Section II: developing a marker-based treatment rule

- Treatment decision rule
- Optimal Treatment Rule
- Estimating optimal treatment decision rule
  - Q-learning (Regression modeling)
  - Direct optimization

Treatment Decision Rule

Treatment Decision Rule

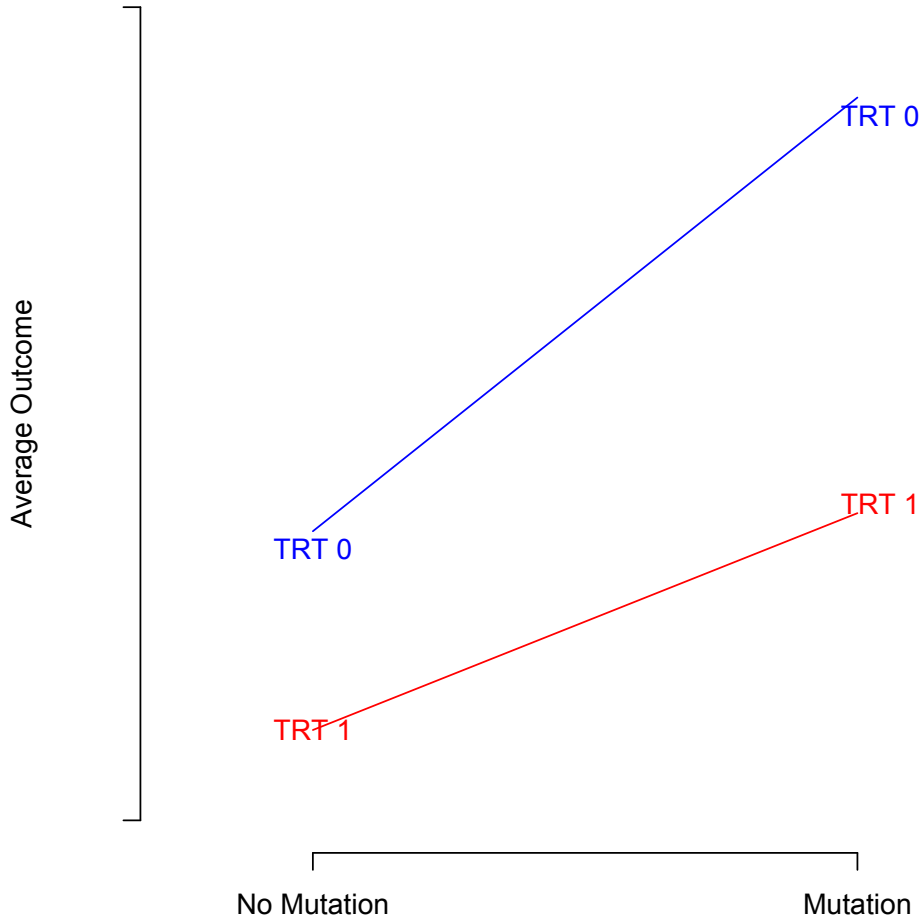
Outcomes are denoted by D,

- Survival time, CD4 count, indicator of no myocardial infarction within 30 days, . . .
- Lower outcomes are better

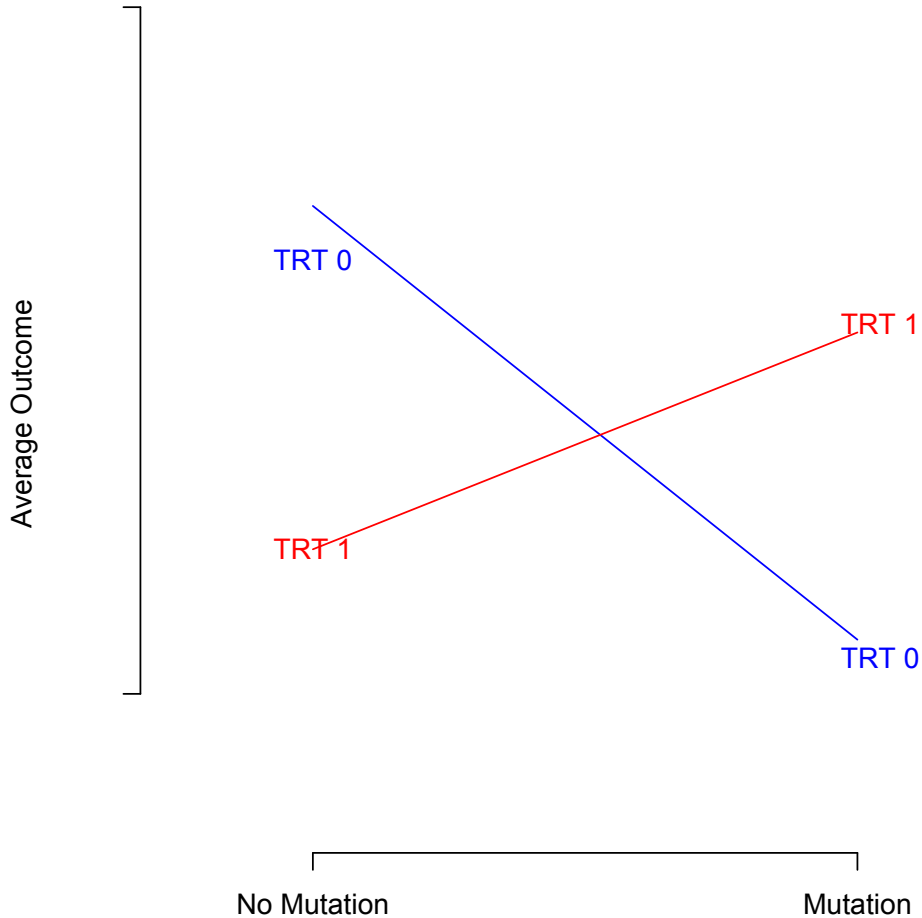
Intuitively: rules should depend on characteristics (variables , covariates), i.e., X, that exhibit a qualitative interaction with treatment

- Tailoring variables/ treatment selection biomarker

Tailoring Variables



Tailoring Variables



Statistical Framework

Simplest setting: A single decision with two treatment options

Observed data: (D_i, X_i, A_i), $i = 1, \ldots, n$, independently and identically distributed (iid)

- D_i outcome, X_i baseline covariates, $A_i = 0, 1$ treatment received

Treatment decision rule: A treatment rule

- A function $d : X \to \{0, 1\}$

Simple example

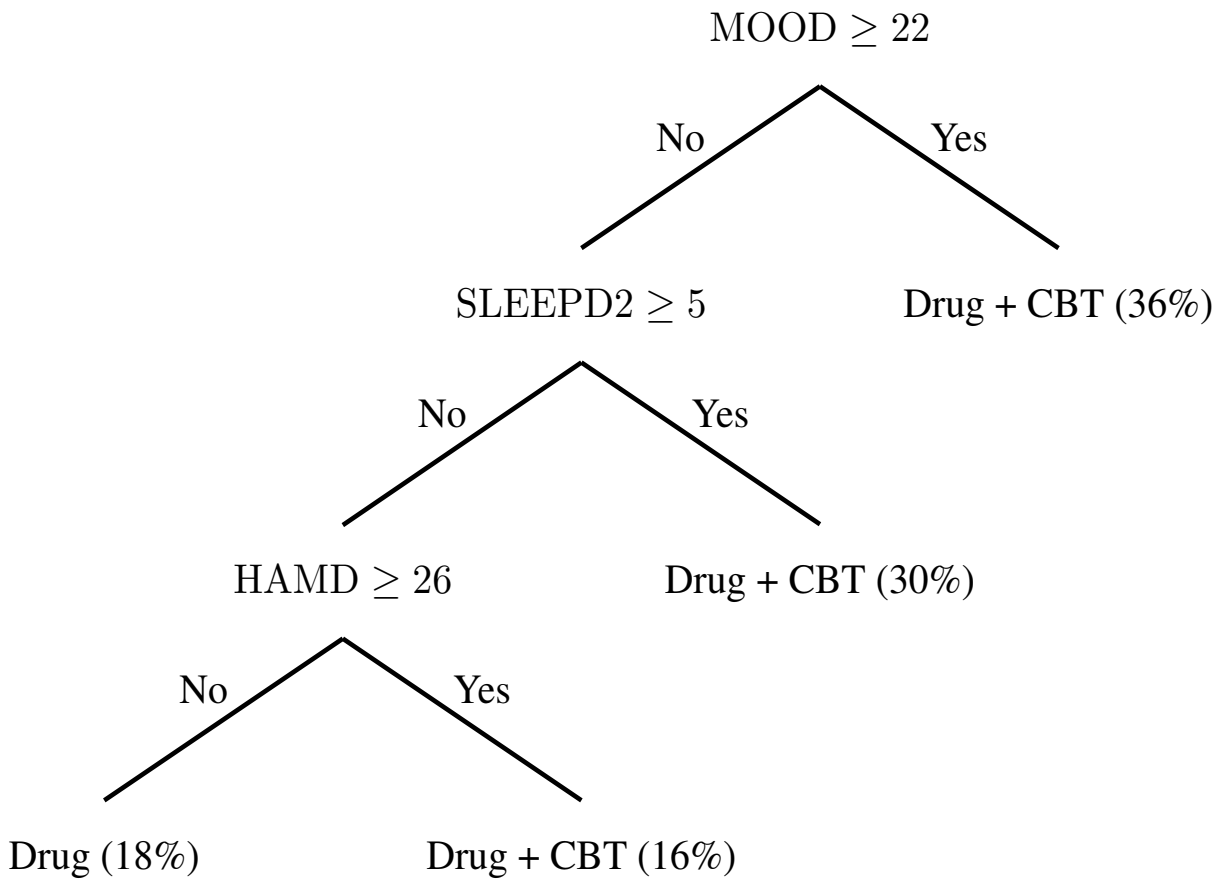
Which treatment to give patients who present with *nonpsychotic Chronic Major Depressive Disorder*?

- *Options:* Nefazodone (Drug) or Drug + Cognitive Behavioral Therapy (CBT)
- *Data:* 681 patients in the Nefazodone-CBASP clinical trial (Keller et al., 2000)
- *Available information:* 50 prognostic variables, e.g., age, baseline depression score
- *Outcome:* Hamilton Rating Scale for Depression

Keller et al. (*NEJM* 2000)

Simple example

A decision rule example:



Simple example

- Even simpler example: If MOOD $\geq 22 \Rightarrow$ Drug + CBT; otherwise $\Rightarrow$ Drug
- *Mathematically:* The formal rule is

$$d(\text{MOOD}) = I(\text{MOOD} > 22)$$

Optimal Treatment Decision Rule

Considerations

- Identify the subset that are good tailoring variables
- Rule $d(X)$: a function of X
- There are many possible rules d:

  $\mathcal{D}$: class of all possible treatment decision rules

- Can we find the optimal treatment decision rule in $\mathcal{D}$?
- Optimal treatment decision rule : If followed by *all patients* in the population, would lead to *smallest expected outcome* among all rules in $\mathcal{D}$

Potential Outcomes

Single decision: Possible treatment options $a \in \{0, 1\}$

- For a *randomly chosen patient* from the population, define the *random variable* $D(a)$ = the outcome the patient *would experience* if s/he *were to receive* treatment option a
- "*Potential outcome*"
- E.g., $D(1)$= the outcome a patient would have if s/he were given treatment 1, and similarly for $D(0)$

Expected outcomes under treatment rules

- Potential outcome for a rule: $D(d)$ = the outcome a patient would have if s/he received treatment *according to a rule* $d \in \mathcal{D}$
- E.g., if the patient has information X

$$D(d) = D(1)I\{d(X) = 1\} + D(0)I\{d(X) = 0\}\}$$

- $E\{D(d)|X = x\}$ is the expected outcome for a patient with information x if s/he were to receive treatment according to rule $d \in \mathcal{D}$.
- $E\{D(d)\} = E[E\{D(d)|X = x\}]$ is the expected outcome for the population if all patients were to receive treatment according to rule $d \in \mathcal{D}$.

Optimal decision rule

- The optimal treatment decision rule $d^* \in \mathcal{D}$ minimizes the expected outcome

$$d^* = \underset{d \in \mathcal{D}}{\operatorname{argmin}}\, E\{D(d)\}$$

- That is, $E\{D(d^*)\} \leq E\{D(d)\}$ for all $d \in \mathcal{D}$
- Also, $E\{D(d^*)|X = x\} \leq E\{D(d)|X = x\}$ for all $d \in \mathcal{D}$ and for all patient subgroups defined by x.
- $d^*(X) = I[E\{D(1)|X\} < +E\{D(0)|X\}]$.

Identifying the optimal treatment decision rule

- We need to discover optimal rules based on data.
- The optimal rule is defined in terms of potential outcomes, not the observed data
- It is possible to discover optimal rule based on the observed data under certain assumptions

Potential Outcomes

Positivity: $P(A = a|X = x)$ strictly positive for all x, i.e., $0 < P(A = 1|X) < 1$ almost surely, usually satisfied in a randomized trial

Consistency: $D(a) = D$ whenever treatment a is actually received, usually satisfied in a randomized trial

No unmeasured confounders: Assume that

$$D(0), D(1) \amalg A|X$$

- X contains all information used to assign treatments in the data
- Automatically satisfied for data from a randomized trial

Potential Outcomes

- Implies that

$$\begin{aligned} E\{D(1)\} &= E[E\{D(1)|X\}] \\ &= E[E\{D(1)|A=1,X\}] \\ &= E\{\,E(D|A=1,X)\,\} \end{aligned}$$

and similarly for $E\{D(0)\}$

Optimal Rule in Terms of Observed Outcomes

- Under positivity, consistency and no unmeasured confounders assumptions:

$$\begin{aligned} E\{D(d)\} &= E[E\{D(d)|X\}] \\ &= E[E(D(1)|A=1,X)I\{d(X)=1\} \\ &\quad +E(D(0)|A=0,X)I\{d(X)=0\}] \\ &= E[E(D|A=1,X)I\{d(X)=1\} \\ &\quad +E(D|A=0,X)I\{d(X)=0\}]. \end{aligned}$$

- Therefore, under these assumptions, the optimal treatment decision rule can be written as a function of observed outcomes, i.e.

$$\begin{aligned} d^*(X) &= \underset{d\in\mathcal{D}}{\operatorname{argmin}}\, E\{D(d)\} \\ &= I\{E(D|A=1,X) < E(D|A=0,X)\} \end{aligned}$$

Example

- $X \sim$Uniform$[-1, 1]$, A is binary {-1, 1} with probability $1/2$, $D \sim N(1 - (X - 1/3)^2 A, 1)$
- Consider the rule $d(x) = I(x - 1/3 \geq 0)$. What is the expected outcome of this rule?

$$\begin{aligned} E\{D(d)\} &= E[E(D|A = 1, X)I(X - 1/3 \geq 0) \\ &\quad + E(D|A = 0, X)I(X - 1/3 < 0)] \\ &= \int_{1/3}^{1} \left\{ \frac{1 - (x - 1/3)^2}{2} \right\} dx + \int_{-1}^{1/3} \frac{1}{2} dx = 73/81 \end{aligned}$$

- What is the optimal treatment rule?

Example

- $d^*(x) = 1$
- What is the expected outcome of the optimal rule?

$$\begin{aligned} E\{D(d)\} &= E[E(D|A=1,X)I\{d(X)=1\} \\ &= \int_{-1}^{1} \frac{1-(x-1/3)^2}{2} dx = 45/81 \end{aligned}$$

Optimal Rule

- Optimal Rule:

$$E(D|X, A = 1) \leq E(D|X, A = 0) \Rightarrow d^*(X) = 1$$
$$E(D|X, A = 1) > E(D|X, A = 0) \Rightarrow d^*(X) = 0$$

- d^* provides a treatment recommendation to every individual given their X

- If $E(D|X, A)$ were known, we could find d^*.

- Problem: $E(D|X, A)$ is unknown.

Estimating optimal treatment decision rule

- Q-learning (Regression modeling)
- Direct optimization

Q-learning (Regression modeling)

- If we had a sample of data $(X_i, A_i, D_i), i = 1, \ldots, n$, we can posit a regression model

$$E(D|A, X) = \mu(A, X; \beta)$$

and estimate $\hat{\beta}$ using e.g. least squares/logistic regression/cox regression.

- The estimator for the optimal treatment decision rule

$$\hat{d}_n(x) = I\{\mu(1, x; \hat{\beta}_n) \leq \mu(0, x; \hat{\beta}_n)\},$$

Regression modeling (Q-learning)

- For a particular rule d, we can estimate $E\{D(d)\}$ by averaging over samples

$$\begin{aligned}\hat{E}(D(d)) &= \hat{E}[\hat{E}(D|A=1,X,d(X)=1)I(d(X)=1) \\ &\quad +\hat{E}(D|A=0,X,d(X)=0)I(d(X)=0)]\end{aligned}$$

With the posited regression model,

$$n^{-1}\sum_{i=1}^{n}[\mu(1,X_i,\hat{\beta}_n)I\{d(X_i)=1\}+\mu(0,X_i,\hat{\beta}_n)I\{d(X_i)=0\}]$$

Regression modeling (Q-learning)

- $\hat{d}_n$ is the minimizer of the estimate of $E\{D(d^*)\}$,

$$n^{-1}\sum_{i=1}^{n}[\mu(1, X_i, \hat{\beta}_n)I\{\hat{d}_n(X_i) = 1\} + \mu(0, X_i, \hat{\beta}_n)I\{\hat{d}_n(X_i) = 0\}],$$

- $\mu(A, X; \beta)$ may be misspecified.

Example Revisited

- $X \sim$Uniform$[-1, 1]$, A is binary {-1, 1} with probability $1/2$, $D \sim N(1 - (X - 1/3)^2 A, 1)$
- What if we use a linear model for the outcome, i.e., $E(D|A, X) = \beta_0 + \beta_1 X + \beta_2 + \beta_3 X$?
- The decision rules considered $d(x) = I(\beta_2 + \beta_3 x \geq 0)$
- Does the optimal rule belong to the this class of rules?

Example Revisited

- The optimal linear rule is $d_L^*(x) = I(x - 2/3 < 0)$.
- The expected outcome of the optimal linear rule is

$$\begin{aligned} E\{D(d_L^*)\} &= E[E(D|A=1,X)I(X-2/3<0) \\ &\quad + E(D|A=0,X)I(X-2/3 \geq 0)] \\ &= \int_{-1}^{2/3} \left\{ \frac{1-(x-1/3)^2}{2} \right\} dx + \int_{2/3}^{1} \frac{1}{2} dx \\ &= 97/162 \\ &> E\{D(d^*)\} \end{aligned}$$

Alternatives

- Use flexible models for the outcome.
- Other methods, e.g., modeling contrast
  - A more robust method for estimating the optimal treatment decision rule
  - One does not need to know the entire function $E(D|A, X)$.
  - It suffices to only consider the contrast function

$$\Delta(X) = E(D|A = 0, X) - E(D|A = 1, X)$$

  - $d^*(x) = I\{\Delta(x) \geq 0\}$.

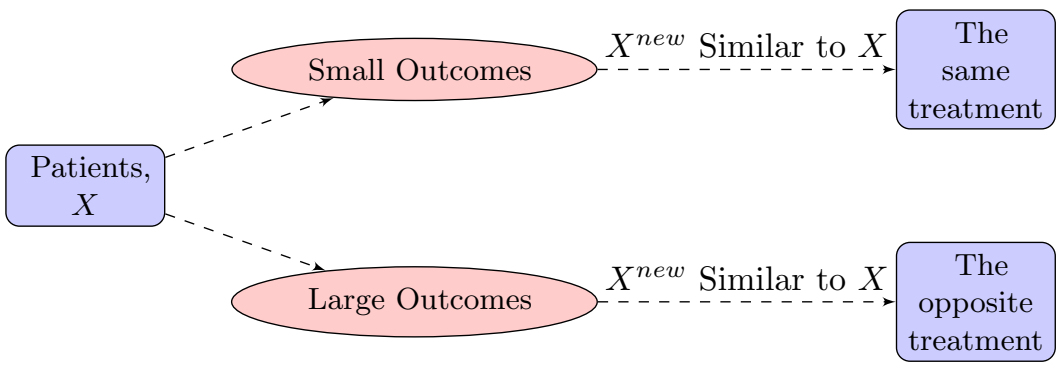
Murphy (*JRSSB*, 2003); Tian et al (*JASA*, 2014)

Direct Optimization: Classification Perspective

Intuition: Classification

Given a new observation X^{new}, predict the class label $d^{*,\text{new}}$.

- No direct information on the true class labels, d^*.
- Can we assign the right treatment based on the observed information?



Directly estimating the Optimal Rule

Thoughts: Minimize a "*good*" estimator for $E\{D(d)\}$

- $\pi(X) = P(A = 1|X)$ is the propensity score for treatment
- $\pi(X)$ known for randomized study; Can also be estimated using the data $(A_i, X_i), i = 1, \ldots, n$, e.g., logistic regression $\pi(X; \gamma)$ and estimate γ by $\hat{\gamma}$.
- The propensity of receiving treatment consistent with $d(X)$

$$\begin{aligned} P\{d(X)|X\} &= P(A = d(X)|X) \\ &= E[Ad(X) + (1 - A)\{1 - d(X)\}|X] \\ &= \pi(X)d(X) + \{1 - \pi(X)\}\{1 - d(X)\}. \end{aligned}$$

Directly estimating the Optimal Rule

Identify estimators for $E\{D(d)\}$:

- Inverse probability weighted estimator

$$IPWE(d) = n^{-1} \sum_{i=1}^{n} \frac{I\{A_i = d(X_i)\} D_i}{P\{d(X_i)|X, \hat{\gamma}\}}. \tag{1}$$

- Consistent for $E\{D(d)\}$ if $\pi(X; \gamma)$, and hence $P\{d(X_i)|X, \hat{\gamma}\}$ is correctly specified

Outcome Weighted Learning (OWL)

- Minimize $IPWE(d)$ (1)
- For any rule d, $2d(X) - 1 = \text{sign}\{f(X)\}$ for some function f.
- Hence, minimize:

$$n^{-1} \sum_{i=1}^{n} \frac{-D_i}{P\{d(X_i)|X, \hat{\gamma}\}} I\{(2A_i - 1) \neq \text{sign}(f(X_i)\}.$$

- Can be treated as recoding $\mathcal{A} = \{-1, 1\}$

Zhao et al. (*JASA* 2012)

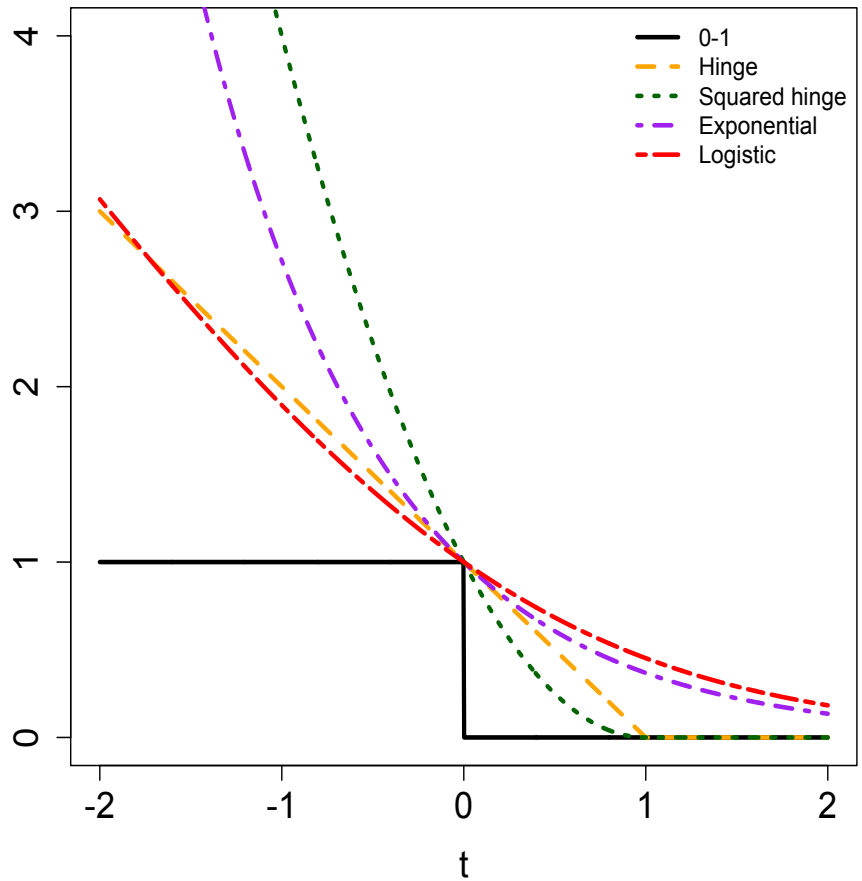
Convex Surrogate Losses for Computation Relaxation

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

Replace 0-1 loss by convex surrogate loss

- Hinge loss, $\phi(t) = \max(1 - t, 0)$.
- Exponential loss, $\phi(t) = e^{-t}$.
- Logistic loss, $\phi(t) = \log(1 + e^{-t})$.
- Squared hinge loss, $\phi(t) = \{\max(1 - t, 0)\}^2$.

Convex Surrogate Losses



- Hinge loss: directly provide the sign of the rule.
- Other losses are smooth. Will use logistic loss for iillustration.

Outcome Weighted Learning: Add Penalties

Objective Function: Regularization Framework

$$\min_f \frac{1}{n}\sum_{i=1}^{n} \frac{-D_i}{P\{d(X_i)|X,\hat{\gamma}\}}\phi\{(2A_i-1)f(X_i))\} + \lambda_n\|f\|^2. \qquad (2)$$

- $\|f\|$ 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.

Outcome Weighted Learning: Add Penalties

- Estimated treatment rule:

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

  where $\hat{f}_n$ is the solution to (2).

- Variable selection is possible, e.g., change $\|f\|^2$ to $\|f\|$.

Generalization of Outcome Weighted Learning

- Residual weighted learning: use residuals (after subtracting main effects) instead of the original outcomes as the weights.
- Efficient augmentation and relaxation learning
  - Doubly robust augmented inverse probability weighted estimator: model both the propensity score and the outcome

$$AIPWE(d) = IPWE(d) - \text{an augmentation term.}$$

  - Consistent if either $\pi(X;\gamma)$ (for the propensity score) or $\mu(A,X;\beta)$ (for the outcome) is correct
  - Outcome weighted learning is a special case.
  - More efficient in estimating $E\{D(d)\}$

Zhou et al. (*JASA* 2015)

Direct optimization: Optimal Restricted Rule

- Optimize the objective within a class of rules

$$d(X, \beta) = I\{\mu(1, X; \beta) < \mu(0, X; \beta)\},$$

indexed by β,

- E.g.,

$$E(D|A, X) = \exp\{1 + X_1 + 2X_2 + 3X_1X_2 + A(1 - 2X_1 + X_2)\}$$

$$\Rightarrow d^*(X) = I(X_2 < 2X_1 - 1)$$

Zhang et al. (*Biometrics* 2012)

Direct optimization: Optimal Restricted Rule

- *Posit*

$$\mu(A, X; \beta) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + A(\beta_3 + \beta_4 X_1 + \beta_5 X_2)$$

- The rules $I\{\mu(1, X; \beta) < \mu(0, X; \beta)\}$ define a *class* $\mathcal{D}_\eta$ with elements

$I(X_2 \geq \eta_1 X_1 + \eta_0)$ or $I(X_2 \leq \eta_1 X_1 + \eta_0)$, $\eta_0 = -\beta_3/\beta_5$, $\eta_1 = -\beta_4/\beta_5$

depending on the sign of β_5

- The optimal rule in this case is contained in $\mathcal{D}_\eta$

Optimal Restricted Rule

Consider directly rules of the form $\mathcal{D}_\eta = \{d(X, \eta)\}$ indexed by η

- Write $d_\eta(X) = d(X, \eta)$, e.g., $d(X, \eta) = I(X_2 \leq \eta_1 X_1 + \eta_0)$
- Defined based on clinical practice, cost, and interpretability, without reference to a regression model.
- d^* may or may not be in $\mathcal{D}_\eta$ but still of interest
- Optimal restricted rule $d^*_\eta(X) = d(X, \eta^*)$,

$$\eta^* = \underset{\eta}{\text{argmin}}\ E\{D(d_\eta)\}$$

- Estimate the optimal restricted rule by estimating η^*

Estimating the Optimal Restricted Rule

- Minimize a "*good*" estimator for $E\{D(d_\eta)\}$ in η:
- Estimators $\hat{\eta}$ for η^* obtained by minimizing $IPWE(d_\eta)$ or $AIPWE(d_\eta)$ in η
- Non-smooth functions of η; must use suitable optimization techniques (RGENOUD package in R)
- Estimators for $E\{D(d_\eta)\}$

  $$IPWE(d_{\hat{\eta}_{ipwe}}) \text{ or } AIPWE(d_{\hat{\eta}_{aipwe}})$$

  Can calculate standard errors
- Performs well when the covariate dimension is not high.

Summary on Direct Optimization Approach

- Direct optimization: conceptual appeal
- How to implement, eg surrogate loss function, form of penalties for variable selection, depends on the context and deserves future research.

Depression Data

- Compare drug therapy ($A = 0$) with drug + behavioral therapy ($A = 1$)
- Five covariates: Age, Gender, HAMABase (pre-treatment total Hamilton Anxiety Rating Scale score), Sleep (sleep disturbance score), Mood (mood cognition score)
- Response: 24-item Hamilton Rating Scale for Depression
- Number of patients: 436

Analyzing Depression Data

- Q-learning: model the depression score using the covariate, the treatment and their interactions

$$D \sim 1 + X + A + XA$$

- Efficient Augmentation and Relaxation Learning: will model both the outcome and the propensity score
  - Logistic loss: $\phi(t) = \log(1 + e^{-t})$
  - Outcome model: $D \sim 1 + X + A + XA$
  - Propensity model: $A \sim X$

Results

- Q-learning: $\hat{d}(X) = I(-0.83 + 0.01 Age - 0.55 Gender + 0.06 HAMABase + 0.01 Sleep - 0.04 Mood < 0)$.
- Efficient Augmentation and Relaxation Learning: $\hat{d}(X) = I(-0.94 + 0.00 Age - 0.33 Gender + 0.05 HAMABase + 0.02 Sleep - 0.01 Mood < 0)$.

Simulation Example

- $X_1, \ldots, X_5 \sim \mathit{Uniform}(-1, 1)$
- $A \sim \{0, 1\}$ w.p. 0.5
- $D \sim 3 + X_1^2 + X_2^2 + (2X_1 + X_3 - 1)A + N(0, 1)$
- The optimal rule: $d^*(x) = I(2x_1 + x_3 < 1)$

Simulation Example

- $X_1, \ldots, X_5 \sim \mathit{Uniform}(-1, 1)$
- $A \sim \{0, 1\}$ w.p. 0.5
- $D \sim 3 + X_1^2 + X_2^2 + (2X_1 + X_3 - 1)A + N(0, 1)$
- The optimal rule: $d^*(x) = I(2x_1 + x_3 < 1)$

Simulation Example

```
set.seed(1111)

n  = 300
p  = 5
X  = matrix(runif(n*p,-1,1),n,p)
A  = rbinom(n,1,0.5)
mX = 3 + X[,1]^2 + X[,2]^2
cX = 2*X[,1]+ X[,3] -1
D  = mX + A*cX + rnorm(n,1)

## optimal rule
dstar  = (cX<0)
> table(dstar)
dstar
FALSE  TRUE
   85   215
```

Simulation Example: Q learning

- Q learning

```
## Q learning (regression modeling)

bReg = lm(D~X*A)
bCoef = bReg$coef
QTrtRec = as.numeric(cbind(1,X) %*% bCoef[7:12]<0)

> bCoef
(Intercept)          X1          X2          X3          X4          X5           A
 4.59108114  0.19191209 -0.43920444 -0.03709295  0.10806789 -0.03144234 -0.89900907
       X1:A        X2:A        X3:A        X4:A        X5:A
 1.70576380  0.53473878  0.94800628  0.20574692 -0.13128873

> table(QTrtRec)
QTrtRec
  0   1
 76 224
```

Simulation Example: Restricted Rule

- R package: DynTxRegime, methods for Estimating Optimal Dynamic Treatment Regimes, including single decision setup
- For restricted regime:

```
## A doubly robust Augmented Inverse Propensity Weighted Estimator (AIPWE) or Inverse
Propensity Weighted Estimator (IPWE) for population mean outcome is optimized over a
restricted class of regimes. Methods are available for both single-decision-point and multiple-
decision-point regimes. This method requires the rgenoud package.

Usage

optimalSeq(..., moPropen, moMain, moCont, data, response, txName, regimes,
           fSet = NULL, refit = FALSE, iter = 0, verbose = TRUE)
```

Simulation Example: OWL/EARL

- For OWL/EARL:

```
##Estimation of optimal treatment regime using efficient augmentation and relaxation
learning (EARL). The method is limited to single-decision-point scenarios with binary
treatment options.

## by setting moMain and moCont to NULL, the function is to estimate the optimal
treatment regime using outcome weighted learning (OWL).


Usage

earl(..., moPropen, moMain, moCont, data, response, txName, regime,
     iter = 0L, lambdas = 0.5, cvFolds = 0L, surrogate = "hinge",
     guess = NULL, verbose = TRUE)
```

- There is also a function on OWL implementation with more features (R function: owl). See help for details.

Simulation Example: Restricted Rule

```
library(DynTxRegime)

# implementation to estimate the optimal restricted rule
# Data Preparation
 data <- data.frame(X, A, D)
 colnames(data) <- c("x1", "x2", "x3", "x4", "x5","a","D")


# Define the propensity for treatment model and methods.
 moPropen<- buildModelObj(model =  ~ x1 + x2 + x3 + x4 + x5,
                                   solver.method = 'glm',
                                   solver.args = list('family'='binomial'),
                                   predict.method = 'predict.glm',
                                   predict.args = list(type='response'))


# Create modelObj object for main effect component
  moMain <- buildModelObj(model = ~ x1 + x2 + x3 + x4 + x5,
                           solver.method = 'lm')

# Create modelObj object for contrast component
  moCont <- buildModelObj(model = ~ x1 + x2 + x3 + x4 + x5,
                           solver.method = 'lm')
```

Simulation Example: Restricted Rule

```
# treatment regime rules at each decision point.
  regimes <- function(a,b,c,d, e, f, data){
               as.numeric( a + b*data$x1 + c*data$x2 + d*data$x3 + e*data$x4 + f*data$x5 > 0)
    }

# genoud requires some additional information
  c1 <- c(-1,-1,-1,-1,-1,-1)
  c2 <- c( 1, 1, 1, 1, 1, 1)
  Domains <- cbind(c1,c2)
  starts <- c(0,0,0,0,0,0)

#!! A LARGER VALUE FOR POP.SIZE IS RECOMMENDED
#!! THIS VALUE WAS CHOSEN TO MINIMIZE RUN TIME OF EXAMPLES
  pop.size <- 50
```

Simulation Example: Restricted Rule

```
 estAIPWE <- optimalSeq(moPropen = moPropen,
                        moMain = moMain,
                        moCont = moCont,
                        data = data,
                        response = -data$D,
                        txName =  "a",
                        regimes = regimes,
                        iter=0L,pop.size = pop.size, starting.values = starts,
                        Domains = Domains, solution.tolerance = 0.0001)

> regimeCoef(estAIPWE)
           a            b            c            d            e            f
4.506975e-01 -7.614161e-01 -5.267877e-05 -5.334575e-01  3.900155e-03 -1.398331e-01

 AIPWTrtRec<- optTx(estAIPWE)

> table(AIPWTrtRec)
AIPWTrtRec
  0   1
 70 230
```

Simulation Example: EARL

```
library(DynTxRegime)

# Data Preparation
 data <- data.frame(X, A, D)
 colnames(data) <- c("x1", "x2", "x3", "x4", "x5","a","D")


# Define the propensity for treatment model and methods.
 moPropen<- buildModelObj(model =  ~ x1 + x2 + x3 + x4 + x5,
                                   solver.method = 'glm',
                                   solver.args = list('family'='binomial'),
                                   predict.method = 'predict.glm',
                                   predict.args = list(type='response'))


# Create modelObj object for main effect component
  moMain <- buildModelObj(model = ~ x1 + x2 + x3 + x4 + x5,
                           solver.method = 'lm')

# Create modelObj object for contrast component
  moCont <- buildModelObj(model = ~ x1 + x2 + x3 + x4 + x5,
                           solver.method = 'lm')
```

Simulation Example: EARL

```
 earlRes <- earl(moPropen = moPropen, moMain = moMain,
                   moCont = moCont,
                   data = data, response = -data$D, txName = "a", surrogate = 'logit',
                   regime = ~ x1 + x2 + x3 + x4 + x5, lambdas=2^seq(-5,5,1), cvFolds = 5)

> regimeCoef(earlRes)
[1]  0.39663271 -0.59853084 -0.14985610 -0.34259186  0.00478191 -0.02726041


EARLTrtRec <-  optTx(earlRes)$optimalTx

EARLTrtRec <- (EARLTrtRec + 1)/2  ## change coding from (-1,1) to (0,1)

> table(EARLTrtRec)
EARLTrtRec
  0   1
 64 236
```

Simulation Example: OWL

```
library(DynTxRegime)

# Data Preparation
 data <- data.frame(X, A, D)
 colnames(data) <- c("x1", "x2", "x3", "x4", "x5","a","D")


# Define the propensity for treatment model and methods.
 moPropen<- buildModelObj(model =  ~ x1 + x2 + x3 + x4 + x5,
                                   solver.method = 'glm',
                                   solver.args = list('family'='binomial'),
                                   predict.method = 'predict.glm',
                                   predict.args = list(type='response'))
```

Simulation Example: OWL

```
owlRes <-  earl(moPropen = moPropen, moMain = NULL, moCont = NULL,
                  data = data, response = -data$D, txName = "a", surrogate = 'logit',
                  regime = ~ x1 + x2 + x3 + x4 + x5, lambdas=2^seq(-5,5,1), cvFolds = 5)
```

```
> regimeCoef(owlRes)
[1]  0.42115454 -0.65789664 -0.25178980 -0.33182440 -0.09571889 -0.03276892
```

```
 OWLTrtRec <-  optTx(owlRes)$optimalTx

 OWLTrtRec <- (OWLTrtRec + 1)/2  ## change coding from (-1,1) to (0,1)
```

```
> table(OWLTrtRec)
OWLTrtRec
  0   1
 67 233
```

Simulation Example: Performance Comparison

- Compare with the optimal rule

```
> table(QTrtRec,dstar)
       dstar
QTrtRec FALSE TRUE
      0    71    5
      1    14  210

> table(AIPWTrtRec, dstar)
          dstar
AIPWTrtRec FALSE TRUE
         0    66    4
         1    19  211

> table(EARLTrtRec,dstar)
          dstar
EARLTrtRec FALSE TRUE
         0    63    1
         1    22  214

> table(OWLTrtRec,dstar)
         dstar
OWLTrtRec FALSE TRUE
        0    61    6
        1    24  209
```

- Can validate the performances on an independent data set.

Summary

- Active research area.
- Regression modeling: easy to implement; model may be misspecified.
- Direct optimization: more robust.
- Other methods are also being developed, e.g., tree based methods.