
Point Decision Rule v.s. Interval-valued Decision Rule

Drawback: difficult to implement and unrealistic (possibly infinite doses).



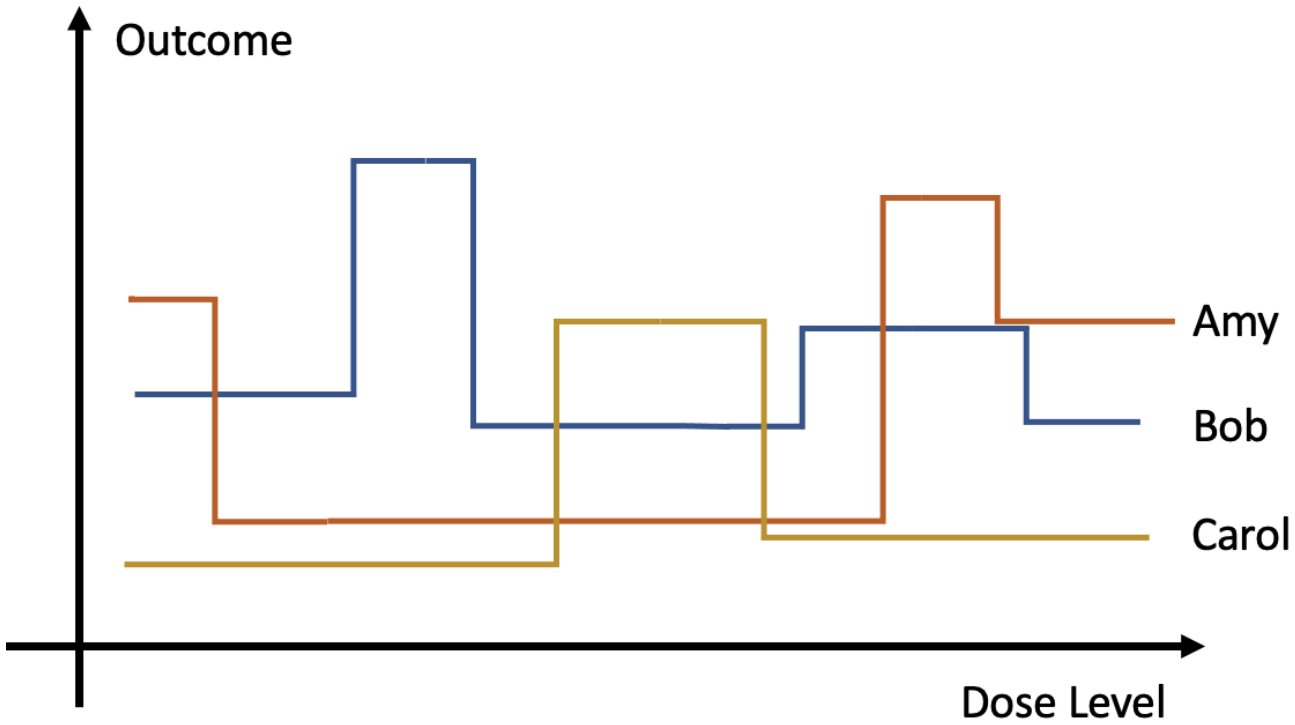
Point Decision Rule v.s. Interval-valued Decision Rule

Given certain continuity of the mean outcome function, arbitrary dose within a certain dose interval could achieve a nearly optimal efficacy.



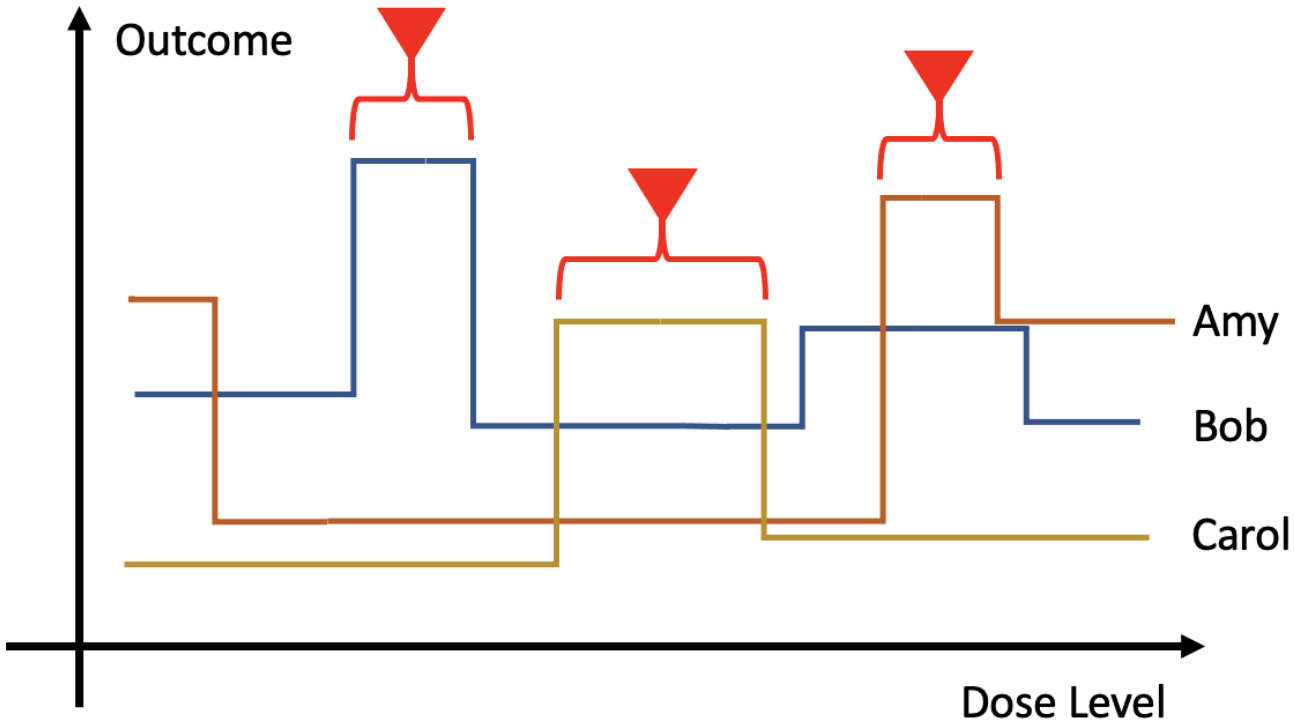
Point Decision Rule v.s. Interval-valued Decision Rule

Consider the mean outcome function as a piecewise constant function of dose level given baseline covariates.



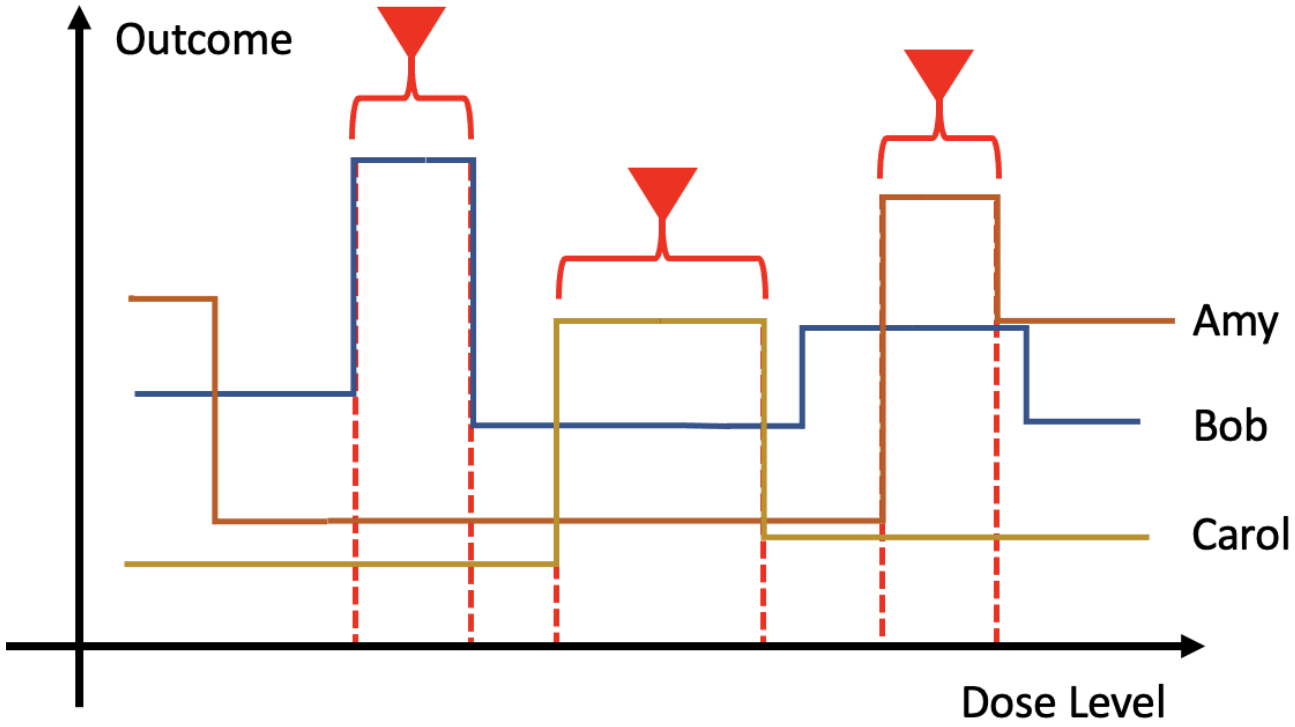
Point Decision Rule v.s. Interval-valued Decision Rule

The optimal dose can be any point within the optimal range of dose level that achieves the highest mean outcome.



Point Decision Rule v.s. Interval-valued Decision Rule

Individualized interval-valued decision rule (I2DR) returns a range of treatment levels based on individuals' baseline information.



Advantages of I2DR

- I2DR gives more options and is thus more flexible to implement in practice.
  - **Arbitrary** dose within the given dose interval could achieve the same efficacy;
  - Patients and clinicians could choose appropriate doses based on **their preference / medicine availability**;
  - Suggestions of interval-valued doses (see e.g., Rotschafer et al. 1982, Kuruvilla & Gurk-Turner 2001).
- I2DR provides instructions for designing the medicine specification to **save cost** on manufacturing dosage.

Advantages of I2DR

- I2DR gives more options and is thus more flexible to implement in practice.
  - **Arbitrary** dose within the given dose interval could achieve the same efficacy;
  - Patients and clinicians could choose appropriate doses based on **their preference / medicine availability**;
  - Suggestions of interval-valued doses (see e.g., Rotschafer et al. 1982, Kuruvilla & Gurk-Turner 2001).
- I2DR provides instructions for designing the medicine specification to **save cost** on manufacturing dosage.

Advantages of I2DR

- I2DR gives more options and is thus more flexible to implement in practice.
  - **Arbitrary** dose within the given dose interval could achieve the same efficacy;
  - Patients and clinicians could choose appropriate doses based on **their preference / medicine availability**;
  - Suggestions of interval-valued doses (see e.g., Rotschafer et al. 1982, Kuruvilla & Gurk-Turner 2001).
- I2DR provides instructions for designing the medicine specification to **save cost** on manufacturing dosage.

Summary of Our Work

- The optimal I2DR recommends an optimal dose range instead of a single optimal dose: more options, more flexible;
- Propose a jump interval-learning (JIL) based on parametric regression (e.g. linear/dose-varying coefficient model) or nonparametric regression (deep neural network model).
- Establish the consistency and convergence rate of the I2DR estimators, and develop a procedure to infer the mean outcome (i.e. the value) under the estimated optimal policy.

Summary of Our Work

- The optimal I2DR recommends an optimal dose range instead of a single optimal dose: more options, more flexible;
- Propose a jump interval-learning (JIL) based on parametric regression (e.g. linear/dose-varying coefficient model) or nonparametric regression (deep neural network model).
- Establish the consistency and convergence rate of the I2DR estimators, and develop a procedure to infer the mean outcome (i.e. the value) under the estimated optimal policy.

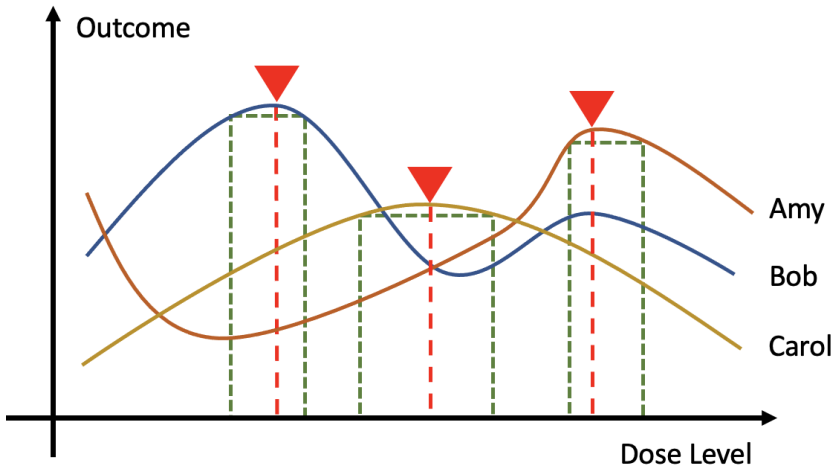
Summary of Our Work

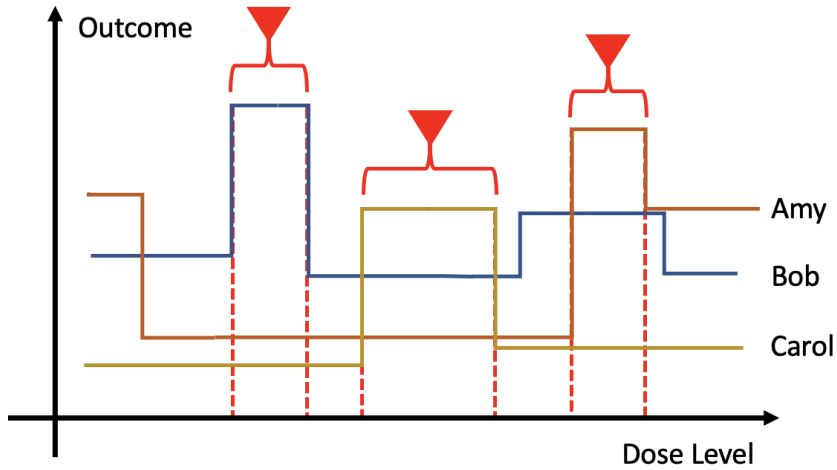
- The optimal I2DR recommends an optimal dose range instead of a single optimal dose: more options, more flexible;
- Propose a jump interval-learning (JIL) based on parametric regression (e.g. linear/dose-varying coefficient model) or nonparametric regression (deep neural network model).
- Establish the consistency and convergence rate of the I2DR estimators, and develop a procedure to infer the mean outcome (i.e. the value) under the estimated optimal policy.

Jump Interval-Learning for Individualized Decision Making with Continuous Treatments

Recap: Problem Setting

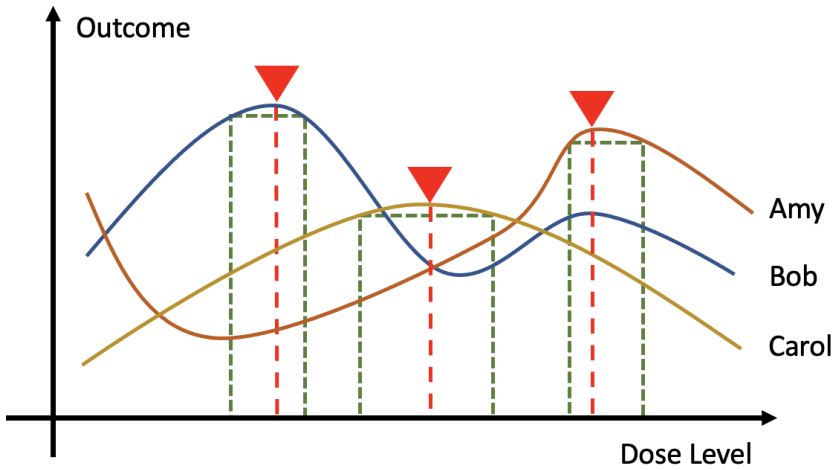
- Data: (X_i, A_i, Y_i), $i = 1, \cdots, n$;
  - $X_i \in \mathcal{X}$: p-dimensional covariates.
  - $A_i \in [0, a_0]$: received dose. w.l.o.g., set $a_0 = 1$.
  - Y_i: outcome of interest, the larger the better.
- Potential outcomes $Y^*(a)$, $a \in [0, 1]$.
- I2DR $d(X) : X \in \mathcal{X} \to \mathcal{I} \subset [0, 1]$, where $\mathcal{I}$ is a subinterval in $[0, 1]$.
- Value function: $V(d) = E\{Y^*(d(X))\}$.
- **Goal**: find the optimal I2DR: $d^{opt} = \arg\max_d V(d)$.

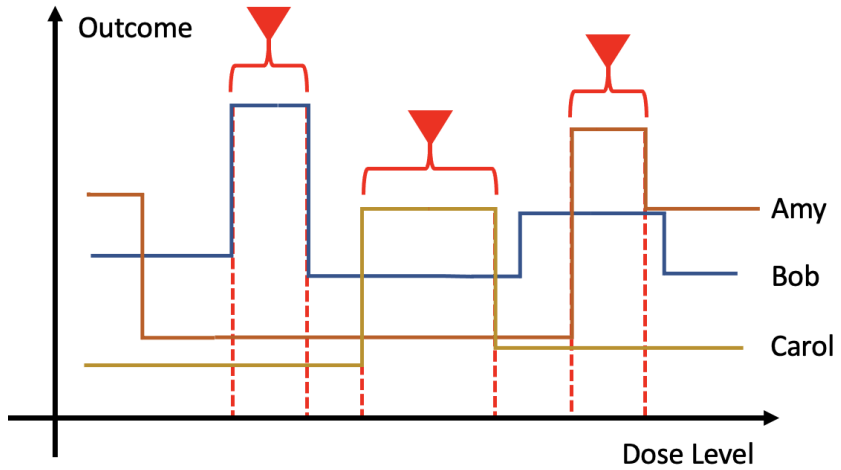




Recap: Problem Setting

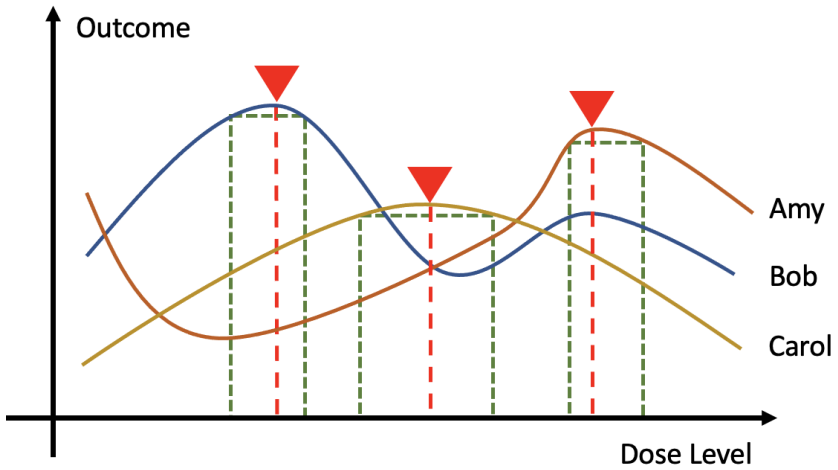
- Data: (X_i, A_i, Y_i), $i = 1, \cdots, n$;
  - $X_i \in \mathcal{X}$: p-dimensional covariates.
  - $A_i \in [0, a_0]$: received dose. w.l.o.g., set $a_0 = 1$.
  - Y_i: outcome of interest, the larger the better.
- Potential outcomes $Y^*(a)$, $a \in [0, 1]$.
- I2DR $d(X) : X \in \mathcal{X} \to \mathcal{I} \subset [0, 1]$, where $\mathcal{I}$ is a subinterval in $[0, 1]$.
- Value function: $V(d) = E\{Y^*(d(X))\}$.
- **Goal**: find the optimal I2DR: $d^{opt} = \arg\max_d V(d)$.

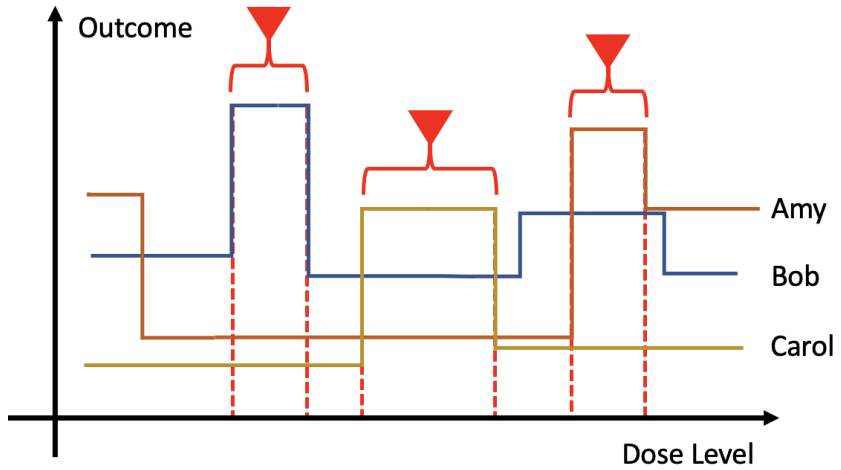




Recap: Problem Setting

- Data: (X_i, A_i, Y_i), $i = 1, \cdots, n$;
  - $X_i \in \mathcal{X}$: p-dimensional covariates.
  - $A_i \in [0, a_0]$: received dose. w.l.o.g., set $a_0 = 1$.
  - Y_i: outcome of interest, the larger the better.
- Potential outcomes $Y^*(a)$, $a \in [0, 1]$.
- I2DR $d(X) : X \in \mathcal{X} \to \mathcal{I} \subset [0, 1]$, where $\mathcal{I}$ is a subinterval in $[0, 1]$.
- Value function: $V(d) = E\{Y^*(d(X))\}$.
- **Goal**: find the optimal I2DR: $d^{opt} = \arg\max_d V(d)$.





Assumptions

A1 Stable Unit Treatment Value Assumption (SUTVA): $Y = \sum_a Y^*(a)\mathbb{I}(A = a)$;

A2 No unmeasured confounders: $\{Y^*(a) : a \in [0, 1]\} \perp A \mid X$;

A3 Positivity: there exists some constant $c_* > 0$ such that $p(a|x) \geq c_*$ for all $x \in \mathcal{X}$ and $a \in [0, 1]$, where $p(\cdot|x)$ denotes the probability density function of A conditional on $X = x$.

Let $Q(x, a) = E(Y|X = x, A = a)$, under above assumptions, do we have

$$V(d) = E\{Q(X, d(X))\}?$$

No, since $d(X)$ returns an interval.

Assumptions

A1 Stable Unit Treatment Value Assumption (SUTVA): $Y = \sum_a Y^*(a)\mathbb{I}(A = a)$;

A2 No unmeasured confounders: $\{Y^*(a) : a \in [0, 1]\} \perp A \mid X$;

A3 Positivity: there exists some constant $c_* > 0$ such that $p(a|x) \geq c_*$ for all $x \in \mathcal{X}$ and $a \in [0, 1]$, where $p(\cdot|x)$ denotes the probability density function of A conditional on $X = x$.

Let $Q(x, a) = E(Y|X = x, A = a)$, under above assumptions, do we have

$$V(d) = E\{Q(X, d(X))\}?$$

No, since $d(X)$ returns an interval.

Value Function under I2DR

- Given a dose interval $\mathcal{I}$, a dose is prescribed by a probability density function $\pi^*(a; x, \mathcal{I})$ such that $\int_{\mathcal{I}} \pi^*(a; x, \mathcal{I}) da = 1$.
- The value function under an I2DR $d(\cdot)$ is defined by

$$V^{\pi^*}(d) = E\left(\int_{d(X)} Q(X, a)\pi^*(a; X, d(X))da\right).$$

- Without knowing π^*, $V^{\pi^*}(d)$ may be difficult to estimate nonparametrically.
- Even a nonparametric estimator of $V^{\pi^*}(d)$ is available, it remains unknown how to efficiently compute the I2DR that maximizes the estimated value.

Value Function under I2DR

- Given a dose interval $\mathcal{I}$, a dose is prescribed by a probability density function $\pi^*(a; x, \mathcal{I})$ such that $\int_{\mathcal{I}} \pi^*(a; x, \mathcal{I})da = 1$.
- The value function under an I2DR $d(\cdot)$ is defined by

$$V^{\pi^*}(d) = E\left(\int_{d(X)} Q(X, a)\pi^*(a; X, d(X))da\right).$$

- Without knowing π^*, $V^{\pi^*}(d)$ may be difficult to estimate nonparametrically.
- Even a nonparametric estimator of $V^{\pi^*}(d)$ is available, it remains unknown how to efficiently compute the I2DR that maximizes the estimated value.

Value Function under I2DR

- Given a dose interval $\mathcal{I}$, a dose is prescribed by a probability density function $\pi^*(a; x, \mathcal{I})$ such that $\int_{\mathcal{I}} \pi^*(a; x, \mathcal{I}) da = 1$.
- The value function under an I2DR $d(\cdot)$ is defined by

$$V^{\pi^*}(d) = E\left(\int_{d(X)} Q(X, a)\pi^*(a; X, d(X))da\right).$$

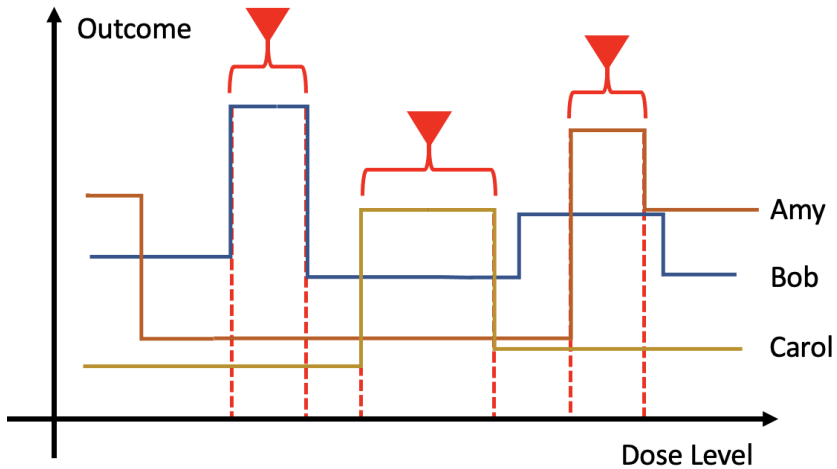
- Without knowing π^*, $V^{\pi^*}(d)$ may be difficult to estimate nonparametrically.
- Even a nonparametric estimator of $V^{\pi^*}(d)$ is available, it remains unknown how to efficiently compute the I2DR that maximizes the estimated value.

Working Model Assumptions

- **Model I (Piecewise Functions).**

$$Q(x,a) = \sum_{\mathcal{I}\in\mathcal{P}_0} q_{\mathcal{I},0}(x)\mathbb{I}(a\in\mathcal{I}) \qquad \forall x\in\mathcal{X}, a\in[0,1]$$

  - $\mathcal{P}_0$ is a finite partition of $[0,1]$, i.e. $\{[\tau_0,\tau_1),[\tau_1,\tau_2),\ldots,[\tau_{K-1},\tau_K]\}$ for some $0=\tau_0<\tau_1<\tau_2<\cdots<\tau_{K-1}<\tau_K=1$.
  - $(q_{\mathcal{I},0})_{\mathcal{I}\in\mathcal{P}_0}$ is collection of continuous functions.

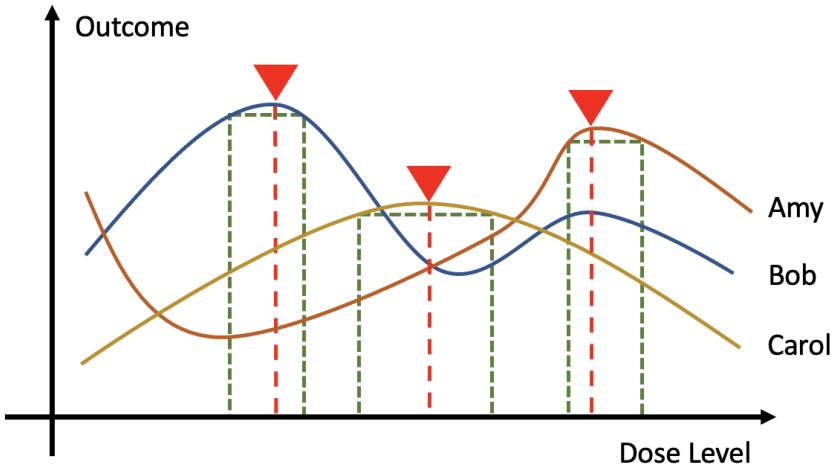


Working Model Assumptions

- **Model II (Continuous Functions).** $Q(x, a)$ is a continuous function of a and x, for any $x \in \mathcal{X}$ and $a \in [0, 1]$.
  - dose-varying coefficient model:

$$Q(x, a) = \bar{x}^T \beta_0(a), \qquad \forall x \in \mathcal{X}, a \in [0, 1].$$

  - $Q(x, a)$ is nonparametric, say a deep neural network model.



Motivation by Model I

- **Optimal I2DR**:

$$d^{opt}(x) = \arg\max_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(x),$$

independent of the preference function π^*.

- **Validation**:

$$V^{\pi^*}(d^{opt}) = \mathsf{E}\Big\{ \sum_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(X) \mathbb{I}(d^{opt}(X) \in \mathcal{I}) \Big\} \geq V^{\pi^*}(d),$$

for any rule d and preference function π^*. Denote $V^{\pi^*}(d)$ by $V(d)$.

- **Q**: How to estimate $\mathcal{P}_0$ and $q_{\mathcal{I},0}(x)$? Jump interval-learning, which works for both Model I and Model II.

Motivation by Model I

- **Optimal I2DR**:

$$d^{opt}(x) = \arg\max_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(x),$$

independent of the preference function π^*.

- **Validation**:

$$V^{\pi^*}(d^{opt}) = \mathsf{E}\Big\{ \sum_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(X) \mathbb{I}(d^{opt}(X) \in \mathcal{I}) \Big\} \geq V^{\pi^*}(d),$$

for any rule d and preference function π^*. Denote $V^{\pi^*}(d)$ by $V(d)$.

- **Q**: How to estimate $\mathcal{P}_0$ and $q_{\mathcal{I},0}(x)$? Jump interval-learning, which works for both Model I and Model II.

Motivation by Model I

- **Optimal I2DR**:

$$d^{opt}(x) = \arg\max_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(x),$$

independent of the preference function π^*.

- **Validation**:

$$V^{\pi^*}(d^{opt}) = \mathsf{E}\Big\{ \sum_{\mathcal{I} \in \mathcal{P}_0} q_{\mathcal{I},0}(X) \mathbb{I}(d^{opt}(X) \in \mathcal{I}) \Big\} \geq V^{\pi^*}(d),$$

for any rule d and preference function π^*. Denote $V^{\pi^*}(d)$ by $V(d)$.

- **Q**: How to estimate $\mathcal{P}_0$ and $q_{\mathcal{I},0}(x)$? Jump interval-learning, which works for both Model I and Model II.

Jump Interval-Learning

- Consider an initial partition $\mathcal{P}_{0,m}$ of $[0,1]$: $\{[\tau_0, \tau_1), \ldots, [\tau_{m-1}, \tau_m]\}$ for some $0 = \tau_0 < \tau_1 < \cdots < \tau_{m-1} < \tau_m = 1$. For example, $\tau_k = k/m$, and m can diverge with n.
- Adaptively determines the optimal partition $\widehat{\mathcal{P}}$ based on jump-penalized regression: each interval in $\widehat{\mathcal{P}}$ corresponds to a union of a set of consecutive intervals in $\mathcal{P}_{0,m}$.
- $\mathcal{B}(m)$ denote the set of all possible partitions $\mathcal{P}$ derived from $\mathcal{P}_{0,m}$.
- Associate to each partition $\mathcal{P} \in \mathcal{B}(m)$ a collection of functions $\{q(\cdot; \theta_{\mathcal{I}})\}_{\mathcal{I} \in \mathcal{P}} \in \prod_{\mathcal{I} \in \mathcal{P}} \mathcal{Q}_{\mathcal{I}}$ for $\mathcal{Q}_{\mathcal{I}}$ as some class of functions, where $\theta_{\mathcal{I}}$ is the underlying parameter associated to interval $\mathcal{I}$.

Jump Interval-Learning

- Consider an initial partition $\mathcal{P}_{0,m}$ of $[0,1]$: $\{[\tau_0, \tau_1), \ldots, [\tau_{m-1}, \tau_m]\}$ for some $0 = \tau_0 < \tau_1 < \cdots < \tau_{m-1} < \tau_m = 1$. For example, $\tau_k = k/m$, and m can diverge with n.
- Adaptively determines the optimal partition $\widehat{\mathcal{P}}$ based on jump-penalized regression: each interval in $\widehat{\mathcal{P}}$ corresponds to a union of a set of consecutive intervals in $\mathcal{P}_{0,m}$.
- $\mathcal{B}(m)$ denote the set of all possible partitions $\mathcal{P}$ derived from $\mathcal{P}_{0,m}$.
- Associate to each partition $\mathcal{P} \in \mathcal{B}(m)$ a collection of functions $\{q(\cdot; \theta_{\mathcal{I}})\}_{\mathcal{I} \in \mathcal{P}} \in \prod_{\mathcal{I} \in \mathcal{P}} \mathcal{Q}_{\mathcal{I}}$ for $\mathcal{Q}_{\mathcal{I}}$ as some class of functions, where $\theta_{\mathcal{I}}$ is the underlying parameter associated to interval $\mathcal{I}$.

Jump Interval-Learning

- Consider an initial partition $\mathcal{P}_{0,m}$ of $[0,1]$: $\{[\tau_0, \tau_1), \ldots, [\tau_{m-1}, \tau_m]\}$ for some $0 = \tau_0 < \tau_1 < \cdots < \tau_{m-1} < \tau_m = 1$. For example, $\tau_k = k/m$, and m can diverge with n.
- Adaptively determines the optimal partition $\widehat{\mathcal{P}}$ based on jump-penalized regression: each interval in $\widehat{\mathcal{P}}$ corresponds to a union of a set of consecutive intervals in $\mathcal{P}_{0,m}$.
- $\mathcal{B}(m)$ denote the set of all possible partitions $\mathcal{P}$ derived from $\mathcal{P}_{0,m}$.
- Associate to each partition $\mathcal{P} \in \mathcal{B}(m)$ a collection of functions $\{q(\cdot; \theta_{\mathcal{I}})\}_{\mathcal{I} \in \mathcal{P}} \in \prod_{\mathcal{I} \in \mathcal{P}} \mathcal{Q}_{\mathcal{I}}$ for $\mathcal{Q}_{\mathcal{I}}$ as some class of functions, where $\theta_{\mathcal{I}}$ is the underlying parameter associated to interval $\mathcal{I}$.

Jump Interval-Learning

Jump-Penalized Regression

$$(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}) =$$

$$\underset{\substack{\mathcal{P}\in\mathcal{B}(m)\\ \{q(\cdot;\theta_{\mathcal{I}})\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\}}}{\arg\min} \Big\{ \sum_{\mathcal{I}\in\mathcal{P}} \Big(\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \big\{ Y_i - q(X_i;\theta_{\mathcal{I}}) \big\}^2 + \lambda_n |\mathcal{I}| \|\theta_{\mathcal{I}}\|_2^2 \Big) + \gamma_n |\mathcal{P}| \Big\},$$

where $|\mathcal{P}|$ denote the number of intervals in $\mathcal{P}$ and $|\mathcal{I}|$ denote the length of the interval $\mathcal{I}$.

- $\gamma_n|\mathcal{P}|$: control the total number of jumps (i.e. intervals).
- $\lambda_n|\mathcal{I}|\|\theta_{\mathcal{I}}\|_2^2$: prevent overfitting in large p problems.
- For a fixed $\mathcal{P}$, solving the optimization problem yields an estimator of the Q-functions $\{\widehat{q}_{\mathcal{I}}\}_{\mathcal{I}\in\mathcal{P}}$, by either a parametric (linear model) or nonparametric (deep learning) regression.

Jump Interval-Learning

Jump-Penalized Regression

$$(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}) =$$

$$\underset{\substack{\mathcal{P}\in\mathcal{B}(m)\\ \{q(\cdot;\theta_{\mathcal{I}})\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\}}}{\arg\min} \Big\{ \sum_{\mathcal{I}\in\mathcal{P}} \Big(\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \big\{ Y_i - q(X_i;\theta_{\mathcal{I}}) \big\}^2 + \lambda_n |\mathcal{I}| \|\theta_{\mathcal{I}}\|_2^2 \Big) + \gamma_n |\mathcal{P}| \Big\},$$

where $|\mathcal{P}|$ denote the number of intervals in $\mathcal{P}$ and $|\mathcal{I}|$ denote the length of the interval $\mathcal{I}$.

- $\gamma_n|\mathcal{P}|$: control the total number of jumps (i.e. intervals).
- $\lambda_n|\mathcal{I}|\|\theta_{\mathcal{I}}\|_2^2$: prevent overfitting in large p problems.
- For a fixed $\mathcal{P}$, solving the optimization problem yields an estimator of the Q-functions $\{\widehat{q}_{\mathcal{I}}\}_{\mathcal{I}\in\mathcal{P}}$, by either a parametric (linear model) or nonparametric (deep learning) regression.

Jump Interval-Learning

Jump-Penalized Regression

$$(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}) =$$

$$\underset{\substack{\mathcal{P}\in\mathcal{B}(m)\\ \{q(\cdot;\theta_{\mathcal{I}})\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\}}}{\arg\min} \Big\{ \sum_{\mathcal{I}\in\mathcal{P}} \Big(\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \big\{ Y_i - q(X_i;\theta_{\mathcal{I}}) \big\}^2 + \lambda_n |\mathcal{I}| \|\theta_{\mathcal{I}}\|_2^2 \Big) + \gamma_n |\mathcal{P}| \Big\},$$

where $|\mathcal{P}|$ denote the number of intervals in $\mathcal{P}$ and $|\mathcal{I}|$ denote the length of the interval $\mathcal{I}$.

- $\gamma_n|\mathcal{P}|$: control the total number of jumps (i.e. intervals).
- $\lambda_n|\mathcal{I}|\|\theta_{\mathcal{I}}\|_2^2$: prevent overfitting in large p problems.
- For a fixed $\mathcal{P}$, solving the optimization problem yields an estimator of the Q-functions $\{\widehat{q}_{\mathcal{I}}\}_{\mathcal{I}\in\mathcal{P}}$, by either a parametric (linear model) or nonparametric (deep learning) regression.

Jump Interval-Learning

Jump-Penalized Regression

$$(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}) =$$

$$\underset{\substack{\mathcal{P}\in\mathcal{B}(m) \\ \{q(\cdot;\theta_{\mathcal{I}})\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\}}}{\arg\min} \Big\{ \sum_{\mathcal{I}\in\mathcal{P}} \Big(\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \big\{ Y_i - q(X_i;\theta_{\mathcal{I}}) \big\}^2 + \lambda_n |\mathcal{I}| \|\theta_{\mathcal{I}}\|_2^2 \Big) + \gamma_n |\mathcal{P}| \Big\},$$

where $|\mathcal{P}|$ denote the number of intervals in $\mathcal{P}$ and $|\mathcal{I}|$ denote the length of the interval $\mathcal{I}$.

- $\gamma_n|\mathcal{P}|$: control the total number of jumps (i.e. intervals).
- $\lambda_n|\mathcal{I}|\|\theta_{\mathcal{I}}\|_2^2$: prevent overfitting in large p problems.
- For a fixed $\mathcal{P}$, solving the optimization problem yields an estimator of the Q-functions $\{\widehat{q}_{\mathcal{I}}\}_{\mathcal{I}\in\mathcal{P}}$, by either a parametric (linear model) or nonparametric (deep learning) regression.

Estimated I2DR and its Value

- **Estimated optimal I2DR**:

$$\widehat{d}(x) = \arg\max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(x), \qquad \forall x \in \mathcal{X},$$

  when the argmax is not unique, $\widehat{d}(\cdot)$ outputs the interval that gives the smallest doses.

- Consider the **generalized propensity score** function $e(\mathcal{I}|x) \equiv \Pr(A \in \mathcal{I}|X = x)$. Let $\widehat{e}(\mathcal{I}|x)$ denote the resulting estimate.

- **Value estimator** (Zhang et al. 2012) of the estimated I2DR:

$$\widehat{V} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\right].$$

Estimated I2DR and its Value

- **Estimated optimal I2DR**:

$$\widehat{d}(x) = \arg\max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(x), \qquad \forall x \in \mathcal{X},$$

  when the argmax is not unique, $\widehat{d}(\cdot)$ outputs the interval that gives the smallest doses.

- Consider the **generalized propensity score** function $e(\mathcal{I}|x) \equiv \Pr(A \in \mathcal{I}|X = x)$. Let $\widehat{e}(\mathcal{I}|x)$ denote the resulting estimate.

- **Value estimator** (Zhang et al. 2012) of the estimated I2DR:

$$\widehat{V} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\right].$$

Estimated I2DR and its Value

- **Estimated optimal I2DR**:

$$\widehat{d}(x) = \arg\max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(x), \qquad \forall x \in \mathcal{X},$$

  when the argmax is not unique, $\widehat{d}(\cdot)$ outputs the interval that gives the smallest doses.

- Consider the **generalized propensity score** function $e(\mathcal{I}|x) \equiv \Pr(A \in \mathcal{I}|X = x)$. Let $\widehat{e}(\mathcal{I}|x)$ denote the resulting estimate.

- **Value estimator** (Zhang et al. 2012) of the estimated I2DR:

$$\widehat{V} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\left\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\right\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}} \widehat{q}_{\mathcal{I}}(X_i)\right].$$

Choice of the $q_{\mathcal{I}}$

- **Linear-JIL**: $q(x, \theta_{\mathcal{I}}) = \bar{x}^\top \theta_{\mathcal{I}}$, where $\bar{x} = (1, x^\top)^\top$.
- **Deep-JIL**: use deep neural networks (DNNs) to present $q_{\mathcal{I}}(x)$.

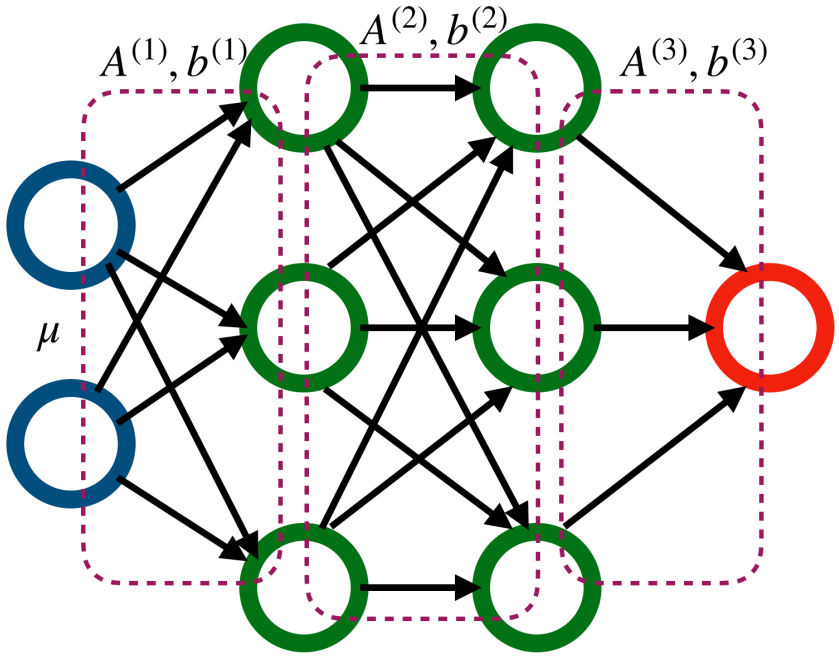


Figure 1: Illustration of DNN with $L = 2$ (layers) and $W = 25$ (parameters).

- Apply the Multi-layer Perceptron (MLP) regressor (Pedregosa et al. 2011) for parameter estimation.

Implementation of JIL

Apply Dynamic Programming (Friedrich et al. 2008) to Find $\widehat{\mathcal{P}}$

- For any interval $\mathcal{I} \subset [0,1]$, define the cost function

$$\text{cost}(\mathcal{I}) = \min_{q(\cdot;\theta_{\mathcal{I}}) \in \mathcal{Q}_{\mathcal{I}}} \left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \{Y_i - q(X_i;\theta_{\mathcal{I}})\}^2 + \lambda_n ||\theta_{\mathcal{I}}||_2^2 \right],$$

- For any integer $1 \leq r \leq m$, define $\mathcal{B}(m,r)$: the set of all possible partitions $\mathcal{P}_r$ of $[0, r/m)$ with grid points $\{j/m : j = 0, 1, \ldots, r\}$. Note $\mathcal{B}(m,m) = \mathcal{B}(m)$.
- Define the Bellman function

$$B(r) = \inf_{\mathcal{P}_r \in \mathcal{B}(m,r)} \left(\sum_{\mathcal{I} \in \mathcal{P}_r} \text{cost}(\mathcal{I}) + \gamma_n(|\mathcal{P}_r| - 1) \right),$$

with $B(0) = -\gamma_n$.

Implementation of JIL

Pruned Exact Linear Time Method (PELT) (Killick et al. 2012)

Calculate the Bellman equation in a recursion formula,

$$B(r) = \min_{j \in \mathcal{R}_r} \{B(j) + \gamma_n + \text{cost}([j/m, r/m))\}, \qquad \forall r \geq 1.$$

where $\mathcal{R}_r$ is the candidate change points list updated by

$$\{j \in \mathcal{R}_{r-1} \cup \{r-1\} : B(j) + \text{cost}([j/m, (r-1)/m)) \leq B(r-1)\},$$

during each iteration with $\mathcal{R}_0 = \{0\}$.

- Given r, search the optimal change point j that minimizes $B(r)$;
- Let $\tau(r)$ be the corresponding minimizer;
- Iteratively compute $B(r)$ and $\tau(r)$ for $r = 1, \ldots, m$;
- The optimal partition $\widehat{\mathcal{P}}$ is determined by the values stored in $\tau(\cdot)$.

Implementation of JIL

Pruned Exact Linear Time Method (PELT) (Killick et al. 2012)

Calculate the Bellman equation in a recursion formula,

$$B(r) = \min_{j \in \mathcal{R}_r} \{B(j) + \gamma_n + \text{cost}([j/m, r/m))\}, \qquad \forall r \geq 1.$$

where $\mathcal{R}_r$ is the candidate change points list updated by

$$\{j \in \mathcal{R}_{r-1} \cup \{r-1\} : B(j) + \text{cost}([j/m, (r-1)/m)) \leq B(r-1)\},$$

during each iteration with $\mathcal{R}_0 = \{0\}$.

- Given r, search the optimal change point j that minimizes $B(r)$;
- Let $\tau(r)$ be the corresponding minimizer;
- Iteratively compute $B(r)$ and $\tau(r)$ for $r = 1, \ldots, m$;
- The optimal partition $\widehat{\mathcal{P}}$ is determined by the values stored in $\tau(\cdot)$.

Results under Model I: Linear-JIL with $q_{\mathcal{I},0}(x) = \bar{x}^\top \theta_{\mathcal{I},0}$

[A4] [Condition on Tails] and [A5] [Margin Condition]

Theorem 1. Convergence Rate of Linear-JIL under Model I

Assume (A1)-(A5) hold. Assume $m \asymp n$, $\lambda_n = O(n^{-1}\log n)$, $\{\gamma_n\}_{n\in\mathbb{N}}$ satisfies $\gamma_n \to 0$ and $\gamma_n n/\log n \to \infty$. There exist constants $\bar{c}_1, \bar{c}_2 > 0$ s.t. the following events hold with probability at least (w.p.a.l.) $1 - O(n^{-2})$:
(i) $|\widehat{\mathcal{P}}| = |\mathcal{P}_0|$.
(ii) $\max_{\tau\in J(\mathcal{P}_0)} \min_{\hat{\tau}\in J(\widehat{\mathcal{P}})} |\hat{\tau} - \tau| \le \bar{c}_1 n^{-1}\log n$.
(iii) $\int_0^1 \|\widehat{\theta}(a) - \theta_0(a)\|_2^2 da \le \bar{c} n^{-1}\log n$.
(iv)

$$V^{opt} - V(\widehat{d}) \le \bar{c}_2 (n^{-1}\log n)^{(1+\gamma)/2} + \bar{c}_2 n^{-1}\log n.$$

Note: $V^{opt} = V(d^{opt})$, and $J(\mathcal{P})$ is the set of change point locations.

Results under Model I: Deep-JIL

Additional assumptions in DNN theories (Farrell et al. 2021): A6 and A7.

Theorem 2. Convergence Rate of Deep-JIL under Model I

Assume (A1)-(A7) hold. Assume $m \asymp n$, $\{\gamma_n\}_{n\in\mathbb{N}}$ satisfies $\gamma_n \to 0$ and $\gamma_n \gg n^{-2\beta/(2\beta+p)} \log^8 n$. There exist a constant $\bar{c} > 0$ and DNN classes $\{\mathcal{Q}_\mathcal{I} : \mathcal{I}\}$ with $L_\mathcal{I} \asymp \log(n|\mathcal{I}|)$ and $W_\mathcal{I} \asymp (n|\mathcal{I}|)^{p/(2\beta+p)} \log(n|\mathcal{I}|)$ s.t. w.p.a.l. $1 - O(n^{-2})$, the Deep-JIL estimator satisfies

(i) $|\widehat{\mathcal{P}}| = |\mathcal{P}_0|$;

(ii) $\max_{\tau\in J(\mathcal{P}_0)} \min_{\hat{\tau}\in J(\widehat{\mathcal{P}})} |\hat{\tau} - \tau| \le \bar{c} n^{-2\beta/(2\beta+p)} \log^8 n$;

(iii) $\mathrm{E}|Q(X,A) - \sum_{\mathcal{I}\in\widehat{\mathcal{P}}} \mathbb{I}(A \in \mathcal{I}) \widehat{q}_\mathcal{I}(X)|^2 da \le \bar{c} n^{-2\beta/(2\beta+p)} \log^8 n$;

(iv) $V(\widehat{d}) \ge V^{opt} - O(1)(n^{-\frac{2\beta}{2\beta+p}} \log^8 n + n^{-\frac{2\beta(1+\gamma)}{(2\beta+p)(2+\gamma)}} \log^{\frac{8+8\gamma}{2+\gamma}} n)$.

Results for the Value Estimator under Model I

- Linear-JIL: $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 1 (iv))
- Deep-JIL: if $4\beta(1+\gamma) > (2\beta+p)(2+\gamma)$, we have $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 2 (iv))

Theorem 3. Asymptotic Normality of Value Estimator under Model I

Assume (A8)($\{\widehat{e}(\mathcal{I}\}$) holds, and for any $\mathcal{I}_1, \mathcal{I}_2 \in \mathcal{P}_0$ with $\mathcal{I}_1 \neq \mathcal{I}_2$, we have $\Pr(q_{\mathcal{I}_1,0}(X) = q_{\mathcal{I}_2,0}(X)) = 0$. For some $\sigma_0^2 > 0$, we have

$$\sqrt{n}(\widehat{V} - V^{opt}) \xrightarrow{d} N(0, \sigma_0^2).$$

Here, σ_0^2 can be estimated by

$$\widehat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i)\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i) - \widehat{V}\right]^2,$$

where $\{\widehat{q}_{\mathcal{I}}(\cdot)\}$ is the value estimations under Linear-JIL or Deep-JIL.

Results for the Value Estimator under Model I

- Linear-JIL: $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 1 (iv))
- Deep-JIL: if $4\beta(1+\gamma) > (2\beta+p)(2+\gamma)$, we have $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 2 (iv))

Theorem 3. Asymptotic Normality of Value Estimator under Model I

Assume (A8)($\{\widehat{e}(\mathcal{I}\}$) holds, and for any $\mathcal{I}_1, \mathcal{I}_2 \in \mathcal{P}_0$ with $\mathcal{I}_1 \neq \mathcal{I}_2$, we have $\Pr(q_{\mathcal{I}_1,0}(X) = q_{\mathcal{I}_2,0}(X)) = 0$. For some $\sigma_0^2 > 0$, we have

$$\sqrt{n}(\widehat{V} - V^{opt}) \xrightarrow{d} N(0, \sigma_0^2).$$

Here, σ_0^2 can be estimated by

$$\widehat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i)\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i) - \widehat{V}\right]^2,$$

where $\{\widehat{q}_{\mathcal{I}}(\cdot)\}$ is the value estimations under Linear-JIL or Deep-JIL.

Results for the Value Estimator under Model I

- Linear-JIL: $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 1 (iv))
- Deep-JIL: if $4\beta(1+\gamma) > (2\beta+p)(2+\gamma)$, we have $V(\widehat{d}) = V^{opt} + o_p(n^{-1/2})$ (by Theorem 2 (iv))

Theorem 3. Asymptotic Normality of Value Estimator under Model I

Assume (A8)($\{\widehat{e}(\mathcal{I}\}$) holds, and for any $\mathcal{I}_1, \mathcal{I}_2 \in \mathcal{P}_0$ with $\mathcal{I}_1 \neq \mathcal{I}_2$, we have $\Pr(q_{\mathcal{I}_1,0}(X) = q_{\mathcal{I}_2,0}(X)) = 0$. For some $\sigma_0^2 > 0$, we have

$$\sqrt{n}(\widehat{V} - V^{opt}) \xrightarrow{d} N(0, \sigma_0^2).$$

Here, σ_0^2 can be estimated by

$$\widehat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\{A_i \in \widehat{d}(X_i)\}}{\widehat{e}(\widehat{d}(X_i)|X_i)}\{Y_i - \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i)\} + \max_{\mathcal{I}\in\widehat{\mathcal{P}}}\widehat{q}_{\mathcal{I}}(X_i) - \widehat{V}\right]^2,$$

where $\{\widehat{q}_{\mathcal{I}}(\cdot)\}$ is the value estimations under Linear-JIL or Deep-JIL.

Results under Model II

Theorem 4. Convergence Rate under Dose-Varying Coefficient Model

Assume (A1)-(A5) and (A9) (Lipschitz continuity) hold. Assume $m \asymp n$, $\lambda_n = O(n^{-1}\log n)$, γ_n satisfies $\gamma_n \asymp (n^{-1}\log n)^{(1+2\alpha_0)/(1+4\alpha_0)}$. There exists $\bar{c}^* > 0$ s.t. the following occurs w.p.a.l. $1 - O(n^{-2})$:

$$V^{opt} - V(\widehat{d}) \leq \bar{c}^* (n^{-1}\log n)^{\alpha_0/(1+4\alpha_0)}.$$

Theorem 5. Consistency under General Continuous Model

Assume (A1)-(A4), (A6)-(A7) hold, $m \asymp n$, and γ_n satisfies $\gamma_n \to 0$ and $\gamma_n \gg n^{-2\beta/(2\beta+p)} \log^8 n$. There exist a constant $\bar{c} > 0$ and DNN classes $\{\mathcal{Q}_{\mathcal{I}} : \mathcal{I}\}$ with $L_{\mathcal{I}} \asymp \log(n|\mathcal{I}|)$ and $W_{\mathcal{I}} \asymp (n|\mathcal{I}|)^{p/(2\beta+p)} \log(n|\mathcal{I}|)$ s.t. w.p.a.l. $1 - O(n^{-2})$, the Deep-JIL estimator satisfies

$$V^{opt} - V(\widehat{d}) = o_p(1).$$

Results under Model II

Theorem 4. Convergence Rate under Dose-Varying Coefficient Model

Assume (A1)-(A5) and (A9) (Lipschitz continuity) hold. Assume $m \asymp n$, $\lambda_n = O(n^{-1}\log n)$, γ_n satisfies $\gamma_n \asymp (n^{-1}\log n)^{(1+2\alpha_0)/(1+4\alpha_0)}$. There exists $\bar{c}^* > 0$ s.t. the following occurs w.p.a.l. $1 - O(n^{-2})$:

$$V^{opt} - V(\widehat{d}) \leq \bar{c}^* (n^{-1}\log n)^{\alpha_0/(1+4\alpha_0)}.$$

Theorem 5. Consistency under General Continuous Model

Assume (A1)-(A4), (A6)-(A7) hold, $m \asymp n$, and γ_n satisfies $\gamma_n \to 0$ and $\gamma_n \gg n^{-2\beta/(2\beta+p)} \log^8 n$. There exist a constant $\bar{c} > 0$ and DNN classes $\{\mathcal{Q}_{\mathcal{I}} : \mathcal{I}\}$ with $L_{\mathcal{I}} \asymp \log(n|\mathcal{I}|)$ and $W_{\mathcal{I}} \asymp (n|\mathcal{I}|)^{p/(2\beta+p)} \log(n|\mathcal{I}|)$ s.t. w.p.a.l. $1 - O(n^{-2})$, the Deep-JIL estimator satisfies

$$V^{opt} - V(\widehat{d}) = o_p(1).$$

Settings under Model I

$$Y|X, A \sim N(Q(X, A), 1), \quad A|X \sim \text{Unif}[0, 1], \quad X^{(1)}, \ldots, X^{(p)} \overset{iid}{\sim} N(0, 1),$$

where $p = 4$ and consider the following two scenarios for $Q(X, A)$:

- **Scenario 1**

$$Q(x, a) = \begin{cases} 1 + x^{(1)}, & a < 0.35, \\ x^{(1)} - x^{(2)}, & 0.35 \leq a < 0.65, \\ 1 - x^{(2)}, & a \geq 0.65. \end{cases}$$

- **Scenario 2**

$$Q(x, a) = \begin{cases} 1 + (x^{(1)})^3, & a < 0.35, \\ x^{(1)} - \log(1.5 + x^{(2)}), & 0.35 \leq a < 0.65, \\ 1 - \sin(0.5dx^{(2)}), & a \geq 0.65. \end{cases}$$

- $J(\mathcal{P}_0) = \{0.35, 0.65\}$ and $|\mathcal{P}_0| = 3$.

Simulation Results I

		Scenario 1, $p = 4$			Scenario 2, $p = 4$		
		$n = 200$	$n = 400$	$n = 800$	$n = 200$	$n = 400$	$n = 800$
Method	V^{opt}	1.34			1.35		
Linear-JIL	$\widehat{V}$	1.436	1.383	1.340	NA	NA	NA
	$\widehat{\sigma}$	0.129	0.091	0.066	NA	NA	NA
	CP	89.80	93.20	95.60	NA	NA	NA
	$\lvert\widehat{\mathcal{P}}\rvert$	2.97	3.01	3.00	NA	NA	NA
Deep-JIL	$\widehat{V}$	1.297	1.338	1.345	1.333	1.331	1.349
	$\widehat{\sigma}$	0.160	0.108	0.060	0.166	0.102	0.060
	CP	90.60	93.60	96.00	95.60	93.80	95.00
	$\lvert\widehat{\mathcal{P}}\rvert$	2.98	3.25	3.18	2.95	3.10	3.08

- Set $m = n/5$, $\lambda_n = 0$, and $\gamma_n = 4n^{-1}\log(n)$.
- Conduct 500 runs for each setting.
- CP, coverage probability for the optimal value function.

Settings for General Models

- **Scenario 3**

$$Q(x,a) = \begin{cases} \sqrt{x^{(1)}/2 + 0.5}, & a < 0.25, \\ \sin(2dx^{(2)}), & 0.25 \le a < 0.5, \\ 0.5 - (x^{(1)} + x^{(2)} - 0.75)^2, & 0.5 \le a < 0.75, \\ 0.5, & a \ge 0.75. \end{cases}$$

- **Scenario 4** $Q(x,a) = \bar{x}^\top \{2|a - 0.5|\theta^*\}, \quad \theta^* = (1, 2, -2, 0_{p-2}^\top)^\top.$
- **Scenario 5**

$$Q(x,a) = 8 + 4x^{(1)} - 2x^{(2)} - 2x^{(3)} - 10(1 + 0.5x^{(1)} + 0.5x^{(2)} - 2a)^2.$$

- Compare with outcome weighted learning (Chen et al. 2016) based on linear function (L-O-L) and Gaussian kernel (K-O-L). Fix the parameter $\phi_n = 0.1$, and select tuning parameters by cross-validation.

Simulation Results II with morec$m = n/10$

	n	50	100	200	400	800
Scenario 1	L-JIL	0.783(0.016)	0.832(0.016)	1.080(0.014)	1.259(0.002)	1.297(0.001)
$V = 1.34$	D-JIL	0.914(0.012)	0.967(0.008)	1.050(0.005)	1.071(0.005)	1.138(0.001)
$p = 20$	L-O-L	0.558(0.004)	0.574(0.004)	0.600(0.005)	0.597(0.005)	0.583(0.005)
	K-O-L	0.335(0.008)	0.415(0.006)	0.441(0.006)	0.457(0.005)	0.489(0.004)
Scenario 2	L-JIL	0.741(0.021)	0.854(0.020)	1.180(0.007)	1.266(0.001)	1.299(0.001)
$V = 1.35$	D-JIL	0.900(0.012)	0.978(0.008)	1.074(0.004)	1.102(0.003)	1.141(0.001)
$p = 20$	L-O-L	0.450(0.009)	0.448(0.006)	0.447(0.005)	0.429(0.004)	0.410(0.003)
	K-O-L	0.115(0.019)	0.213(0.010)	0.229(0.007)	0.241(0.004)	0.276(0.002)
Scenario 3	L-JIL	0.227(0.020)	0.268(0.013)	0.372(0.008)	0.432(0.003)	0.511(0.002)
$V = 0.76$	D-JIL	0.453(0.019)	0.469(0.009)	0.511(0.005)	0.526(0.004)	0.545(0.002)
$p = 20$	L-O-L	0.002(0.010)	-0.009(0.008)	-0.060(0.006)	-0.090(0.005)	-0.107(0.004)
	K-O-L	-0.268(0.026)	-0.233(0.015)	-0.260(0.009)	-0.251(0.006)	-0.233(0.003)
Scenario 4	L-JIL	0.553(0.013)	0.564(0.011)	0.630(0.011)	0.806(0.006)	0.882(0.002)
$V = 1.28$	D-JIL	0.612(0.014)	0.651(0.008)	0.684(0.004)	0.653(0.006)	0.801(0.001)
$p = 20$	L-O-L	0.525(0.016)	0.458(0.010)	0.375(0.004)	0.300(0.002)	0.237(0.001)
	K-O-L	0.236(0.007)	0.260(0.004)	0.252(0.003)	0.244(0.001)	0.246(0.001)
Scenario 5	L-JIL	5.82(0.05)	6.41(0.02)	6.80(0.01)	7.02(0.01)	7.16(0.01)
$V = 8.00$	D-JIL	5.57(0.06)	5.79(0.03)	5.97(0.02)	6.10(0.01)	6.26(0.01)
$p = 20$	L-O-L	5.92(0.07)	6.75(0.03)	7.32(0.02)	7.66(0.01)	7.81(0.01)
	K-O-L	6.70(0.02)	7.05(0.02)	7.38(0.01)	7.58(0.01)	7.56(0.01)

Data Analysis: Warfarin Dose Data

- Total 3848 patients with 6 covariates: age, height, weight, gender (male=1), the VKORC1.AG genotype and VKORC1.AA genotype.
- Warfarin dose: 10mg to 100mg per week. Dose A is scaled into $[0, 1]$.
- The response $Y = -|\text{INR} - 2.5|$.
- Linear-JIL found three dose intervals: [0, 0.05), [0.05, 0.17), [0.17, 1].
- Our value is $-0.332 > -0.344$ by K-O-L (Chen et al., 2016).
- The estimated $\widehat{\theta}$'s are given by

	Intercept	Age	Weight	Height	Gender	VKORC1.AG	VKORC1.AA
$\widehat{\theta}_1$	-1.673	0.025	0.006	0.006	-0.158	-0.364	-0.349
$\widehat{\theta}_2$	-1.741	0.029	0.004	0.006	-0.201	0.057	-0.051
$\widehat{\theta}_3$	-0.488	0.012	0.001	0.001	-0.033	-0.002	-0.120

- Some findings (by fixing all the other variables):
  - Patients with VKORC1 as **AG or AA** should receive higher doses.
  - **Younger** patients should be recommended for lower doses.
  - **Male** patients should be recommended for higher doses.

Data Analysis: Warfarin Dose Data

- Total 3848 patients with 6 covariates: age, height, weight, gender (male=1), the VKORC1.AG genotype and VKORC1.AA genotype.
- Warfarin dose: 10mg to 100mg per week. Dose A is scaled into $[0, 1]$.
- The response $Y = -|\text{INR} - 2.5|$.
- Linear-JIL found three dose intervals: [0, 0.05), [0.05, 0.17), [0.17, 1].
- Our value is $-0.332 > -0.344$ by K-O-L (Chen et al., 2016).
- The estimated $\widehat{\theta}$'s are given by

	Intercept	Age	Weight	Height	Gender	VKORC1.AG	VKORC1.AA
$\widehat{\theta}_1$	-1.673	0.025	0.006	0.006	-0.158	-0.364	-0.349
$\widehat{\theta}_2$	-1.741	0.029	0.004	0.006	-0.201	0.057	-0.051
$\widehat{\theta}_3$	-0.488	0.012	0.001	0.001	-0.033	-0.002	-0.120

- Some findings (by fixing all the other variables):
  - Patients with VKORC1 as **AG or AA** should receive higher doses.
  - **Younger** patients should be recommended for lower doses.
  - **Male** patients should be recommended for higher doses.

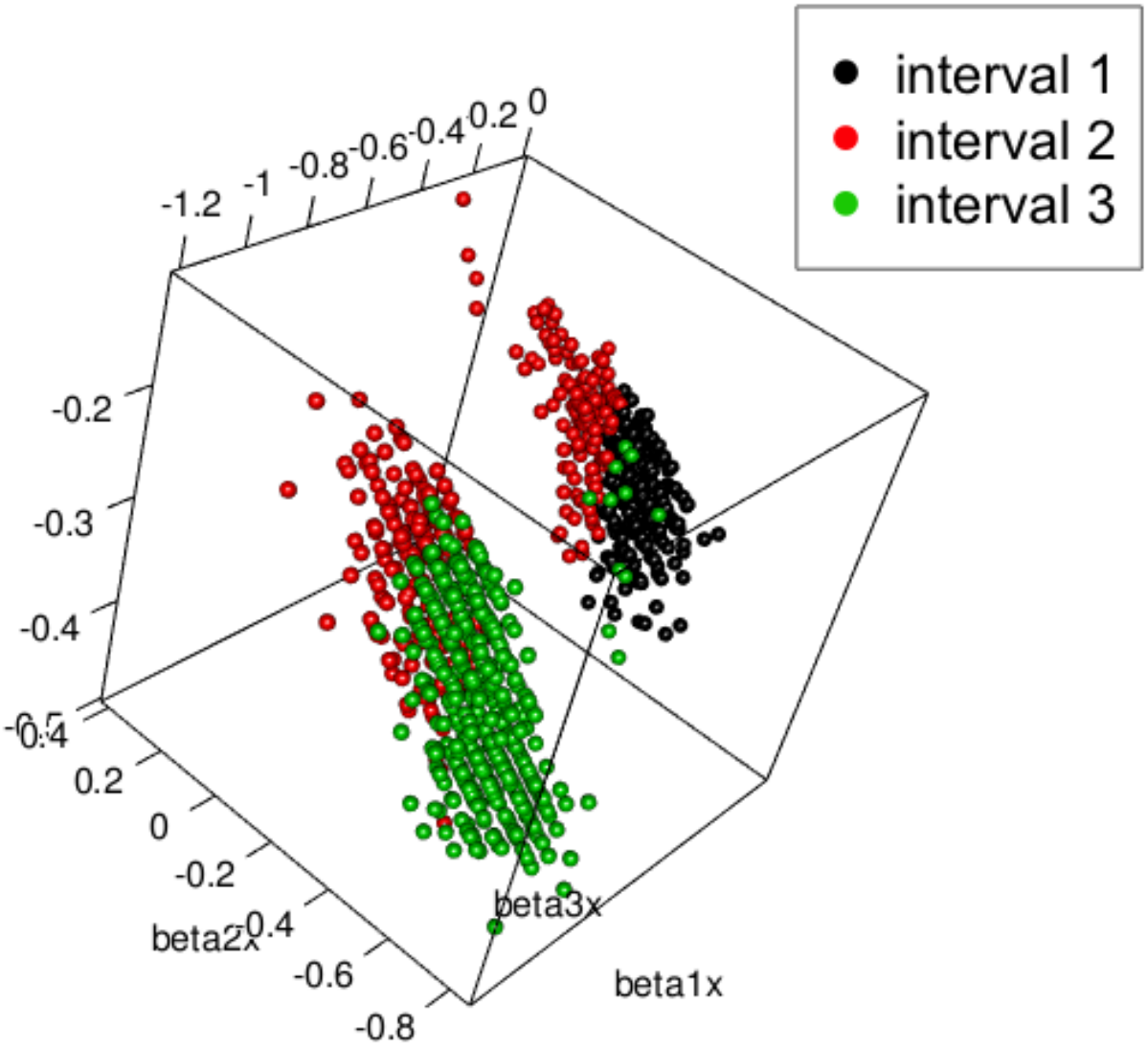
Data Analysis: Warfarin Dose Data

- Total 3848 patients with 6 covariates: age, height, weight, gender (male=1), the VKORC1.AG genotype and VKORC1.AA genotype.
- Warfarin dose: 10mg to 100mg per week. Dose A is scaled into $[0, 1]$.
- The response $Y = -|\text{INR} - 2.5|$.
- Linear-JIL found three dose intervals: [0, 0.05), [0.05, 0.17), [0.17, 1].
- Our value is $-0.332 > -0.344$ by K-O-L (Chen et al., 2016).
- The estimated $\widehat{\theta}$'s are given by

	Intercept	Age	Weight	Height	Gender	VKORC1.AG	VKORC1.AA
$\widehat{\theta}_1$	-1.673	0.025	0.006	0.006	-0.158	-0.364	-0.349
$\widehat{\theta}_2$	-1.741	0.029	0.004	0.006	-0.201	0.057	-0.051
$\widehat{\theta}_3$	-0.488	0.012	0.001	0.001	-0.033	-0.002	-0.120

- Some findings (by fixing all the other variables):
  - Patients with VKORC1 as **AG or AA** should receive higher doses.
  - **Younger** patients should be recommended for lower doses.
  - **Male** patients should be recommended for higher doses.

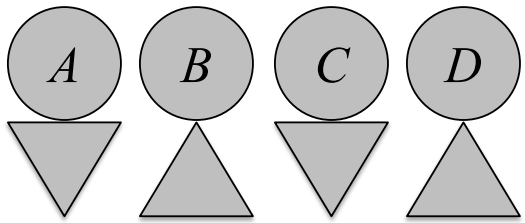
Visualization for the Estimated I2DR



Challenges (2): Policy Evaluation in Continuous Treatment Domains

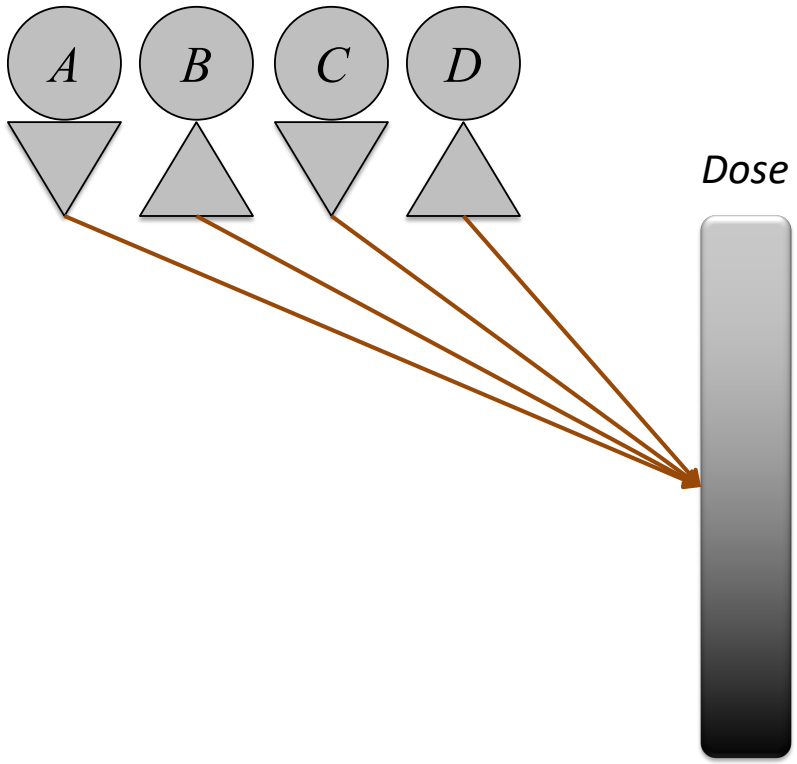
Challenges (2): Policy Evaluation in Continuous Treatment Domains

Consider a decision making problem in a continuous domain:



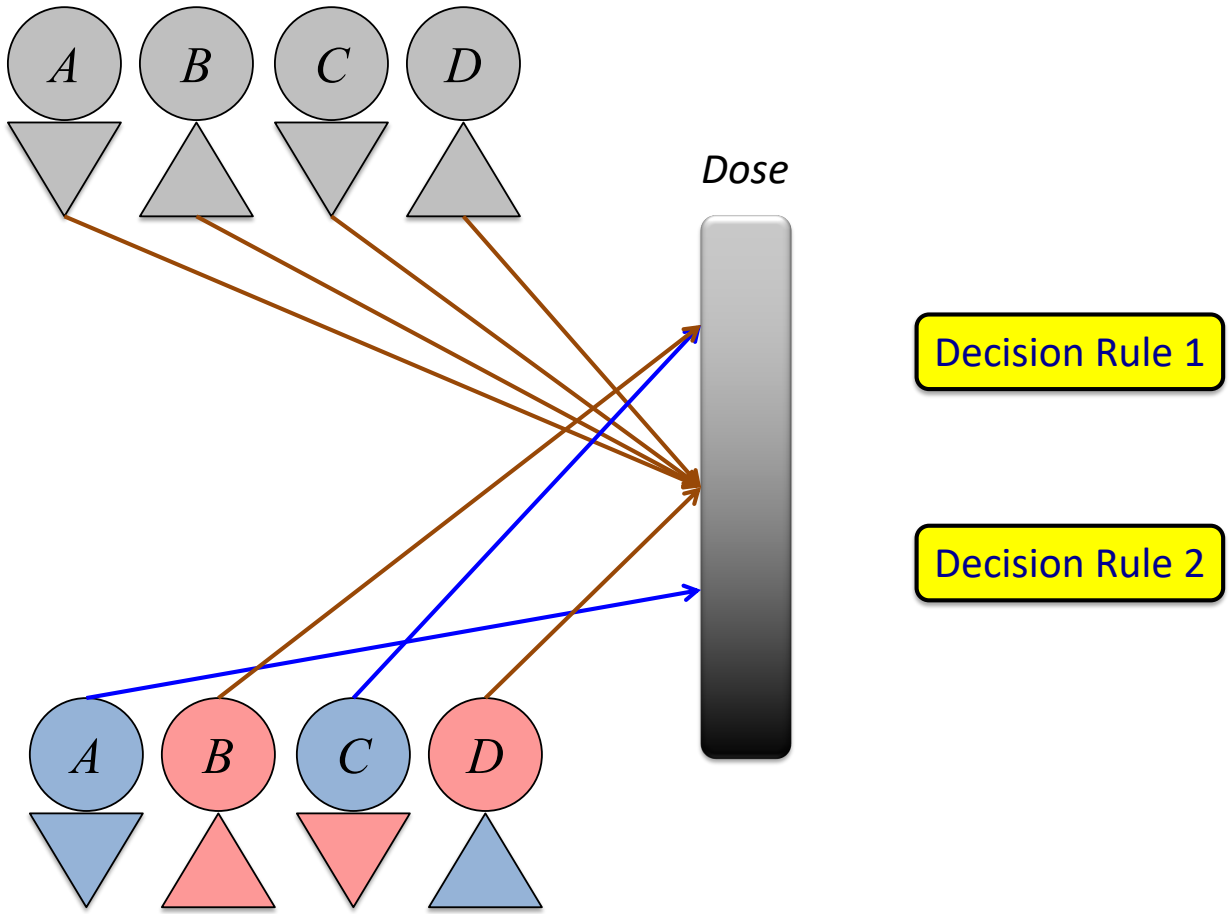
Challenges (2): Policy Evaluation in Continuous Treatment Domains

Decision 1: a simple decision rule that always assigns individuals to a fixed best treatment option.



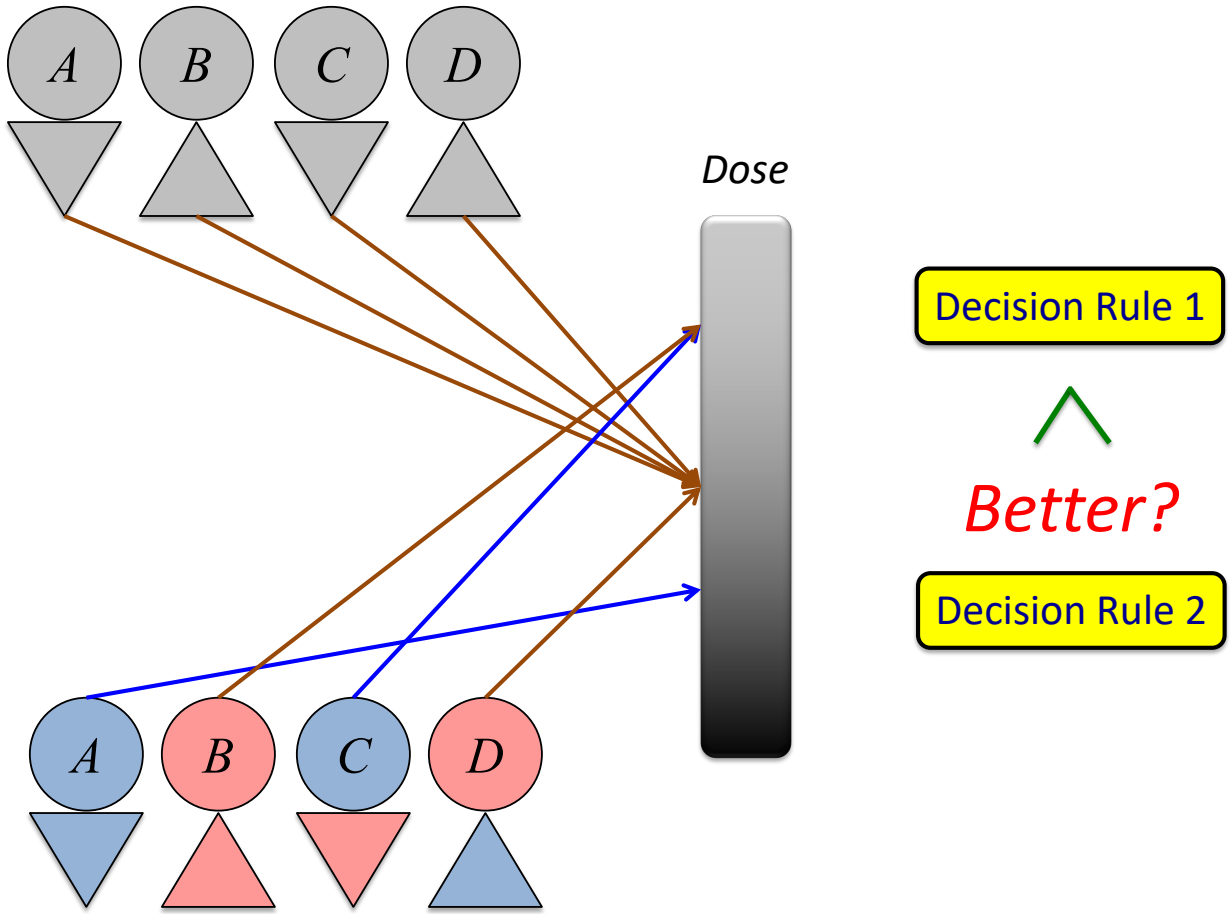
Challenges (2): Policy Evaluation in Continuous Treatment Domains

Decision 2: an individualized decision rule (IDR) that assigns individuals with treatments according to their baseline covariates.



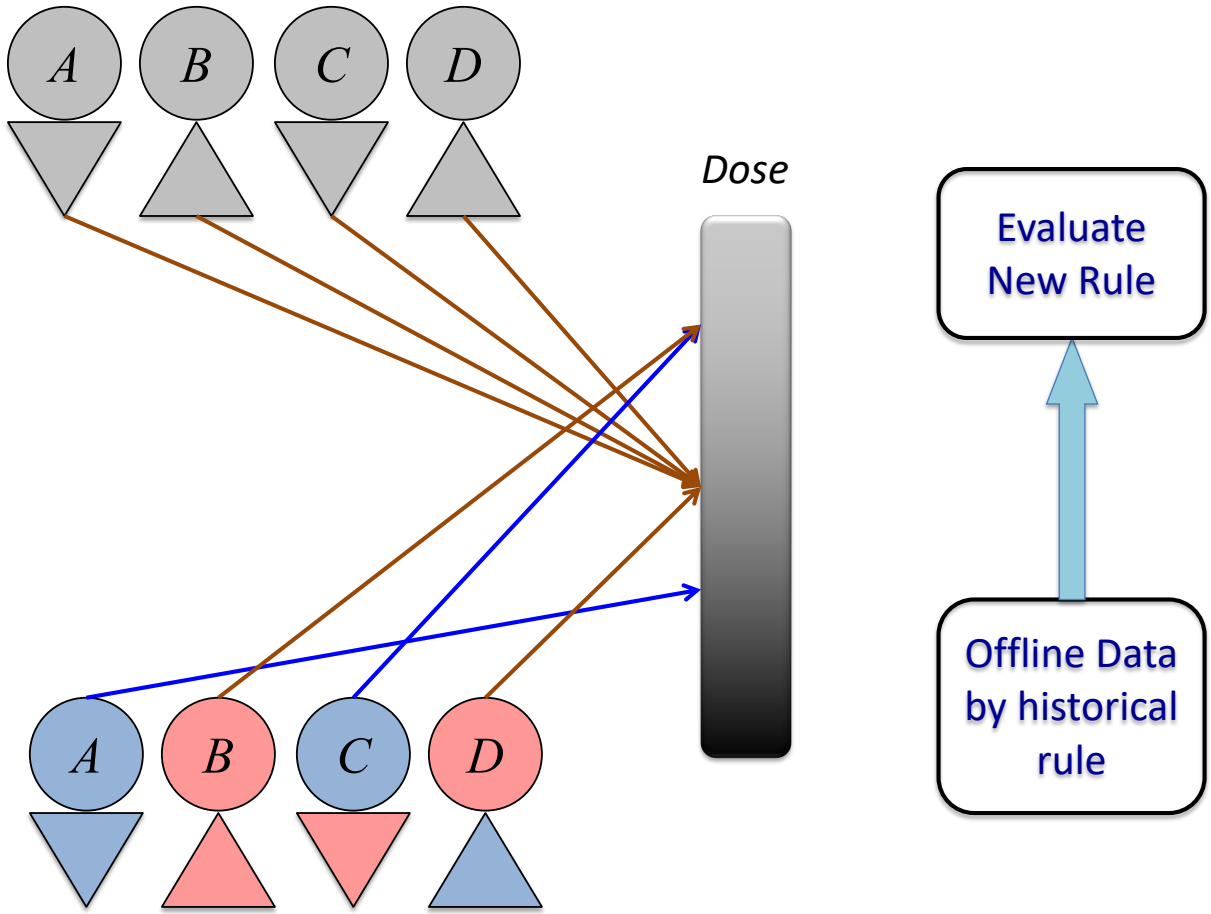
Challenges (2): Policy Evaluation in Continuous Treatment Domains

Prior to adopting any decision rule in practice, it is crucial to know the impact of implementing such a rule.



Challenges (2): Policy Evaluation in Continuous Treatment Domains

It is risky to apply an IDR online to estimate its mean outcome. Policy evaluation proposes to use the offline data from a different historical rule.



Problem Setting

- Data: (X_i, A_i, Y_i), $i = 1, \cdots, n$;
  - $X_i \in \mathcal{X}$: p-dimensional covariates.
  - $A_i \in [0, a_0]$: received dose. w.l.o.g., set $a_0 = 1$.
  - Y_i: outcome of interest, the larger the better.
- Potential outcomes $Y^*(a)$, $a \in [0, 1]$.
- Individualized decision rule (IDR) $d(X)$:
  - $\mathcal{X} \to [0, 1]$.
  - $X \in \mathcal{X} \to \mathcal{I} \subset [0, 1]$, where $\mathcal{I}$ is a subinterval in $[0, 1]$.
- Assume SUTVA, no unmeasured confounders, and the positivity.
- Value: $V(d) = E[Q\{X, d(X)\}]$ with $Q(x, a) = E(Y|X = x, A = a)$. (Also see value function under I2DR.)
- **Goal**: estimate the value $V(d)$ given any target IDR d based on the observed data.

Related Works

- Most of current works on personalized decision making focus on policy optimization not policy evaluation;
  - See e.g.,Chakraborty et al. (2010), Song et al. (2015), Shi et al. (2018).
- Majority of offline policy evaluation methods focus on binary/finite treatment options.
  - See e.g., Wang et al. (2012), Zhang et al. (2012), Chakraborty et al. (2014), Luedtke & Van Der Laan (2016).
- A doubly robust (DR) estimator of $V(d)$ for discrete treatments (see e.g., Zhang et al. 2012):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{\mathbb{I}\{A_i = d(X_i)\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

where $\mathbb{I}(\bullet)$ denotes the indicator function, $\widehat{Q}$ and $\widehat{p}$ denote some estimators for the Q-function and the propensity score function.

Related Works

- Most of current works on personalized decision making focus on policy optimization not policy evaluation;
  - See e.g.,Chakraborty et al. (2010), Song et al. (2015), Shi et al. (2018).
- Majority of offline policy evaluation methods focus on binary/finite treatment options.
  - See e.g., Wang et al. (2012), Zhang et al. (2012), Chakraborty et al. (2014), Luedtke & Van Der Laan (2016).
- A doubly robust (DR) estimator of $V(d)$ for discrete treatments (see e.g., Zhang et al. 2012):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{\mathbb{I}\{A_i = d(X_i)\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

where $\mathbb{I}(\bullet)$ denotes the indicator function, $\widehat{Q}$ and $\widehat{p}$ denote some estimators for the Q-function and the propensity score function.

Related Works

- Most of current works on personalized decision making focus on policy optimization not policy evaluation;
  - See e.g.,Chakraborty et al. (2010), Song et al. (2015), Shi et al. (2018).
- Majority of offline policy evaluation methods focus on binary/finite treatment options.
  - See e.g., Wang et al. (2012), Zhang et al. (2012), Chakraborty et al. (2014), Luedtke & Van Der Laan (2016).
- A doubly robust (DR) estimator of $V(d)$ for discrete treatments (see e.g., Zhang et al. 2012):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{\mathbb{I}\{A_i = d(X_i)\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

where $\mathbb{I}(\bullet)$ denotes the indicator function, $\widehat{Q}$ and $\widehat{p}$ denote some estimators for the Q-function and the propensity score function.

Related Works

- Available methods for continuous treatments rely on the use of a kernel function.
- A DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal.

Related Works

- Available methods for continuous treatments rely on the use of a kernel function.
- A DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal.

Related Works

- Available methods for continuous treatments rely on the use of a kernel function.
- A DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal.

Related Works

- Available methods for continuous treatments rely on the use of a kernel function.
- A DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal.

Summary of Our Work

- Propose deep jump evaluation method for continuous treatments by integrating **multi-scale change point detection**, **deep learning**, and the **doubly-robust** value estimators in discrete domains;
- Our method does not require kernel bandwidth selection, by adaptively discretizing the treatment space using deep discretization;
- Our method has a better convergence rate, allowing the conditional mean outcome to be either a continuous or piecewise function of the treatment.

Summary of Our Work

- Propose deep jump evaluation method for continuous treatments by integrating **multi-scale change point detection**, **deep learning**, and the **doubly-robust** value estimators in discrete domains;
- Our method does not require kernel bandwidth selection, by adaptively discretizing the treatment space using deep discretization;
- Our method has a better convergence rate, allowing the conditional mean outcome to be either a continuous or piecewise function of the treatment.

Summary of Our Work

- Propose deep jump evaluation method for continuous treatments by integrating **multi-scale change point detection**, **deep learning**, and the **doubly-robust** value estimators in discrete domains;
- Our method does not require kernel bandwidth selection, by adaptively discretizing the treatment space using deep discretization;
- Our method has a better convergence rate, allowing the conditional mean outcome to be either a continuous or piecewise function of the treatment.

Deep Jump Learning for Off-Policy Evaluation in Continuous Treatment Settings

Recap: Limitations of Kernel-based Evaluation Methods

- Recall the DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.

- Limitation 1: Require the mean outcome to be smooth over the treatment space;
  - In dynamic pricing, the expected demand for a product has jump discontinuities as a function of the charged price (den Boer & Keskin 2020).
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal;
  - when the second-order derivative of the conditional mean function has an abrupt change in the treatment space.

Recap: Limitations of Kernel-based Evaluation Methods

- Recall the DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
  - In dynamic pricing, the expected demand for a product has jump discontinuities as a function of the charged price (den Boer & Keskin 2020).
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal;
  - when the second-order derivative of the conditional mean function has an abrupt change in the treatment space.

Recap: Limitations of Kernel-based Evaluation Methods

- Recall the DR estimator of $V(d)$ for continuous treatments (see e.g., Kallus & Zhou 2018, Colangelo & Lee 2020):

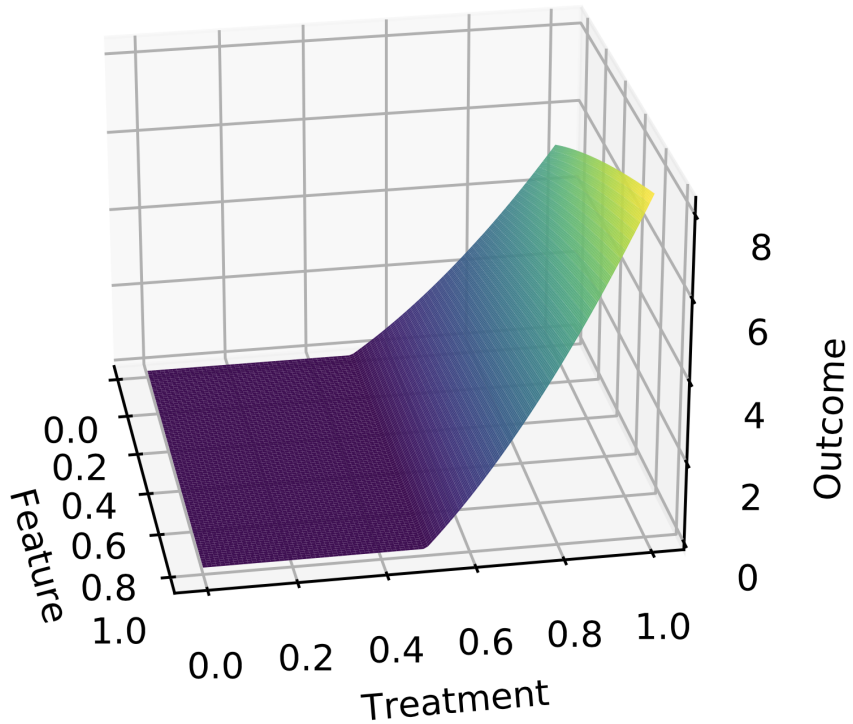
$$\frac{1}{n}\sum_{i=1}^{n}\left[\widehat{Q}\{X_i, d(X_i)\} + \frac{K\{\frac{A_i - d(X_i)}{h}\}}{\widehat{p}(A_i|X_i)}\{Y_i - \widehat{Q}(X_i, A_i)\}\right],$$

  where $K(\cdot)$ is a kernel function and h is the kernel bandwidth.
- Limitation 1: Require the mean outcome to be smooth over the treatment space;
  - In dynamic pricing, the expected demand for a product has jump discontinuities as a function of the charged price (den Boer & Keskin 2020).
- Limitation 2: Use a single bandwidth parameter, which may be sub-optimal;
  - when the second-order derivative of the conditional mean function has an abrupt change in the treatment space.

Toy Example

Consider a **smooth** function $Q(x, a) = 10 \max(a^2 - 0.25, 0) \log(x + 2)$ for any $x, a \in [0, 1]$: with different patterns when the treatment belongs to different intervals:

- For $a \in [0, 0.5]$, $Q(x, a)$ is constant as a function of a.
- For $a \in (0.5, 1]$, $Q(x, a)$ depends quadratically in a.



Sub-optimality of Kernel-Based Method in Toy Example

Target policy: $d(x) = x$; the value $V(d) = V^{(1)}(d) + V^{(2)}(d)$ where

- $V^{(1)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) \leq 0.5\}]$;
- $V^{(2)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) > 0.5\}]$.

Bias (SD)	Indicator	Kernel with $h = 0.4$	Kernel with $h = 1$
$V^{(1)}(d)$	$\mathbb{I}\{d(X) \leq 0.5\}$	0.50 (0.08)	0.40 (0.05)
$V^{(2)}(d)$	$\mathbb{I}\{d(X) > 0.5\}$	0.16 (0.20)	1.09 (0.09)

Due to the use of a single bandwidth, the kernel-based estimator suffers from either a large bias or a large variance.

- By Theorem 1 of Kallus & Zhou (2018), the leading term of bias:

$$h^2 \frac{\int u^2 K(u) du}{2} \mathsf{E}\left\{ \frac{\partial^2 Q(X, a)}{\partial a^2}\bigg|_{a=d(X)} \right\}.$$

Sub-optimality of Kernel-Based Method in Toy Example

Target policy: $d(x) = x$; the value $V(d) = V^{(1)}(d) + V^{(2)}(d)$ where

- $V^{(1)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) \leq 0.5\}]$;
- $V^{(2)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) > 0.5\}]$.

Bias (SD)	Indicator	Kernel with $h = 0.4$	Kernel with $h = 1$
$V^{(1)}(d)$	$\mathbb{I}\{d(X) \leq 0.5\}$	0.50 (0.08)	0.40 (0.05)
$V^{(2)}(d)$	$\mathbb{I}\{d(X) > 0.5\}$	0.16 (0.20)	1.09 (0.09)

Due to the use of a single bandwidth, the kernel-based estimator suffers from either a large bias or a large variance.

- By Theorem 1 of Kallus & Zhou (2018), the leading term of bias:

$$h^2 \frac{\int u^2 K(u) du}{2} \mathsf{E}\left\{ \frac{\partial^2 Q(X, a)}{\partial a^2}\bigg|_{a=d(X)} \right\}.$$

Sub-optimality of Kernel-Based Method in Toy Example

Target policy: $d(x) = x$; the value $V(d) = V^{(1)}(d) + V^{(2)}(d)$ where

- $V^{(1)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) \leq 0.5\}]$;
- $V^{(2)}(d) = \mathsf{E}[Q\{X, d(X)\}\mathbb{I}\{d(X) > 0.5\}]$.

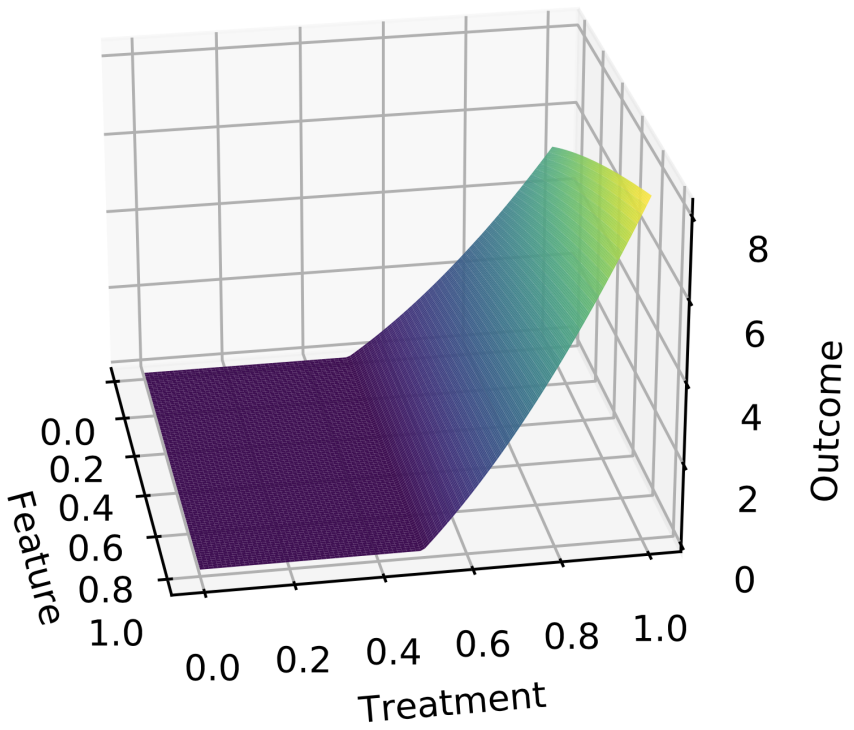
Bias (SD)	Indicator	Kernel with $h = 0.4$	Kernel with $h = 1$
$V^{(1)}(d)$	$\mathbb{I}\{d(X) \leq 0.5\}$	0.50 (0.08)	0.40 (0.05)
$V^{(2)}(d)$	$\mathbb{I}\{d(X) > 0.5\}$	0.16 (0.20)	1.09 (0.09)

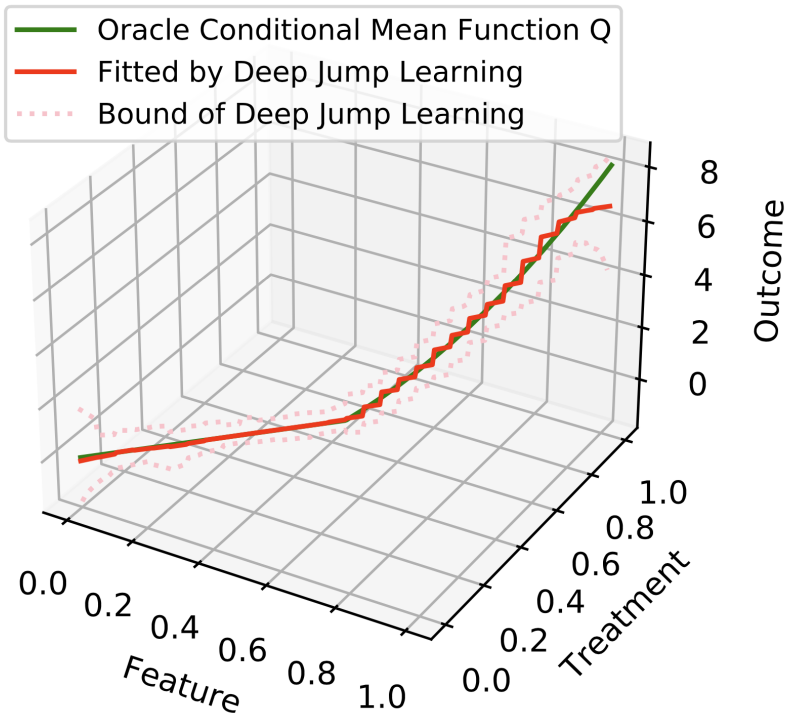
Due to the use of a single bandwidth, the kernel-based estimator suffers from either a large bias or a large variance.

- By Theorem 1 of Kallus & Zhou (2018), the leading term of bias:

$$h^2 \frac{\int u^2 K(u) du}{2} \mathsf{E}\left\{ \frac{\partial^2 Q(X, a)}{\partial a^2}\bigg|_{a=d(X)} \right\}.$$

Motivation from Toy Example: Adaptive Discretization

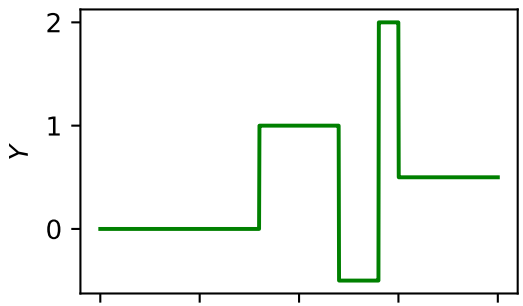


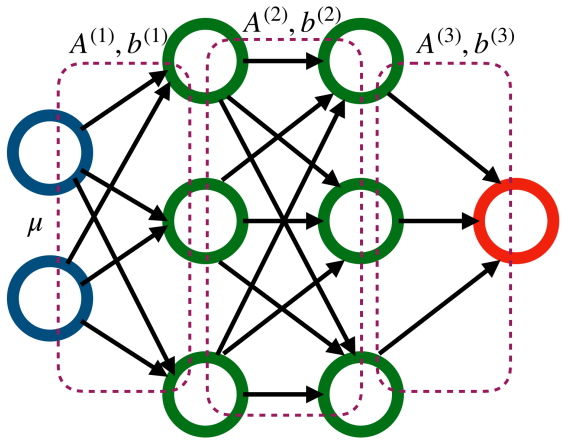


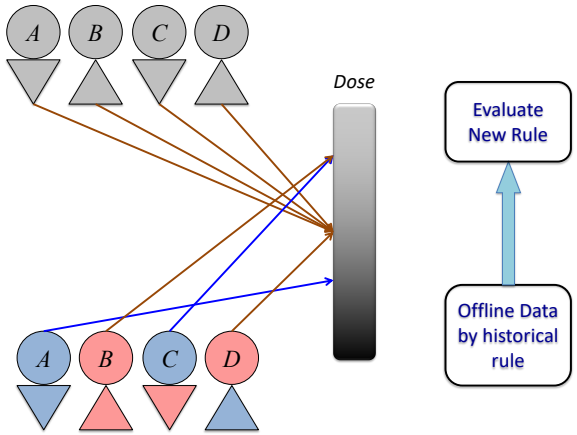
Bias (SD)	Indicator	Deep Jump Learning	Kernel with $h = 0.4$	Kernel with $h = 1$
$V^{(1)}(\pi)$	$\mathbb{I}\{\pi(X) \leq 0.5\}$	0.31 (0.06)	0.50 (0.08)	0.40 (0.05)
$V^{(2)}(\pi)$	$\mathbb{I}\{\pi(X) > 0.5\}$	0.09 (0.19)	0.16 (0.20)	1.09 (0.09)

Deep Jump Evaluation

Deep jump evaluation integrates **multi-scale change point detection**, **deep learning**, and the **doubly-robust** value estimators in discrete domains.







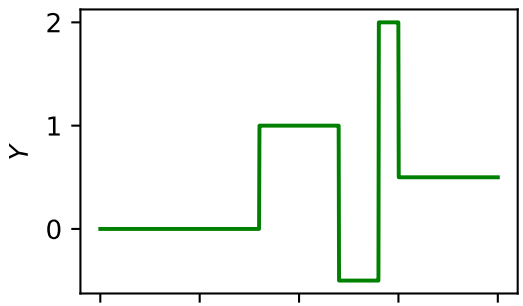
Deep jump evaluation works for both Model I and Model II:

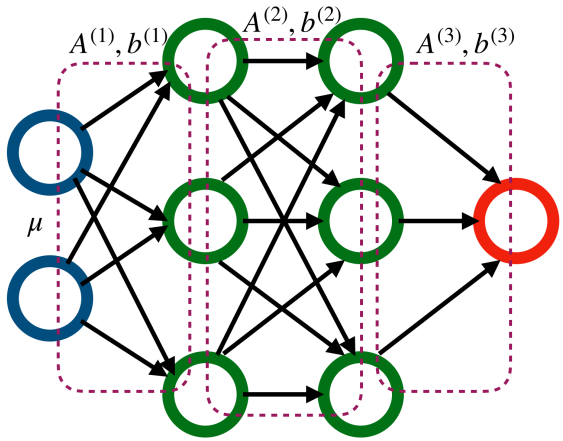
Model I: Piecewise function: $Q(x,a) = \sum_{\mathcal{I}\in\mathcal{P}_0}\{q_{\mathcal{I},0}(x)\mathbb{I}(a\in\mathcal{I})\}$, for some partition $\mathcal{P}_0$ of $[0,1]$ and a collection of functions $\{q_{\mathcal{I},0}\}_{\mathcal{I}\in\mathcal{P}_0}$.

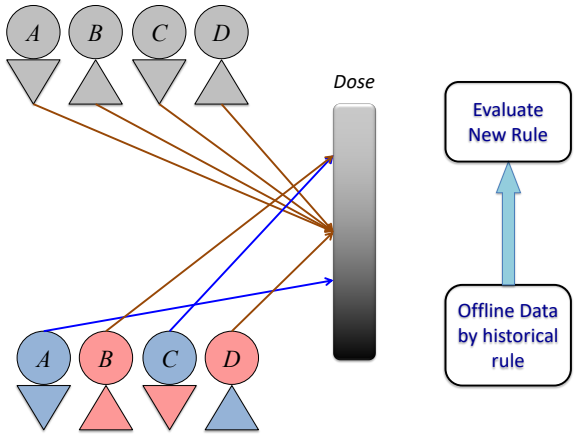
Model II: Continuous function: Q is a continuous function of a and x.

Deep Jump Evaluation

Deep jump evaluation integrates **multi-scale change point detection**, **deep learning**, and the **doubly-robust** value estimators in discrete domains.







Deep jump evaluation works for both Model I and Model II:

Model I: Piecewise function: $Q(x,a) = \sum_{\mathcal{I}\in\mathcal{P}_0} \{q_{\mathcal{I},0}(x)\mathbb{I}(a \in \mathcal{I})\}$, for some partition $\mathcal{P}_0$ of $[0,1]$ and a collection of functions $\{q_{\mathcal{I},0}\}_{\mathcal{I}\in\mathcal{P}_0}$.

Model II: Continuous function: Q is a continuous function of a and x.

Step 1: Deep Discretization

- Recall notations in jump interval learning.
- Model these $q_{\mathcal{I}}$ in some function class of deep neural networks $\mathcal{Q}_{\mathcal{I}}$, to capture the complex dependence between the outcome and features.
- **Estimate Discretization by**:

$$\left(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}\right) = \underset{(\mathcal{P}\in\mathcal{B}(m), \{q_{\mathcal{I}}\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\})}{\arg\min} \left(\sum_{\mathcal{I}\in\mathcal{P}} \left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \{Y_i - q_{\mathcal{I}}(X_i)\}^2 \right] + \gamma_n |\mathcal{P}| \right),$$

for some regularization parameter γ_n.

Step 1: Deep Discretization

- Recall notations in jump interval learning.
- Model these $q_{\mathcal{I}}$ in some function class of deep neural networks $\mathcal{Q}_{\mathcal{I}}$, to capture the complex dependence between the outcome and features.
- **Estimate Discretization by**:

$$\left(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}\right) = \underset{(\mathcal{P}\in\mathcal{B}(m), \{q_{\mathcal{I}}\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\})}{\arg\min} \left(\sum_{\mathcal{I}\in\mathcal{P}} \left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \{Y_i - q_{\mathcal{I}}(X_i)\}^2 \right] + \gamma_n |\mathcal{P}| \right),$$

for some regularization parameter γ_n.

Step 1: Deep Discretization

- Recall notations in jump interval learning.
- Model these $q_{\mathcal{I}}$ in some function class of deep neural networks $\mathcal{Q}_{\mathcal{I}}$, to capture the complex dependence between the outcome and features.
- **Estimate Discretization by**:

$$\left(\widehat{\mathcal{P}}, \{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}\right) = \underset{(\mathcal{P}\in\mathcal{B}(m), \{q_{\mathcal{I}}\in\mathcal{Q}_{\mathcal{I}}:\mathcal{I}\in\mathcal{P}\})}{\arg\min} \left(\sum_{\mathcal{I}\in\mathcal{P}} \left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(A_i \in \mathcal{I}) \{Y_i - q_{\mathcal{I}}(X_i)\}^2 \right] + \gamma_n |\mathcal{P}| \right),$$

for some regularization parameter γ_n.

Step 2: Policy Evaluation

Doubly Robust Estimator under Deep Jump Evaluation

Given $\widehat{\mathcal{P}}$ and $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$, the value for any decision rule of interest d is

$$\widehat{V}(d) = \frac{1}{n} \sum_{\mathcal{I} \in \widehat{\mathcal{P}}} \sum_{i=1}^{n} \left(\mathbb{I}\{d(X_i) \in \mathcal{I}\} \left[\frac{\mathbb{I}(A_i \in \mathcal{I})}{\widehat{p}_{\mathcal{I}}(X_i)} \{Y_i - \widehat{q}_{\mathcal{I}}(X_i)\} + \widehat{q}_{\mathcal{I}}(X_i) \right] \right),$$

where $\widehat{p}_{\mathcal{I}}(x)$ is some estimator of the generalized propensity score function $\Pr(A \in \mathcal{I} | X = x)$.

The complete algorithm consists of:

- Data Splitting: use different subsets of data samples to estimate the discretization and to construct the value estimator.
- Deep Discretization: apply PELT method (Killick et al. 2012) in multi-scale change point detection.
- Cross-fitting: to remove the bias induced by overfitting.

Step 2: Policy Evaluation

Doubly Robust Estimator under Deep Jump Evaluation

Given $\widehat{\mathcal{P}}$ and $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$, the value for any decision rule of interest d is

$$\widehat{V}(d) = \frac{1}{n} \sum_{\mathcal{I} \in \widehat{\mathcal{P}}} \sum_{i=1}^{n} \left(\mathbb{I}\{d(X_i) \in \mathcal{I}\} \left[\frac{\mathbb{I}(A_i \in \mathcal{I})}{\widehat{p}_{\mathcal{I}}(X_i)} \{Y_i - \widehat{q}_{\mathcal{I}}(X_i)\} + \widehat{q}_{\mathcal{I}}(X_i) \right] \right),$$

where $\widehat{p}_{\mathcal{I}}(x)$ is some estimator of the generalized propensity score function $\Pr(A \in \mathcal{I} | X = x)$.

The complete algorithm consists of:

- Data Splitting: use different subsets of data samples to estimate the discretization and to construct the value estimator.
- Deep Discretization: apply PELT method (Killick et al. 2012) in multi-scale change point detection.
- Cross-fitting: to remove the bias induced by overfitting.

Step 2: Policy Evaluation

Doubly Robust Estimator under Deep Jump Evaluation

Given $\widehat{\mathcal{P}}$ and $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$, the value for any decision rule of interest d is

$$\widehat{V}(d) = \frac{1}{n} \sum_{\mathcal{I} \in \widehat{\mathcal{P}}} \sum_{i=1}^{n} \left(\mathbb{I}\{d(X_i) \in \mathcal{I}\} \left[\frac{\mathbb{I}(A_i \in \mathcal{I})}{\widehat{p}_{\mathcal{I}}(X_i)} \{Y_i - \widehat{q}_{\mathcal{I}}(X_i)\} + \widehat{q}_{\mathcal{I}}(X_i) \right] \right),$$

where $\widehat{p}_{\mathcal{I}}(x)$ is some estimator of the generalized propensity score function $\Pr(A \in \mathcal{I} | X = x)$.

The complete algorithm consists of:

- Data Splitting: use different subsets of data samples to estimate the discretization and to construct the value estimator.
- Deep Discretization: apply PELT method (Killick et al. 2012) in multi-scale change point detection.
- Cross-fitting: to remove the bias induced by overfitting.

Step 2: Policy Evaluation

Doubly Robust Estimator under Deep Jump Evaluation

Given $\widehat{\mathcal{P}}$ and $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$, the value for any decision rule of interest d is

$$\widehat{V}(d) = \frac{1}{n} \sum_{\mathcal{I} \in \widehat{\mathcal{P}}} \sum_{i=1}^{n} \left(\mathbb{I}\{d(X_i) \in \mathcal{I}\} \left[\frac{\mathbb{I}(A_i \in \mathcal{I})}{\widehat{p}_{\mathcal{I}}(X_i)} \{Y_i - \widehat{q}_{\mathcal{I}}(X_i)\} + \widehat{q}_{\mathcal{I}}(X_i) \right] \right),$$

where $\widehat{p}_{\mathcal{I}}(x)$ is some estimator of the generalized propensity score function $\Pr(A \in \mathcal{I} | X = x)$.

The complete algorithm consists of:

- Data Splitting: use different subsets of data samples to estimate the discretization and to construct the value estimator.
- Deep Discretization: apply PELT method (Killick et al. 2012) in multi-scale change point detection.
- Cross-fitting: to remove the bias induced by overfitting.

Convergence Rates

Recall assumptions in DNN theories (Farrell et al. 2021): A6 and A7.

Theorem 1 (under Model I (Piecewise Function))

Suppose $m \asymp n$, γ_n *satisfies* $\gamma_n \to 0$ *and* $\gamma_n \gg n^{-2\beta/(2\beta+p)} \log^8 n$. *There exist some classes of DNNs s.t. for any decision rule* d,

$$\widehat{V}(d) = V(d) + O_p\{n^{-2\beta/(2\beta+p)} \log^8 n\} + O_p(n^{-1/2}).$$

Theorem 2 (under Model II (Continuous Function))

Suppose $m \asymp n$ *and* γ_n *is proportional to* $\max\{n^{-3/5}, n^{-2\beta/(2\beta+p)} \log^9 n\}$. *Then for any decision rule* d,

$$\widehat{V}(d) - V(d) = O_p(n^{-1/5}) + O_p\{n^{-2\beta/(6\beta+3p)} \log^3 n\}.$$

When $4\beta > 3p$, *the convergence rate is given by* $O_p(n^{-1/5})$.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Theoretical Comparison with Kernel-Based Methods

Assume the symmetric kernel; set oracle values Q and b; and $4\beta > 3p$.

- Suppose Model I holds.
  - Convergence rate of kernel-based methods is $O_p(n^{-1/3})$ with **optimal bandwidth selection**.
  - The proposed estimator converges at a faster rate of $O_p(n^{-1/2})$.
- Suppose Model II holds.
  - Convergence rate of kernel-based methods is $O_p(h) + O_p(n^{-1/2}h^{-1/2})$.
  - When the second-order derivative of Q has **an abrupt change** in the treatment space, they suffer from either a large bias, or a large variance.
  - When h is either much larger than $n^{-1/5}$ or much smaller than $n^{-3/5}$, our estimator converges at a faster rate of $O_p(n^{-1/5})$.
- Kernel-based estimators could converge at a faster rate when Q has a uniform degree of smoothness over the entire treatment space and the optimal bandwidth parameter is correctly identified.

Simulation Settings

$Y|X, A \sim N\{Q(X, A), 1\}$, $A|X \sim \text{Unif}[0, 1]$ $X^{(1)}, \ldots, X^{(p)} \overset{iid}{\sim} \text{Unif}[-1, 1]$, where $p = 20$ and consider the following four scenarios for $Q(X, A)$:

- **Scenario 1**

$$\begin{aligned} Q(x, a) =& (1 + x^{(1)})\mathbb{I}(a < 0.35) + (x^{(1)} - x^{(2)})\mathbb{I}(0.35 \leq a < 0.65) \\ &+ (1 - x^{(2)})\mathbb{I}(a \geq 0.65). \end{aligned}$$

- **Scenario 2**

$$\begin{aligned} Q(x, a) =& \mathbb{I}(a < 0.25) + \sin(2\pi x^{(1)})\mathbb{I}(0.25 \leq a < 0.5) \\ &+ \{0.5 - 8(x^{(1)} - 0.75)^2\}\mathbb{I}(0.5 \leq a < 0.75) + 0.5\mathbb{I}(a \geq 0.75). \end{aligned}$$

- **Scenario 3 (toy example)**

$$Q(x, a) = 10 \max\{a^2 - 0.25, 0\} \log(x^{(1)} + 2).$$

- **Scenario 4**

$$Q(x, a) = 0.2(8 + 4x^{(1)} - 2x^{(2)} - 2x^{(3)}) - 2(1 + 0.5x^{(1)} + 0.5x^{(2)} - 2a)^2.$$

Target policy: the optimal policy that achieves the highest mean outcome.

Simulation Results I

- SLOPE by Su et al. (2020): adopt the Lepski's method for bandwidth selection.
- Kallus & Zhou (2018): compute h^* using data with sample size $n_0 = 50$, and adjust h^* by setting $h^*\{n_0/n\}^{0.2}$ for different n.
- Colangelo & Lee (2020): manually select the best bandwidth by $h = c\sigma_A n^{-0.2}$ with $c \in \{0.5, 0.75, 1.0, 1.5\}$.

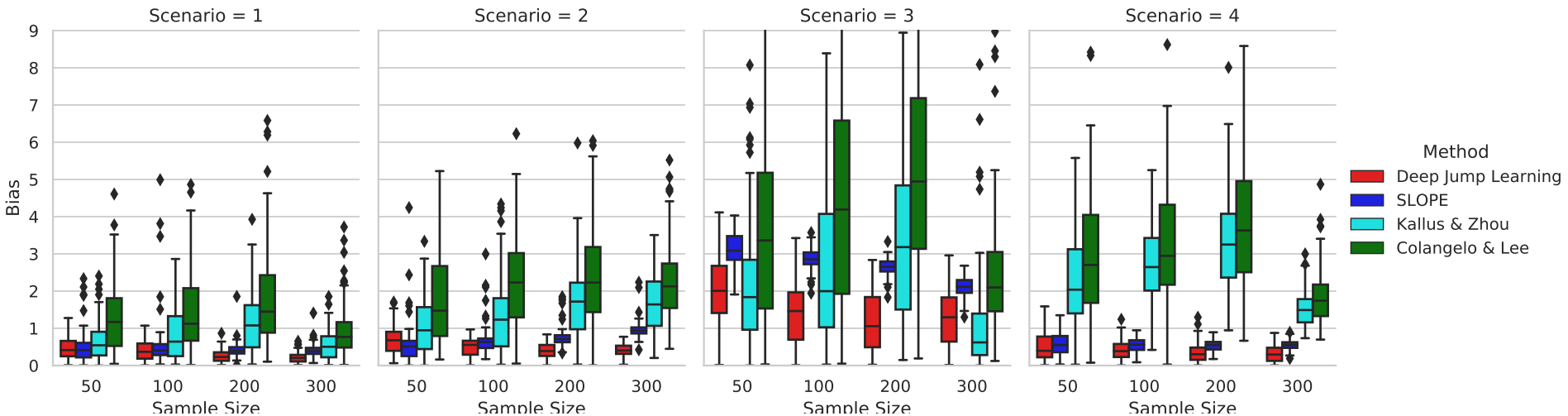


Figure 3: The target values are 1.33, 1, 4.86 and 1.6, respectively.

Simulation Results I

- SLOPE by Su et al. (2020): adopt the Lepski's method for bandwidth selection.
- Kallus & Zhou (2018): compute h^* using data with sample size $n_0 = 50$, and adjust h^* by setting $h^*\{n_0/n\}^{0.2}$ for different n.
- Colangelo & Lee (2020): manually select the best bandwidth by $h = c\sigma_A n^{-0.2}$ with $c \in \{0.5, 0.75, 1.0, 1.5\}$.

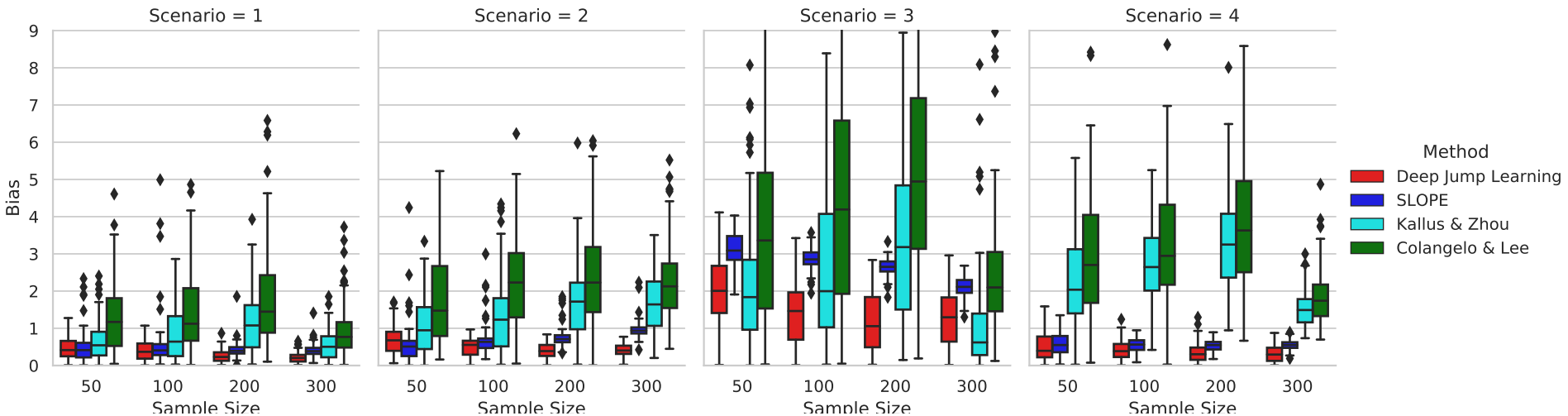


Figure 3: The target values are 1.33, 1, 4.86 and 1.6, respectively.

Simulation Results II

Table 1: The averaged computational cost (in minutes) for Scenario 1.

Methods	DJL	SLOPE (Su et al. 2020)	Kallus & Zhou (2018)	Colangelo & Lee (2020)
$n = 50$	< 1	< 1	365	< 1
$n = 100$	3	< 1	773	< 1
$n = 200$	7	1	> 1440 (24 hours)	< 1
$n = 300$	14	2	> 2880 (48 hours)	< 1

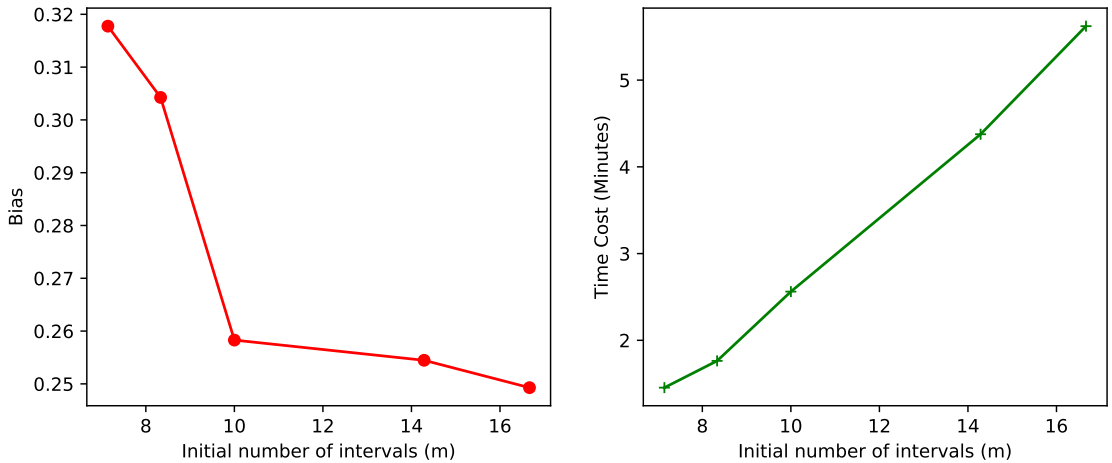


Figure 4: The bias and the time cost of DJL with different m for $n = 100$ in S1.

Real Data Analysis: Warfarin Dosing

- Use $p = 81$ baseline covariates.
- Calibrate data for evaluation:
  - Fit $\widehat{Q}(X, A)$ based on DNN. Randomly sample (a_j, x_j) from $\{(A_1, X_1), \cdots, (A_n, X_n)\}$ with replacement;
  - For each j, generate y_j according to $N\{\widehat{Q}(x_j, a_j), \widehat{\sigma}^2\}$, where $\widehat{\sigma}$ is the standard deviation of the fitted residual $\{Y_i - \widehat{Q}(X_i, A_i)\}_i$.
- Decision rule of interest: $d^\star(X) \equiv \arg\max_{a \in [0,1]} \widehat{Q}(X, a)$, with target value as -0.278.

Methods	Bias	Standard deviation	Mean squared error
Deep Jump Learning	0.259	0.416	0.240
SLOPE (Su et al. 2020)	0.611	0.755	0.943
Kallus & Zhou (2018)	0.662	0.742	0.989
Colangelo & Lee (2020)	0.442	1.164	1.550

Real Data Analysis: Warfarin Dosing

- Use $p = 81$ baseline covariates.
- Calibrate data for evaluation:
  - Fit $\widehat{Q}(X, A)$ based on DNN. Randomly sample (a_j, x_j) from $\{(A_1, X_1), \cdots, (A_n, X_n)\}$ with replacement;
  - For each j, generate y_j according to $N\{\widehat{Q}(x_j, a_j), \widehat{\sigma}^2\}$, where $\widehat{\sigma}$ is the standard deviation of the fitted residual $\{Y_i - \widehat{Q}(X_i, A_i)\}_i$.
- Decision rule of interest: $d^\star(X) \equiv \arg\max_{a \in [0,1]} \widehat{Q}(X, a)$, with target value as -0.278.

Methods	Bias	Standard deviation	Mean squared error
Deep Jump Learning	0.259	0.416	0.240
SLOPE (Su et al. 2020)	0.611	0.755	0.943
Kallus & Zhou (2018)	0.662	0.742	0.989
Colangelo & Lee (2020)	0.442	1.164	1.550

Thank You!

Appendix: Summary of JIL Algorithm

Global: data $\{(X_i, A_i, Y_i) : i = 1, \ldots, n\}$; sample size n; covariates dimension p; number of initial intervals m; penalty terms γ_n.

Local: integers $l, r \in \mathbb{N}$; cost dictionary $\mathcal{C}$; a vector of integers $\tau \in \mathbb{N}^m$; Bellman function $B \in \mathbb{R}^m$; a set of candidate point lists $\mathcal{R}$.

Output: $\widehat{\mathcal{P}}$ and $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$.

I. Initialization.

1. Set $B(0) \leftarrow -\gamma_n$; $\widehat{\mathcal{P}} \leftarrow Null$; $\tau \leftarrow Null$; $\mathcal{R}(0) \leftarrow \{0\}$;
2. Define the cost function $\mathcal{C}(\mathcal{I})$:
   (i). If $\mathcal{C}(\mathcal{I}) \leftarrow NULL$:
      (a). Apply Linear / MLP regression: $\widehat{q}_{\mathcal{I}}(\cdot) \leftarrow \arg\min_q \sum_i \mathbb{I}(A_i \in \mathcal{I})\{Y_i - q(X_i)\}^2$;
      (b). Calculate the cost: $\mathcal{C}(\mathcal{I}) \leftarrow \sum_i \mathbb{I}(A_i \in \mathcal{I})\{\widehat{q}_{\mathcal{I}}(X_i) - Y_i\}^2$;
   (ii). Return $\mathcal{C}(\mathcal{I})$.

II. Apply the PELT method. For $r = 1, \ldots, m$:

1. $B(r) = \min_{j \in \mathcal{R}(r)}\{B(j) + \mathcal{C}([j/m, r/m)) + \gamma_n\}$;
2. $j^* \leftarrow \arg\min_{j \in \mathcal{R}(r)}\{B(j) + \mathcal{C}([j/m, r/m)) + \gamma_n\}$;
3. $\tau(r) \leftarrow \{j^*, \tau(j^*)\}$;
4. $\mathcal{R}(r) \leftarrow \{j \in \mathcal{R}(r-1) \cup \{r-1\} : B(j) + \mathcal{C}([j/m, (r-1)/m)) \le B(r-1)\}$;

III. Get Partitions. $\tau^* \leftarrow \tau(m)$; $r \leftarrow m$; $l \leftarrow \tau^*[r]$; While $r > 0$:

1. Let $\mathcal{I} = [l/m, r/m)$ if $r < m$ else $\mathcal{I} = [l/m, 1]$;
2. $\widehat{\mathcal{P}} \leftarrow \widehat{\mathcal{P}} \cup \mathcal{I}$;
3. $\widehat{q}_{\mathcal{I}}(\cdot) \leftarrow \arg\min_q \sum_i \mathbb{I}(A_i \in \mathcal{I})\{Y_i - q(X_i)\}^2$;
4. $r \leftarrow l$; $l \leftarrow \tau^*[r]$;

return $\widehat{\mathcal{P}}$ *and* $\{\widehat{q}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{P}}\}$.

Appendix: Tuning Parameters Selection

- For **initial number of intervals** m, we recommend to set $m = n/c$ with some constant $c > 0$ such that m and n are of the same order.
- For **linear-JIL**, we choose γ_n and λ_n simultaneously via cross-validation. We develop an algorithm to facilitate the computation.
- For **deep-JIL**, find that the MLP regressor is not overly sensitive to the choice of λ_n, so we set $\lambda_n = 0$. The parameter γ_n is chosen based on cross-validation.

Appendix: Technical Assumptions

A4 [Condition on Tails]There exists some constant $\omega > 0$ such that $\|X^{(j)}\|_{\psi_2|A} \leq \omega$, for any $j \in \{1, \ldots, p\}$ and $\|Y\|_{\psi_2|A} \leq \omega$ almost surely. Here, $\|Z\|_{\psi_2|A}$ denotes the conditional Orlicz norm of Z given the dose level A.

A5 [Margin Condition] For any $\mathcal{I}_1, \mathcal{I}_2 \in \mathcal{P}_0$, there exist some constants $\gamma, \delta_0 > 0$ such that

$$\Pr(0 < |q_{\mathcal{I}_1,0}(X) - q_{\mathcal{I}_2,0}(X)| \leq t) = O(t^{\gamma}),$$

where the big-O term is uniform in $0 < t \leq \delta_0$.

A6 Suppose $Q(x, a)$ and $p(a|x)$ belong to the class of β-smooth functions in terms of x, for any a.

A7 Functions $\{\widehat{q}_{\mathcal{I}}\}_{\mathcal{I} \in \widehat{\mathcal{P}}}$ are uniformly bounded.

Appendix: Technical Assumptions

A8 $[\mathrm{E}\{\widehat{e}(\mathcal{I}|X) - e(\mathcal{I}|X)\}^2]^{1/2} = o(n^{-1/4})$ and $\widehat{e}(\mathcal{I}|X)$ belongs to the class of VC-type functions with VC-index upper bounded by $O(n^{1/2})$. $\{\widehat{e}_{\mathcal{I}}\}_{\mathcal{I}\in\widehat{\mathcal{P}}}$ are uniformly bounded away from zero.

A9 Suppose there exist some constants $L > 0$, $0 < \alpha_0 \leq 1$ such that $\theta_0(\cdot)$ satisfies $\sup_{a_1,a_2\in[0,1]} \|\theta_0(a_1) - \theta_0(a_2)\|_2 \leq L|a_1 - a_2|^{\alpha_0}$.

Appendix: Simulation Results III: Deep-JIL with Different $m = n/c$

	n	50	100	200	400	800
Scenario 1	$c = 6$	0.941(0.012)	0.972(0.008)	1.028(0.004)	1.065(0.004)	1.127(0.001)
$V = 1.34$	$c = 8$	0.973(0.016)	0.990(0.008)	1.030(0.004)	1.053(0.005)	1.136(0.001)
$p = 20$	$c = 10$	0.914(0.012)	0.967(0.008)	1.050(0.005)	1.071(0.005)	1.138(0.001)
Scenario 2	$c = 6$	0.943(0.013)	0.980(0.008)	1.037(0.004)	1.087(0.003)	1.129(0.001)
$V = 1.35$	$c = 8$	1.002(0.015)	1.012(0.008)	1.039(0.004)	1.076(0.003)	1.137(0.001)
$p = 20$	$c = 10$	0.900(0.012)	0.978(0.008)	1.074(0.004)	1.102(0.003)	1.141(0.001)
Scenario 3	$c = 6$	0.475(0.018)	0.480(0.009)	0.481(0.006)	0.493(0.004)	0.521(0.002)
$V = 0.76$	$c = 8$	0.416(0.019)	0.497(0.009)	0.493(0.006)	0.506(0.003)	0.532(0.002)
$p = 20$	$c = 10$	0.453(0.019)	0.469(0.009)	0.511(0.005)	0.526(0.004)	0.545(0.002)
Scenario 4	$c = 6$	0.624(0.014)	0.655(0.008)	0.686(0.004)	0.687(0.005)	0.801(0.001)
$V = 1.28$	$c = 8$	0.622(0.014)	0.651(0.008)	0.684(0.004)	0.676(0.005)	0.801(0.001)
$p = 20$	$c = 10$	0.612(0.014)	0.651(0.008)	0.684(0.004)	0.653(0.006)	0.801(0.001)
Scenario 5	$c = 6$	5.49(0.06)	5.69(0.03)	5.82(0.02)	5.97(0.01)	6.12(0.01)
$V = 8.00$	$c = 8$	5.58(0.05)	5.77(0.03)	5.91(0.02)	6.04(0.01)	6.20(0.01)
$p = 20$	$c = 10$	5.57(0.06)	5.79(0.03)	5.97(0.02)	6.10(0.01)	6.26(0.01)

Appendix: Summary of DJL Algorithm

Global: data $\{(X_i, A_i, Y_i)\}_{1\le i\le n}$; number of initial intervals m; penalty term γ_n; target policy π.
Local: an upper triangular matrix of cost $\mathcal{C} \in \mathbb{R}^{m(m+1)/2}$; Bellman function Bell $\in \mathbb{R}^m$; partitions $\widehat{\mathcal{D}}$;
DNN functions $\{\widehat{q}_{\mathcal{I}}, \widehat{b}_{\mathcal{I}} : \mathcal{I} \in \widehat{\mathcal{D}}\}$; a vector $\tau \in \mathbb{N}^m$; a set of candidate point lists $\mathcal{R}$.
Output: the value estimator for target policy $\widehat{V}(\pi)$.
I. Split all n samples into $\mathcal{L}$ subsets as $\{\mathbb{L}_1, \cdots, \mathbb{L}_{\mathcal{L}}\}$; $\widehat{V}(\pi) \leftarrow 0$;
II. Initialize an even segment on the action space with m pieces:
$\{\mathcal{I}\} = \{[0, 1/m), [1/m, 2/m), \ldots, [(m-1)/m, 1]\}$;
III. For $\ell = 1, \cdots, \mathcal{L}$:
1. Set the training dataset as $\mathbb{L}_\ell^c = \{1, 2, \cdots, n\} - \mathbb{L}_\ell$;
2. Bell$(0) \leftarrow -\gamma_n$; $\widehat{\mathcal{D}} = [0, 1]$; $\tau \leftarrow Null$; $\mathcal{R}(0) \leftarrow \{0\}$;
3. Collect cost function:
For $r = 1, \ldots, m$: For $l = 0, \ldots, (r-1)$:
(i). Let $\mathcal{I} = [l/m, r/m)$ if $r < m$ else $\mathcal{I} = [l/m, 1]$;
(ii). Fit a DNN regressor: $\widehat{q}_{\mathcal{I}}(\cdot) \leftarrow \mathbb{I}(i \in \mathbb{L}_\ell^c)\mathbb{I}(A_i \in \mathcal{I})Y_i \sim \mathbb{I}(A_i \in \mathcal{I})DNN(X_i)$;
(iii). Calculate the cost: $\mathcal{C}(\mathcal{I}) \leftarrow \sum_{i\in\mathbb{L}_\ell^c} \mathbb{I}(A_i \in \mathcal{I})\{\widehat{q}_{\mathcal{I}}(X_i) - Y_i\}^2$;
4. Apply the pruned exact linear time method to get partitions: For $v^* = 1, \ldots, m$:
(i).Bell$(v^*) = \min_{v\in\mathcal{R}(v^*)}\{\text{Bell}(v) + \mathcal{C}([v/m, v^*/m)) + \gamma_n\}$;
(ii). $v^1 \leftarrow \arg\min_{v\in\mathcal{R}(v^*)}\{\text{Bell}(v) + \mathcal{C}([v/m, v^*/m)) + \gamma_n\}$;
(iii). $\tau(v^*) \leftarrow \{v^1, \tau(v^1)\}$;
(iv). $\mathcal{R}(v^*) \leftarrow \{v \in \mathcal{R}(v^*-1) \cup \{v^*-1\} : \text{Bell}(v) + \mathcal{C}([v/m, (v^*-1)/m)) \le \text{Bell}(v^*-1)\}$;
5. Construct the DR value estimator: $r \leftarrow m$; $l \leftarrow \tau[r]$; While $r > 0$:
(i) Let $\mathcal{I} = [l/m, r/m)$ if $r < m$ else $\mathcal{I} = [l/m, 1]$; $\widehat{\mathcal{D}} \leftarrow \widehat{\mathcal{D}} \cup \mathcal{I}$;
(ii) Recall fitted DNN: $\widehat{q}_{\mathcal{I}}(\cdot) \leftarrow \mathbb{I}(i \in \mathbb{L}_\ell^c)\mathbb{I}(A_i \in \mathcal{I})Y_i \sim \mathbb{I}(A_i \in \mathcal{I})DNN(X_i)$;
(iii) Fit propensity score: $\widehat{b}_{\mathcal{I}}(\cdot) \leftarrow \mathbb{I}(i \in \mathbb{L}_\ell^c)\mathbb{I}(A_i \in \mathcal{I}) \sim \mathbb{I}(A_i \in \mathcal{I})DNN(X_i)$;
(iv) $r \leftarrow l$; $l \leftarrow \tau(r)$;
6. Evaluation using testing dataset $\mathbb{L}_\ell$:
$\widehat{V}(\pi) += \sum_{\mathcal{I}\in\widehat{\mathcal{D}}}\left(\sum_{i\in\mathbb{L}_\ell}\mathbb{I}(A_i \in \mathcal{I})\left[\frac{\mathbb{I}\{\pi(X_i)\in\mathcal{I}\}}{\widehat{b}_{\mathcal{I}}(X_i)}\{Y_i - \widehat{q}_{\mathcal{I}}(X_i)\} + \widehat{q}_{\mathcal{I}}(X_i)\right]\right)$;
return $\widehat{V}(\pi)/n$.

Chakraborty, B., Laber, E. B. & Zhao, Y.-Q. (2014), 'Inference about the expected performance of a data-driven dynamic treatment regime', *Clinical Trials* **11**(4), 408–417.

Chakraborty, B., Murphy, S. & Strecher, V. (2010), 'Inference for non-regular parameters in optimal dynamic treatment regimes', *Stat. Methods Med. Res.* **19**(3), 317–343.

Chen, G., Zeng, D. & Kosorok, M. R. (2016), 'Personalized dose finding using outcome weighted learning', *Journal of the American Statistical Association* **111**(516), 1509–1521.

Colangelo, K. & Lee, Y.-Y. (2020), 'Double debiased machine learning nonparametric inference with continuous treatments', *arXiv preprint arXiv:2004.03036* .

Consortium, I. W. P. (2009), 'Estimation of the warfarin dose with clinical and pharmacogenetic data', *New England Journal of Medicine* **360**(8), 753–764.

den Boer, A. V. & Keskin, N. B. (2020), 'Discontinuous demand functions: estimation and pricing', *Management Science* .

Farrell, M. H., Liang, T. & Misra, S. (2021), 'Deep neural networks for estimation and inference', *Econometrica* **89**(1), 181–213.

Friedrich, F., Kempe, A., Liebscher, V. & Winkler, G. (2008), 'Complexity penalized m-estimation: fast computation', *Journal of Computational and Graphical Statistics* **17**(1), 201–224.

Kallus, N. & Zhou, A. (2018), 'Policy evaluation and optimization with continuous treatments', *arXiv preprint arXiv:1802.06037* .

Killick, R., Fearnhead, P. & Eckley, I. A. (2012), 'Optimal detection of changepoints with a linear computational cost', *Journal of the American Statistical Association* **107**(500), 1590–1598.

Kuruvilla, M. & Gurk-Turner, C. (2001), 'A review of warfarin dosing and monitoring', *Proceedings (Baylor University. Medical Center)* **14**(3), 305.

Laber, E. B. & Zhao, Y.-Q. (2015), 'Tree-based methods for individualized treatment regimes', *Biometrika* **102**(3), 501–514.

Luedtke, A. R. & Van Der Laan, M. J. (2016), 'Statistical inference for the mean outcome under a possibly non-unique optimal treatment strategy', *Annals of statistics* **44**(2), 713.

Murphy, S. A. (2003), 'Optimal dynamic treatment regimes', *J. R. Stat. Soc. Ser. B Stat. Methodol.* **65**(2), 331–366.
URL: *https://doi.org/10.1111/1467-9868.00389*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011), 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research* **12**, 2825–2830.

Rich, B., Moodie, E. E. & Stephens, D. A. (2014), 'Simulating sequential multiple assignment randomized trials to generate optimal personalized warfarin dosing strategies', *Clinical trials* **11**(4), 435–444.

Rotschafer, J. C., Crossley, K., Zaske, D., Mead, K., Sawchuk, R. & Solem, L. (1982), 'Pharmacokinetics of vancomycin: observations in 28 patients and dosage recommendations.', *Antimicrobial Agents and Chemotherapy* **22**(3), 391–394.

Shi, C., Fan, A., Song, R. & Lu, W. (2018), 'High-dimensional a-learning for optimal dynamic treatment regimes', *Annals of statistics* **46**(3), 925.

Song, R., Wang, W., Zeng, D. & Kosorok, M. R. (2015), 'Penalized q-learning for dynamic treatment regimens', *Statistica Sinica* **25**(3), 901.

Su, Y., Srinath, P. & Krishnamurthy, A. (2020), Adaptive estimator selection for off-policy evaluation, *in* 'International Conference on Machine Learning', PMLR, pp. 9196–9205.

Wang, L., Rotnitzky, A., Lin, X., Millikan, R. E. & Thall, P. F. (2012), 'Evaluation of viable dynamic treatment regimes in a sequentially randomized trial of advanced prostate cancer', *Journal of the American Statistical Association* **107**(498), 493–508.

Watkins, C. & Dayan, P. (1992), 'Q-learning', *Mach. Learn.* **8**, 279–292.

Zhang, B., Tsiatis, A. A., Laber, E. B. & Davidian, M. (2012), 'A robust method for estimating optimal treatment regimes', *Biometrics* **68**, 1010–1018.

Zhang, B., Tsiatis, A. A., Laber, E. B. & Davidian, M. (2013), 'Robust estimation of optimal dynamic treatment regimes for sequential treatment decisions', *Biometrika* **100**(3), 681–694.

Zhao, Y.-Q., Zeng, D., Laber, E. B. & Kosorok, M. R. (2015), 'New statistical learning methods for estimating optimal dynamic treatment regimes', *J. Amer. Statist. Assoc.* **110**(510), 583–598.

Zhao, Y., Zeng, D., Rush, A. J. & Kosorok, M. R. (2012), 'Estimating individualized treatment rules using outcome weighted learning', *J. Amer. Statist. Assoc.* **107**(499), 1106–1118.

Zhu, L., Lu, W., Kosorok, M. R. & Song, R. (2020), Kernel assisted learning for personalized dose finding, *in* 'Proceedings of the 26th ACM

SIGKDD International Conference on Knowledge Discovery & Data Mining', pp. 56–65.