Multiple treatment or Continuous treatment

Causal Inference and Missing Values

Ayoub Tabaai, Benjamin Cohen, Oumaima BOUTHER, Jean Pachebat, Hamza Oulhaj

Supervisor :
Julie josse

April 22, 2022

Overview

Abstract

Several methods to estimate the average effect (IPW, g-estimators, AIPW) and heterogeneous effects in the context of binary treatment already exist. **Purpose**: make a state of the art of the available methods to manage multiple treatments or even continuous treatments (doses).

$\Longrightarrow$ list the main estimators and their characteristics.
$\Longrightarrow$ list their implementations to compare them empirically by reproducing simulations of a real dataset.

Imputation or IPW on the propensity score can reduce the initial covariate bias between the treatment and control groups. It will help us to create recommendations to the users. We will prove some results empirically using Python and also try to suggest improvements if needed.

Notations and reminders on Binary Treatment

N $<$ [total number of people in the super-population].
$\forall$ i $\in$ (1,...,N), (Y_i, X_i, T_i): observed outcome, set of covariates and binary treatment assignment of subject i.

1. τ: treatment space. Binary treatment: $\tau = t_1, t_2$.
2. (n_{t_1}, n_{t_2}): number of subjects receiving treatments t_1 and t_2.
3. $e_{t_1,t_2}(X) = P(T = t_1|X)$ the propensity score(PS), and $\hat{e}_{t_1,t_2}(X)$ be the estimated PS.
4. $I_i(t)$ = (1 if $T_i = t$, 0 otherwise).

Every subject $\rightarrow$ unique treatment at a specific point in time.
This means that we observe either $Y_i(t_1)$ or $Y_i(t_2)$ for each subject.
Rubin Causal Model (RCM) $\rightarrow$ SUTVA to define the potential outcomes.

Definition : Stable Unit Treatment Value Assumption (SUTVA)

The potential outcome observation on one unit should be unaffected by the particular assignment of treatments to the other units.

Estimands of superpopulation binary effects

Population average treatment effect: $\text{PATE}_{t_1,t_2} = \text{E}(\text{Y}_i(\text{t}_1) - \text{Y}_i(\text{t}_2))$
Population average treatment effect among those receiving t_1:
$\text{PATT}_{t_1,t_2} = \text{E}(\text{Y}_i(\text{t}_1) - \text{Y}_i(\text{t}_2)|\text{T}_i = \text{t}_1)$
Approximated by the sample average treatment effects:

$$\text{SATE}_{t_1,t_2} = \frac{1}{N}\sum_{i=1}^{N}(Y_i(t_1) - Y_i(t_2))$$

$$SATT_{t_1,t_2} = \frac{1}{n_{t_1}}\sum_{i=1}^{N}(Y_i(t_1) - Y_i(t_2)) \times I_{(T_i=t_1)}$$

Number of possible estimands grows with increasing treatment options.
$\tau = (t_1, t_2, ..., t_Z)$ the treatment support for Z treatments.
$Y_i = (Y_i(t_1), Y_i(t_2), ..., Y_i(t_Z))$ the set of potential outcomes for subject i.
(w_1, w_2): subgroups of treatments with $(w_1, w_2) \subseteq \tau^2$ and $w_1 \cap w_2 = \emptyset$.
$(|w_1|, |w_2|)$: cardinalities of w_1 and w_2.

Estimands for multiple treatments as $PATE_{w_1,w_2}$ and $PATT_{w_1|w_1,w_2}$:

$$PATE_{w_1,w_2} = E\left[\frac{\sum_{t \in w_1} Y_i(t)}{|w_1|} - \frac{\sum_{t \in w_2} Y_i(t)}{|w_2|}\right]$$

$$PATT_{w_1|w_1,w_2} = E\left[\frac{\sum_{t \in w_1} Y_i(t)}{|w_1|} - \frac{\sum_{t \in w_2} Y_i(t)}{|w_2|} | T_i \in w_1\right]$$

Series of binomial comparisons

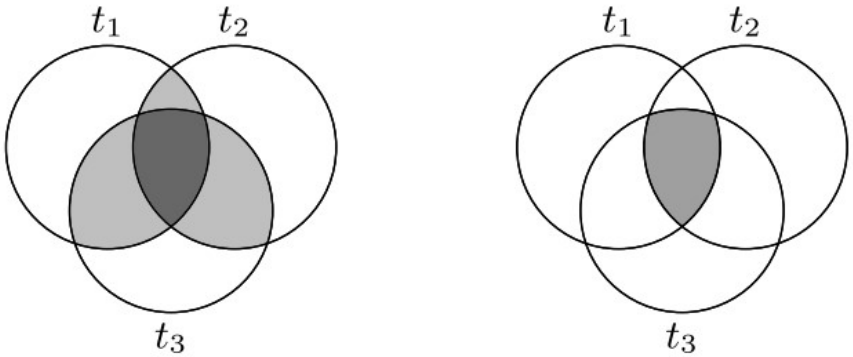


FIG. 2. *Two scenarios of eligible subjects with three treatments shaded areas represent subjects included in a matched analysis.*

Unique common support regions for Z = 3 when using series of binomial comparisons (SBC).
The treatment effects reflect different subsets of the population.
$e_{(t_1,t_2)}(X,T = t_1)$ and $e_{(t_1,t_2)}(X,T = t_2)$: the vector of binary propensity.

Series of binomial comparisons

SBC estimates the causal effect of treatment t_1 versus treatment t_2, among those on t_1:

$$PATT_{E_2(t_1|t_1,t_2)} = E[Y_i(t_1) - Y_i(t_2)|\, T_i = 1, E_{2i}(t_1, t_2) = 1]$$

$E_{2i}(t_1, t_2)$: indicator for subject i having a binary propensity score for treatments t_1 and t_2 within the common support:

$$E_{2i}(t_1, t_2) = 1 \text{ if } e_{(t_1,t_2)}(X_i) \in e_{(t_1,t_2)}(X, T = t_1) \cap e_{(t_1,t_2)}(X, T = t_2)$$

0 otherwise.

Here $\tau = (t_1, t_2, t_3)$ with $n_{t_1} = \min(n_{t_1}, n_{t_2}, n_{t_3})$, with t_1 reference group.

1. $\forall$ subject receiving (t_1, t_2) or (t_1, t_3) we use logistic regression to estimate respectively $e_{t_1,t_2}(X)$ and $e_{t_1,t_3}(X)$.
2. Matching step: each pair of units receiving t_1 or t_2 are matched using $\hat{e}_{t_1,t_2}(X)$ and pairs of units receiving t_1 or t_3 are matched using $\hat{e}_{t_1,t_3}(X)$.
3. Construction of matched triplets with the patients receiving t_1 matched to both a unit receiving t_2 and a unit receiving t_3, with their matches.
   Matched pairs from treatments t_1 and t_3 are dropped if the unit getting t_1 did not match a unit on treatment t_2, and pairs of units receiving t_1 and t_2 are dropped when there is no match for the reference unit to a unit receiving t_3.

Notation:

E_{3i}: indicator of getting two pairwise treatments.

$$E_{3i} = \begin{cases} 1 \ \textit{if} \ E_{2i}(t_1, t_2) = 1 \ \textit{and} \ E_{2i}(t_1, t_3) = 1 \\ 0 \ \textit{otherwise} \end{cases}$$

We then estimate the average difference in the potential outcomes:

$$PATT_{E_3(t_1|t_1,t_2)} = E(Y_i(t_1) - Y_i(t_2)|T_i = t_1, E_{3i} = 1)$$

$$PATT_{E_3(t_1|t_1,t_3)} = E(Y_i(t_1) - Y_i(t_3)|T_i = t_1, E_{3i} = 1)$$

$$PATT_{E_3(t_1|t_2,t_3)} = E(Y_i(t_2) - Y_i(t_3)|T_i = t_1, E_{3i} = 1)$$

This method can underestimate the sampling variance, because we do not take into account the variability caused by the matching procedure.

Need for a relaxed version of weak unconfoundedness.
IPW needs that $\forall t \in \tau$, $P(I_i(t) = 1|Y_i(t), X_i) = P(I_i(t) = 1|X_i)$.
We then get the following estimands with the generalized propensity score (GPS), r(t,X) = Pr(T = t|X = x):

$$PATE_{t_1,t_2} = E[Y_i(\hat{t}_1)] - E[Y_i(\hat{t}_2)]$$

with

$$E[Y_i(\hat{t}_1)] = (\sum_{i=1}^{N} \frac{Y_i I(T_i = t_1)}{r(t_1, X_i)}) \times (\sum_{i=1}^{N} \frac{I(T_i = t_1)}{r(t_1, X_i)})^{-1}$$

$$E[Y_i(\hat{t}_2)] = (\sum_{i=1}^{N} \frac{Y_i I(T_i = t_2)}{r(t_2, X_i)}) \times (\sum_{i=1}^{N} \frac{I(T_i = t_2)}{r(t_2, X_i)})^{-1}$$

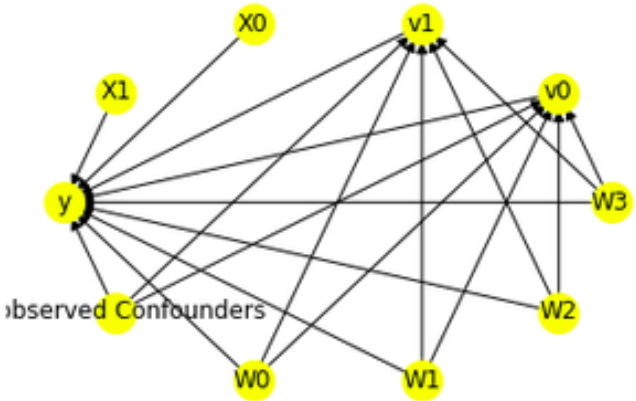
Weights $\approx 0 \rightarrow$ estimates with large variances.
Solved by pruning subjects with extreme weights $\implies$ reducing variance but increasing bias.
Doubly robust methods, covariate balancing propensity score, generalized boosted models are other ways to work against those extreme weights.

Causal Inference in Python

Python Packages for causal inference:

Causallib: provides a suit of causal methods with a scikit-learn API for regression and classification tasks.

DoWhy: Developped by Microsoft, DoWhy is an end-to-end libray for Causal Inference, which provides a principled four-step interface for causal inference that focuses on explicitly modeling causal assumptions and validating them as much as possible.



Implementation for multiple treatment IPW

Dataset: Generated dataset available at:
https://github.com/shuyang1987/multilevelMatching
The data consist of 300 entries with 3 different treatments and six associated covariates. The outcome, y, is the outcome on which we want to perform estimation of the PATE.

	Unnamed: 0	outcome	treatment	covar1	covar2	covar3	covar4	covar5	covar6
0	1	-5.126366	1	-0.865533	0.236547	0.229628	-2.891290	0.211508	0
1	2	-3.029197	1	0.270518	-0.352855	-0.402111	-2.214653	0.070739	1
2	3	3.050304	1	1.421589	1.316865	-1.199193	0.055687	1.258639	1
3	4	-6.091895	1	-1.391808	-1.072550	1.120137	-2.361204	0.003611	0
4	5	-2.455256	1	-1.146590	0.946788	0.510067	-2.678355	0.069594	0

Implementation for multiple treatment IPW

The model was implemented using the **Causallib** package.

```
# Train:
learner = LogisticRegression(solver="liblinear")
ipw = IPW(learner)
ipw.fit(X, a)
```

```
# We can now predict the weight of each individual:
ipw.compute_weights(X, a).head()
```

```
0    2.048790
1    2.702934
2    3.103907
3    2.301524
4    1.812628
dtype: float64
```

```
Estimated PATE_t1,t2: 0.047893245540196 28
Estimated PATE_t1,t3: 0.3819777857796844
Estimated PATE_t2,t3: 0.3340845402394881
```

The region of common support is the region of overlap between treatment groups

$$r(t,X)^{(low)} = max(min(r(t,X|T=t_1)),...,min(r(t,X|T=t_Z)))$$
$$r(t,X)^{(high)} = min(max(r(t,X|T=t_1)),...,max(r(t,X|T=t_Z)))$$

Let E_{4i} be the indicator for all treatments eligibility, where

$$E_{4i} = \begin{cases} 1 & \text{if} r(t,X_i) \in [r(t,X)^{(low)}, r(t,X)^{(high)}], \forall t \in \mathbb{T} \\ 0 & otherwise \end{cases}$$

Using t_1 as a reference treatment, PATT 's among subjects eligible for all treatments are defined as follows.

$$PATT_{E4}(t_1|t_1,t_2) = E[Y_i(t_1)Y_i(t_2)|T_i = t_1, E_{4i} = 1]$$
$$PATT_{E4}(t_1|t_1,t_3) = E[Y_i(t_1)Y_i(t_3)|T_i = t_1, E_{4i} = 1]$$
$$\ldots = \ldots$$
$$PATT_{E4}(t_1|t_1,t_Z) = E[Y_i(t_1)Y_i(t_Z)|T_i = t_1, E_{4i} = 1]$$

matching:

1. Classify all units using KMC on the logit transform of $\hat{R}_{t,t'}(X)$, where $\hat{R}_{t,t'}(X) = (r(l, X), \forall l \neq t, t')$. This forms K strata of subjects, with similar $Z - 2$ GPS scores (not including $r(t, X)$ or $r(t', X)$in each $k \in K$).
2. Within each strata $k \in K$, use 1:1 matching to match those receiving t to those receiving t' on $logit(r(t, X))$. Matching is performed with replacement using a caliper of $\epsilon SD(logit(r(t, X)))$,where $\epsilon = 0.25$.
3. Subjects receiving t who were matched to subjects receiving all treatments $l \neq t$, along with their matches receiving the other treatments, compose the final matched cohort.

inference:
inferences using vector matching can be obtained by contrasting those matched using a weighted average, with weights proportional to ψ_i, where ψ_i is the number of times subject i is part of a matched set. Let n_{trip} be the number of matched sets:

$$SATT_{E_4(t_1|t_1,t_2)} = \frac{\sum_{i \in E_4} Y_i I(T_i = t_1)\psi_i - Y_i I(T_i = t_2)\psi_i}{n_{trip}}$$

$$SATT_{E_4(t_1|t_1,t_2)} = \frac{\sum_{i \in E_4} Y_i I(T_i = t_1)\psi_i - Y_i I(T_i = t_3)\psi_i}{n_{trip}}$$

$$\ldots = \ldots$$

$$SATT_{E_4(t_1|t_1,t_2)} = \frac{\sum_{i \in E_4} Y_i I(T_i = t_1)\psi_i - Y_i I(T_i = t_Z)\psi_i}{n_{trip}}$$

Propensity Score Subclassification

• The goal of the subclassification method used is to divide the sample into a number of subclasses by the value of the propensity score $r(t, x)$ such that $r(t, x) = \mathbb{P}(T = t|X = x)$.

• We are interested in $\tau(t, t')$ for some pair of treatment levels t and t'. $\tau(t, t') = \mathbb{E}[Y_i(t)] - \mathbb{E}[Y_i(t')]$ To estimate these expectations we construct subclasses or **strata** based on $r(t, x)$.

Steps:

We estimate the first term by :

Step 1: we estimate the value of $Y_i(t)$ on the **jth strata**:

$$\hat{\mu}_{jt} = \frac{1}{N_{jt}} \sum_{i: q_{j-1}^{p(t|x)} \leq p(t|X_i) \leq q_j^{p(t|x)}, T_i = t} Y_i^{obs} \tag{1}$$

Where $q_j^{r(t,x)}$ is the quantile of the empirical distribution $p(t|X_i)$ in the sample, and N_{jt} is the number of units.

Step 2: The overall average of $Y_i(t)$ is then estimated as

$$\hat{\mathbb{E}}[Y_i(t)] = \sum_{j=1}^{5} \frac{N_t}{N} . \hat{\mu}_{jt}$$

Continuous treatement Methods :

- Covariate balancing propensity score for continuous treatment :
- Parametric Covariate balancing propensity score for continuous treatment
- Non parametric Covariate balancing propensity score for continuous treatment
- Generative Adversarial De-confounding (GAD) algorithm

Covariate balancing propensity score

- The causal effect of treatment can be captured by the Average Dose Response Function (ADRF). $ADRF(t) = \mathbb{E}[Y_i(t)]$.

- The CBGPS adapts covariate balancing condition for continuous treatment that $\mathbb{E}[P(T_i|X_i)T_iX_i] = \mathbb{E}(T_i)\mathbb{E}(X_i) = 0$, where X and T are centralized and orthogonalized in preprocessing.

Parametric Covariate balancing propensity score

The generalized propensity score as a density function is given by conditional normal density:

$$f_\theta(T_i^*|X_i^*) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(T_i^* - X_i^{*\top}\beta)^2) \quad (2)$$

Where $T_i^* = s_T^{-1/2}(T_i - \bar{T})$ where s_T and $\bar{T}$ are the sample mean and the sample covariance matrix of T_i. Also, $X_i^* = S_X^{-1/2}(X_i - \bar{X})$ and $\theta = (\beta, \sigma^2)$.
Note that the scaling by the covariance matrix makes the covariates indepenedent.

The stabilizing weight is given by :

$$w_i = \frac{f(T_i^*)}{f_\theta(T_i^*|X_i^*)} = \sigma exp\left[-\frac{1}{2\sigma^2}(T_i^* - X_i^{*\top}\beta)^2 - \frac{T_i^2}{2}\right]$$

$$\mathbb{E}(w_i T_i^* X_i^*) = \mathbb{E}(T_i^*)\mathbb{E}(X_i^*) = 0$$

$$\mathbb{E}(w_i T_i^*) = \mathbb{E}(T_i^*) = 0,\ \mathbb{E}(w_i X_i^*) = \mathbb{E}(X_i^*) = 0 \tag{3}$$

We can estimate $\theta = (\beta, \sigma^2)$ using the following equation:

$$\mathbb{E}\begin{pmatrix} \frac{1}{\sigma^2}(T_i^* - X_i^{*\top}\beta)^2 - 1 \\ \sigma exp\left[\frac{1}{2\sigma^2}(T_i^* - X_i^{*\top}\beta)^2 - \frac{T_i^2}{2}\right] T_i^* X_i^* \end{pmatrix} = 0$$

Non parametric Covariate balancing propensity score

The weights $w_i = \frac{f(T_i^*)}{f_\theta(T_i^*|X_i^*)}$ has a mean of 1:

$$\mathbb{E}(w_i) = \int\int \frac{f(T_i^*)}{f_\theta(T_i^*|X_i^*)} f(T_i^*, X_i^*) dT_i^* dX_i^* = 1$$

Also,

$$\mathbb{E}(w_i T_i^* X_i^*) = \mathbb{E}(T_i^*)\mathbb{E}(X_i^*) = 0$$

And

$$\mathbb{E}(w_i T_i^*) = \mathbb{E}(T_i^*) = 0, \ \mathbb{E}(w_i X_i^*) = \mathbb{E}(X_i^*) = 0 \tag{4}$$

$$\sum_{i=1}^{N} w_i g(X_i^*, T_i^*) = 0, \ \sum_{i=1}^{N} w_i - N = 0 \tag{5}$$

where $g(X_i^*, T_i^*) = (X_i^*, T_i^*, X_i^* T_i^*)^\top$

In this setting, we can express the joint density of each observation in relation to the weights as: $f(T_i^*, X_i^*) = \frac{1}{w_i} f(T_i^*) f(X_i^*)$. Now, our goal is to maximize the empirical likelihood of the data by choosing w_i, but also require w_i to satisfy the constraints listed above. Thus we maximize :

$$\prod_{i=1}^{N} f(T_i^*, X_i^*) = \prod_{i=1}^{N} \frac{1}{w_i} f(T_i^*) f(X_i^*)$$

Subject to :

$$\sum_{i=1}^{N} w_i g(X_i^*, T_i^*) = 0, \ \sum_{i=1}^{N} w_i - N = 0$$

Which is equivalent to maximizing:

$$arg \min_{w \in \mathbb{R}^N} \sum_{i=1}^{N} log(w_i)$$

The problem we have is a maximization problem under constraints, thus we can follow an approach similar to the standard Lagrange multiplier technique for numerically solving this optimization problem.
The Lagrangian is :
$\mathcal{L}(w_i, \lambda, \gamma) = \sum_{i=1}^{N} log(w_i) + \lambda(N - \sum_{i=1}^{N} w_i) + \gamma^\top \sum_{i=1}^{N} w_i g(X_i^*, T_i^*)$
Using the first order conditions we find that $\lambda = 1$ and $w_i = \frac{1}{1-\gamma^\top g(X_i^*, T_i^*)}$.
Therefore, our constrained optimization problem is solved by the unconstrained maximization,

$$arg \max_{\gamma \in \mathbb{R}^K} \sum_{i=1}^{N} log(1 - \gamma^\top g(X_i^*, T_i^*)) \tag{6}$$

Generative Adversarial De-confounding (GAD)

Goals :

$\Longrightarrow$ Make the covariates X independent from the treatment T by randomly shuffling the value of each covariate $X_{.,i}$ over all samples in $D_{obs} = \{T, X\}$;the shuffled covariates would become independent with the treatment T if sample size n$\rightarrow +\infty$.

$\Longrightarrow$ Develop a Generative Adversarial Network to learn a sample weight w on the observed data D_{obs} such that the distribution of weighted observed data would be similar even identical with the "calibration" data D_{cal}, formally $wP(T, X) = P(T, X')$.

The loss is :

$$L(\mathbf{w}, d) = \mathbb{E}_{(t,x)\sim D_{cal}}[l(d(t,x),1)] \tag{7}$$

$$+ \mathbb{E}_{(t,x)\sim D_{obs}}[w_{(t,x)} l(d(t,x),0)], \tag{8}$$

$$s.t; \mathbb{E}_{(t,x)\sim D_{cal}}[w_{(t,x)}] = 1, \mathbf{w} \succeq 0 \tag{9}$$

Where $d(.)$ is our descriminator.
The term $\mathbb{E}_{(t,x)\sim D_{cal}}[w_{(t,x)}] = 1$ avoids all sample weight to be zero, and $\mathbf{w} \succeq 0$ constrains each sample weight to be non-negative.
by solving the following optimization problem:

$$\hat{\mathbf{w}} = arg \max_{\mathbf{w}} (\min_{d} L(\mathbf{w}, d))$$

Algorithm

Input: Observed Data $D_{obs} = \{T, X\}$, stopping criterion $h(D_{obs}, D_{target}, w)$, optimizer for discriminator, $SGD(, L_d(w, d))$, and optimizer for w, $Ranger(w, L_w(w, d))$

Output:sample weight **w**

for $i = 1, 2, , p$ do:

Generating shuffled covariate $X'_{.,i}$ by randomly permuting the elements in $X_{.,i}$ **end for**

Generate target $D_{cal} = \{T, X'\}$ and Initialize $w^0 = [1, 1, ..., 1]$

Initialize d(·) with parameter $_0$

Initialize the iteration variable $t = 0$

Repeat:

$\theta^t \overset{S}{\leftarrow} GD(\theta^{t-1}, L_d(w, d))$

Update sample weight $w^t \overset{R}{\leftarrow} ange(w^{t-1}, L_w(w^{t-1}, d))$

Limit mean of sample weight $w_i^t \overset{n}{\leftarrow} w_i^t / \sum_{i=1}^n w_i^t,\ i = 1, ..., n$

until $h(D_{obs}, D_{cal}, w^t)$ satisfied or max iteration is reached

return sample weight w

Thank you

Questions?