task2

Keyu Mao

2023/6/8

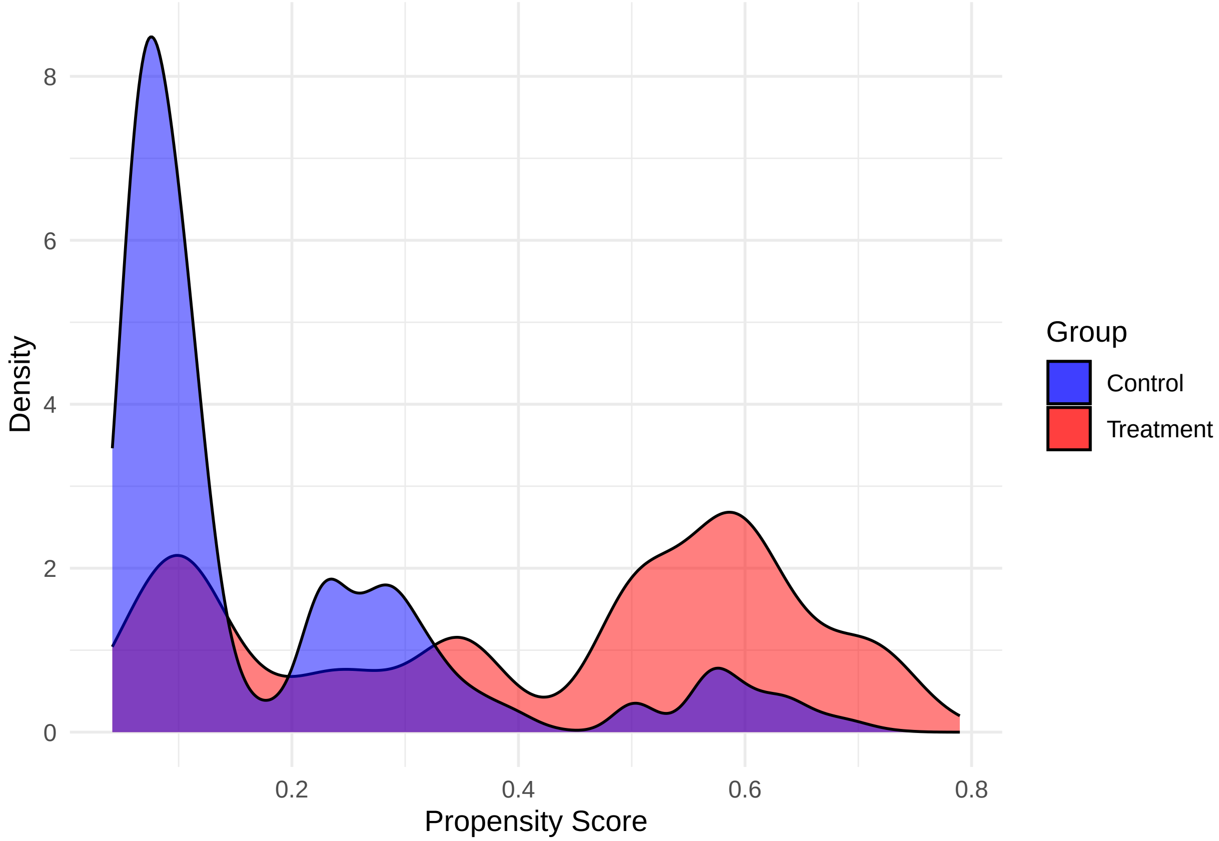
In the task, we will build a model of propensity scores using logisitic regression.

```
data <- read.csv("final_data.csv")
head(data)
```

```
##   X Country AQI.Value AQI.Category CO.AQI.Value CO.AQI.Category Ozone.AQI.Value
## 1 0     123        51            1            1               0              36
## 2 1      71        41            0            1               0               5
## 3 2     108        66            1            1               0              39
## 4 3       0        34            0            1               0              34
## 5 4      37        22            0            0               0              22
## 6 5     164        54            1            1               0              14
##   Ozone.AQI.Category NO2.AQI.Value NO2.AQI.Category PM2.5.AQI.Value
## 1                  2             0                0              51
## 2                  0             1                0              41
## 3                  2             2                1              66
## 4                  2             0                0              20
## 5                  1             0                0               6
## 6                  0            11                2              54
##   PM2.5.AQI.Category
## 1                  1
## 2                  0
## 3                  1
## 4                  0
## 5                  0
## 6                  1
```

```
library(MatchIt)
library(ggplot2)
ps_formula <- CO.AQI.Category~Ozone.AQI.Category+NO2.AQI.Category+Country
ps_model <- matchit(ps_formula, data = data, method = "nearest")
ps_scores <- ps_model$distance
data$propensity_score <- ps_scores
# Subset data for treatment and control groups
treatment_data <- data[data$CO.AQI.Category == 1, ]
control_data <- data[data$CO.AQI.Category == 0, ]

# Create density plots for propensity scores
ggplot() +
  geom_density(data = treatment_data, aes(x = propensity_score, fill = "Treatment"), alpha = 0.5) +
  geom_density(data = control_data, aes(x = propensity_score, fill = "Control"), alpha = 0.5) +
  labs(x = "Propensity Score", y = "Density", fill = "Group") +
  scale_fill_manual(values = c("blue", "red")) +
  theme_minimal()
```



2.3

In this task we will try to do matching. We can use package matchit and use propensity scores which are gained in the last section.

```
match.method = matchit(CO.AQI.Category~propensity_score, data=data, method="nearest",ratio = 1)
match_data = match.data(match.method)
```

```
summary(match.method, un = FALSE)
```

```
##
## Call:
## matchit(formula = CO.AQI.Category ~ propensity_score, data = data,
##     method = "nearest", ratio = 1)
##
## Summary of Balance for Matched Data:
##                  Means Treated Means Control Std. Mean Diff. Var. Ratio
## distance                0.4214        0.3309          0.3726     1.3615
## propensity_score        0.4123        0.3406          0.3233     1.3687
##                  eCDF Mean eCDF Max Std. Pair Dist.
## distance            0.0465   0.2575           0.373
## propensity_score    0.0465   0.2575           0.324
##
## Sample Sizes:
##           Control Treated
## All          17601    5434
## Matched       5434    5434
## Unmatched    12167       0
## Discarded        0       0
```

We may take a look at the smd of each covariate.

```
smd <- function(variate, match_data.=match_data){
  ind_treat = which(match_data.$CO.AQI.Category == 1)
  ind_control = which(match_data.$CO.AQI.Category == 0)
  treat = data[[variate]][ind_treat]
  control = data[[variate]][ind_control]
  smd = abs(mean(treat)-mean(control))/sqrt((var(treat)+var(control))/2)
  smd
}
cat(c("The smd of Ozone.AQI.Category:", smd('Ozone.AQI.Category')))
```

```
## The smd of Ozone.AQI.Category: 0.0148481209367743
```

```
cat(c("\nThe smd of NO2.AQI.Category:", smd('NO2.AQI.Category')))
```

```
##
## The smd of NO2.AQI.Category: 0.00649384120916875
```

```
cat(c("\nThe smd of Country:", smd('Country')))
```

```
##
## The smd of Country: 0.00373530624958449
```

As we can see in the above results, the smd of each covariate is fewer than 0.1. Indicating that the covariate balance has to some degree been achieved.

Then we can compute ATT

```
att <- function(match_data){
  ind_treat = which(match_data$CO.AQI.Category == 1)
  ind_control = which(match_data$CO.AQI.Category == 0)
  treat = match_data$PM2.5.AQI.Value[ind_treat]
  control = match_data$PM2.5.AQI.Value[ind_control]
  mean(treat)-mean(control)
}

cat("The att of matched data is:", att(match_data))
```

```
## The att of matched data is: 44.95381
```

```
att_v = att(match_data)
```

Then we can compute ATC similarly.

```
new_data = data
new_data$CO.AQI.Category = 1 - data$CO.AQI.Category
match.method = matchit(CO.AQI.Category~propensity_score, data=new_data, method="nearest",ratio = 1)
```

```
## Warning: Fewer control units than treated units; not all treated units will get
## a match.
```

```
match_data = match.data(match.method)

cat(c("The smd of Ozone.AQI.Category:", smd('Ozone.AQI.Category')))
```

```
## The smd of Ozone.AQI.Category: 0.00230964558086092
```

```
cat(c("\nThe smd of NO2.AQI.Category:", smd('NO2.AQI.Category')))
```

```
##
## The smd of NO2.AQI.Category: 0.0324733152585385
```

```
cat(c("\nThe smd of Country:", smd('Country')))
```

```
##
## The smd of Country: 0.0246270334303466
```

```
summary(match.method)
```

```
##
## Call:
## matchit(formula = CO.AQI.Category ~ propensity_score, data = new_data,
##     method = "nearest", ratio = 1)
##
## Summary of Balance for All Data:
##                  Means Treated Means Control Std. Mean Diff. Var. Ratio
## distance                0.8214        0.5786          1.5374     0.4226
## propensity_score        0.1814        0.4123         -1.4518     0.5150
##                  eCDF Mean eCDF Max
## distance            0.2822    0.529
## propensity_score    0.2822    0.529
##
## Summary of Balance for Matched Data:
##                  Means Treated Means Control Std. Mean Diff. Var. Ratio
## distance                0.9162        0.5786          2.1379     0.0003
## propensity_score        0.0641        0.4123         -2.1896     0.0017
##                  eCDF Mean eCDF Max Std. Pair Dist.
## distance            0.5661   0.9308          2.1379
## propensity_score    0.5661   0.9308          2.1896
##
## Sample Sizes:
##           Control Treated
## All          5434   17601
## Matched      5434    5434
## Unmatched       0   12167
## Discarded       0       0
```

We can see that all the smd of all covariates are lower than 0.1.

```
atc <- function(match_data){
  ind_treat = which(match_data$CO.AQI.Category == 0)
  ind_control = which(match_data$CO.AQI.Category == 1)
  treat = match_data$PM2.5.AQI.Value[ind_treat]
  control = match_data$PM2.5.AQI.Value[ind_control]
  mean(treat)-mean(control)
}
cat("The atc of matched data is:", atc(match_data))
```

```
## The atc of matched data is: 69.1452
```

```
atc_v = atc(match_data)
```

Then we can compute ATE:

```r
prob = mean(data$CO.AQI.Category)
ATE = prob*att_v+(1-prob)*atc_v
cat("ATE is:", ATE)
```

```
## ATE is: 63.4384
```

2.4

In this task we will using weighting to reweigh our samples.

```r
data$weights <- ifelse(data$CO.AQI.Category==1, 1/data$propensity_score, 1/(1-data$propensity_score))

E_Y_1 <- sum(data$PM2.5.AQI.Value*data$weights*data$CO.AQI.Category)/sum(data$weights*data$CO.AQI.Catego

E_Y_0 <- sum(data$PM2.5.AQI.Value*data$weights*(1-data$CO.AQI.Category))/sum(data$weights*(1-data$CO.AQI

cat("The ACE computed under weighted is:", E_Y_1-E_Y_0)
```

```
## The ACE computed under weighted is: 49.53684
```

```r
# Example propensity score model using stratification
model.stra <- matchit(ps_formula, data = data, method = "subclass")
data$weights<-NULL
matched_data <- match.data(model.stra)
strata <- unique(matched_data$subclass)
ace <- NULL

for (i in 1:length(strata)) {
  stratum <- strata[i]

  treatment_group <- matched_data[matched_data$subclass == stratum & matched_data$CO.AQI.Category == 1,
  control_group <- matched_data[matched_data$subclass == stratum & matched_data$CO.AQI.Category == 0, ]

  mean_treatment <- mean(treatment_group$PM2.5.AQI.Value)
  mean_control <- mean(control_group$PM2.5.AQI.Value)

  ace[i] <- mean_treatment - mean_control
}
weighted_ace <- weighted.mean(ace, weights =table(matched_data$subclass))
weighted_ace
```

```
## [1] 55.67434
```