gestimation: R package for estimating causal effects in a tidy way

Submitted by
Md. Eraf Hossain
Roll of the submitter
FH-1100

Supervised by
Md. Mynul Islam
Assistant Professor, ISRT

This project report submitted in partial fulfillment of the requirement
for the
degree of M.S in Applied Statistics.

Institute of Statistical Research and Training(I.S.R.T)
University of Dhaka
August, 2023

Declaration

I certify that the project report entitled as "gestimation: R package for estimating causal effects in a tidy way" submitted as a partial requirement for the degree of M.S in Applied Statistics is the result of my own work, except where otherwise acknowledged, and that this report in whole or in part has not been submitted for an award, including a higher degree, to any other university or institution.

Name: Md. Eraf Hossain

Signature: .....

Date:.....

Acknowledgement

It is my proud privilege to release the feelings of my gratitude to several persons who helped me directly or indirectly to conduct this project work.

At first I would like to thank Almighty, the most Gracious, Merciful and beneficial, for showing me the way to acquire knowledge in Applied Statistics and helping me to complete this project successfully .

I am highly indebted to my project mentor, Md. Mynul Islam, Assistant Professor, ISRT, for his support throughout the tenure of my project in spite of his tighted schedule.

Dedicated
to
My Parents

Abstract

The 'gestimation' R package streamlines effect estimation via exposure and outcome modeling-based standardization, a doubly robust estimator, and bootstrap confidence intervals. Notably, it seamlessly integrates data manipulation, akin to 'dplyr', enhancing usability and efficacy in effect estimation workflows.

Contents

Chapter 1

Introduction

Welcome to the 'gestimation' package vignette. This vignette provides an overview of the main functions offered by the 'gestimation' package, which is designed to facilitate effect estimation using the 'g' method. The package offers three key functions: exposure_modelling_standardization, outcome_modelling_standardization, and dr (doubly robust estimator), along with a utility function bootstrap_ci for obtaining 95% bootstrap confidence intervals. Let's dive into each function and its usage.

1.1 A unique advantage:

One of the standout features of the 'gestimation' package is its seamless integration with data manipulation, offering an experience similar to using popular 'dplyr' functions. While other packages may focus solely on estimation techniques, 'gestimation' goes a step further by providing users with the ability to manipulate their dataset effortlessly within the estimation workflow. This unique advantage sets 'gestimation' apart from its counterparts in the field.

Please install and load the package

```
#devtools::install_github('eraf/gestimation')

library(gestimation)
```

Chapter 2

Exposure modelling based standardization

2.1 Exposure or Treatment Model

The exposure or treatment model is defined as follows, where the probability of treatment A being 1 is determined by a logistic function of covariates L:

$$E(A|L) = P(A = 1|L) = \exp(\beta_0+\beta_1 L_1+\cdots+\beta_q L_q)/(1+\exp(\beta_0+\beta_1 L_1+\cdots+\beta_q L_q)) = e(L)$$

2.2 Average Treatment Effect (ATE)

The average treatment effect is defined as the difference in the expected outcomes between the treated $A = 1$ and untreated $A = 0$ groups:

$$ATE = E(Y|A = 1) - E(Y|A = 0)$$

The expected value of the product of treatment A and outcome Y, given the exposure model, is equivalent to the expected outcome for the treated group:

$$E[AYe(L)] = E(Y|A = 1) \tag{2.1}$$

Similarly, the expected value of the product of $(1 - A)$ and Y, given the exposure model, is equivalent to the expected outcome for the untreated group:

$$E[(1 - A)Ye(L)] = E(Y|A = 0) \tag{2.2}$$

2.3 Standardization of ATE with Exposure Modeling

The ATE can be expressed using the exposure model and the estimated probabilities of treatment. The formula for the ATE becomes:

$$ATE = \sum[A_iY_i/\hat{e}(L_i)]/\sum[A_i/\hat{e}(L_i)] - \sum[(1-A_i)Y_i/(1-\hat{e}(L_i))]/\sum[(1-A_i)/(1-\hat{e}(L_i))] \tag{2.3}$$

This formula involves weights $W_i(1)$ and $W_i(0)$, which are calculated based on the estimated probabilities of treatment and the complement of those probabilities.

2.4 Summarized Formulas

Exposure Model:

$$E(A|L) = e(L)$$

$$ATE = \sum[A_i Y_i/\hat{e}(L_i)]/\sum[A_i/\hat{e}(L_i)] - \sum[(1-A_i)Y_i/(1-\hat{e}(L_i))]/\sum[(1-A_i)/(1-\hat{e}(L_i))]$$

$$Weights: W_i(1) = A_i/\sum[A_i/\hat{e}(L_i)], W_i(0) = (1-A_i)/\sum[(1-A_i)/(1-\hat{e}(L_i))]$$

These formulas describe the standardization process with exposure modeling and how it can be used to estimate the average treatment effect while accounting for the treatment assignment probabilities based on the exposure model. (Hernán and Robins [2010])

2.5 Function description

The exposure_modelling_standardization function calculates the effect of a binary exposure (treatment) on the outcome using the exposure-modeling approach. It takes the following inputs:

data: A data frame containing variables specified in the provided exposure_model. exposure_model: The exposure model. y: The outcome variable. The function returns a tibble containing risks for exposure levels 0 and 1, Risk Difference, Risk Ratio, and Odds Ratio.

2.6 R example

The BRFSS 2019 data has been modified and included in the package for the examples. (Centers for Disease Control and Prevention [2019])

```
library(dplyr)
#>
#> Attaching package: 'dplyr'
#> The following objects are masked from 'package:stats':
#>
#>     filter, lag
#> The following objects are masked from 'package:base':
#>
#>     intersect, setdiff, setequal, union

brfss0 <- brfss %>% filter(gt65 == 0)



exposure_mod <- glm(insured ~ female + whitenh + blacknh + hisp + multinh +
                      gthsedu + rural, family = "binomial",data = brfss0)

exposure_modeling_standardization(data = brfss0,
                                  exposure_model =  exposure_mod,
                                  y = flushot)
```

```
#> # A tibble: 1 x 5
#>   risk0 risk1 risk_diff risk_ratio odds_ratio
#>   <dbl> <dbl>     <dbl>      <dbl>      <dbl>
#> 1 0.199 0.440     0.241       2.21       3.16
```

Chapter 3

Outcome Modeling Based Standardization

3.1 Theoretical backgroud

It is a methodology for estimating mean outcomes in the context of causal inference using a parametric modeling approach. The goal is to estimate the conditional mean of an outcome variable Y given specific conditions on treatment (A), confounders (L), and adjusting for selection bias (C). This is achieved by standardizing the mean outcome to the distribution of the confounders. Here's a summary of the theory and the associated formulas:

3.2 Estimating Conditional Mean via Parametric Modeling

For high-dimensional data with numerous confounders, nonparametric estimation of conditional means might be infeasible due to limited observations. Therefore, a parametric

approach is used. The conditional mean is modeled using a regression function:

$$E(Y|A = a, C = 0, L = l) = f(a, l)$$

This regression function might include terms such as squared continuous variables of L and interactions.

3.3 Standardization of Mean Outcome

The standardized mean outcome is defined as a weighted average of the conditional means, where the weights are based on the probability mass or density function that an individual belongs to a specific stratum (L). This approach avoids the need for nonparametric estimation of the distribution of L.

$$StandardizedMean = \sum[P(L = l) \times E(Y|A = a, C = 0, L = l)]$$

3.4 Steps for Computing Standardized Mean Outcomes

Step 1: Fit the conditional regression model to estimate the conditional means.

Step 2: Create a copy of the original data and set $A = 1$ for all individuals. Calculate the estimated mean for this modified dataset.

$$EstimatedCounterfactualMean(A = 1, C = 0) = (1/n) \times \sum[E(Y|A = 1, C = 0, L = l)]$$

Step 3: Repeat Step 2, but now set $A = 0$ for all individuals.

$$EstimatedCounterfactualMean(A = 0, C = 0) = (1/n) \times \sum[E(Y|A = 0, C = 0, L = l)]$$

These formulas and steps outline a methodology for estimating mean outcomes while accounting for confounders and treatment effects through a parametric modeling approach. The goal is to provide an alternative to nonparametric estimation, especially in cases of high-dimensional data where direct estimation of conditional means might be challenging. (Hernán and Robins [2010])

3.5 Function description

The outcome_modelling_standardization function calculates the effect of a binary exposure (treatment) on the outcome using the outcome-modeling approach. It takes the following inputs:

data: A data frame containing variables specified in the provided outcome_model. outcome_model: The outcome model. exposure: The exposure (treatment) variable. The function returns a tibble containing risks for exposure levels 0 and 1, Risk Difference, Risk Ratio, and Odds Ratio.

3.6 Example in R

```
library(dplyr)
```

```r
brfss0 <- brfss %>% filter(gt65 == 0)



outcome_mod <- glm(flushot ~ insured + female + whitenh + blacknh + hisp +
                     multinh + gthsedu + rural, family = "binomial",
                   data = brfss0)



outcome_modeling_standardization(data = brfss0, outcome_model =  outcome_mod,
                                 exposure =  insured)
#> # A tibble: 1 x 5
#>   risk0 risk1 risk_diff risk_ratio odds_ratio
#>   <dbl> <dbl>     <dbl>      <dbl>      <dbl>
#> 1 0.215 0.440     0.225       2.04       2.86
```

Chapter 4

Doubly Robust Estimator

Doubly Robust estimator is a technique used in causal inference to account for missing data and potential biases in parameter estimates. It combines two approaches: modeling the relationship between covariates and the outcome and modeling the probabilities of missingness given the covariates.

4.1 Formulas and Steps

The goal is to estimate the conditional mean of an outcome variable Y, given specific treatment (A), confounder (L), and selection bias (C) circumstances.

The Doubly Robust estimator involves two models: The treatment model:

$$e(L) = P(A = 1|L) \quad \text{(probability of treatment given covariates } L\text{)}$$

The outcome model:

$$E(Y|L, A = 1) \quad \text{(conditional mean of Y given covariates } L \text{ and } A = 1)$$

The estimator is applied in the following steps:

Calculate the estimated mean for the treated group ($A = 1$):

$$E(\hat{Y}_{a=1}) = (1/n) \times \sum [A_i Y_i / \hat{e}(L_i) - A_i - \hat{e}(L_i)/\hat{e}(L_i) \times E(Y_i|L_i, A_i = 1)]$$

Similarly, calculate the estimated mean for the untreated group ($A = 0$):

$$E(\hat{Y}_{a=0}) = (1/n) \times \sum [(1 - A_i)Y_i/(1 - \hat{e}(L_i)) - A_i - \hat{e}(L_i)/(1 - \hat{e}(L_i)) \times E(Y_i|L_i, A_i = 0)]$$

The Doubly Robust estimator combines these estimates to provide a consistent estimate of the causal effect, even if only one of the models (treatment or outcome) is correctly specified.

4.2 Summary

The Doubly Robust estimator is a powerful technique that addresses missing data and covariate relationships simultaneously, leading to more robust causal inference. It allows for accurate parameter estimation by leveraging both the treatment model and the outcome model, resulting in consistent estimates of causal effects. (Bang and Robins [2005])

4.3 Function description

The dr function provides a doubly robust estimator to estimate the effect. It takes the following inputs:

data: A data frame containing variables specified in both the exposure_model and outcome_model. exposure: The exposure (treatment) variable. y: The outcome variable. exposure_model: The exposure model. outcome_model: The outcome model. id: The unique identifier of subjects. The function returns a tibble containing risks for exposure levels 0 and 1, Risk Difference, Risk Ratio, and Odds Ratio.

4.4 R example

```
library(dplyr)
library(broom)
brfss0 <- brfss %>% filter(gt65 == 0)

exposure_mod <- glm(insured ~ female + whitenh + blacknh + hisp + multinh + gthsedu + ru
outcome_mod <- glm(flushot ~ insured + female + whitenh + blacknh + hisp + multinh + gth
                   data = brfss0)

brfss0 %>%
  mutate(id = row_number()) %>%
  dr(exposure = insured, y = flushot, outcome_model = outcome_mod,
     exposure_model = exposure_mod, id = "id")
```

```
#> # A tibble: 1 x 5
#>      r1    r0    rd    rr    or
#>   <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 0.440 0.187 0.252  2.34  3.40
```

Chapter 5

Bootstrap Confidence Intervals

The bootstrap_ci function calculates 95% bootstrap confidence intervals for the estimated effect. It takes the following inputs:

df: A data frame containing variables specified in the provided formula. f: The g-estimation method. outmod: The outcome model. trtmod: The exposure model. x: The exposure/treatment variable. y: The outcome variable. id: The unique identifier variable of subjects. The function returns a tibble containing bootstrap confidence intervals for Risk Difference, Risk Ratio, and Odds Ratio.

5.1 R example

```
library(dplyr)

library(broom)



brfss0 <- brfss %>% filter(gt65 == 0)
```

```r
exposure_mod <- glm(insured ~ female + whitenh + blacknh + hisp + multinh + gthsedu + ru
outcome_mod <- glm(flushot ~ insured + female + whitenh + blacknh + hisp + multinh + gth

bootstrap_ci(df = brfss0, f = exposure_modeling_standardization,
  trtmod = exposure_mod, y = flushot)
#>             mean   sd conf.low conf.high
#> risk diff  0.24 0.00     0.23      0.25
#> risk ratio 2.20 0.04     2.12      2.28
#> odds ratio 3.15 0.07     3.00      3.30

brfss0 %>%
  mutate(id = row_number()) %>%
  bootstrap_ci(f = dr, trtmod = exposure_mod, outmod = outcome_mod,
  x = insured, y = flushot, id = 'id')
#>             mean   sd conf.low conf.high
#> risk diff  0.25 0.01     0.24      0.27
#> risk ratio 2.35 0.10     2.16      2.54
#> odds ratio 3.41 0.17     3.07      3.75
```

Chapter 6

Conclusion

In summary, the 'gestimation' package provides a suite of functions for effect estimation using the 'g' method. You can choose between exposure modeling, outcome modeling, or a doubly robust estimator, and then obtain bootstrap confidence intervals for your estimates. We hope this vignette helps you effectively utilize the 'gestimation' package for your estimation needs.

Bibliography

Heejung Bang and James M Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

Centers for Disease Control and Prevention. Behavioral risk factor surveillance system: 2019 codebook report, 2019. URL https://www.cdc.gov/brfss/annual_data/2019/pdf/codebook19_llcp-v2-508.HTML.

Miguel A Hernán and James M Robins. Causal inference, 2010.