統計的機械学習 （応用計量分析２）第6回

条件付き平均因果効果の推定法（参考pdf 5.3節、6章）

本日の内容

CATE推定法の中でも決定木系の手法について学ぶ
ベイズ推論についてもとりあげる

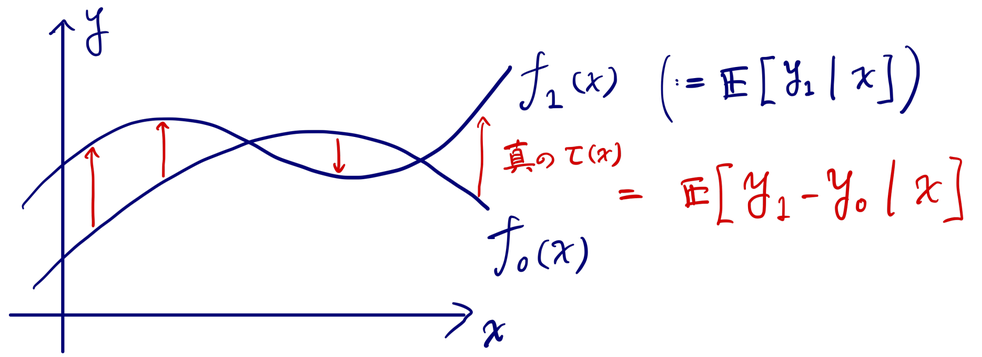
- 1. ガイダンス・因果推論と機械学習の概論
- 2. 意思決定理論（期待効用理論）の復習、因果推論との関係
- 3. 潜在結果モデルに基づく因果推論の枠組み
- 4. 平均因果効果の推定法
- **5. 条件付き平均因果効果（CATE）の推定法１：メタ学習器**
  - **CATEの推定法２：二重機械学習**
- **6. CATEの推定法３：決定木と決定森**
  - **深層学習に基づく方法**
- 7. 構造方程式モデルとバックドア基準
- 8. 因果探索
- 9. 発展的な因果推論手法：フロントドア調整、操作変数法、回帰不連続デザイン、代理変数法
- 10. 続き
- 11. 発展的な意思決定理論
- 12. 強化学習
- 13. オフライン強化学習
- 14. バンディット
- 15. まとめ

T-Learnerの課題（X-Learnerへの伏線）

出欠コード
XXXX

結果モデル f_a は複雑、曝露群が少ないとき推定精度悪化

- T-Learnerは曝露群・対照群両方の
  結果モデル推定精度に依存
  - $\hat{\tau}(x) = \hat{f}_1(x) - \hat{f}_0(x)$
- 一方で、真の因果効果 $\tau(x)$ は単純かもしれない
  - x に依存しない一定値 $\tau(x) = \tau$ かもしれない
  - x の一部の変数にしか依存しないかもしれない
  - にもかかわらず、結果モデル全体 f_a の推定精度に依存してしまう
  - とくに、曝露群は小さい（$\mu(a=1|x)$が小さい）ことが多く、$\hat{f}_1$は信頼できない
- 因果効果自体のデータはないので $\tau(x)$ を直接学習することはできない
  - しかし、$\hat{f}_0$ は比較的信頼できる。ならば、$\hat{\tau}^i = y_1^i - \hat{f}_0(x^i)$ と推定し、これを学習すればよいのではないか？
  - → これを、x ごとにサンプルの多い方を”信頼”して最終的な推定を行うように一般化したのが **X-Learner**



出欠コード
XXXX

X-Learner：群ごとの効果モデルの重み付き平均

- X-Learnerの手順

1. T-Learner と同様に、行動ごとの潜在結果予測モデルを学習

   $\hat{f}_1(x), \hat{f}_0(x)$

2. 行動ごとのデータに対し上記モデルをプラグインして
   個別の効果推定値を計算

   $\hat{\tau}_0^i = \hat{f}_1(x^i) - y_0^i$ (for $i : a^i = 0$)

   $\hat{\tau}_1^i = y_1^i - \hat{f}_0(x^i)$ (for $i : a^i = 1$)

   - 欠損している方を潜在結果モデルで補完

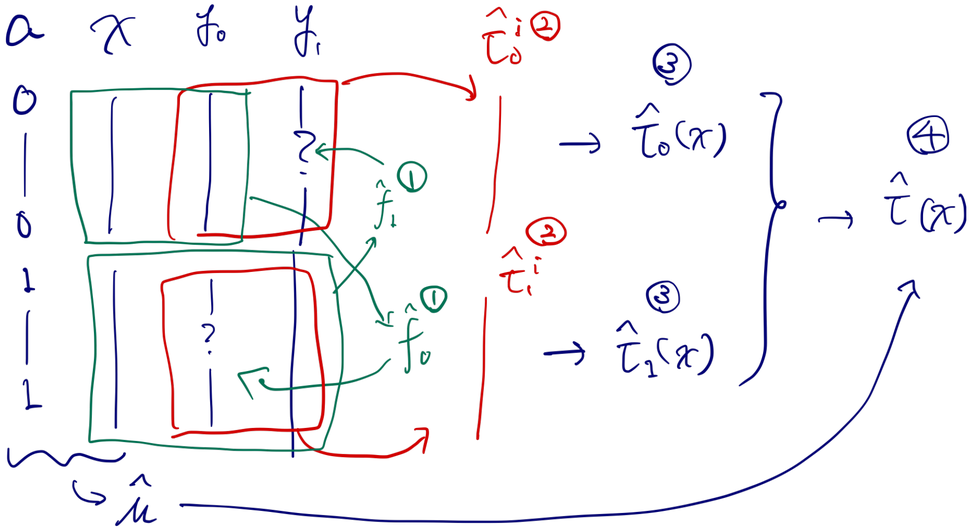
3. 行動ごとの効果モデルを学習

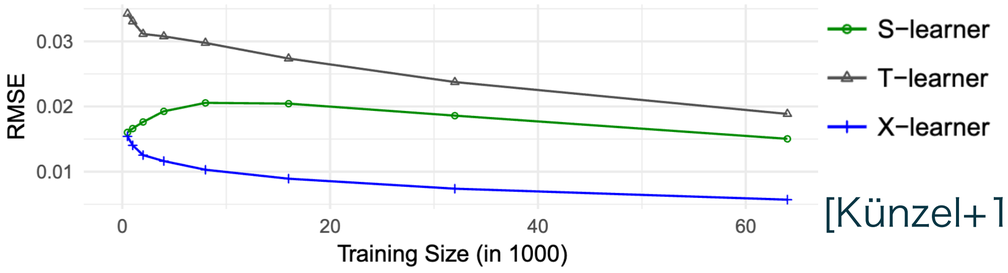
   $\hat{\tau}_0 = \mathrm{Learn}(\{\hat{\tau}_0^i, x^i\}_i),\ \hat{\tau}_1 = \mathrm{Learn}(\{\hat{\tau}_1^i, x^i\}_i)$

4. 最終的に傾向スコアで重み付きアンサンブル（足し合わせ）

   $\hat{\tau}(x) = \hat{\mu}(x)\hat{\tau}_0(x) + (1 - \hat{\mu}(x))\hat{\tau}_1(x)$

   - $\mu(x)$ が小さい＝曝露群が小さい x では $\hat{f}_0$ を途中で用いた $\hat{\tau}_1(x)$ を信頼





[Künzel+19]

CATE推定のためのメタ学習器（3/4）

目的変数変換法：ホーヴィッツ＝トンプソン推定量の応用
傾向スコアによるノイズ付き個別因果効果を用いた直接回帰

- 目的変数変換法：ATE推定のためのホーヴィッツ＝トンプソン推定の応用
  1. 傾向スコア推定モデル $\hat{\mu}(x)$ を学習
  2. 各サンプル $(x^i, a^i, y^i_{a^i})$ に対し、目的変数を以下により変換

$$z^i = y^i_1 \frac{a^i}{\hat{\mu}\left(x^i\right)} - y^i_0 \frac{\left(1 - a^i\right)}{1 - \hat{\mu}\left(x^i\right)}$$

     - $\hat{\mu}$ が正しければ、z^iの条件付き期待値はCATEになる $\mathbb{E}[z^i \mid x^i] = \tau(x^i)$
  3. z を x に回帰

$$\hat{\tau} = \mathrm{Learn}(\{x^i, z^i\}_i)$$

- 利点：結果モデル $f(x, a)$ の推定を経ることなく直接的に $\hat{\tau}$ のモデルを学習できる
- 欠点：傾向スコアの推定 $\hat{\mu}$ の値が0や1に近い場合、z^i に加わるノイズが大きく推定分散が高い

CATE推定のためのメタ学習器（4/4）

DR-Learner：二重頑健推定量の応用
結果予測モデルか傾向スコアモデルのどちらかが良ければ良い

- DR-Learner

1. T-Learnerと同様、行動ごとの潜在結果モデルを学習

   $\hat{f}_1(x), \hat{f}_0(x)$

2. DR推定量の結果変換を適用

$$z^i = \hat{f}_1(x^i) - \hat{f}_0(x^i) + \frac{y_1^i - \hat{f}_1(x^i)}{\hat{\mu}(x^i)} a^i - \frac{y_0^i - \hat{f}_0(x^i)}{1 - \hat{\mu}(x^i)}(1 - a^i)$$

   - 直接法による予測＋
     残差を目的変数変換法と同様逆傾向スコア重み付け

3. その差をxに回帰して効果モデルを学習

   $\hat{\tau} = \mathrm{Learn}(\{x^i, z^i\}_i)$

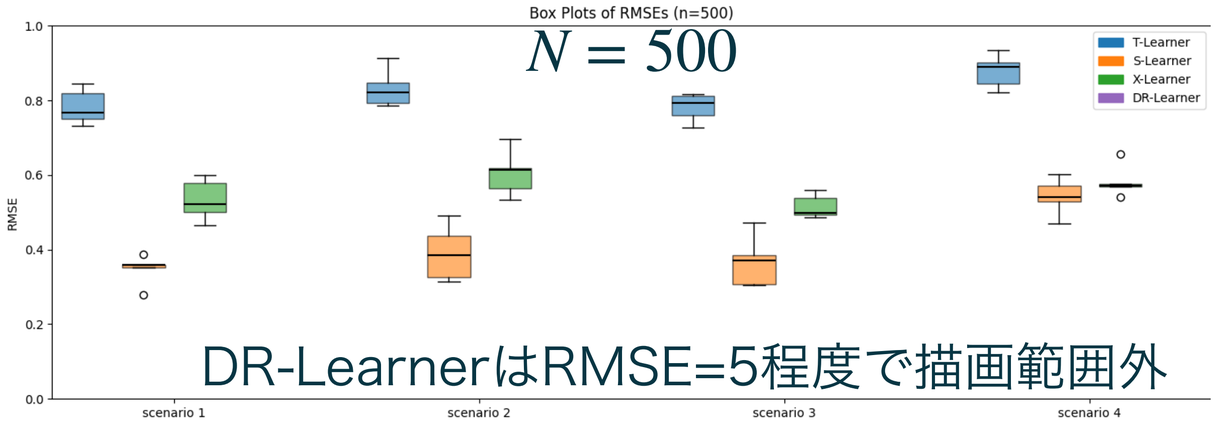
- 利点
  - 結果モデルと傾向スコアモデルの**いずれか**が正しければ後段の $\hat{\tau}$ の推定への悪影響が少ない
  - 理論的には、全体の収束レートが2つのモデルの収束レートの**積**になる
    - 片方の仮説集合が間違っていて $O(1)$ レートでも、もう片方が正しくて $O(N^{-1/2})$ レートであればよい
    - 両方のモデルがそれぞれ $O(N^{-1/4})$ 以上であれば効率的なレート $O(N^{-1/2})$ が達成される
- 欠点
  - $\hat{\mu}$ の値が0や1に近い場合、残差なのでマシとはいえやはりノイズが大きく推定が不安定

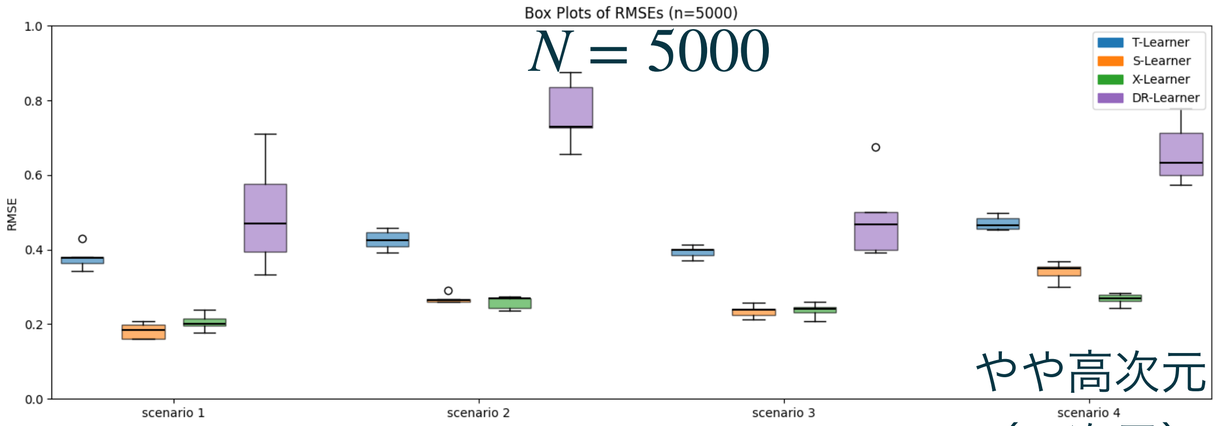
性能比較例

出欠コード
XXXX

DR-Learnerは逆数をとるので不安定になりがち

- 実際の精度は諸条件に依存するので参考まで
  - サンプルサイズN、次元p、μの偏り、真のモデル$f(x,a)$の複雑さ、ベース学習器、等
- 理論では $N \to \infty$ の漸近的な性質を議論しがち（理論的に言えることが他にあまりない）だが、DR-Learnerなど推定値の逆数をとる推定法はサンプルサイズが大きくないとき不安定になりがち





（30次元）

4つのデータ生成モデルで実行した例

（ベース学習器：勾配ブースティング決定木 GBDT）

二重機械学習

二値行動の一般化：行動線形モデル

二値/連続値行動向けセミパラメトリック法：DML（1/3）

出欠コード
xxxx

二重機械学習：連続的行動 $a \in \mathbb{R}$ 等にも適用可能

- **二重機械学習（Double/Debiased Machine Learning; DML、R-Learner）**
  - $y = \theta(x) \cdot a + g(x, w) + \varepsilon \qquad \left(\mathbb{E}[\varepsilon \mid x, w] = 0\right) \quad \cdots (1)$
    - 説明変数のうち、aの係数に関与する部分（**効果修飾因子**）をx、関与しない部分をw
    - 行動aに関して線形性を仮定、その他の部分 g は非線形を許容（セミパラメトリック）
  - $a = h(x, w) + \eta \qquad (\mathbb{E}[\eta \mid x, w] = 0)$
  - 部分線形モデルが正しい（誤指定ではない）場合、無視可能性から $\mathbb{E}[\eta \cdot \varepsilon \mid x, w] = 0$
- 残差を線形モデルで学習
  - (1) の期待値をとると $\mathbb{E}[y \mid x, w] = \theta(x) \cdot \mathbb{E}[a \mid x, w] + g(x, w) \cdots (2)$
  - (1) の両辺から (2) の両辺を引くと、
  - $y - \mathbb{E}[y \mid x, w] = \theta(x) \cdot (a - \mathbb{E}[a \mid x, w]) + \varepsilon$
    “普段”の売上との差　　“普段”の価格との差
- 結果を(x, w)に回帰した残差を行動の残差に回帰する

← 推定の邪魔だった $g(x, w)$ が消えている！

Robinsonの分解とも呼ばれることから
DMLによるCATE推定法のことをR-Learnerとも呼ぶ

DML（2/3）イメージ

行動以外の背景因子の効果を差っ引くことで背景因子ごとに原点をずらす

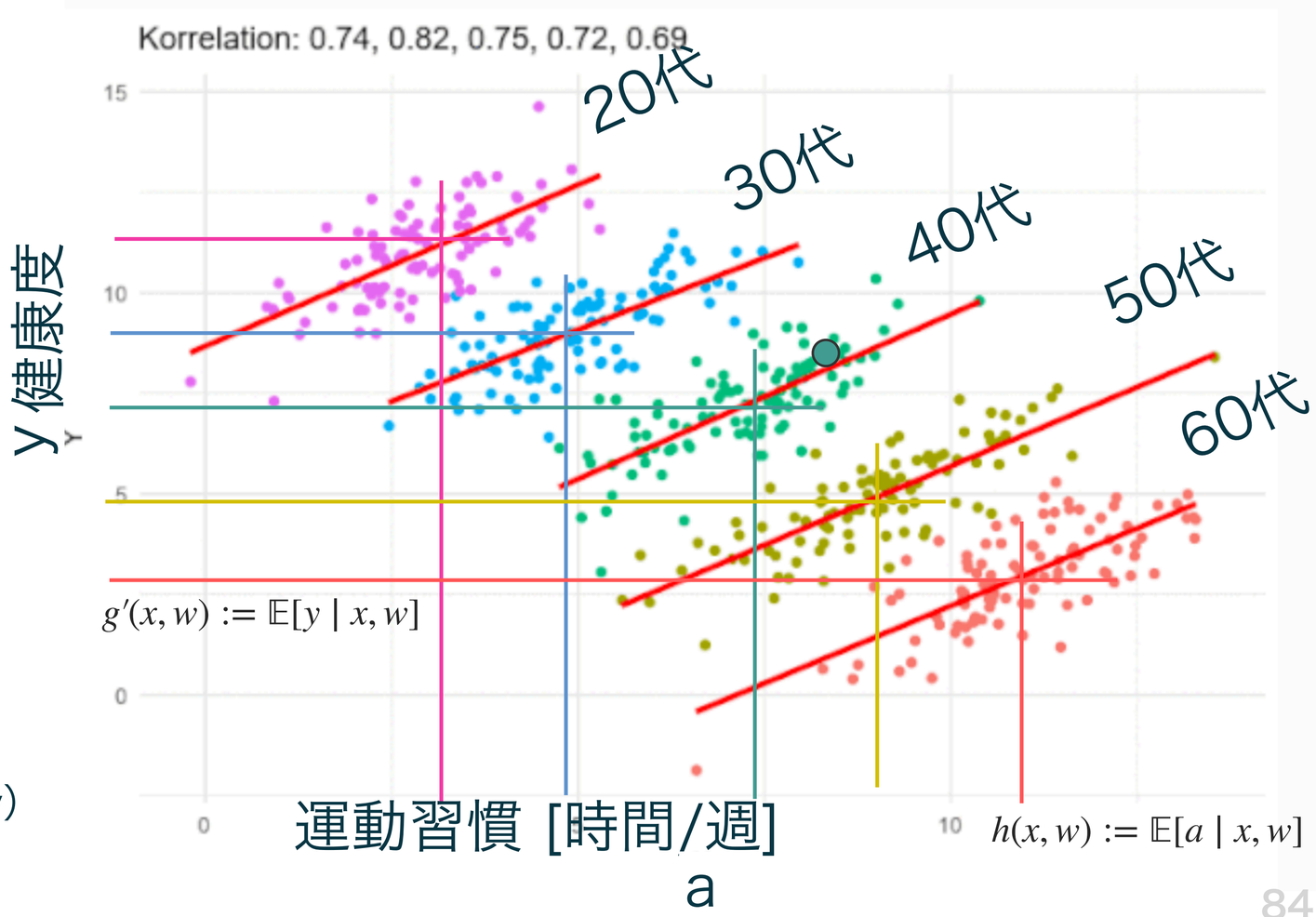
- $y - \mathbb{E}[y \mid x, w] = \theta(x) \cdot (a - \mathbb{E}[a \mid x, w]) + \varepsilon$
- 属性 (x, w)ごとに「原点」を設定

  そこからの差分にフォーカス
- g' と h の推定誤差が θ の推定誤差

  に影響する度合いが小さい
  - セミパラメトリック推論の理論
  - 誤差 Δ_θ を $(\Delta_h, \Delta_{g'})$ についてテイラー

    展開した１次の係数がゼロ
    - ネイマン直交性（Neyman orthogonality）

      という



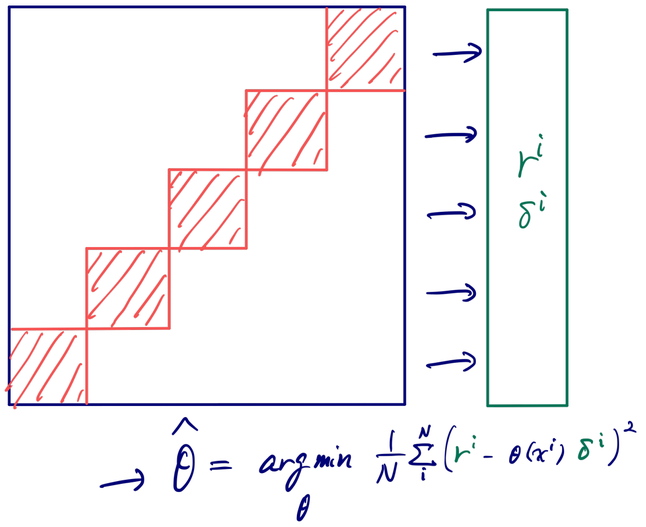
DML（3/3）データ分割法（クロスフィット）

残差を偏りなく推定するため、K-foldに分割して推定

- 学習用データと同じデータで残差を評価すると過小評価になる
  - 訓練誤差はテスト誤差より通常小さい
- データをK分割し、未学習データで残差 r^i, δ^i を算出
- これらを集め、$\left(x^i, r^i, \delta^i\right)$から$\theta(x)$ を推定
  - $\hat{\theta} = \arg\min_{\theta} \frac{1}{N} \sum_{i}^{N} \left(r^i - \theta(x^i)\delta^i\right)^2$
- （または各foldで学習したモデルを平均する）

$$r^i = y^i - \hat{g}'(x^i, w^i)$$

$$\delta^i = a^i - \hat{h}(x^i, w^i)$$



（参考）Neyman直交性

モーメント条件が迷惑モデルの推定誤差に関して直交

- 真のパラメータ θ_0, 真の迷惑モデル $\eta_0 = (g', h)$ に対して、

  損失関数がモーメント条件 $E_P\left[\psi\left(o; \theta_0, \eta_0\right)\right] = 0$ を満たすと仮定する

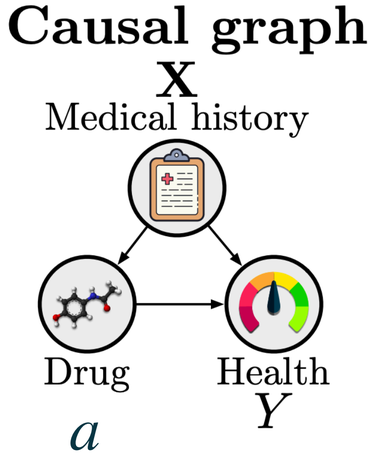
  - ψ はスコア関数。損失関数のθ微分だと思えばOK
- Neyman直交性の定義
  - $\partial_\eta E_P\left[\psi\left(o; \theta_0, \eta_0\right)\right]\left[\eta - \eta_0\right] = 0, \quad \text{for all } \eta \in \mathcal{T}_N$
  - 微小なズレ $\Delta_\eta \ll 1$ では期待モーメントは0から変化しない

    →真のパラメータへの影響も小さい
- なお、Doubly RobustならNeyman直交だが、その逆は必ずしも真ではない
  - R-Learnerは二重頑健ではない

（参考）DMLの構造的行動への拡張

構造のある行動空間の特徴付けに対して線形なモデル

- Causal Effect Inference for Structured Treatments (NeurIPS '21)
- 積効果モデル: $y = g(\mathbf{x})^\top h(\mathbf{a}) + \varepsilon$
  - cf) DML: $y = \theta(x) \cdot a + g(x, w) + \varepsilon$　→ aは二値か1次元か1-hotベクトルくらい
  - 行動を特徴付ける空間 h を同時学習する
  - $e^h(x) := \mathbb{E}\left[h(a) \,|\, x\right]$
  - 一般化Robinson分解
    - $y - m(x) = g(\mathbf{x})^\top \left(h(\mathbf{a}) - e^h(x)\right) + \varepsilon$
- 学習ステップ
  - $m(x) \approx \mathbb{E}[y \,|\, x]$ を学習
  - while not converged
    - g, h をK回更新
    - e^h を更新



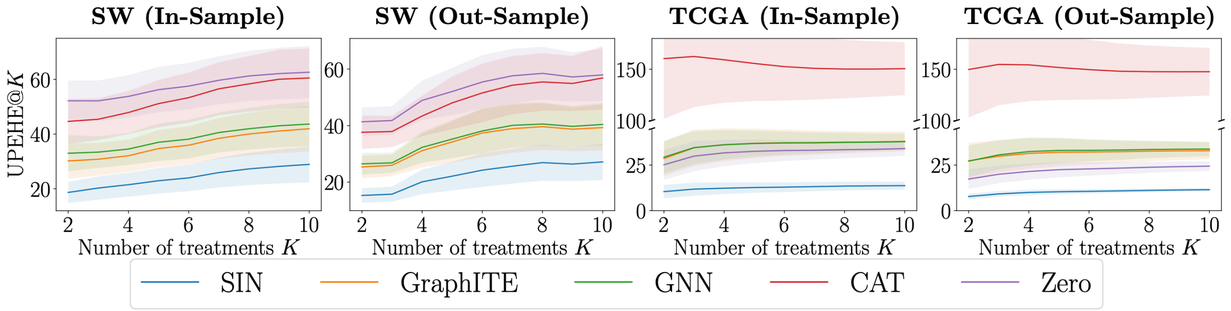


Figure 3: UPEHE@K for $K \in \{2, \ldots, 10\}$.

CATE推定のためのメタ学習とDML

- メタ学習器
  - S-Learnerは行動を入力変数に含めてモデル化
  - T-Learnerは行動ごとに分けてモデル化
  - X-LearnerはT-Learnerをベースに曝露/対照群間の偏りによって結果モデルの推定精度がネックにならないよう群ごとに$\hat{\tau}$を学習して重みつき平均
  - DR-LearnerはAIPW法の擬似結果を回帰する方法で二重頑健性を保証
- 二重機械学習（DML）
  - 行動が二値$a \in \{0,1\}$の場合に加えて連続値の場合に対して線形モデルとした

    セミパラメトリックモデル（a以外に関しては非線形を許容）
  - 結果yを説明変数（a以外）に回帰したモデルの残差を、行動を説明変数に回帰したモデルの残差に回帰
  - 残差が過小推定とならないようにデータ分割、クロスフィットを行う

CATE推定のためのメタ学習とDML

- メタ学習器
  - S-Learnerは行動を入力変数に含めてモデル化
  - T-Learnerは行動ごとに分けてモデル化
  - X-LearnerはT-Learnerをベースに曝露/対照群間の偏りによって結果モデルの推定精度がネックにならないよう群ごとに$\hat{\tau}$を学習して重みつき平均
  - DR-LearnerはAIPW法の擬似結果を回帰する方法で二重頑健性を保証
- 二重機械学習（DML）
  - 行動が二値$a \in \{0,1\}$の場合に加えて連続値の場合に対して線形モデルとした

    セミパラメトリックモデル（a以外に関しては非線形を許容）
  - 結果yを説明変数（a以外）に回帰したモデルの残差を、行動を説明変数に回帰したモデルの残差に回帰
  - 残差が過小推定とならないようにデータ分割、クロスフィットを行う

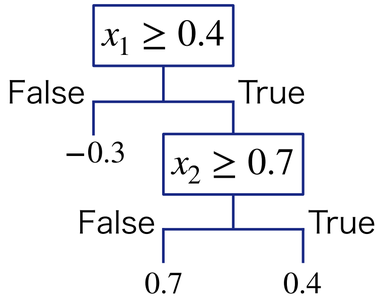
出欠コード
XXXX

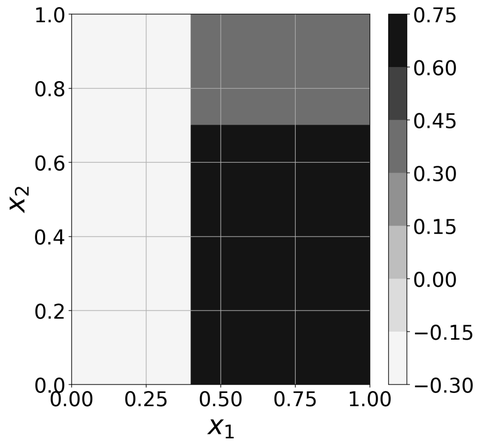
決定木は各次元で領域分割した上で定数で予測するモデル

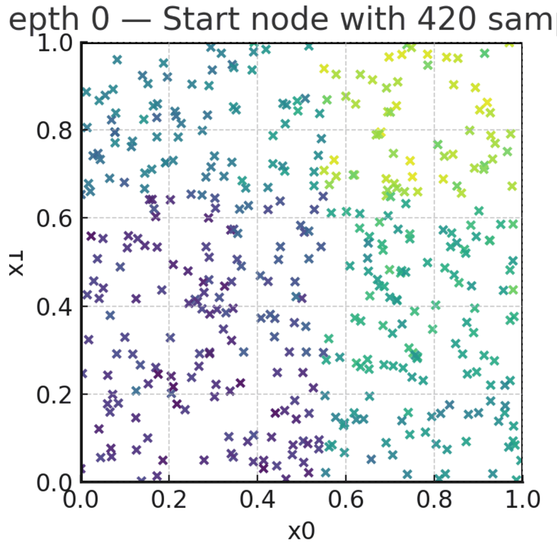
- 決定木は説明変数の各次元の値の大小によって場合分けした領域ごとに定数を予測値として出力
  - 分割ごとの変数としきい値、領域ごとの予測値が学習可能なパラメタ
- 決定木の基本的な学習法

1. 現状の領域分割に追加して最も目的関数（損失）を改善する分割を逐次的に探す
   - 二乗誤差による回帰では、領域の予測値は平均値とする
2. 停止条件に合致したら停止
   - 木の深さが一定になるか、領域内のサンプルサイズが一定以下になったらその領域は分割終了

- 各分割は説明変数軸に沿って行われる
  - 「斜め切り」はしない。ただ、表データは各軸が意味的に独立していることが多く、現実には有益な制約であることが多いとされる







因果推論向けの決定木

因果木：データ分割の工夫と因果効果の直接推定

- 以下は割付確率pが一定のランダム化試験（0<p<1）
  - 決定木＝領域内は同一の値で近似
    - xの属する領域を$\ell(x)$として、　（※葉ノード＝リーフのL。損失関数のℓとは全く別）

      $\tau(x) = \mathbb{E}[y_1 \mid x] - \mathbb{E}[y_0 \mid x] \approx \mathbb{E}[y_1 \mid \ell(x)] - \mathbb{E}[y_0 \mid \ell(x)]$
  - 領域内の曝露群と対照群の平均差を使う（観察データでは交絡調整が必要）
- 領域分割の最適化に用いる訓練データ$\mathcal{D}^{\mathrm{tr}}$、予測値の計算に用いる推定用データを分ける：**Honestアプローチ**
  - 説明変数の空間$\mathcal{X}$の領域$\ell_k \subset \mathcal{X}$によるある分割 $\Pi = \{\ell_k\}_k$ $(\cup_k \ell_k = \mathcal{X})$ に対して推定用データ$\mathcal{D}^{\mathrm{est}}$を用いて
  - $\hat{\tau}(x; \mathcal{D}^{\mathrm{est}}, \Pi) = \dfrac{1}{|\mathcal{D}_1^{\ell(x)}|} \sum_{i \in \mathcal{D}_1^{\ell(x)}} y^i - \dfrac{1}{|\mathcal{D}_0^{\ell(x)}|} \sum_{i' \in \mathcal{D}_0^{\ell(x)}} y^{i'}$　と推定する。ここで$\mathcal{D}_a^{\ell(x)}$は$\mathcal{D}^{\mathrm{est}}$のうち$\ell(x)$に属し行動が$a$のサンプル
- 目的関数も推定分散を考慮して修正
  - 以下の目的関数を**最大化**するように領域Πを学習
  - $-\mathrm{EMSE}_\tau(\Pi) = \mathbb{E}_x[\tau^2(x;\Pi)] - \mathbb{E}_x\left[\mathbb{V}_{\mathcal{D}^{\mathrm{est}}}\left[\hat{\tau}(x; \mathcal{D}^{\mathrm{est}}, \Pi) \mid x\right]\right]$　の推定として以下の目的関数を得る
  - $-\widehat{\mathrm{EMSE}}_\tau\left(\mathcal{D}^{\mathrm{tr}}, \Pi\right) := \dfrac{1}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{D}^{\mathrm{tr}}} \hat{\tau}^2\left(x^i; \mathcal{D}^{\mathrm{tr}}, \Pi\right) - \left(\dfrac{1}{N^{\mathrm{tr}}} + \dfrac{1}{N^{\mathrm{est}}}\right) \cdot \sum_{\ell \in \Pi} \left(\dfrac{(S_1^{\mathrm{tr}}(\ell))^2}{p} + \dfrac{(S_0^{\mathrm{tr}}(\ell))^2}{1-p} \right)$　ただし$(S_a^{\mathrm{tr}}(\ell))^2$は領域内の各群の結果の経験分散
  - $\hat{\tau}$は平均が入るので、それが領域ごとに異なっているほどxごとの異質性を捉えており近似精度が高い（第１項）
    かつ領域内の分散は小さいほど推定分散が小さい（第２項）

（参考）因果木の目的関数の導出

推定用データと訓練データの独立性を利用して訓練の楽観バイアスと推定分散を推定できる

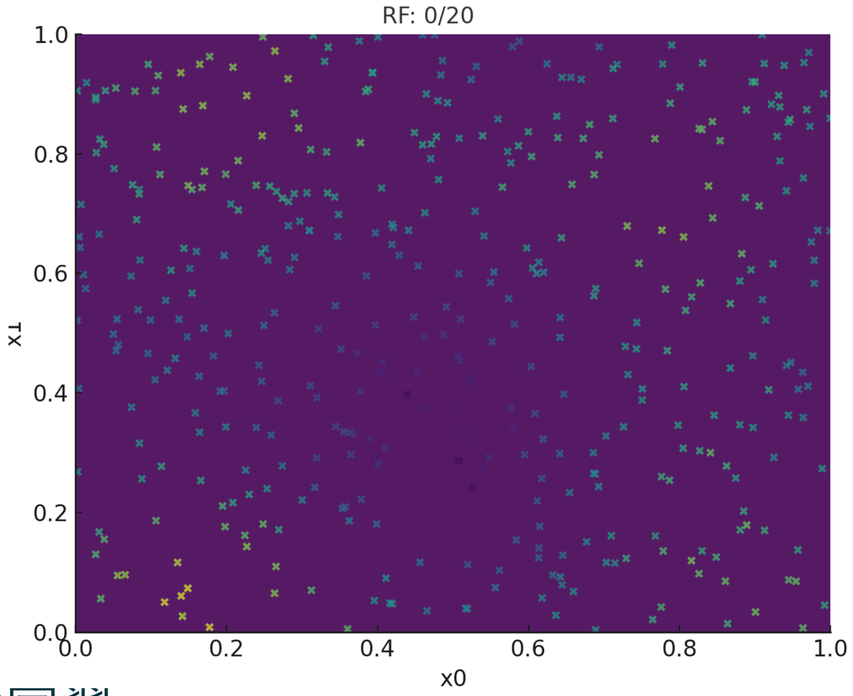
- 固定した分割Πに関して、$\mathscr{D}^{\mathrm{est}}$を用いて前ページのように$\hat{\tau}$を定義し、これを$\mathscr{D}^{\mathrm{te}}$で評価した場合のMSEを考える
  - $$\mathrm{MSE}_{\tau}(\mathscr{D}^{\mathrm{te}}, \mathscr{D}^{\mathrm{est}}, \Pi) := \frac{1}{N^{\mathrm{te}}} \sum_{i \in \mathscr{D}^{\mathrm{te}}} \left\{ \left(\tau^i - \hat{\tau}(x^i; \mathscr{D}^{\mathrm{est}}, \Pi)\right)^2 - (\tau^i)^2 \right\}$$
  - ここで最後の $-(\tau^i)^2$ は解析の都合で付与した、モデルによらない定数
- $\mathscr{D}^{\mathrm{est}}$、$\mathscr{D}^{\mathrm{te}}$に関して期待値をとる
  - $$\mathrm{EMSE}_{\tau}(\Pi) := \mathbb{E}_{\mathscr{D}^{\mathrm{te}}, \mathscr{D}^{\mathrm{est}}} \left[\mathrm{MSE}_{\tau}(\mathscr{D}^{\mathrm{te}}, \mathscr{D}^{\mathrm{est}}, \Pi) \right]$$
- 負号をつけて式展開する。領域$\mathscr{\ell}(x)$における真の平均因果効果を$\tau(x^i; \Pi)$として分解して
- $$\begin{aligned}
-\mathrm{EMSE}_{\tau}(\Pi) &= -\mathbb{E}_{(x^i, \tau^i), \mathscr{D}^{\mathrm{est}}} \left[(\tau^i - \hat{\tau}(x^i; \mathscr{D}^{\mathrm{est}}, \Pi))^2 - (\tau^i)^2 \right] \\
&= -\mathbb{E}_{(x^i, \tau^i)} \left[(\tau^i - \tau(x^i; \Pi))^2 - (\tau^i)^2 \right] \\
&\quad -\mathbb{E}_{x^i, \mathscr{D}^{\mathrm{est}}} \left[\left(\hat{\tau}(x^i; \mathscr{D}^{\mathrm{est}}, \Pi) - \tau(x^i; \Pi) \right)^2 \right] \\
&= -\mathbb{E}_{x^i} \left[-2\mathbb{E}[\tau^i \mid x^i] \tau(x^i; \Pi) + (\tau(x^i; \Pi))^2 \right] \\
&\quad -\mathbb{E}_{x^i} \left[\mathbb{V}_{\mathscr{D}^{\mathrm{est}}} \left[\hat{\tau}(x^i; \mathscr{D}^{\mathrm{est}}, \Pi) \mid x^i \right] \right] \\
&= \mathbb{E}_{x^i} \left[\tau^2 \left(x^i; \Pi \right) \right] - \mathbb{E}_{x^i} \left[\mathbb{V}_{\mathscr{D}^{\mathrm{est}}} \left[\hat{\tau}(x^i; \mathscr{D}^{\mathrm{est}}, \Pi) \mid x^i \right] \right]
\end{aligned}$$

$\leftarrow \mathbb{E}\left[\tau^i \mid \mathscr{\ell}(x^i)\right] = \tau(x^i; \Pi)$

出欠コード
XXXX

因果フォレスト：因果木のアンサンブル

- ランダム森（教師あり学習法）
  - 決定木をベース学習器とするアンサンブル法
    1. データを重複ありで再抽出し、
       分割候補の変数もランダムに選んで決定木を学習する
       - 木の多様性を高める（木どうしの相関を下げる）ため
    2. これを繰り返し、出力は各木の出力の平均とする
- 因果フォレスト（Causal Forest）
  - 因果木をベース学習器$\hat{\tau}_j(x)$とし、重複なしのサブサンプルで学習
  - $$\hat{\tau}(x) = \frac{1}{m}\sum_{j=1}^{m}\hat{\tau}_j(x)$$



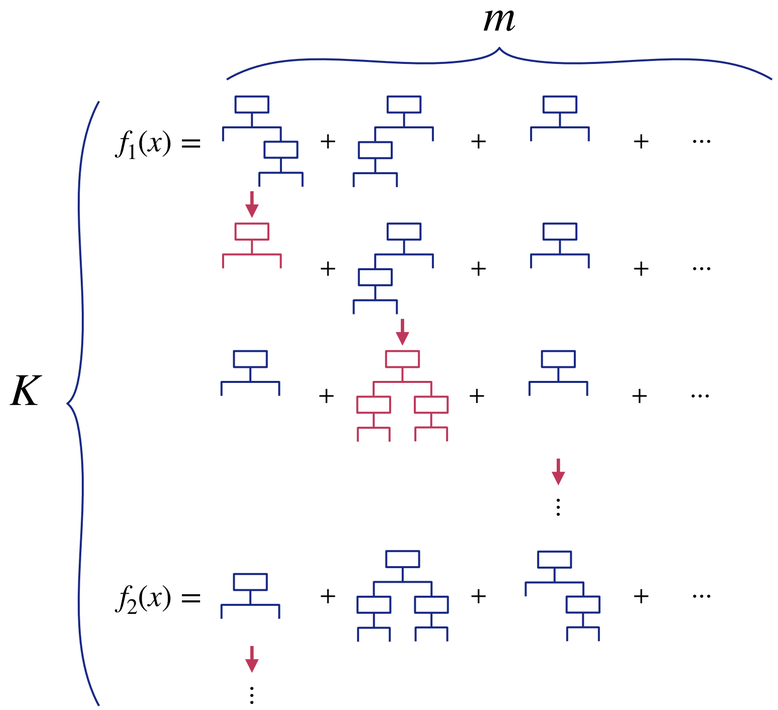
ベイズ的決定森

BARTはベイズ推論（MCMC）による決定森
もともとは因果推論用ではないが、因果推論にもよく使われる

- ベイズ推論
  - ベイズの定理を用いてデータからパラメタの分布を推論する
    - $p(\theta \mid D) = \dfrac{p(D \mid \theta)}{\int p(D \mid \theta')p(\theta')\mathrm{d}\theta'} p(\theta) \quad \cdots (\bigstar)$
    - 予測時はパラメタに関して期待値をとる： $\hat{y} = \mathbb{E}_{\theta \sim p(\theta \mid D)} \mathbb{E}[y \mid x, \theta]$
      - 実際には$p(\theta \mid D)$からサンプリングしたパラメタ集合$\{\theta_1, \ldots, \theta_K\}$の各モデルが返す予測値を平均する
  - BARTでは(★)式の分母の積分を直接計算するのが困難
    → マルコフ連鎖モンテカルロ（MCMC）法
    - あるパラメタθ_{t-1}から対称な提案分布で候補θ^*に遷移するか止まるかを

      受容確率 $\min\left\{1, \dfrac{p(\theta^* \mid D)}{p(\theta_{t-1} \mid D)}\right\}$に従って決める（対称提案のメトロポリス法）。適切な条件のもとで$\{\theta_t\}_t$は

      $p(\theta \mid D)$を定常分布とする連鎖になる
    - MCMC Interactive Gallery
- BART（Bayesian Additive Regression Trees）
  - 複数の木の和を1つのモデルとし、そのパラメタ θ をMCMCで推定する回帰モデル
  - これをS-Learnerとして因果推論に用いることが提案されている [Hill 2011]

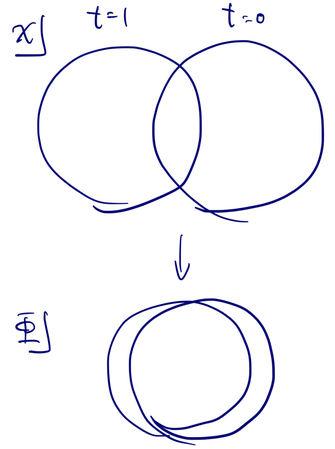


木を1つずつ更新する
（backfitting MCMC）

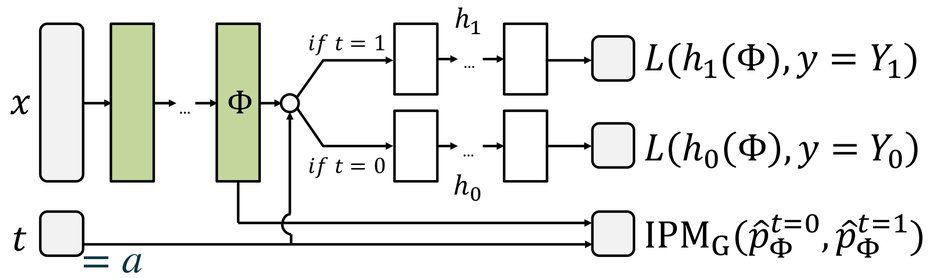
Hill, Jennifer L. "Bayesian nonparametric modeling for causal inference." *Journal of Computational and Graphical Statistics* 20.1 (2011): 217-240.

深層学習による敵対的ドメイン適応として解く

- 実績分布が傾向スコア $\mu(a|x)$ によって偏ることが問題
- Counterfactual Regression（CFR）
  - 行動 a によって分布変化しない表現 $\phi_x = \phi(x)$ を抽出し、
    $p(\phi_x|a=0) \simeq p(\phi_x|a=1)$ になるようにして、
    ϕ_x 空間上で構築したモデルの誤差低減を目指す（均衡化だけでは保証されない）



- CFRは積分確率計量 Integral Probability Metric (IPM) によって
  分布間不一致を測り、これを同時最小化（損失に加える）
  - $$\mathrm{IPM}_G(p_1, p_2) := \sup_{g \in G} \left| \int_\Phi g(\phi)(p_1(\phi) - p_2(\phi)) \mathrm{d}\phi \right| .$$
  - 関数クラスGの設定は1-リップシッツ関数などが用いられる
    - この場合のIPMはワッサースタイン計量とも呼ばれる
  - 群間の差を測る関数gをGの制約のもとで最大化する（敵対的学習）
    - （上記supの双対問題と呼ばれる別問題を毎回解く方法もある）



（参考）敵対的ドメイン適応の理論

群間のx 分布による損失の差を sup で上から抑える

- 理論（引用式のtは行動a、Φは表現抽出器φ）
  - 表現抽出器 ϕ が逆関数を持つなら、反事実損失 $\mathrm{MSE}_{\mathrm{CF}}$ が、
    各群の事実損失の重み付き和 R_F と分布間IPMを用いて、下記の条件のもとで上から抑えられる
  - → R_F + B_Φ IPM を最小化して $\mathrm{MSE}_{\mathrm{CF}}$ の低減を目指す

反事実（観測されなかった潜在結果）の誤差

$$\mathrm{MSE}_{CF}(h,\Phi) \le \underbrace{(1-u)\mathrm{MSE}_F^{t=1}(h,\Phi) + u\mathrm{MSE}_F^{t=0}(h,\Phi)}_{\text{各群の事実損失の重み付き和（通常の事実損失とは重みが逆）}} + B_\Phi \cdot \mathrm{IPM}_{\mathrm{G}}\left(p_\Phi^{t=1}, p_\Phi^{t=0}\right),$$

（$\downarrow p(a=1)$）

- Φ による表現上の予測モデル h と損失 L (MSE) に対する期待損失
  を正の定数 B_Φ で割った関数が、両行動でGに入ると仮定。

$$\ell_{h,\Phi}(x,t) = \int_{\mathcal{Y}} L(Y_t, h(\Phi(x),t)) p(Y_t|x) dY_t$$

- 未知の期待損失 $\ell_{h,\Phi}$ の群間の差を sup で抑える

決定森と深層学習を用いた手法

- 因果木は出力をτとし分割学習用と予測値推定用にデータ分割した決定木
- 因果フォレストは因果木をベース学習器とするランダム森
  - ランダム森はデータをサブサンプリングして学習した決定木を平均したもの
- BARTはベイズ推論を複数の木の和のモデルに適用したもの
- 深層学習を用いるとIPMによる正則化を敵対的学習で実現可
  - 重み付けとは異なるアプローチ