Causal Inference

Wakeel Kasali

```r
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
```

```r
library(ggplot2)
library(readr)
library(boot)
library(cobalt)  # required for love plots
```

```
##  cobalt (Version 4.5.5, Build Date: 2024-04-02)
```

```r
library(MatchIt)
```

```
##
## Attaching package: 'MatchIt'

## The following object is masked from 'package:cobalt':
##
##     lalonde
```

Project Objective

This project aims to estimate the causal effect of a binary exposure variable `X` on a continuous outcome variable `Y`, adjusting for potential confounders `C`. Specifically, we estimate the causal contrast:

$$\Delta = \mathbb{E}\left[\mathbb{E}(Y \mid X = 1, C)\right] - \mathbb{E}\left[\mathbb{E}(Y \mid X = 0, C)\right]$$

To accomplish this, we implement and compare the following four estimation strategies:

1. **Regression Estimator**

2. **Inverse Probability Weighted (IPW) Estimator**

3. **Double-Robust Estimator**

4. **Stratification Estimator (based on quintiles of the propensity score)**

Each estimator is implemented in R. Where applicable, standard errors are computed to assess the precision of the estimates. This project provides a practical comparison of causal inference methods in observational data settings.

The goal is to better understand how different estimators behave when confounding is present and to demonstrate reproducible causal analysis using real-world data.

Although the data used in this study is basically about attrition; implying whether the employee left the company. However, it's been widely used for survival analysis and for that reason we would consider using it because of some of the variables of interest present. The six variables which we would use in the dataset are `Overtime`, `MonthlyIncome`, `JobLevel`, `JobRole`, `PerformanceRating` and `MaritalStatus`. Below are the reasons why they fit for the analytical method in our statistical problem.

- MonthlyIncome is a continuous variable which is the employee's base financial compensation.
- Overtime is a **treatment** as it binary (Yes/No) referring to whether an employee works overtime or not. It reflects organizational policy i.e. actionably, the management could reduce overtime requirements or enforce equitable pay policies.
- `JobLevel`, `JobRole`, `PerformanceRating` and `MaritalStatus` are employee characteristics and being background factors that might confound the relationship between `MonthlyIncome` and `Overtime`. The relationship signifies medium of allowing leadership to determine if overtime is truly being compensated fairly.

Below are the reasons for including the other factors as confounders:

- In `JobLevel`, senior-level employees may work longer hours and earn higher income regardless of overtime.
- In `Jobrole`, different roles (e.g., Sales Executive) might have different expectations and compensation structures.
- In `PerformanceRating`, high-performing employees may be encouraged to work overtime and may also earn bonuses.
- In `MaritalStatus`, personal obligations might affect the likelihood of working overtime, and companies may compensate differently depending on family responsibilities.

With that being said, we have established the statistical structure necessary for the solution to causal analysis problems.

```r
Causal <- read.csv("data.csv", header = TRUE)

table(Causal$OverTime)
```

```
##
##   No  Yes
## 1054  416
```

```r
Causal <- read.csv("data.csv")

# Select the relevant variables for the analysis
causal_data <- Causal %>%
  select(
    OverTime,            # Treatment
    MonthlyIncome,       # Outcome
    JobLevel,            # Confounder 1
    JobRole,             # Confounder 2
    PerformanceRating,   # Confounder 3
    MaritalStatus        # Confounder 4
  ) %>%
  mutate(
    OverTime = ifelse(OverTime == "Yes", 1, 0),  # Convert to binary 0/1
    JobRole = as.factor(JobRole), # Discrete variable
    MaritalStatus = as.factor(MaritalStatus)
  )

# Check treatment balance
table(causal_data$OverTime)
```

```
##
##    0    1
## 1054  416
```

```
table(causal_data$JobLevel)
```

```
##
##   1   2   3   4   5
## 543 534 218 106  69
```

```
table(causal_data$JobRole)
```

```
##
## Healthcare Representative           Human Resources     Laboratory Technician
##                       131                        52                       259
##                   Manager    Manufacturing Director         Research Director
##                       102                       145                        80
##        Research Scientist           Sales Executive      Sales Representative
##                       292                       326                        83
```

```
table(causal_data$PerformanceRating)
```

```
##
##    3    4
## 1244  226
```

```
table(causal_data$MaritalStatus)
```

```
##
## Divorced  Married   Single
##      327      673      470
```

```
summary(causal_data$MonthlyIncome)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1009    2911    4919    6503    8379   19999
```

Regression Estimator + Bootstrap Standard Error

In this part, we would model the conditional expectation of Y (income) as a function of treatment X and covariates Ci.

After which, we would estimate:

$$\widehat{\Delta}_R = \frac{1}{n} \sum_{i=1}^{n} [\widehat{m}_1(c_i) - \widehat{m}_0(c_i)]$$

Where $\widehat{m}_x(c_i) = \mathbb{E}(Y \mid X = x, C = c_i)$, estimated using a **regression model (RM)**.

```r
# Fit outcome model
outcome_model <- lm(MonthlyIncome ~ OverTime + JobLevel +
                      JobRole + PerformanceRating +
                      MaritalStatus, data = causal_data)

# Predict outcomes
causal_data$m1 <- predict(outcome_model,
                          newdata = causal_data %>%
                            mutate(OverTime = 1))

causal_data$m0 <- predict(outcome_model,
                          newdata = causal_data %>%
                            mutate(OverTime = 0))

# Find the estimate of RM
delta_reg <- mean(causal_data$m1 - causal_data$m0)
cat("Regression estimate is:", round(delta_reg, 2), "\n")
```

```
## Regression estimate is: 81.19
```

```r
# Standard error
boot_fun <- function(data, indices) {
  d <- data[indices, ]
  mod <- lm(MonthlyIncome ~ OverTime + JobLevel +
              JobRole + PerformanceRating +
              MaritalStatus, data = d)
  m1 <- predict(mod, newdata = d %>%
                  mutate(OverTime = 1))
  m0 <- predict(mod, newdata = d %>%
                  mutate(OverTime = 0))
  return(mean(m1 - m0))
}

set.seed(1995)
boot_reg <- boot(data = causal_data,
                 statistic = boot_fun, R = 999)

# Estimate SE
se_reg <- round(sd(boot_reg$t), 2)
cat("Bootstrap SE is:", round(se_reg, 3), "\n")
```

```
## Bootstrap SE is: 63.41
```

This means that, after adjusting for job level, job role, performance, and marital status, employees who work overtime earn on average 81.1872601 more per month than their counterparts who don't. The standard error of 63.41 measures the uncertainty due to sampling variability and model estimation in this approach. The assumption is that it is only valid if our model for $Y \mid X, C$ is correctly specified. If not, this estimate may be biased

IPW Estimator + Bootstrap SE

In this method, the estimator is given by:

$$\widehat{\Delta}_{\text{IPW}} = \frac{1}{n}\sum_{i=1}^{n} y_i \left(\frac{x_i}{\hat{\pi}(c_i)} - \frac{1 - x_i}{1 - \hat{\pi}(c_i)} \right)$$

Where:

$$\pi(C) = \Pr(X = 1 \mid C = c_i)$$

is the estimated propensity score.

```r
# Estimate Propensity Scores
ps_model <- glm(OverTime ~ JobLevel + JobRole +
                  PerformanceRating +
                  MaritalStatus,
                data = causal_data, family = binomial)

causal_data$pscore <- predict(ps_model, type = "response")


# Compute IPW estimate
x <- causal_data$OverTime
y <- causal_data$MonthlyIncome
e <- causal_data$pscore # Let that represent a propensity score here

delta_ipw <- mean(y * x / e - y * (1 - x) / (1 - e))
cat("IPW estimate is:", round(delta_ipw, 2), "\n")
```

```
## IPW estimate is: 84.51
```

```r
# Bootstrap for SE
boot_ipw <- function(data, indices) {
  d <- data[indices, ]
  psmod <- glm(OverTime ~ JobLevel + JobRole +
                 PerformanceRating + MaritalStatus,
               data = d, family = binomial)
  pscore <- predict(psmod, type = "response")
  x <- d$OverTime
  y <- d$MonthlyIncome
  mean(y * x / pscore - y * (1 - x) / (1 - pscore))
}
```

```r
set.seed(1995)
boot_out <- boot(data = causal_data, statistic = boot_ipw, R = 999)
se_ipw <- round(sd(boot_out$t), 2)
cat("Bootstrap SE is:", round(se_ipw, 3), "\n")
```

```
## Bootstrap SE is: 67.58
```

In this method, after weighting for the inverse of estimated treatment probabilities, we estimate that overtime work increases monthly income by about 84.5071576, with standard error measuring variability of about 67.58. These values are a bit above those for Regression method.

Double-Robust (DR) Estimator + Bootstrap SE

In this model, two other models are combined:

- An **outcome regression model** $\hat{m}_x(c)$, and
- A **propensity score model** $\hat{\pi}(c) = \Pr(X = 1 \mid C = c)$.

```r
# Fit the DR model
outcome_model <- lm(MonthlyIncome ~ OverTime +
                      JobLevel + JobRole +
                      PerformanceRating + MaritalStatus, data = causal_data)
causal_data$m1 <- predict(outcome_model, newdata = causal_data %>% mutate(OverTime = 1))
causal_data$m0 <- predict(outcome_model, newdata = causal_data %>% mutate(OverTime = 0))

# Propensity model part
ps_model <- glm(OverTime ~ JobLevel + JobRole + PerformanceRating +
                  MaritalStatus, data = causal_data, family = binomial)
causal_data$pscore <- predict(ps_model, type = "response")

# Double Robust estimate
x <- causal_data$OverTime
y <- causal_data$MonthlyIncome
e <- causal_data$pscore # For propensity score
m1 <- causal_data$m1
m0 <- causal_data$m0

dr_part1 <- (y*x - (x-e)*m1)/e
dr_part0 <- (y*(1-x) + (x-e)*m0)/(1-e)

# dr_part1 <- (x * (y - m1)) / e + m1
# dr_part0 <- ((1 - x) * (y - m0)) / (1 - e) + m0
delta_dr <- mean(dr_part1 - dr_part0)
cat("Double Robust estimate is:", round(delta_dr, 2), "\n")
```

```
## Double Robust estimate is: 79.65
```

```r
# Bootstrap SE
boot_dr <- function(data, indices) {
  d <- data[indices, ]

  outmod <- lm(MonthlyIncome ~ OverTime + JobLevel +
               JobRole + PerformanceRating +
               MaritalStatus, data = d)
  m1 <- predict(outmod,
              newdata = d %>% mutate(OverTime = 1))
  m0 <- predict(outmod,
              newdata = d %>% mutate(OverTime = 0))

  psmod <- glm(OverTime ~ JobLevel + JobRole + PerformanceRating +
               MaritalStatus, data = d, family = binomial)
  pscore <- predict(psmod, type = "response")

  x <- d$OverTime
  y <- d$MonthlyIncome

  part1 <- (x * (y - m1)) / pscore + m1
  part0 <- ((1 - x) * (y - m0)) / (1 - pscore) + m0
  mean(part1 - part0)
}

set.seed(1995)
boot_out <- boot(data = causal_data, statistic = boot_dr, R = 999)
se_dr <- round(sd(boot_out$t), 2)
cat("Bootstrap SE is:", round(se_dr, 3), "\n")
```

```
## Bootstrap SE is: 63.72
```

From this model, the estimated increase in monthly income due to overtime work is 79.6515388, and even if only one of the two models (outcome or propensity) is correctly specified, this estimate is still valid. The estimate of standard error is 63.72 accounting for variability in both models.

Estimator based on grouping the data according to quintiles of propensity

```r
# Create quintiles
causal_data <- causal_data %>%
  mutate(quintile = ntile(pscore, 5))  # quintile 1 to 5

# Estimate changes within each quintile
stratified_estimates <- causal_data %>%
  group_by(quintile) %>%
  summarise(
    n_group = n(),
    diff = mean(MonthlyIncome[OverTime == 1]) - mean(MonthlyIncome[OverTime == 0]),
    .groups = "drop"
  )
```

```r
# Weighted average of stratum effects
delta_strat <- weighted.mean(stratified_estimates$diff,
                             stratified_estimates$n_group)
cat("Stratified estimate is:", round(delta_strat, 2), "\n")
```

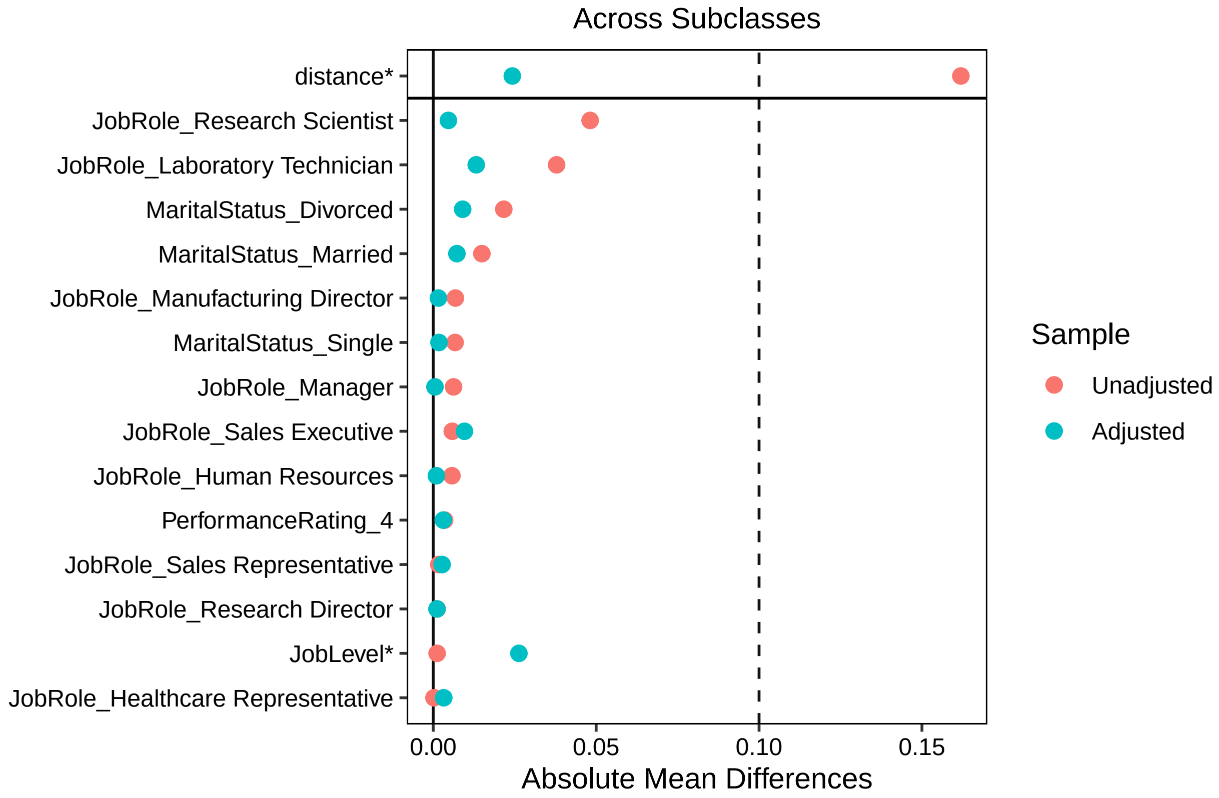
```
## Stratified estimate is: 187.27
```

```r
# Bootstrap SE
boot_strat <- function(data, indices) {
  d <- data[indices, ]
  ps_model <- glm(OverTime ~ JobLevel + JobRole + PerformanceRating + MaritalStatus,
                  data = d, family = binomial)
  d$pscore <- predict(ps_model, type = "response")
  d$quintile <- ntile(d$pscore, 5)

  strat_diffs <- d %>%
    group_by(quintile) %>%
    summarise(
      n_group = n(),
      diff = mean(MonthlyIncome[OverTime == 1]) - mean(MonthlyIncome[OverTime == 0]),
      .groups = "drop"
    )

  weighted.mean(strat_diffs$diff, strat_diffs$n_group)
}

set.seed(1995)
boot_out <- boot(data = causal_data, statistic = boot_strat, R = 999)
se_strat <- sd(boot_out$t)
cat("Bootstrap SE is:", round(se_strat, 3), "\n")
```

```
## Bootstrap SE is: 120.846
```

The values seem not in alignment with other preceeding estimators. We would further explore in regards to covariate balance before and after stratification

```r
# Create a MatchIt object just for the balance checking
m.out <- matchit(OverTime ~ JobLevel + JobRole +
                   PerformanceRating + MaritalStatus,
                 data = causal_data,
                 method = "subclass", subclass = 5)  # 5 quintiles

# Generate love plot
love.plot(m.out, stats = "mean.diffs",
          thresholds = c(m = 0.1),
          abs = TRUE, var.order = "unadjusted",
          title = "", stars = "std" )
```



With the love plot above, green circles are close to zero for most of the variables, which suggest a good match. That suggests there is no evidence of residual confounding after adjustment. Hence we can conclude that within each propensity score quintile, the treated (Overtime = Yes) and control (Overtime = No) units have similar distributions for JobLevel, JobRole, MaritalStatus, PerformanceRating.

Normalized IPW (Stabilized Weights)

In the normalized (or stabilized) IPW estimator, each weighted mean is divided by the sum of its weights and it is given by;

$$\widehat{\Delta}_{IPW2} = \frac{\sum_{i=1}^{n} y_i \left(\frac{x_i}{\hat{\pi}(c_i)}\right)}{\sum_{i=1}^{n} \left(\frac{x_i}{\hat{\pi}(c_i)}\right)} - \frac{\sum_{i=1}^{n} y_i \left(\frac{1-x_i}{1-\hat{\pi}(c_i)}\right)}{\sum_{i=1}^{n} \left(\frac{1-x_i}{1-\hat{\pi}(c_i)}\right)}$$

Compared to the standard IPW estimator which is estimated by

$$\widehat{\Delta}_{IPW} = \frac{1}{n} \sum_{i=1}^{n} y_i \left(\frac{x_i}{\hat{\pi}(c_i)} - \frac{1-x_i}{1-\hat{\pi}(c_i)} \right)$$

# We can split the parts of the formula in the code

```
# Numerators
num_treat <- sum(causal_data$MonthlyIncome * causal_data$OverTime / causal_data$pscore)
num_control <- sum(causal_data$MonthlyIncome * (1 - causal_data$OverTime) / (1 - causal_data$pscore))
```

```r
# Denominators
den_treat <- sum(causal_data$OverTime / causal_data$pscore)
den_control <- sum((1 - causal_data$OverTime) / (1 - causal_data$pscore))

# Normalized IPW estimate
delta_ipw2 <- round(num_treat / den_treat - num_control / den_control, 2)
print(delta_ipw2)
```

```
## [1] 84.45
```

```r
set.seed(1995)
B <- 999
n <- nrow(causal_data)
ipw2_estimates <- numeric(B)

for (b in 1:B) {
  idx <- sample(1:n, n, replace = TRUE)
  df_b <- causal_data[idx, ]

  ps_b <- predict(glm(OverTime ~ JobLevel + JobRole +
                        PerformanceRating + MaritalStatus,
                      data = df_b, family = binomial), type = "response")

  num_treat <- sum(df_b$MonthlyIncome * df_b$OverTime / ps_b)
  den_treat <- sum(df_b$OverTime / ps_b)

  num_control <- sum(df_b$MonthlyIncome * (1 - df_b$OverTime) / (1 - ps_b))
  den_control <- sum((1 - df_b$OverTime) / (1 - ps_b))

  ipw2_estimates[b] <- num_treat / den_treat - num_control / den_control
}

se_ipw2 <- round(sd(ipw2_estimates), 2)
```

The estimates of the two methods are very close, except in the standard error. It could be noted that while the standard IPW averages over all subjects with their weights, the normalized version (IPW2) rather estimates the weighted average within each group, which still converges to the same target quantity under correct assumptions. In other words, both estimators are just two algebraically similar ways of solving the same estimation problem.

The basic idea behind both

$$\widehat{\Delta}_{\text{IPW}}$$

and

$$\widehat{\Delta}_{\text{IPW2}}$$

is to estimate the average treatment effect (ATE)** by reweighting individuals to mimic a randomized experiment.

The standard IPW estimator applies individual-level weights to each observation and averages their contributions directly. In contrast, the IPW2 version normalizes the weights within each group (treated and control) so that the sum of the weights equals one for each group. This yields weighted group means, and then computes the difference in those means — but crucially, the target is still the same as below:

$$\Delta = \mathbb{E}\left[\mathbb{E}(Y \mid X = 1, C)\right] - \mathbb{E}\left[\mathbb{E}(Y \mid X = 0, C)\right]$$