# Matching and ATT Estimators

David M. Vock

PubH 7485/8485

## Overview

- Define matching as "any method that aims to equate (or"balance") the distribution of covariates in the treated and control groups"
- Any study of the effect of some intervention has two stages (1) design and (2) outcome analysis; matching belongs in the first stage
- Note that this can be done in both non-experimental and experimental studies (blocking)

# Two major settings

1. Outcome not available but want to select a subset of subjects for (expensive) followup
2. All outcome data available but the goal of matching is to reduce bias in estimation of some treatment effect

Importantly, regardless of whether the outcome data are available, they should not be used in the matching process

# Key Decision in Matching

1. Define "closeness" - the distance measure used to determine whether an individual is a good match for another.
2. Implement a matching algorithm (e.g., nearest neighbor, optimal matching, etc.)
3. Asses the quality of the matching (perhaps iterate between 1 and 2 to improve)
4. Analysis of outcome data

## Distance Metrics Commonly used

Let $D_{ij}$ be the distance between the $i^{th}$ and $j^{th}$ subjects

1. Exact: $D_{ij} = 0$ if $X_i = X_j$ and $D_{ij} = \infty$ otherwise
2. Mahalanobis: $D_{ij} = (X_i - X_j)^T \Sigma^{-1}(X_i - X_j)$ If interest is in the ATT, $\Sigma$ is the variance matrix of $X$ is the full control group, if interest is in the ATE then $\Sigma$ is the variance matrix in the pooled treatment and control group. Categorical variables should be converted to series of binary ones
3. Propensity Score: $D_{ij} = |e_i - e_j|$ where $e_k$ is the propensity score of the $k^{th}$ individual (note: some use logit of the propensity score)

Big idea of a distance metric is to take a large number of covariates and summarize the "closeness" among individuals with a single number

# Propensity Score as Balancing Score

- Why does matching on propensity score "work"?
- Recall that the propensity score is a balancing score - the distribution of covariates in the treated and control groups is the same at each level of the propensity score

# Notes on distance metrics

- Exact matching is difficult with more than a handful of categorical covariates - several subjects could be unmatched leading to bias
- When exact matching is not possible for even a small number of covariates, matching algorithms that lead to fine balance (e.g., proportion of a categorical covariate in treated and controls the same) may be used
- Distance metrics can be combined! Frequently, use exact matching for a small number of covariates and Mahalanobis or propensity score for the rest
- Exact and Mahalanobis distances tend to not perform well when the covariate dimension is high

# Matching Algorithms

- Why need an algorithm?

# k:1 nearest neighbor matching

- Subjects in the treatment group are matched to the k subjects in the control with the closest distance
- Always estimates the ATT as it matches control subjects to the treated ones and discards controls who are not selected for matches
- Need to choose k. Larger k reduces variability (using more data) but smaller k gives better matches reducing bias
- Sometimes use caliper - i.e., if no match is within $c$ of the treated subject then that subject is not matched. Hard to interpret if lots of subjects are not matched
- Can be done with replacement (better matches, but more challenging inference in the analysis phase) or without replacement

# (Sparse) Optimal Matching

- Subjects in the treatment group are matched to the subjects in the control
- Always estimates the ATT as it matches control subjects to the treated ones and discards controls who are not selected for matches
- Look at all possible pairs of matches (possibly within subgroups defined by exact matches) and pick the matches that lead to some optimal criteria (e.g., smallest average distance)
- Can be cast as a network flow optimization problem

# Subclassification Matching

- Propensity score stratification is one type but can cast this as a matching algorithm
- Groups of subjects who are similar (e.g., have similar propensity score) are "matched"
- Can be used to estimate the ATE or ATT

# Full Matching

- More refined version of subclassification matching, selects the subclasses automatically
- Full matching creates a series of matched sets, where each matched set contains at least one treated individual and at least one control individual (and each matched set may have many from either group).
- Can be used to estimate the ATE or ATT

# Assessing Matching

- Propensity score matching only guarantees balance on average (i.e., with large samples). Still may have imbalance in some covariates after matching
- Important to examine the standardized mean difference before and after matching
- Those covariates with SMD $> 0.1$ after matching - can adjust for in the analysis or categorize them and exact match on those categories
- Some advocate looking at other moments (e.g., ratio of the variance) or empirical distribution

# Analysis of Outcome

- Matching by itself does not give a causal estimate
- Matching methods are not designed to compete with modeling adjustments such as linear regression, and, in fact, the two methods have been shown to work best in combination – double robustness
- Similar to regression adjustment in randomized experiments, where the regression adjustment is used to "clean up" small residual covariate imbalance between the groups
- Debate about whether the analysis needs to account for the matched pair nature of the data – more common to simply pool all the matches into matched treated and control groups and run analyses using the groups as a whole, rather than using the individual matched pairs

# Variance Estimation

- Highly debated topic
- Key question is whether to account for uncertainty in the propensity score estimation and the matching procedure
- As in inverse probability weighting, ignoring estimation of the coefficients in the propensity score is conservative. Typically do not account for uncertainty in covariate estimation in the propensity score
- Not a lot of good methods to account for variability in the matching methods - some bootstrap and empirical formulas but not widely used
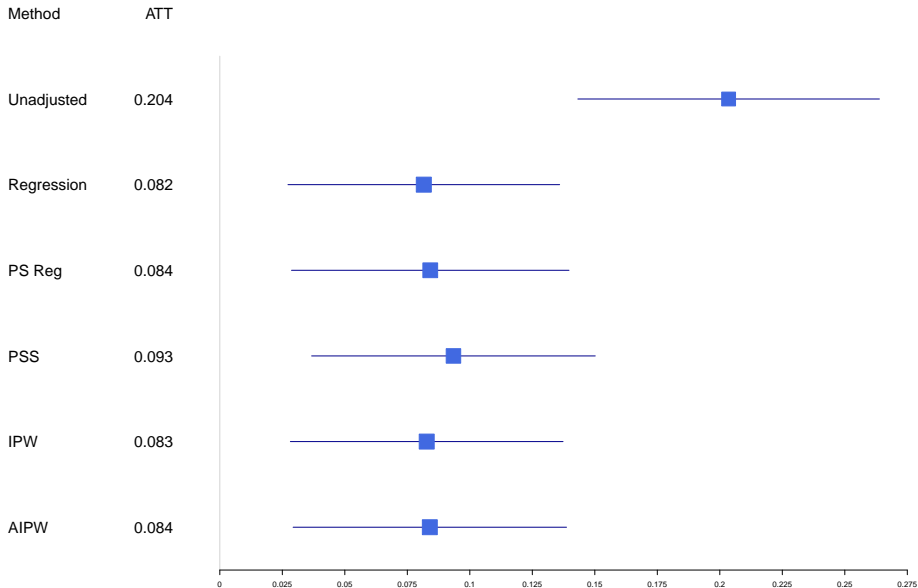
# Average Treatment Effect Among Treated

- Want to use nearest neighbor matching on the get out the vote dataset
- Always estimates the ATT as it matches control subjects to the treated ones and discards controls who are not selected for matches
- Will briefly describe how we can adapt other methods (regression adjustment, inverse weighting, etc.)

# Average Treatment Effect Among Treated

- Regression adjustment: Get predicted value for each individual in the TREATED GROUP assuming that they are (a) in the treatment group and (b) in the control group and then take the difference in the mean predicted value to get estimate of ATT.
- Propensity score stratification: Form strata using quintiles of ps in TREATED GROUP. Weight by $n_{Aj}/n_A$ where $n_A$ is number in treated group and $n_{Aj}$ is number of treated in stratum $j$.
- IPW: Weights for treated subjects are 1 and for control are $\pi(X_i)(1 - A_i)/\{1 - \pi(X_i)\}$ (must use IPW2 here)

# Putting it All Together: ATT



| Method | ATT |
|--------|-----|
| Unadjusted | 0.204 |
| Regression | 0.082 |
| PS Reg | 0.084 |
| PSS | 0.093 |
| IPW | 0.083 |
| AIPW | 0.084 |

# Propensity Score Matching

- k:1 propensity score matching where we will alter k. Use exact matching on WARD
- Nearest neighbor matching on the logit scale without replacement. Treated subjects matched in order of the propensity score (largest to smallest).
- No caliper used
- Use MatchIt package in R

# Unmatched Differences in Key Variables Between Groups

`knitr::kable(t1)`

|  | level | Not Contacted | Contacted | SMD |
|---|---|---|---|---|
| n |  | 10582 | 247 |  |
| Voted in 1998 (%) | No | 5881 (55.6) | 87 (35.2) | 0.418 |
|  | Yes | 4701 (44.4) | 160 (64.8) |  |
| Voters in household (%) | 1 Voter | 5269 (49.8) | 119 (48.2) | 0.032 |
|  | 2+ Voters | 5313 (50.2) | 128 (51.8) |  |
| Age (years) (mean (SD)) |  | 49.43 (18.73) | 58.31 (19.85) | 0.460 |
| Voted in 1996 (%) | No | 4965 (46.9) | 71 (28.7) | 0.382 |
|  | Yes | 5617 (53.1) | 176 (71.3) |  |
| New voter (%) | Previous Voter | 8452 (79.9) | 219 (88.7) | 0.243 |
|  | New Voter | 2130 (20.1) | 28 (11.3) |  |
| Party affilation (%) | Republican | 2701 (25.5) | 49 (19.8) | 0.136 |
|  | Democrat | 7881 (74.5) | 198 (80.2) |  |
| Ward of residence (%) | 2 | 317 ( 3.0) | 3 ( 1.2) | 0.565 |
|  | 3 | 273 ( 2.6) | 3 ( 1.2) |  |
|  | 4 | 234 ( 2.2) | 2 ( 0.8) |  |
|  | 5 | 200 ( 1.9) | 4 ( 1.6) |  |
|  | 6 | 435 ( 4.1) | 5 ( 2.0) |  |
|  | 7 | 337 ( 3.2) | 3 ( 1.2) |  |
|  | 8 | 360 ( 3.4) | 7 ( 2.8) |  |
|  | 9 | 387 ( 3.7) | 9 ( 3.6) |  |
|  | 10 | 452 ( 4.3) | 16 ( 6.5) |  |
|  | 11 | 451 ( 4.3) | 19 ( 7.7) |  |
|  | 12 | 364 ( 3.4) | 9 ( 3.6) |  |
|  | 13 | 435 ( 4.1) | 10 ( 4.0) |  |
|  | 14 | 383 ( 3.6) | 6 ( 2.4) |  |
|  | 15 | 329 ( 3.1) | 8 ( 3.2) |  |
|  | 16 | 240 ( 2.3) | 5 ( 2.0) |  |

# 1:1 Matching

```
library(MatchIt)
rownames(imai) <- 1:nrow(imai)
mod_match <- matchit(PHN.C1 ~ PERSONSF + VOTE96.1F + NEWF +
        MAJORPTYF + AGE, distance = "logit",
    method = "nearest", data = imai, exact = c("WARD"),
    ratio = 1)
print(tail(mod_match$match.matrix))
```

```
##        [,1]
## 2898 "3556"
## 2899 "4890"
## 2900 "6786"
## 2901 "9832"
## 2902 "9781"
## 2903 "10766"
```

# Check for Balance 1:1 Matching

```
imai_match <- data.frame(rbind(imai[imai$PHN.C1 == 1, ],
    imai[as.numeric(mod_match$match.matrix), ]))
tabmatched1.1 <- CreateTableOne(vars = vars, strata = "PHN.C1"
    data = imai_match, test = FALSE)
t2 <- print(tabmatched1.1, smd = TRUE, showAllLevels = TRUE,
                        varLabels = TRUE)
```

# 1:1 Matching: Differences in Key Variables Between Groups

`knitr::kable(t2)`

|  | level | 0 | 1 | SMD |
|---|---|---|---|---|
| n |  | 247 | 247 |  |
| VOTED98F (%) | No | 94 (38.1) | 87 (35.2) | 0.059 |
|  | Yes | 153 (61.9) | 160 (64.8) |  |
| PERSONSF (%) | 1 Voter | 127 (51.4) | 119 (48.2) | 0.065 |
|  | 2+ Voters | 120 (48.6) | 128 (51.8) |  |
| AGE (mean (SD)) |  | 57.93 (19.40) | 58.31 (19.85) | 0.019 |
| VOTE96.1F (%) | No | 65 (26.3) | 71 (28.7) | 0.054 |
|  | Yes | 182 (73.7) | 176 (71.3) |  |
| NEWF (%) | Previous Voter | 220 (89.1) | 219 (88.7) | 0.013 |
|  | New Voter | 27 (10.9) | 28 (11.3) |  |
| MAJORPTYF (%) | Republican | 63 (25.5) | 49 (19.8) | 0.136 |
|  | Democrat | 184 (74.5) | 198 (80.2) |  |
| WARD (%) | 2 | 3 ( 1.2) | 3 ( 1.2) | <0.001 |
|  | 3 | 3 ( 1.2) | 3 ( 1.2) |  |
|  | 4 | 2 ( 0.8) | 2 ( 0.8) |  |
|  | 5 | 4 ( 1.6) | 4 ( 1.6) |  |
|  | 6 | 5 ( 2.0) | 5 ( 2.0) |  |
|  | 7 | 3 ( 1.2) | 3 ( 1.2) |  |
|  | 8 | 7 ( 2.8) | 7 ( 2.8) |  |
|  | 9 | 9 ( 3.6) | 9 ( 3.6) |  |
|  | 10 | 16 ( 6.5) | 16 ( 6.5) |  |
|  | 11 | 19 ( 7.7) | 19 ( 7.7) |  |
|  | 12 | 9 ( 3.6) | 9 ( 3.6) |  |
|  | 13 | 10 ( 4.0) | 10 ( 4.0) |  |
|  | 14 | 6 ( 2.4) | 6 ( 2.4) |  |
|  | 15 | 8 ( 3.2) | 8 ( 3.2) |  |
|  | 16 | 5 ( 2.0) | 5 ( 2.0) |  |

# Graphically Assess Matching

```r
plot(mod_match, type = "hist")
```
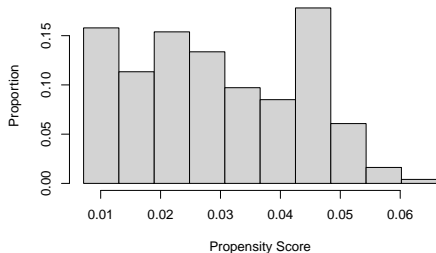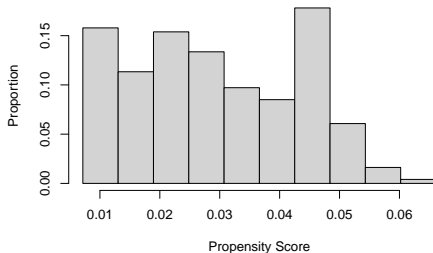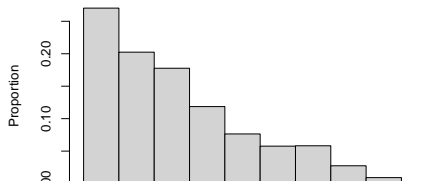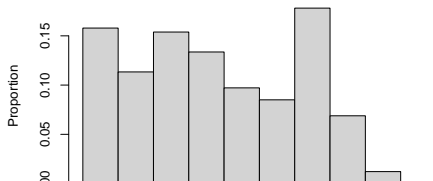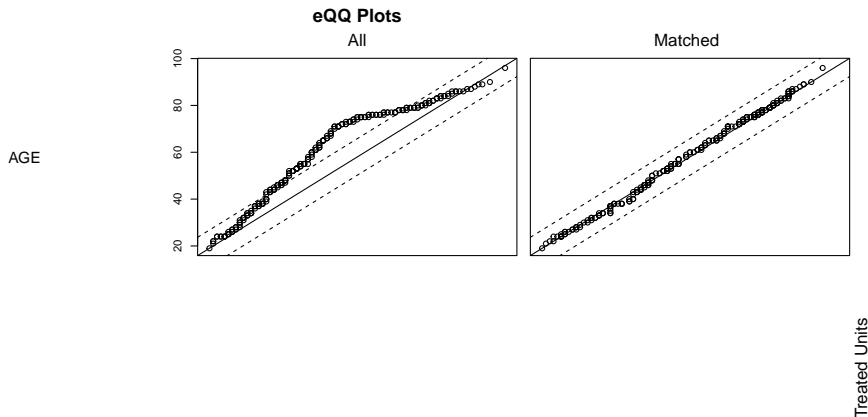
# Graphically Assess Matching

```
plot(mod_match, type = "QQ", which.xs = c("AGE"))
```

# Results 1:1 Matching

```
## [1] "1:1 Match ATT"

## [1] 0.0283

## [1] "Standard Error"

##      1
## 0.0433

## [1] "95% CI"

## [1] -0.0566  0.1133
```

# Check for Balance 4:1 Matching

```
mod_match <- matchit(PHN.C1 ~ PERSONSF + VOTE96.1F + NEWF +
        MAJORPTYF + AGE, distance = "logit",
    method = "nearest", data = imai, exact = c("WARD"),
    ratio = 4)
print(tail(mod_match$match.matrix))

imai_match <- data.frame(rbind(imai[imai$PHN.C1 == 1, ],
    imai[as.numeric(mod_match$match.matrix), ]))
tabmatched4.1 <- CreateTableOne(vars = vars, strata = "PHN.C1"
    data = imai_match, test = FALSE)
t3 <- print(tabmatched4.1, smd = TRUE, showAllLevels = TRUE,
                        varLabels = TRUE)
```

# 4:1 Matching: Differences in Key Variables Between Groups

`knitr::kable(t3)`

|  | level | 0 | 1 | SMD |
|---|---|---|---|---|
| n |  | 988 | 247 |  |
| VOTED98F (%) | No | 390 (39.5) | 87 (35.2) | 0.088 |
|  | Yes | 598 (60.5) | 160 (64.8) |  |
| PERSONSF (%) | 1 Voter | 494 (50.0) | 119 (48.2) | 0.036 |
|  | 2+ Voters | 494 (50.0) | 128 (51.8) |  |
| AGE (mean (SD)) |  | 58.24 (19.68) | 58.31 (19.85) | 0.003 |
| VOTE96.1F (%) | No | 289 (29.3) | 71 (28.7) | 0.011 |
|  | Yes | 699 (70.7) | 176 (71.3) |  |
| NEWF (%) | Previous Voter | 863 (87.3) | 219 (88.7) | 0.041 |
|  | New Voter | 125 (12.7) | 28 (11.3) |  |
| MAJORPTYF (%) | Republican | 205 (20.7) | 49 (19.8) | 0.023 |
|  | Democrat | 783 (79.3) | 198 (80.2) |  |
| WARD (%) | 2 | 12 ( 1.2) | 3 ( 1.2) | <0.001 |
|  | 3 | 12 ( 1.2) | 3 ( 1.2) |  |
|  | 4 | 8 ( 0.8) | 2 ( 0.8) |  |
|  | 5 | 16 ( 1.6) | 4 ( 1.6) |  |
|  | 6 | 20 ( 2.0) | 5 ( 2.0) |  |
|  | 7 | 12 ( 1.2) | 3 ( 1.2) |  |
|  | 8 | 28 ( 2.8) | 7 ( 2.8) |  |
|  | 9 | 36 ( 3.6) | 9 ( 3.6) |  |
|  | 10 | 64 ( 6.5) | 16 ( 6.5) |  |
|  | 11 | 76 ( 7.7) | 19 ( 7.7) |  |
|  | 12 | 36 ( 3.6) | 9 ( 3.6) |  |
|  | 13 | 40 ( 4.0) | 10 ( 4.0) |  |
|  | 14 | 24 ( 2.4) | 6 ( 2.4) |  |
|  | 15 | 32 ( 3.2) | 8 ( 3.2) |  |
|  | 16 | 20 ( 2.0) | 5 ( 2.0) |  |

# Results 4:1 Matching

```
## [1] "4:1 Match ATT"

## [1] 0.0425

## [1] "Standard Error"

##       1
## 0.0341

## [1] "95% CI"

## [1] -0.0244  0.1094
```

# Check for Balance 10:1 Matching

```
mod_match <- matchit(PHN.C1 ~ PERSONSF + VOTE96.1F + NEWF +
        MAJORPTYF + AGE, distance = "logit",
    method = "nearest", data = imai, exact = c("WARD"),
    ratio = 10)
print(tail(mod_match$match.matrix))

imai_match <- data.frame(rbind(imai[imai$PHN.C1 == 1, ],
    imai[as.numeric(mod_match$match.matrix), ]))
tabmatched10.1 <- CreateTableOne(vars = vars, strata = "PHN.C1
    data = imai_match, test = FALSE)
t4 <- print(tabmatched10.1, smd = TRUE, showAllLevels = TRUE,
                        varLabels = TRUE)
```

# 10:1 Matching: Differences in Key Variables Between Groups

|  | level | 0 | 1 | SMD |
|---|---|---|---|---|
| n |  | 2470 | 247 |  |
| VOTED98F (%) | No | 1050 (42.5) | 87 (35.2) | 0.150 |
|  | Yes | 1420 (57.5) | 160 (64.8) |  |
| PERSONSF (%) | 1 Voter | 1125 (45.5) | 119 (48.2) | 0.053 |
|  | 2+ Voters | 1345 (54.5) | 128 (51.8) |  |
| AGE (mean (SD)) |  | 58.01 (19.37) | 58.31 (19.85) | 0.015 |
| VOTE96.1F (%) | No | 753 (30.5) | 71 (28.7) | 0.038 |
|  | Yes | 1717 (69.5) | 176 (71.3) |  |
| NEWF (%) | Previous Voter | 2188 (88.6) | 219 (88.7) | 0.003 |
|  | New Voter | 282 (11.4) | 28 (11.3) |  |
| MAJORPTYF (%) | Republican | 506 (20.5) | 49 (19.8) | 0.016 |
|  | Democrat | 1964 (79.5) | 198 (80.2) |  |
| WARD (%) | 2 | 30 ( 1.2) | 3 ( 1.2) | <0.001 |
|  | 3 | 30 ( 1.2) | 3 ( 1.2) |  |
|  | 4 | 20 ( 0.8) | 2 ( 0.8) |  |
|  | 5 | 40 ( 1.6) | 4 ( 1.6) |  |
|  | 6 | 50 ( 2.0) | 5 ( 2.0) |  |
|  | 7 | 30 ( 1.2) | 3 ( 1.2) |  |
|  | 8 | 70 ( 2.8) | 7 ( 2.8) |  |
|  | 9 | 90 ( 3.6) | 9 ( 3.6) |  |
|  | 10 | 160 ( 6.5) | 16 ( 6.5) |  |
|  | 11 | 190 ( 7.7) | 19 ( 7.7) |  |
|  | 12 | 90 ( 3.6) | 9 ( 3.6) |  |
|  | 13 | 100 ( 4.0) | 10 ( 4.0) |  |
|  | 14 | 60 ( 2.4) | 6 ( 2.4) |  |
|  | 15 | 80 ( 3.2) | 8 ( 3.2) |  |
|  | 16 | 50 ( 2.0) | 5 ( 2.0) |  |

# Results 10:1 Matching

```
## [1] "10:1 Match ATT"

## [1] 0.0729

## [1] "Standard Error"

##      1
## 0.032

## [1] "95% CI"

## [1] 0.0102 0.1356
```

# Plot of SMD

```r
p1 <- glm(PHN.C1 ~ (PERSONS + VOTE96.1 + NEW + MAJORPTY +
        AGE + WARD), data = imai, family = "binomial")
ps <- predict(p1, type = "response")
imai$weight <- imai$PHN.C1 + (1-imai$PHN.C1)/(1-ps)*ps
imaiSvy <- svydesign(ids = ~ 1, data = imai, weights = ~ weight)
tabWeighted <- svyCreateTableOne(vars = vars, strata = "PHN.C1F",
  data = imaiSvy, test = FALSE)
dataPlot <- data.frame(variable  = rownames(ExtractSmd(tabUnmatched)),
    Unweighted = as.numeric(ExtractSmd(tabUnmatched)),
    Weighted_logit  = as.numeric(ExtractSmd(tabWeighted)),
    Matched4_1 = as.numeric(ExtractSmd(tabmatched1.1)),
    Matched4_1 = as.numeric(ExtractSmd(tabmatched4.1)),
    Matched10_1 = as.numeric(ExtractSmd(tabmatched10.1)))
dataPlot <- dplyr::filter(dataPlot,
                          variable != "VOTED98F")

## Create long-format data for ggplot2
dataPlotMelt <- melt(data          = dataPlot,
    id.vars       = c("variable"),
    variable.name = "Method",
    value.name    = "SMD")

## Order variable names by magnitude of SMD
varNames <- as.character(dataPlot$variable)[order(dataPlot$Unweighted)]

## Order factor levels in the same order
dataPlotMelt$variable <- factor(dataPlotMelt$variable,
    levels = varNames)

## Plot using ggplot2
ggplot(data = dataPlotMelt,
    mapping = aes(x = variable, y = SMD, group = Method, color = Method)) +
    geom_line() +
    geom_point() +
```
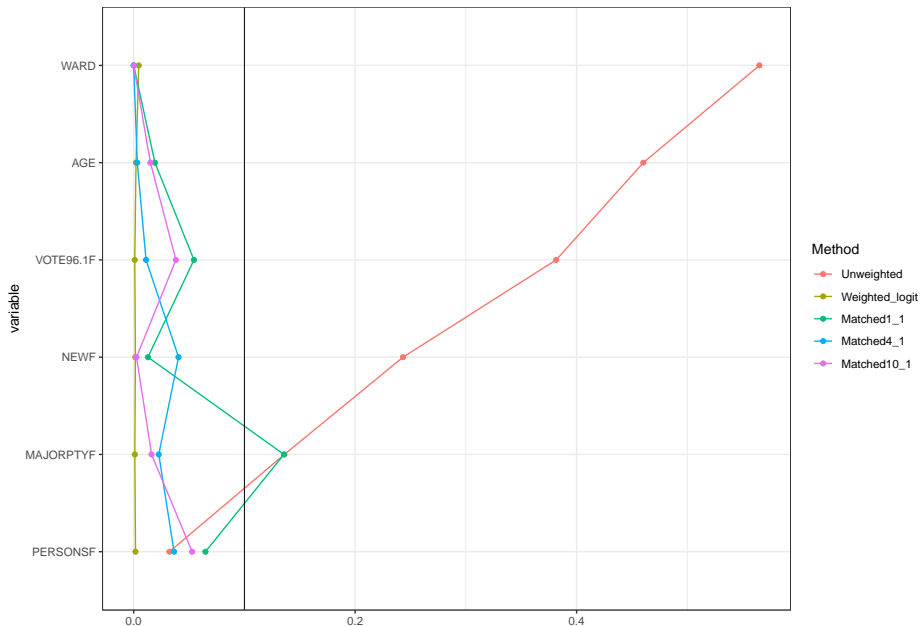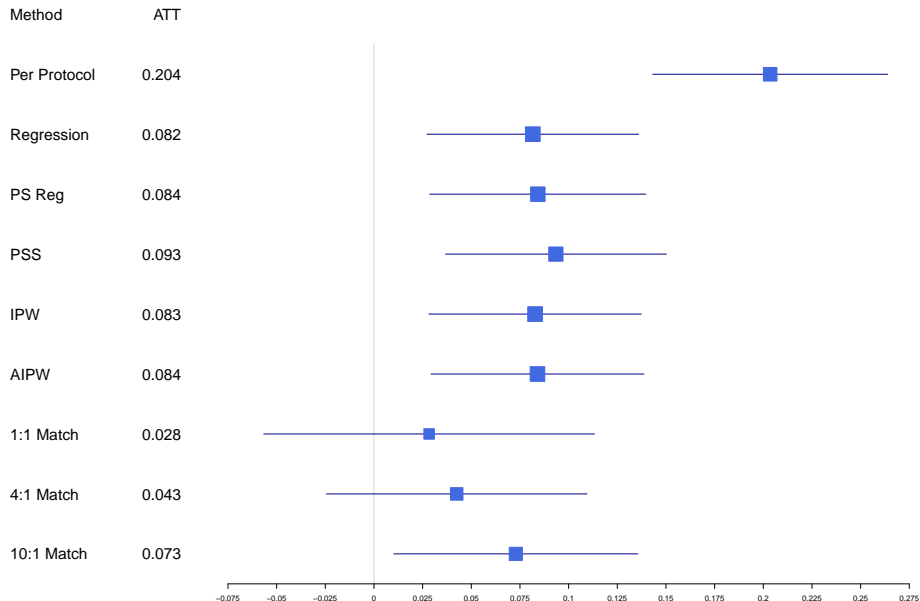
# Plot of SMD

# Putting it All Together: ATT



| Method | ATT |
|--------|-----|
| Per Protocol | 0.204 |
| Regression | 0.082 |
| PS Reg | 0.084 |
| PSS | 0.093 |
| IPW | 0.083 |
| AIPW | 0.084 |
| 1:1 Match | 0.028 |
| 4:1 Match | 0.043 |
| 10:1 Match | 0.073 |

# PS Matching Results: Key Assumptions

Identifying

1. Consistency
2. No Unmeasured confounding

Modeling

1. Propensity score model (given all confounders) correctly specified.