Estimation of Optimal Dynamic Treatment Regimes

Yingqi Zhao
University of Wisconsi-Madison

2014 IMPACT Symposium III

November 21, 2014

Acknowledgements

- Joint work with Eric B. Laber.
- Symposium Organizers

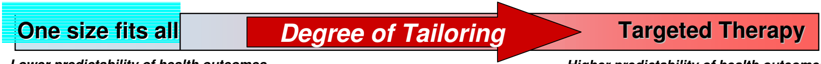
Table of Contents

1. Overview: Dynamic Treatment Regimes
   - What are DTRs?
   - Why use DTRs?
2. Constructing Optimal DTRs from Data: Q-learning
   - Introduction to Q-learning
   - Q-learning: Pros and Cons
3. Value Maximization Methods

Tailored Therapies

“Providing meaningful *improved health outcomes for patients* by delivering the right drug at the right dose at the right time.”

***Goal*: Improve individual patient outcomes and health outcome predictability through tailoring drug, dose, timing of treatment, and relevant information.**



One size fits all

Degree of Tailoring

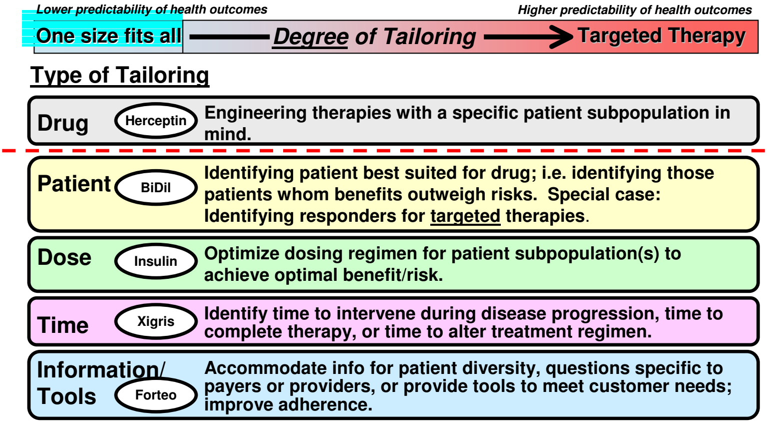
Targeted Therapy

Lower predictability of health outcomes (e.g. most pharma products today)

assess spectrum of patient response to therapy;
stratify patient populations;
optimize benefit/risk.

Higher predictability of health outcomes (e.g. oncology products comprising drug and companion diagnostic)

Tailored Therapies



Lower predictability of health outcomes

Higher predictability of health outcomes

One size fits all → ***Degree of Tailoring*** → Targeted Therapy

Type of Tailoring

Drug	Herceptin	Engineering therapies with a specific patient subpopulation in mind.
Patient	BiDil	Identifying patient best suited for drug; i.e. identifying those patients whom benefits outweigh risks. Special case: Identifying responders for targeted therapies.
Dose	Insulin	Optimize dosing regimen for patient subpopulation(s) to achieve optimal benefit/risk.
Time	Xigris	Identify time to intervene during disease progression, time to complete therapy, or time to alter treatment regimen.
Information/ Tools	Forteo	Accommodate info for patient diversity, questions specific to payers or providers, or provide tools to meet customer needs; improve adherence.

Can apply one or more scenarios to a compound.
Scenarios can often be interdependent.

Dynamic Treatment Regime

- At any decision point
    - Input: available information on the patient to that point.
    - Output: next treatment.
- Dynamic treatment regimes (DTRs) are sequential *decision rules* for individual patients that can adapt over time to an evolving illness.
    - One decision rule for each time point.
    - Each rule: recommends the treatment action at that point as a function of accrued historical information.
    - The rules determine an algorithm for treating any patient.
    - Aim to optimize some cumulative clinical outcome.

Table of Contents

1. Overview: Dynamic Treatment Regimes
   - What are DTRs?
   - Why use DTRs?
2. Constructing Optimal DTRs from Data: Q-learning
   - Introduction to Q-learning
   - Q-learning: Pros and Cons
3. Value Maximization Methods

Why Do We Need DTRs?

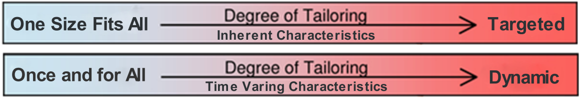
- Heterogeneity

- Multiple active treatments available.
- Heterogeneity in responses:
  1. Across patients: what works for one may not work for another.
  2. Within a patient: what works now may not work later.

- Chronic or Waxing and Waning Course
- More is not always better

DTR Goals

Learn adaptive treatment strategies: tailor (sequences of) treatments based on patient characteristics.



One Size Fits All → Degree of Tailoring, Inherent Characteristics → Targeted

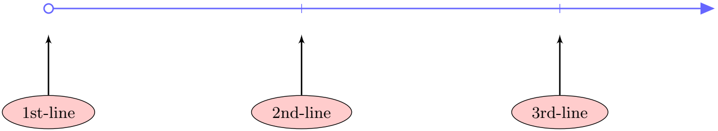
Once and for All → Degree of Tailoring, Time Varing Characteristics → Dynamic

Maximize the benefit of dynamic treatment regimes:

- Well chosen tailoring variables.
- Well conceived decision rules.

Examples: Late Stage Non-Small Cell Lung Cancer

In treating advanced non-small cell lung cancer, patients typically experience two or more lines of treatment, and many studies demonstrate that three lines of treatment can improve survival for patients.



NSCLC: Important clinical questions

1. Among many approved 1st-line treatments, what treatment to administer?
2. Then, at the end of the 1st-line treatment
   - Among approved 2nd-line treatments, what treatment to administer?
   - When to begin the 2nd-line of treatment?
3. Goal: Improve survival.

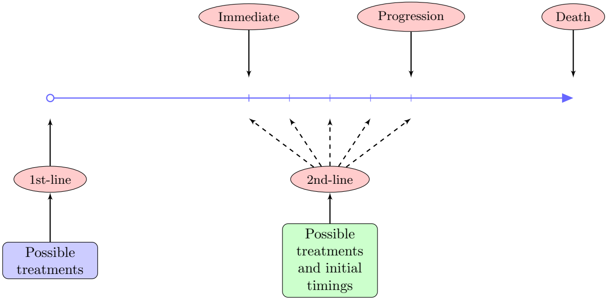


Table of Contents

1. Overview: Dynamic Treatment Regimes
    - What are DTRs?
    - Why use DTRs?
2. Constructing Optimal DTRs from Data: Q-learning
    - Introduction to Q-learning
    - Q-learning: Pros and Cons
3. Value Maximization Methods

Sequential Multiple Assignment Randomized Trials (SMARTs) for DTRs

These are multi-stage trials; each stage corresponds to a critical decision and a randomization takes place at each critical decision.

Goal

Inform the construction of dynamic treatment regimes.

Dynamic Treatment Regime

Observe data on n individuals, T stages for each individual,

$$X_1, A_1, R_1, X_2, A_2, \ldots, X_T, A_T, R_T, X_{T+1}$$

X_t: Patient covariates available at stage t.
A_t: Treatment at stage t, $A_t \in \{-1, 1\}$.
R_t: Outcome following stage t.
H_t: History available at stage t, $H_t = \{X_1, A_1, R_1, \ldots, A_{t-1}, R_{t-1}, X_t\}$.

A DTR is a sequence of decision rules:

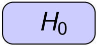
$$\mathcal{D} = (d_1(H_1), \ldots, d_T(H_T)), d_t(H_t) \in \{-1, 1\}.$$

- The regime, $\mathcal{D}$, should have high Value: $V^{\mathcal{D}} = E^{\mathcal{D}}\left(\sum_t R_t\right)$
  - The value corresponds to the average outcome if all patients are assigned treatment according to $\mathcal{D}$
  - Optimal decision rule $\mathcal{D}^{\mathrm{opt}}$ satisfies

$$E^{\mathcal{D}^{\mathrm{opt}}}(\sum_t R_t) = \sup_{\mathcal{D}} E^{\mathcal{D}}(\sum_t R_t)$$

Dynamic Programming

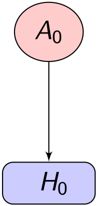
- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



H_0

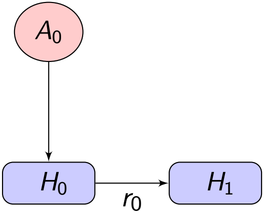
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



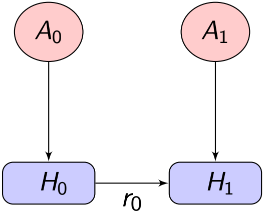
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



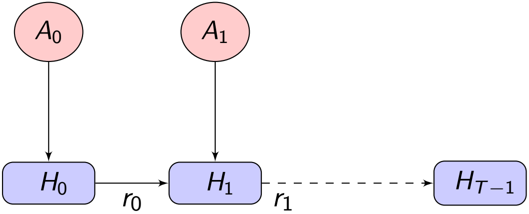
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



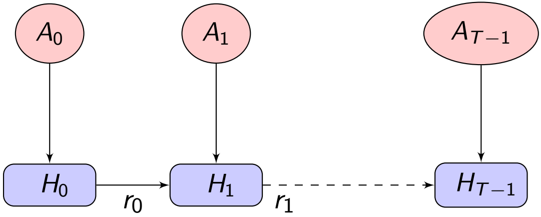
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



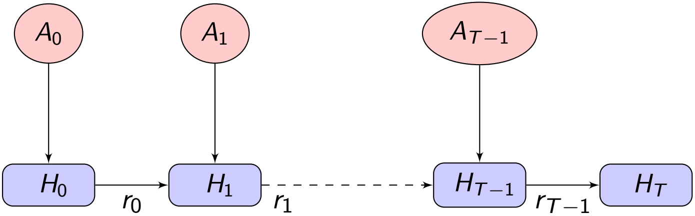
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



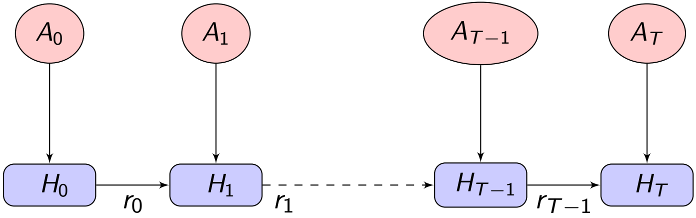
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



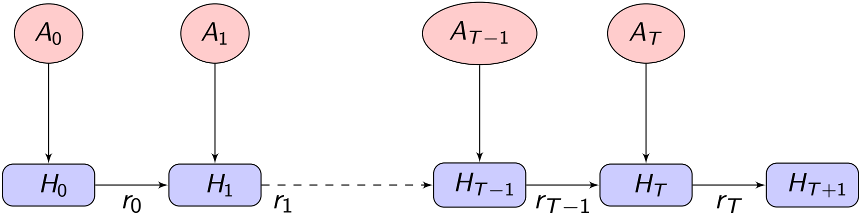
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



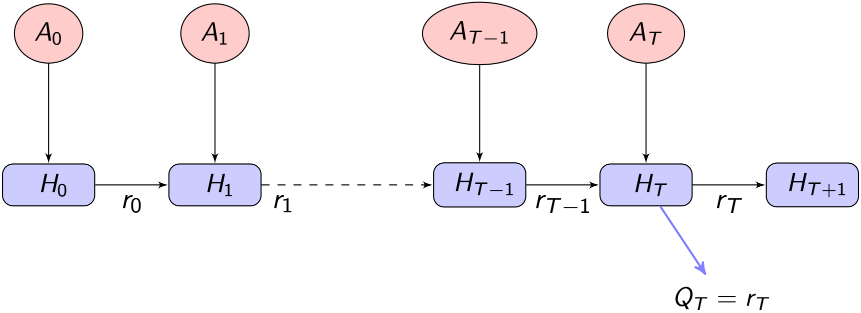
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



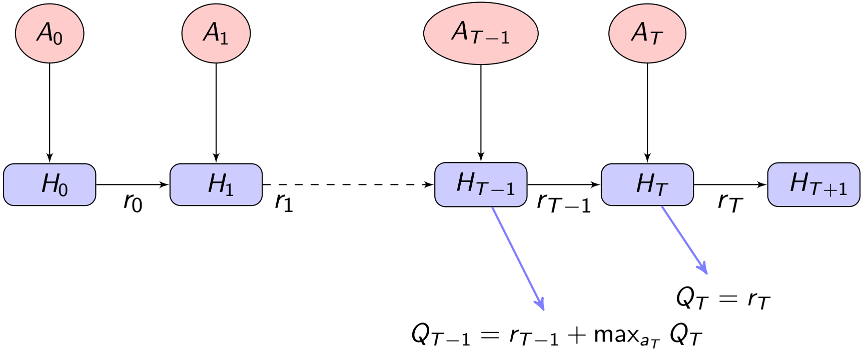
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



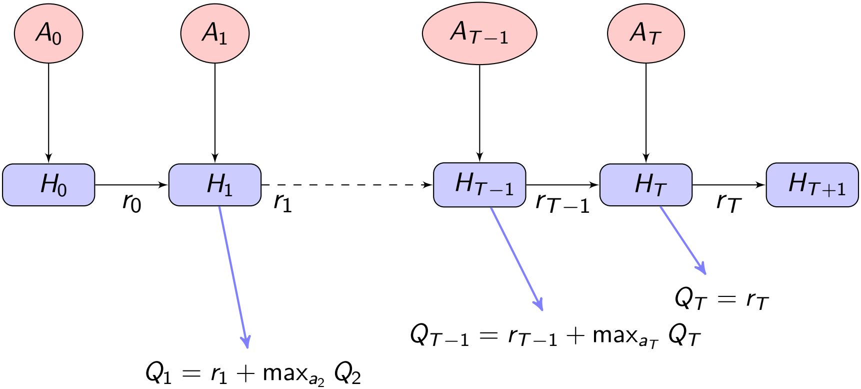
Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



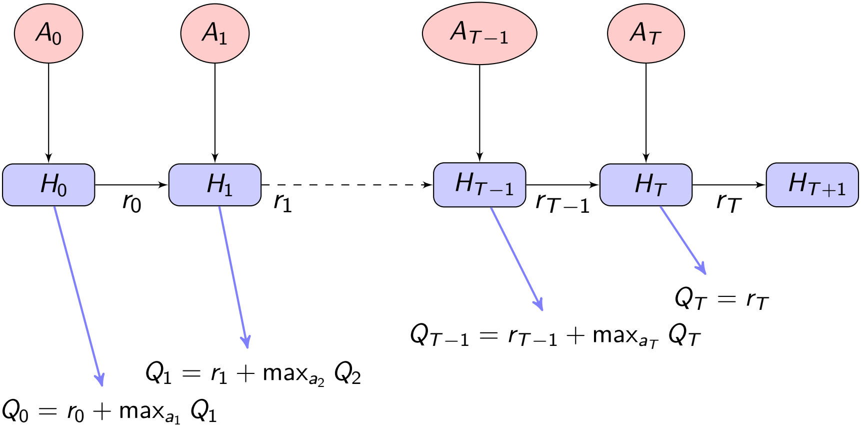
$Q_T = r_T$

$Q_{T-1} = r_{T-1} + \max_{a_T} Q_T$

$Q_1 = r_1 + \max_{a_2} Q_2$

Dynamic Programming

- Estimate $\mathcal{D}^*$ if one knows the complete probability distribution of data generation.



$Q_T = r_T$

$Q_{T-1} = r_{T-1} + \max_{a_T} Q_T$

$Q_1 = r_1 + \max_{a_2} Q_2$

$Q_0 = r_0 + \max_{a_1} Q_1$

Constructing a DTR from Data: Q-learning

- Data-driven analog of dynamic programming.
- Backwards and recursively estimates the following Q-function:

$$Q_j(h_j, a_j) = E(R_j + \max_{a_{j+1} \in \{-1,1\}} Q_{j+1}(H_{j+1}, a_{j+1}) | H_j = h_j, A_j = a_j),$$

where $Q_{T+1} = 0$, and $h_j \in \mathcal{O}_j, a_j \in \mathcal{A}_j, j = 1, \ldots, T$.

- The estimated optimal sequence of decision rules

$$\hat{d}_j(h_j) = \operatorname*{argmax}_{a_j \in \{-1,1\}} \hat{Q}_j(h_j, a_j).$$

- Q learning with regression: estimate the Q-functions from data using regression and then find the optimal DTR.
- An extension of regression to sequential treatments.

Constructing a DTR from Data: *Q*-learning

- First, do a regression at stage 2 to learn about more deeply tailored second-line treatment.
  - Outcome: second stage outcomes;
  - Predictors: history information: characteristics of the participant at baseline and outcome during first-line treatment
- Second, do a regression to learn about more deeply tailored first-line treatment.
  - Outcome: an estimate of the outcome under the second-line treatment that yields the best outcome.
    —- already taken into account future optimal treatment;
  - Predictors: baseline characteristics

Q-Learning: Two Stages

Two stages, $t = 1, 2$; binary treatments denoted by $A_t \in \{0, 1\}$, final outcome R, H_t features of patient history:

- Stage 2 regression: Regress R on H_2 to obtain $\hat{Q}_2(H_2, A_2) = \hat{\beta}_{21}^T H_2 + \hat{\beta}_{22}^T H_2 A_2$
  - $\hat{d}_2(H_2) = \arg\max_{a_2 \in \{0,1\}} \hat{Q}_2(H_2, a_2) = \arg\max_{a_2 \in \{0,1\}} \hat{\beta}_{22}^T H_2 a_2$

Q-Learning: Two Stages

Two stages, $t = 1, 2$; binary treatments denoted by $A_t \in \{0, 1\}$, final outcome R, H_t features of patient history:

- Stage 2 regression: Regress R on H_2 to obtain $\hat{Q}_2(H_2, A_2) = \hat{\beta}_{21}^T H_2 + \hat{\beta}_{22}^T H_2 A_2$
    - $\hat{d}_2(H_2) = \arg\max_{a_2 \in \{0,1\}} \hat{Q}_2(H_2, a_2) = \arg\max_{a_2 \in \{0,1\}} \hat{\beta}_{22}^T H_2 a_2$
- $\tilde{R} = \hat{\beta}_{21}^T H_2 + \max_{a_2 \in \{0,1\}} \hat{\beta}_{22}^T H_2 a_2$
    - $\tilde{R}$ is a predictor of $\max_{a_2 \in \{0,1\}} Q_2(H_2, a_2)$

Q-Learning: Two Stages

Two stages, $t = 1, 2$; binary treatments denoted by $A_t \in \{0, 1\}$, final outcome R, H_t features of patient history:

- Stage 2 regression: Regress R on H_2 to obtain $\hat{Q}_2(H_2, A_2) = \hat{\beta}_{21}^T H_2 + \hat{\beta}_{22}^T H_2 A_2$
  - $\hat{d}_2(H_2) = \arg\max_{a_2 \in \{0,1\}} \hat{Q}_2(H_2, a_2) = \arg\max_{a_2 \in \{0,1\}} \hat{\beta}_{22}^T H_2 a_2$

- $\tilde{R} = \hat{\beta}_{21}^T H_2 + \max_{a_2 \in \{0,1\}} \hat{\beta}_{22}^T H_2 a_2$
  - $\tilde{R}$ is a predictor of $\max_{a_2 \in \{0,1\}} Q_2(H_2, a_2)$

- Stage 1 regression: Regress $\tilde{R}$ on H_1 to obtain $\hat{Q}_1(H_1, A_1) = \hat{\beta}_{11}^T H_1 + \hat{\beta}_{12}^T H_1 A_1$
  - $\hat{d}_1(H_1) = \arg\max_{a_1 \in \{0,1\}} \hat{Q}_1(H_1, a_1) = \arg\max_{a_1 \in \{0,1\}} \hat{\beta}_{12}^T H_1 a_1$

Table of Contents

1. Overview: Dynamic Treatment Regimes
   - What are DTRs?
   - Why use DTRs?
2. Constructing Optimal DTRs from Data: Q-learning
   - Introduction to Q-learning
   - Q-learning: Pros and Cons
3. Classification Perspective for Estimating Optimal DTRs

Q-learning Positives

- Natural approximate dynamic programming approach
- Linear models are common but non-essential
  - Parsimonious and interpretable
  - More flexible models can be used to define the Q-functions (e.g., boosting, random forests, etc.)
- Regression models are well-understood
  - Diagnostic and validation tools exist
  - EDA is straightforward

Q-learning . . . Opportunities

- Non-smooth non-monotone max-operator
  - Linear models are rarely correctly specified for Q_1
  - Non-smoothness induces non-regularity so that standard methods for inference, e.g., the bootstrap and taylor series arguments, are invalid
  - Non-monotone transformations are difficult to model
- Q-learning indirectly estimates d^{opt} through the conditional mean functions
  - Recall, $d_t^{\mathrm{opt}} = \arg\max_{a_t} Q_k(h_t, a_t)$ which depends only on the sign of $Q_t(h_t, 1) - Q_t(h_t, 0)$.
  - Analog in classification: logistic classification vs. large-margin classification

Linear Models are Rarely Correctly Specified for Q_1

- Toy generative model

$$\begin{array}{ll} X_1 \sim \mathrm{Normal}(0,1), & \xi \sim \mathrm{Normal}(0,1/2), \\ X_2 = \zeta X_1 + \xi, & A_t \sim \mathrm{Uniform}\{0,1\}, t = 1,2, \\ \phi \sim \mathrm{Normal}(0,1/2), & R = 1.25 A_1 A_2 + A_2 X_2 - A_1 X_1 + \phi, \end{array}$$

  ζ governs the correlation between X_1 and X_2

- Linear model is correct for Q_2

$$Q_2(H_2, A_2) = 1.25 A_1 A_2 + A_2 X_2 - A_2 X_1$$

- Nonlinear model required for Q_1

$$Q_1(H_1, A_1) = \frac{1}{2\sqrt{2\pi}} \exp\left\{-2(1.25A_1 + \zeta X_1)^2\right\} \\ + (1.25A_1 + \zeta X_1)\Phi\left(2(1.25A_1 + \zeta X_1)\right)$$

Linear Models are Rrarely . . . cont'd

- Nonlinear model required for Q_1

$$Q_1(H_1, A_1) = \frac{1}{2\sqrt{2\pi}} \exp\left\{-2(1.25A_1 + \zeta X_1)^2\right\} \\ + (1.25A_1 + \zeta X_1)\Phi\left(2(1.25A_1 + \zeta X_1)\right)$$

- This is an idealized setting, yet:
  - Linear model assumption holds only when $\zeta = 0$, but this is unlikely in practice
  - Even seasoned data analysts would likely have trouble identifying the correct functional form given limited data

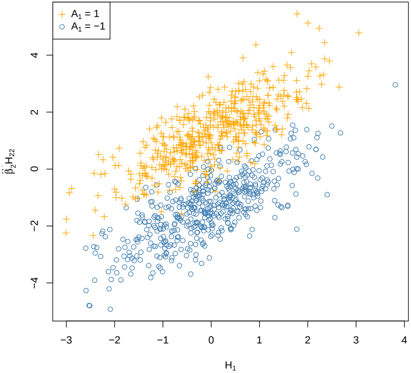
Non-smooth Non-monotone Transformations

- Recall $\tilde{R} = \max_{a_2} \hat{Q}_2(H_2, a_2) = \hat{\beta}_{21}^T H_{21} + \max(\hat{\beta}_{22}^T H_{22}, 0)$

Non-smooth Non-monotone Transformations

- Recall $\tilde{R} = \max_{a_2} \hat{Q}_2(H_2, a_2) = \hat{\beta}_{21}^T H_{21} + \max(\hat{\beta}_{22}^T H_{22}, 0)$

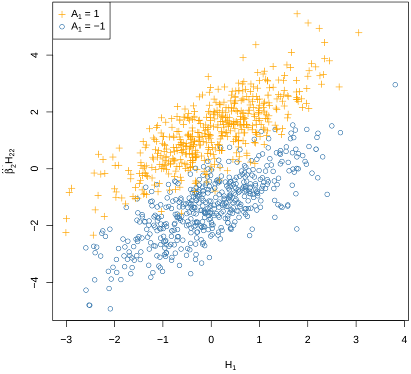
Before maximization



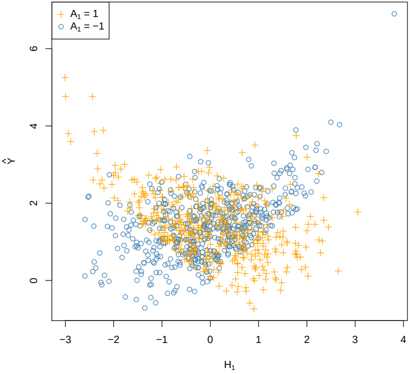
Non-smooth Non-monotone Transformations

- Recall $\tilde{R} = \max_{a_2} \hat{Q}_2(H_2, a_2) = \hat{\beta}_{21}^T H_{21} + \max(\hat{\beta}_{22}^T H_{22}, 0)$

Before maximization



After maximization



Q-learning Indirectly Estimates d^{opt}

- $d_t^{\mathrm{opt}}(h_t) = \arg\max_{a_t} Q_t(h_t, a_t) = \mathbf{1}_{Q_t(h_k,1)-Q_t(h_t,0)>0}$

- Thus, $d_t^{\mathrm{opt}}(h_t)$ depends only on the sign of contrast $Q_t(h_t, 1) - Q_t(h_t, 0)$

    - Q-learning estimates $Q_t(h_t, a_t)$, hence does not directly target d^{opt}

    - A-learning (Murphy, 2003) targets $Q_t(h_t, 1) - Q_t(h_t, 0)$, is closer but still indirect

- Recent classification-based estimators of Zhao et al. (2012) and Zhang et al. (2012) directly target d^{opt}

Table of Contents

1. Overview: Dynamic Treatment Regimes
   - What are DTRs?
   - Why use DTRs?
2. Constructing Optimal DTRs from Data: Q-learning
   - Introduction to Q-learning
   - Q-learning: Pros and Cons
3. Value Maximization Methods

Value Maximization Methods

- Augmented inverse probability-weighting
- Marginal structural mean models
- Outcome weighted learning

Classification Estimators: One Stage

- For clarity, simplify development of Zhao et al. (2012)
  - Assume R is nonnegative
  - Assume A are randomly assigned, recoded to take values in $\{-1, 1\}$
- For any policy d the value equals

$$E^d R = E\left[\frac{I(A = d(X))}{P(A|X)} R\right].$$

Outcome Weighted Learning (OWL)

Optimal Individualized Treatment Rule d^*

Maximize the value   Minimize the risk

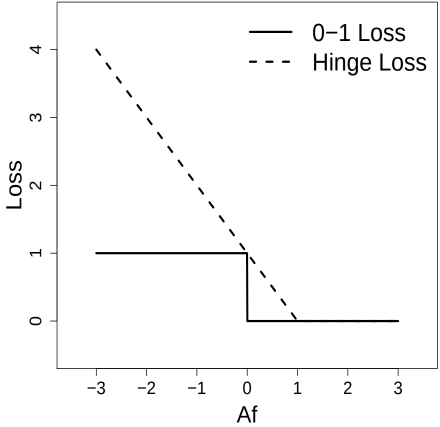
$$E\left[\frac{I(A=d(X))}{P(A|X)}R\right] \quad E\left[\frac{I(A\neq d(X))}{P(A|X)}R\right]$$

- For any rule d, $d(X)=\text{sign}(f(X))$ for some function f.
- Empirical approximation to the risk function:

$$n^{-1}\sum_{i=1}^{n}\frac{R_i}{P(A_i|X_i)}I(A_i\neq \text{sign}(f(X_i))).$$

- Computation challenges: non-convexity and discontinuity of 0-1 loss.

Convex Surrogate Loss: Hinge Loss



Hinge Loss: $\phi(Af(X)) = (1 - Af(X))^{+}$, where $x^{+} = \max(x, 0)$

Outcome Weighted Support Vector Machine (SVM)

Objective Function: Regularization Framework

$$\min_f \left\{ \frac{1}{n} \sum_{i=1}^{n} \frac{R_i}{P(A_i|X_i)} \phi(A_i f(X_i)) + \lambda_n \|f\|^2 \right\}. \qquad (1)$$

- $\|f\|$ is some norm for f, and λ_n controls the severity of the penalty on the functions.
- A linear decision rule: $f(X) = X^T \beta + \beta_0$, with $\|f\|$ as the Euclidean norm of β.
- Estimated individualized treatment rule:

$$\hat{d}_n(X) = \text{sign}(\hat{f}_n(X)),$$

where $\hat{f}_n$ is the solution to (1).

Backward Outcome Weighted Learning (BOWL)

- This is similar to Q-learning but we target value functions directly.
- Assume $P(A_1 = 1) = P(A_2 = 1) = 1/2$, then

$$\mathcal{V}_{\mathcal{D}} = 4E[(R_1 + R_2)I(A_1 = \mathcal{D}_1(H_1))I(A_2 = \mathcal{D}_2(H_2))].$$

- At Stage 2, we obtain $\hat{\mathcal{D}}_2(H_2)$ with objective to minimize

$$E(R_2 I(A_2 \neq \mathcal{D}_2(H_2)))$$

  using OWL.
- At Stage 1, we obtain $\hat{\mathcal{D}}_1(H_1)$ with objective to minimize

$$E([(R_1 + R_2)I(A_2 = \hat{\mathcal{D}}_2(H_2))]I(A_1 \neq \mathcal{D}_1(H_1))),$$

  using OWL.

The estimation restricted to the subset of patients who have been assigned to the estimated optimal treatments in stage 2.

Wrap-up

- This is an extremely active area of research
- Tools for estimation and inference exist and are continually being improved
- There is no panacea, choosing the proper statistical tool depends critically on the goals of the analysis