Denis Ostroushko - HW3

Introduction

Imputation and Variable Processing

For the purpose of this assignment we retain the same imputation schemes we used in the previous two assignments. We will use imputation with the median of observed values and replace missing values with modes for categorical predictors.

Since we will need to create two models that have all possible confounders in the data set, we need to be careful with variable inclusion. We drop variable `Hisp` because it is highly correlated with other variables that contain race and ethnic information. We also `Drug.Add` due to the issues with its imputation. When imputed with the most common level "No", which indicated no drug use, this variable has one unique level. Such zero variance predictors can cause problems with fitting models, so we will avoid using it in our analyses.

Variables `BMI`, `BL.Cig.Day`, `BL.Drks.Day`, `N.living.kids` are imputed with medians like `N.prev.preg`, `Birthweight` in the previous assignments.

`Use.Alc` is imputed with a mode like `Race_ethnicity`, `Use.Tob`

Problem 1

Instead of writing equations for each model, I provide formatted coke chunks that contain model statements.

Logistic regression - A

```
logistic_regression_a <-
  glm(
    I(data$Group == "T") %>% as.numeric() ~

        Race_ethnicity + Public.Asstce +
        Use.Tob + Live.PTB +

        N.prev.preg + BL.GE + BL..BOP + BL..PD.4 + BL..CAL.3,

    data = data,
    family = "binomial"
  )
```

Logistic regression - B

For these non-linear terms, I selected polynomial degrees that minimize AIC, to some reasonable degree.

```
logistic_regression_b <-
  glm(
    I(data$Group == "T") %>% as.numeric() ~

        Race_ethnicity + Public.Asstce +
        Use.Tob + Live.PTB +

        poly(N.prev.preg,2) +
        poly(BL.GE,5) +
        poly(BL..BOP,5) +
        poly(BL..PD.4,5) +
        poly(BL..CAL.3, 2),

    data = data,
    family = "binomial"
  )
```

Logistic Regresson - C

I decided to use forward selection. After a few trials I determined that forward variable selection includes less variables into the final output, while providing similar AIC. Also, when I compared SMD balance, forward stepping variable selection resulted in smaller SMD of the balanced covariates sets.

```
logistic_regression_c_full <-
  glm(
    I(data$Group == "T") %>% as.numeric() ~

        (Race_ethnicity + Public.Asstce +
        Use.Tob + Live.PTB +

        N.prev.preg + BL.GE + BL..BOP + BL..PD.4 + BL..CAL.3)^2,

    data = data,
    family = "binomial"
  )


logistic_regression_c_lower <-
  glm(I(data$Group == "T") %>% as.numeric() ~ 1,
      data, family = "binomial")

logistic_regression_c <-
  MASS::stepAIC(logistic_regression_c_lower,
                direction = "forward",
                trace = 0,
                scope = list(upper = logistic_regression_c_full,
                             lower = logistic_regression_c_lower)
                )
```

Logistic Regression - D

```
# remove variables that are other outcomes, or IDs
data2 =
  data %>%
    select(- PID, - Birth.outcome, -
             GA.at.outcome, -Preg.ended...37.wk,
```

```
            -Birthweight) %>%
  mutate(Race_ethnicity = as.factor(Race_ethnicity))

logistic_regression_d_full <-
  glm(
    I(data$Group == "T") %>% as.numeric() ~ .,
    data = data2,
    family = "binomial"
  )

logistic_regression_d <-
  MASS::stepAIC(
    logistic_regression_c_lower,
    direction = "forward",
    trace = 0,
    scope = list(upper = logistic_regression_d_full,
                 lower = logistic_regression_c_lower)
    )
```

Flexible Regression - Random Forest

```
rf = randomForest(
  Group ~ .,
  data = data2 ,
  ntree = 1000
)
```

Having obtained five sets of propensity scores, we can created five sets of weighted SMDs for all covariates, i.e. potential confounders. We contrast five sets of SMDs and weighted SMD on Figure 1.

A black vertical line is set at 0.1, as we consider any SMD beyond 0.1 to be a sign of possible imbalance.

A blue horizontal line separates covariates we explicitly called in some models from the rest of confounding values. We expect that covariates not explicitly used in confounding control will exhibit higher levels of imbalance. It appears that this is true. It is especially true for logistic regressions models A and B, which used only some predictors. To reiterate, all those predictors are listed below the vertical blue line.

Lastly, due to some fitted probabilities being equal to 1 or 0, I replace these extreme values with numbers that are close to them. For 0, the replacement is `0.1 * 10^-15`, while for 1 the replacement is `1 - 0.1*10^-15`.

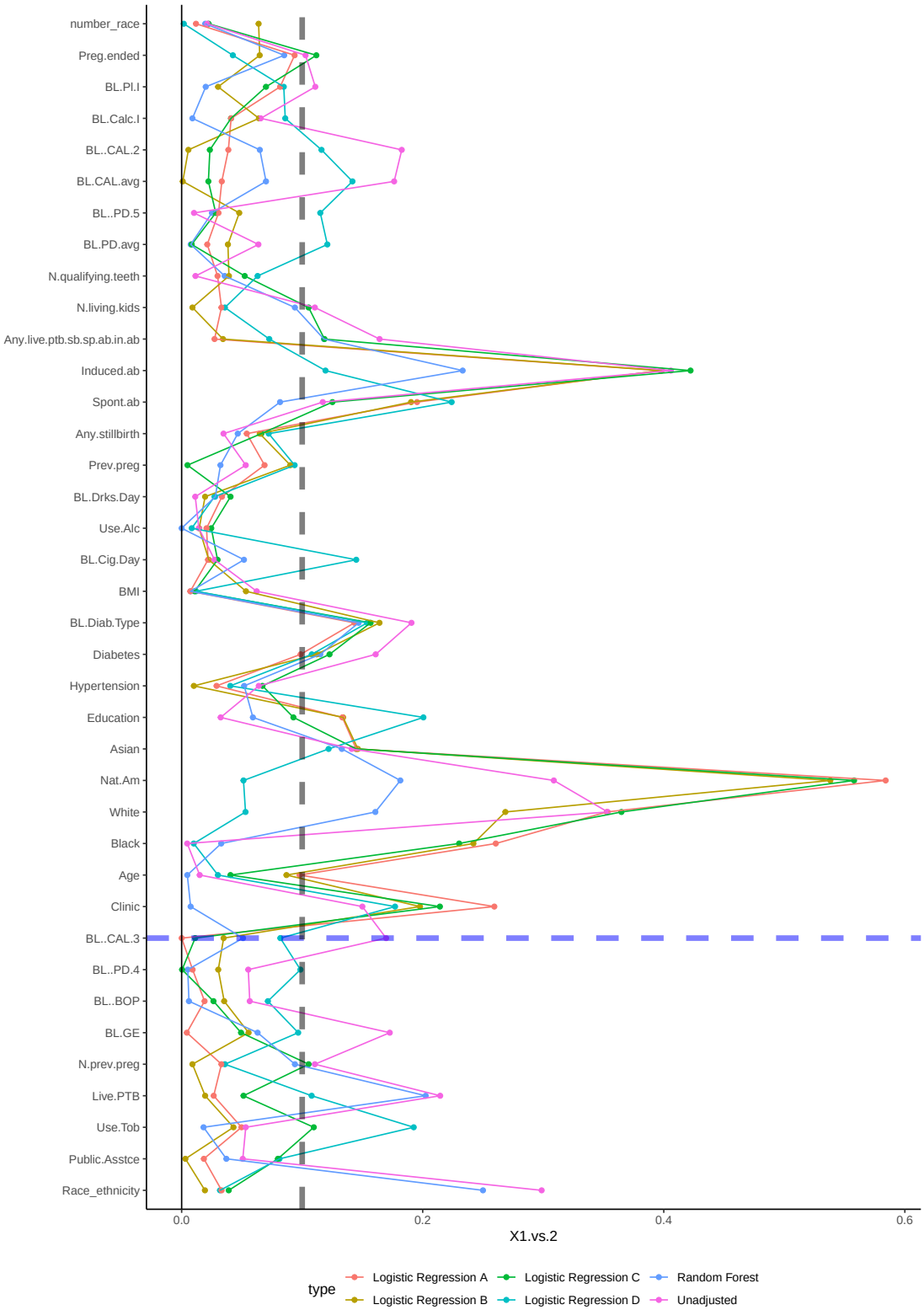

Figure 1: Comparion of confounding adjustment methods and their impact on SMDs

Part B

In order to estimate Treatment Effect using a doubly robust, I will propensity scores from logistic regression specified in part A. Additionally, I specify two outcome models, one for each outcome of interest. Definition of two models are given in the code chunks below:

```
pregnancy_model <-
  glm(
    `Preg.ended...37.wk` ~
      Group + Race_ethnicity + Public.Asstce + Use.Tob + N.prev.preg +
      Live.PTB + BL.GE + BL..BOP + BL..PD.4 + BL..CAL.3,

    data = data,
    family = "binomial"
  )

borthweight_model <-
  lm(
    Birthweight ~
    Group *(Race_ethnicity + Public.Asstce + Use.Tob +
              N.prev.preg + Live.PTB + BL.GE + BL..BOP +
              BL..PD.4 + BL..CAL.3),

    data = data
  )
```

We consider less flexible regression methods in order to allow a higher chance at specifying at least one model for the doubly robust estimator. We wish to observe that a point estimate for ATE will be consistent with IPW2, but will have a smaller bootstrapped SE. My code for obtaining 500 bootstrap iterations is given in an appendix.

My code for calculating ATE using AIPW is given below. I am giving a step by step estimation of ATE for both outcomes variables, and will comment of the results one at a time.

```
# Step 1: get weigths
prop_scores = logistic_regression_a$fitted.values
data_1 <- data_0 <- data

data_1$Group = "T"
data_0$Group = "C"

preg_y_1 = predict(object = pregnancy_model, newdata = data_1, type = "response")
```

```r
preg_y_0 = predict(object = pregnancy_model, newdata = data_0, type = "response")

bw_y_1 = predict(object = borthweight_model, newdata = data_1)
bw_y_0 = predict(object = borthweight_model, newdata = data_0)

### record actual outcomes and treatment flags
preg_y = ifelse(data$Preg.ended...37.wk == "Yes", 1, 0)
bw_y = data$Birthweight
trt_ind = ifelse(data$Group == "T", 1, 0)

  ### Pregnancy ATE
  trt_ind * preg_y / prop_scores -
    ((trt_ind - prop_scores)/prop_scores) * preg_y_1 -> E_Y_1

  (1 - trt_ind) * preg_y / (1 - prop_scores) -
    ((1 - trt_ind) - (1 - prop_scores))/
    (1 - prop_scores) * preg_y_0 -> E_Y_0

  AIPW_preg_ate = mean(E_Y_1 - E_Y_0)

  ### Birthweight ATE

  trt_ind * bw_y / prop_scores -
    ((trt_ind - prop_scores)/prop_scores) * bw_y_1 -> E_Y_1

  (1 - trt_ind) * bw_y / (1 - prop_scores) -
    ((1 - trt_ind) - (1 - prop_scores))/
    (1 - prop_scores) * bw_y_0 -> E_Y_0

  AIPW_bw_ate = mean(E_Y_1 - E_Y_0)
```

Pre-term pregnancy AIPW

- Pre-term pregnancy rate reduction estimate is -0.03, suggesting that the program was responsible for a -3% reduction in the rates of pre-term pregnancies as a result of program intervention.
- Bootstrapped standard error is 0.03
- A normal 95% confidence interval for the treatment effect is given by (-0.08, 0.03)

Birth weight AIPW

- Average birth weight of newborns increased by 73.81 as a result of program intervention.
- Bootstrapped standard error is 55.61
- A normal 95% confidence interval for the treatment effect is given by (-35.17, 182.8)

Pre-term pregnancies: comparison with previous results

Figure 2 shows that the variance of all estimators we have considered over the past few weeks. We observe that the variance of Augmented IPW is similar to the IPW2 estimator we considered in the previous assignment. This means that the outcome model was not correctly specified. This tells us that we have either omitted important predictors, or we did not consider a correct amount of non-linear or interaction terms to model the likelihood of pre-term pregnancies.

We also observe that the point estimate matches previous results, showing that the estimator is consistent if we specify just one of the two models. We have high confidence that a glm for pregnancy outcome was misspecified, which means that the propensity score model is likely correctly specified.

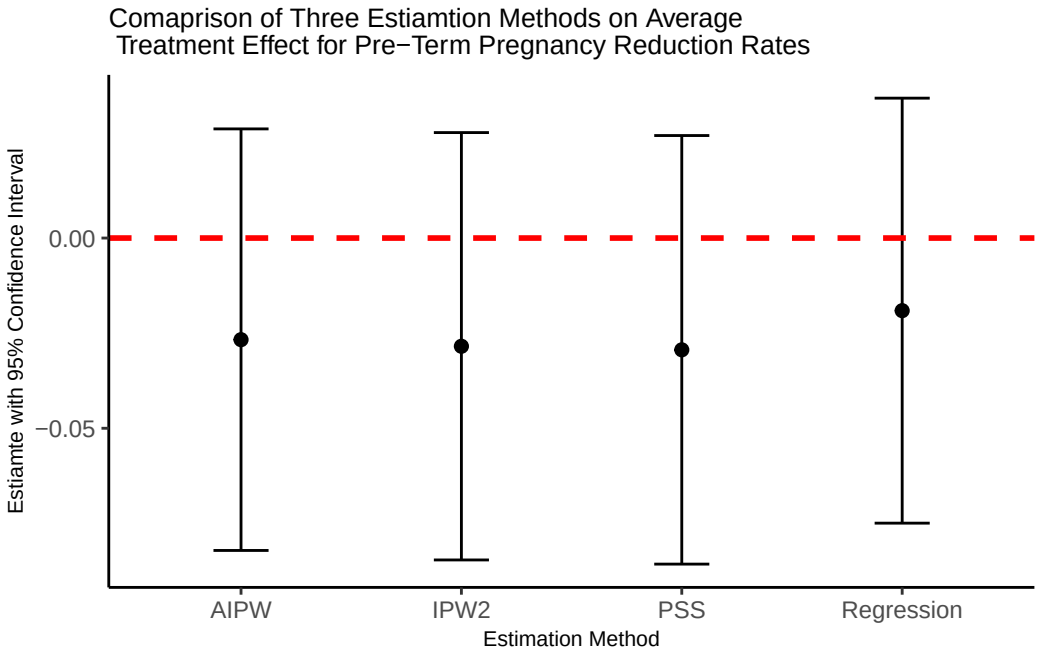

Figure 2: Comparison of Estiamtion Method Varinances for Pre-term pregnancy rate reduction

Birth weights: comparison with previous results

Figure 3 shows that we likely have the same issue with the AIPW estimator for the birthweights ATE. Variance being similar to IPW2 suggests that the model for the average birhtweight was misspecified.

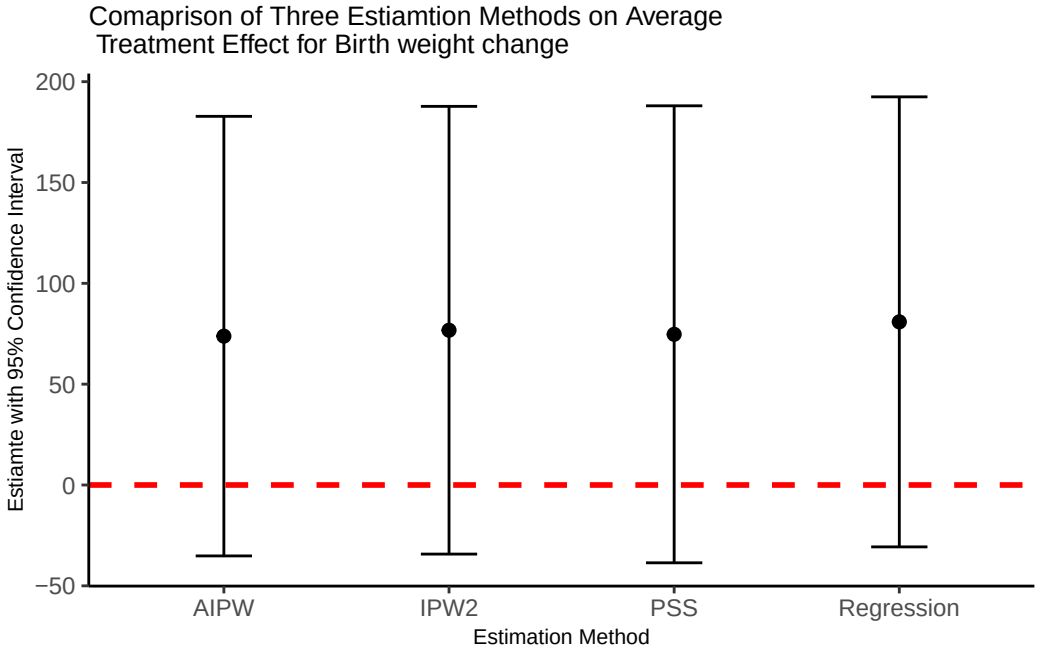

Figure 3: Comparison of Estiamtion Method Varinances for Average Birth Weight Increase

Appendix: bootsrap code

```
K = 500

res <-
  data.frame(
    i = 1:K,
    AIPW_preg_ate = NA,
    AIPW_bw_ate = NA
  )

set.seed(718297)
for(i in 1:K){

  print(i)
  iter_data = data[sample(1:nrow(data), replace = T), ]

  #### train propensity score, pregnancy, and birth weights models
  iter_prop_score <-
    glm(I(iter_data$Group == "T") %>% as.numeric() ~

        Race_ethnicity + Public.Asstce +
        Use.Tob + Live.PTB +

        N.prev.preg + BL.GE + BL..BOP + BL..PD.4 + BL..CAL.3,

    data = iter_data,
    family = "binomial"
  )

  iter_pregnancy_model <-
    glm(
      `Preg.ended...37.wk` ~
        Group + Race_ethnicity + Public.Asstce + Use.Tob + N.prev.preg +
        Live.PTB + BL.GE + BL..BOP + BL..PD.4 + BL..CAL.3,

      data = iter_data,
      family = "binomial"
    )

  iter_borthweight_model <-
```

```
  lm(
    Birthweight ~
    Group *(Race_ethnicity + Public.Asstce + Use.Tob +
              N.prev.preg + Live.PTB + BL.GE + BL..BOP +
              BL..PD.4 + BL..CAL.3),

    data = iter_data
  )

#### propensity scores, counterfactuals
prop_scores = iter_prop_score$fitted.values

iter_data_1 <- iter_data_0 <- iter_data

iter_data_1$Group = as.factor("T")
iter_data_0$Group = as.factor("C")

preg_y_1 = predict(object = iter_pregnancy_model, newdata = iter_data_1, type = "respons
preg_y_0 = predict(object = iter_pregnancy_model, newdata = iter_data_0, type = "respons

bw_y_1 = predict(object = iter_borthweight_model, newdata = iter_data_1)
bw_y_0 = predict(object = iter_borthweight_model, newdata = iter_data_0)

### record actual outcomes and treatment flags
preg_y = ifelse(iter_data$Preg.ended...37.wk == "Yes", 1, 0)
bw_y = iter_data$Birthweight
trt_ind = ifelse(iter_data$Group == "T", 1, 0)

### Pregnancy ATE
trt_ind * preg_y / prop_scores -
  ((trt_ind - prop_scores)/prop_scores) * preg_y_1 -> E_Y_1

(1 - trt_ind) * preg_y / (1 - prop_scores) -
  ((1 - trt_ind) - (1 - prop_scores))/
  (1 - prop_scores) * preg_y_0 -> E_Y_0

res$AIPW_preg_ate[i] = mean(E_Y_1 - E_Y_0)

### Birthweight ATE

trt_ind * bw_y / prop_scores -
```

```
    ((trt_ind - prop_scores)/prop_scores) * bw_y_1 -> E_Y_1

  (1 - trt_ind) * bw_y / (1 - prop_scores) -
    ((1 - trt_ind) - (1 - prop_scores))/
    (1 - prop_scores) * bw_y_0 -> E_Y_0

  res$AIPW_bw_ate[i] = mean(E_Y_1 - E_Y_0)

}

write.csv(res, "Augment IPW bootstrap results.csv")
```