Propensity Scores

Theory & Practice

Mabel Carabali

EBOH, McGill University

2024/10/01 (updated: 2024-11-07)

Expected competencies

- Knows the definition and mechanisms of confounding.
- Knows and understand the principles of logistic regression and prediction.
- Know and understand considerations for causal inference.

Objectives

- Provide an overview of what are and how to estimate propensity scores (PS).
- Illustrate the use and application of PS on epidemiological inference.

Map of propensity score lecture

1. Theoretical background [5:18]
2. Propensity score methods (parametric and semi-parametric)
   - 2.1 Overview [19:35]
   - 2.2 Stratification [37:41]
   - 2.3 **Matching [43:67]**
   - 2.4 **Weighting [69:84]**
3. Propensity scores (miscellaneous topics) [>85]

Part 1 - Theoretical Background

Experiments are great, but....

Assuming successful randomization to treatment and control, you **know** it's the treatment that's causing the effect.

Can't do everything

- Ethics
- External validity
- Often non-representative
- Some treatments are hard or impossible to assign randomly
  - Motherhood
  - Divorce
  - Boycotts
  - Smoking

What do we want to know?

We really care about the difference between Y^0 and Y^1.

Let $\delta_i = y_i^1 - y_i^0$ (observed values)

$$E[\delta] = E[Y^1 - Y^0]$$

$$E[\delta] = E[Y^1] - E[Y^0]$$

This is the definition of a **Treatment Effect (TE)**.

Recall...

In a properly executed experiment, no association between potential outcome variables and treatment assignment

$$E[Y^0|T=0] \simeq E[Y^0]$$

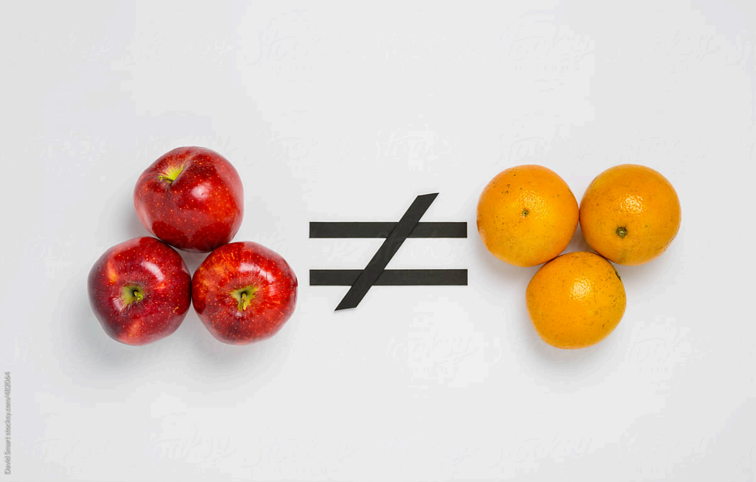
$$E[Y^1|T=1] \simeq E[Y^1]$$

So...

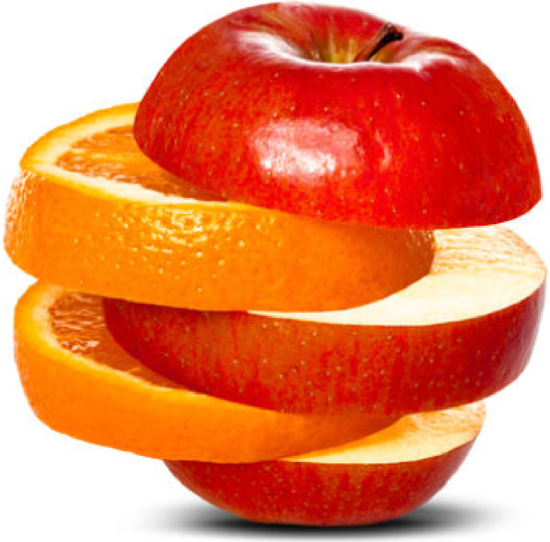
$$E[\delta] = E[Y^1] - E[Y^0] = E[Y|T=1] - E[Y|T=0]$$

Treatment effect = Δ (Delta = Change) between **observed** treatment and control averages

But more often than not, we end up comparing...



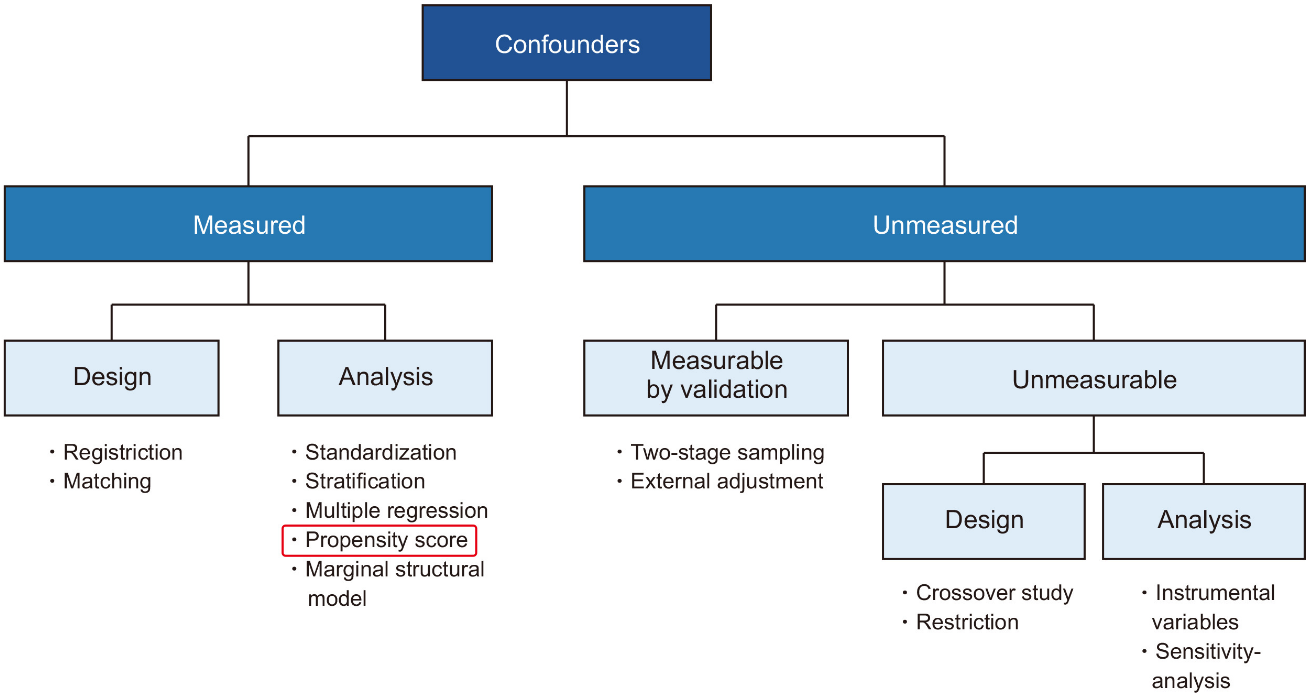
Or forcing matches...



To control of confounding we have:

Observational study

- Registry studies
- Cohort studies
- Case-control studies
- Cross-sectional studies
- Case-series



Confounders

Measured

Design

- Registriction
- Matching

Analysis

- Standardization
- Stratification
- Multiple regression
- Propensity score
- Marginal structural model

Unmeasured

Measurable by validation

- Two-stage sampling
- External adjustment

Unmeasurable

Design

- Crossover study
- Restriction

Analysis

- Instrumental variables
- Sensitivity-analysis

The point of propensity scores?

1. A **propensity score** is a subject's probability of receiving the treatment/intervention.

2. A **propensity score** is intended to make data points (observations) comparable.

3. A **propensity score** is a solution to the problem of *sparseness*. If we can't find exact matches for each case because of high dimensional (or fine-grained) data, can we reduce the complexity of the data and find "good enough" matches?

The point of propensity scores

- The propensity score (PS) is defined as the **conditional probability** of receiving a treatment given pre-treatment covariates {Z}:

  PS(Z) = Pr(X = 1 | Z)

- Exchangeability. The propensity score PS(X) balances the distribution of all X between the treatment groups:

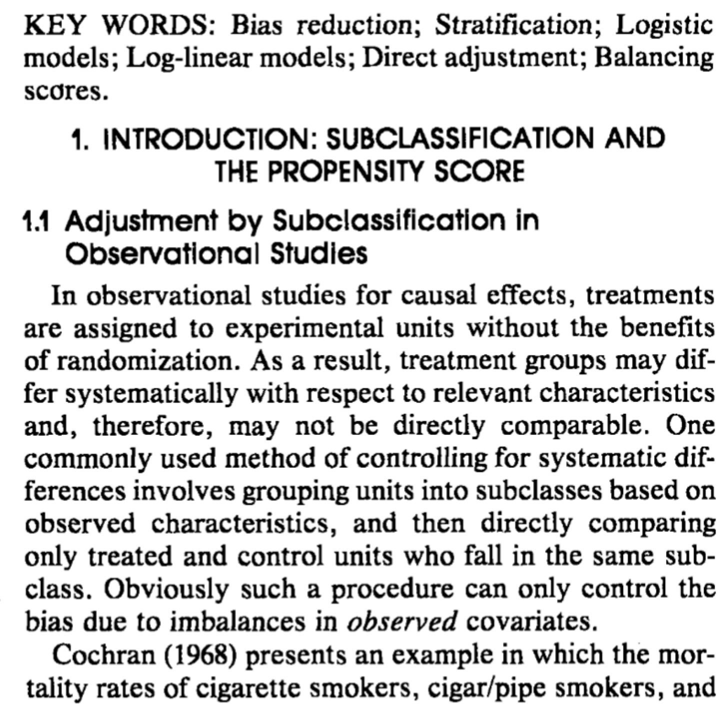
  X ⊥ Z | PS(Z)

- *in the DAG $X \leftarrow Z \rightarrow Y$, the arrow between Z and X is broken*

- The propensity score balances the observed covariates, but does not generally balance unobserved covariates

1. Bias-variance trade-off between modeling $PS(X)$ and directly modeling the outcome $Pr(Y(t)|X)$

Propensity Scores (PS) are not a "new" method



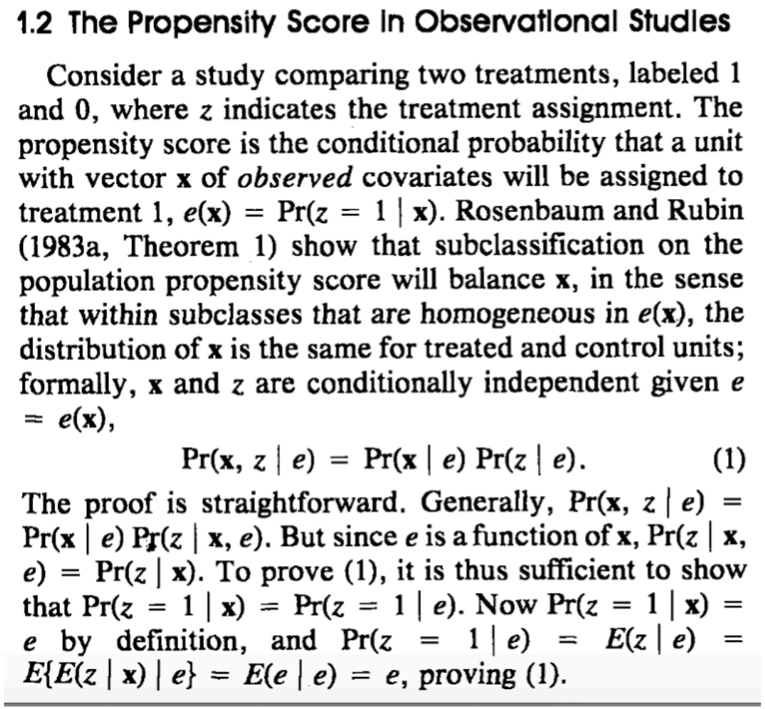
KEY WORDS: Bias reduction; Stratification; Logistic models; Log-linear models; Direct adjustment; Balancing scores.

1. INTRODUCTION: SUBCLASSIFICATION AND THE PROPENSITY SCORE

1.1 Adjustment by Subclassification in Observational Studies

In observational studies for causal effects, treatments are assigned to experimental units without the benefits of randomization. As a result, treatment groups may differ systematically with respect to relevant characteristics and, therefore, may not be directly comparable. One commonly used method of controlling for systematic differences involves grouping units into subclasses based on observed characteristics, and then directly comparing only treated and control units who fall in the same subclass. Obviously such a procedure can only control the bias due to imbalances in *observed* covariates.

Cochran (1968) presents an example in which the mortality rates of cigarette smokers, cigar/pipe smokers, and



1.2 The Propensity Score in Observational Studies

Consider a study comparing two treatments, labeled 1 and 0, where z indicates the treatment assignment. The propensity score is the conditional probability that a unit with vector $\mathbf{x}$ of *observed* covariates will be assigned to treatment 1, $e(\mathbf{x}) = \Pr(z = 1 \mid \mathbf{x})$. Rosenbaum and Rubin (1983a, Theorem 1) show that subclassification on the population propensity score will balance $\mathbf{x}$, in the sense that within subclasses that are homogeneous in $e(\mathbf{x})$, the distribution of $\mathbf{x}$ is the same for treated and control units; formally, $\mathbf{x}$ and z are conditionally independent given $e = e(\mathbf{x})$,

$$\Pr(\mathbf{x}, z \mid e) = \Pr(\mathbf{x} \mid e) \Pr(z \mid e). \qquad (1)$$

The proof is straightforward. Generally, $\Pr(\mathbf{x}, z \mid e) = \Pr(\mathbf{x} \mid e) \Pr(z \mid \mathbf{x}, e)$. But since e is a function of $\mathbf{x}$, $\Pr(z \mid \mathbf{x}, e) = \Pr(z \mid \mathbf{x})$. To prove (1), it is thus sufficient to show that $\Pr(z = 1 \mid \mathbf{x}) = \Pr(z = 1 \mid e)$. Now $\Pr(z = 1 \mid \mathbf{x}) = e$ by definition, and $\Pr(z = 1 \mid e) = E(z \mid e) = E\{E(z \mid \mathbf{x}) \mid e\} = E(e \mid e) = e$, proving (1).

Rosenbaum, P. R., & Rubin, D. B. (1984). Reducing Bias in Observational Studies Using Subclassification on the Propensity Score. Journal of the American Statistical Association, 79(387), 516–524. https://doi.org/10.1080/01621459.1984.10478078

Two common Propensity Scores questions

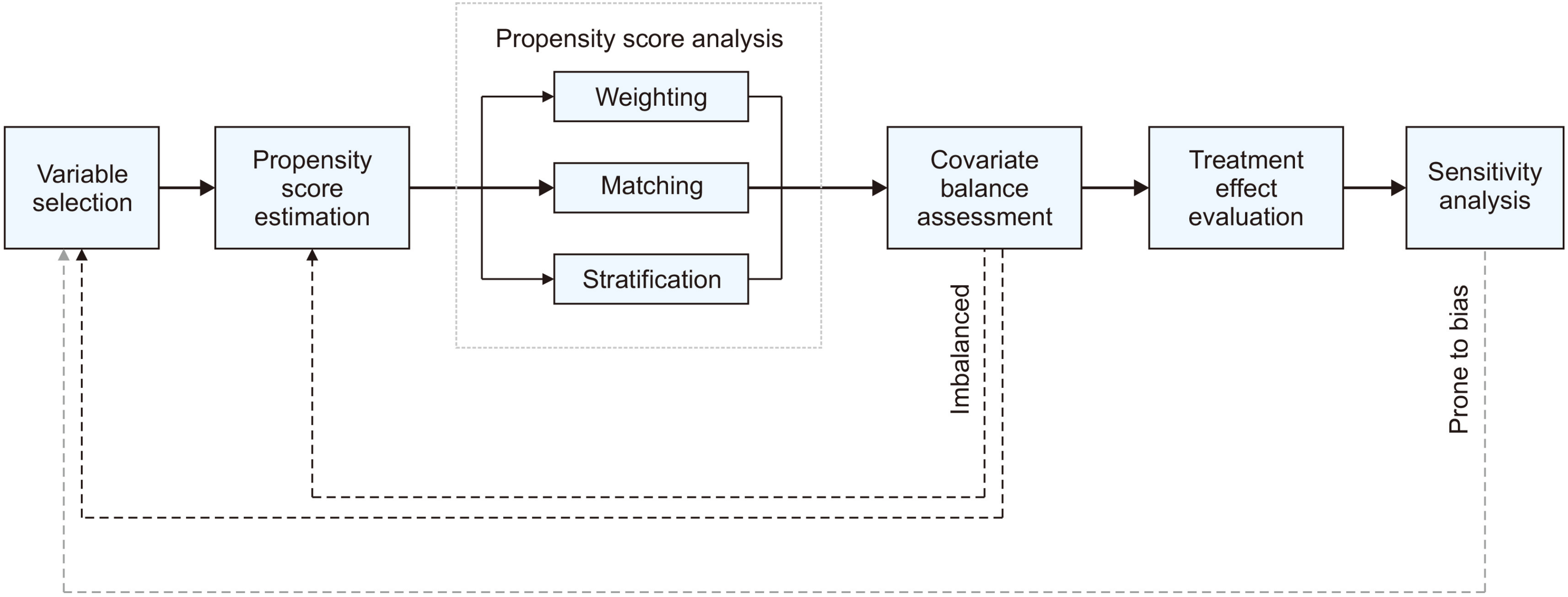
How do I estimate propensity scores?

- **Logistic regression**
- Probit regression
- Covariate-balancing-type of propensity scores
- Machine learning algorithms
- any other classifier...

How to use them for causal inference?

- Stratification ("interval matching")
- Matching
- Weighting
- Regression
- any combination of the above

PS (standard) workflow*


Propensity score analysis
Variable selection
Propensity score estimation
Weighting
Matching
Stratification
Covariate balance assessment
Treatment effect evaluation
Sensitivity analysis
Imbalanced
Prone to bias

*Step 1 and 2 can be iterative, and before the empirical application one should plan ahead.

Estimating the propensity score

Most often estimate the PS using logistic regression:

$$E(X|Z) = \beta_0 + \beta_1 Z_1 + \beta_2 Z_2 + \ldots + \beta_n Z_n$$

Here, X (a.k.a. the treatment, exposure or intervention, becomes the "outcome of interest" for the PS)

To properly estimate the PS, one need to ask beforehand, **which covariates should be included in Z and what is the target estimand**.

Variable selection

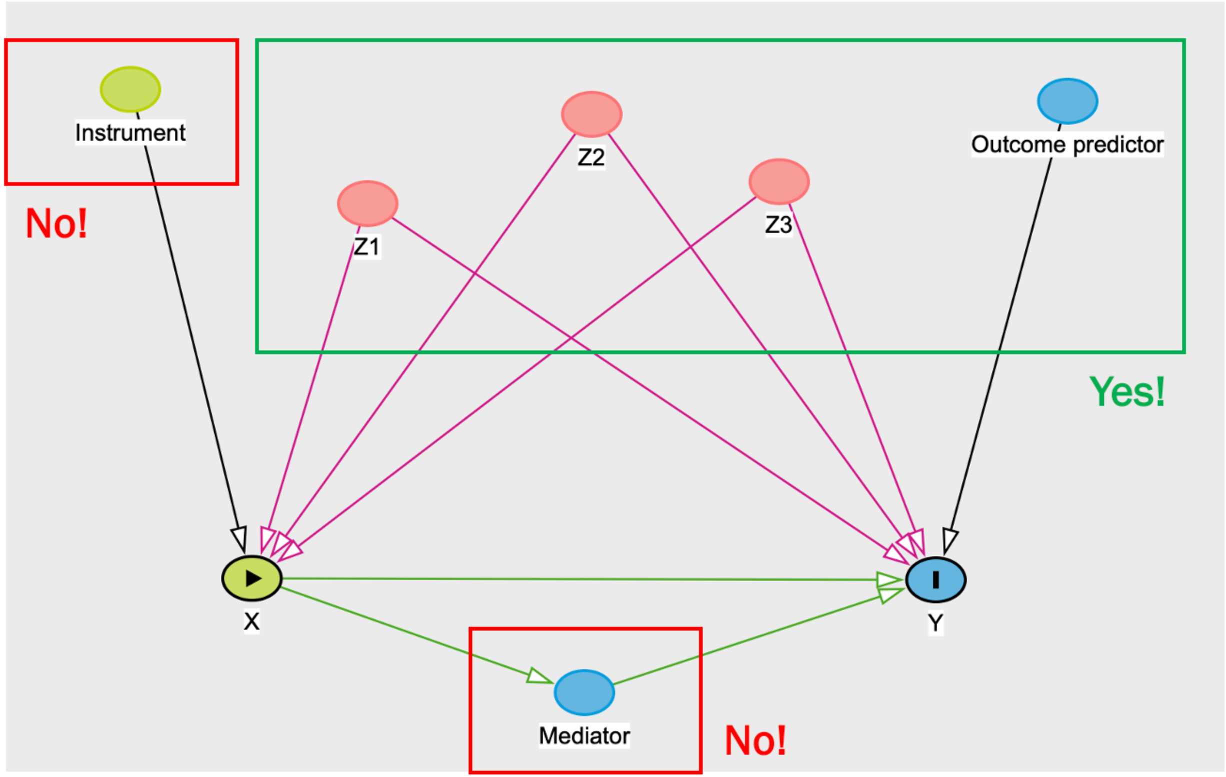
We should **include**:

- Variables associated with **both the exposure and outcome** (confounders)
  - Because these exposure-variable relationships are the ones we're trying to break
- Variables associated with the **outcome only**
  - Improves precision

We should **not** include:

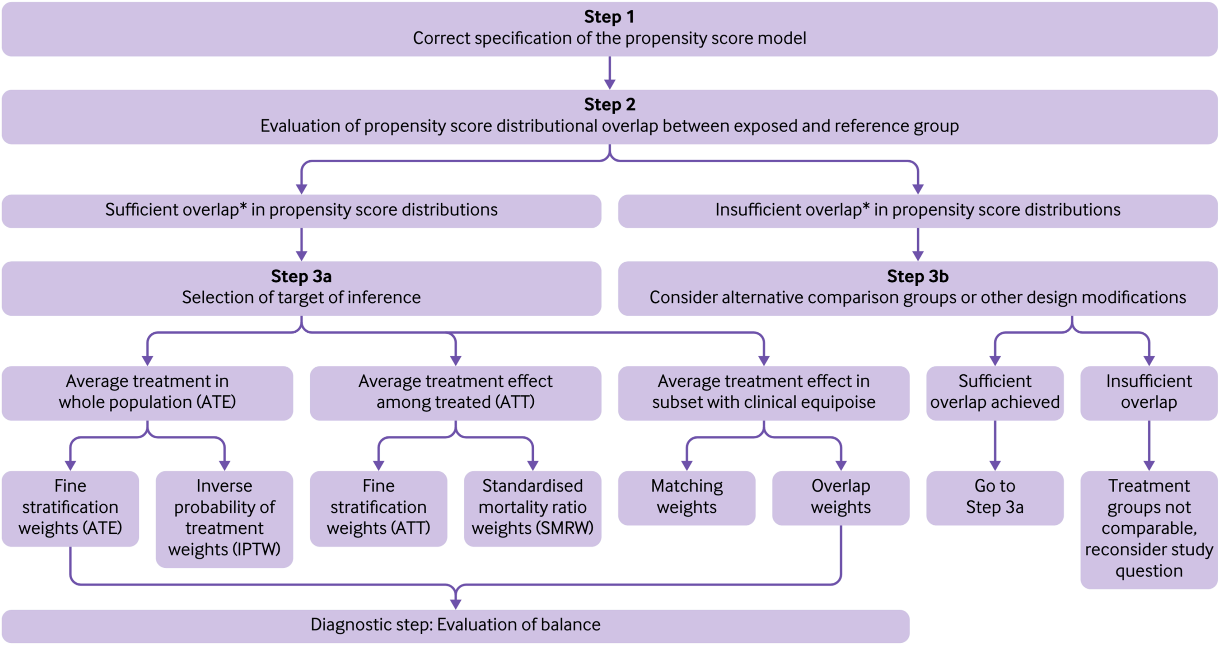
- Variables on the **causal path**
- Variables that are **predicted by both the exposure and outcome** (colliders)
  - Opens backdoor pathways
- Variables associated with the exposure that **do not form a causal path** (instrumental variables)

Variable selection



Which estimand?

After balance assessment, choosing the estimand involves choosing optimal population and matching approach. See: Choosing the Causal Estimand for Propensity Score Analysis of Observational Studies



Step 1
Correct specification of the propensity score model

Step 2
Evaluation of propensity score distributional overlap between exposed and reference group

Sufficient overlap* in propensity score distributions

Insufficient overlap* in propensity score distributions

Step 3a
Selection of target of inference

Step 3b
Consider alternative comparison groups or other design modifications

Average treatment in whole population (ATE)

Average treatment effect among treated (ATT)

Average treatment effect in subset with clinical equipoise

Sufficient overlap achieved

Insufficient overlap

Fine stratification weights (ATE)

Inverse probability of treatment weights (IPTW)

Fine stratification weights (ATT)

Standardised mortality ratio weights (SMRW)

Matching weights

Overlap weights

Go to Step 3a

Treatment groups not comparable, reconsider study question

Diagnostic step: Evaluation of balance

- Weighting is generally preferred over matching & stratification

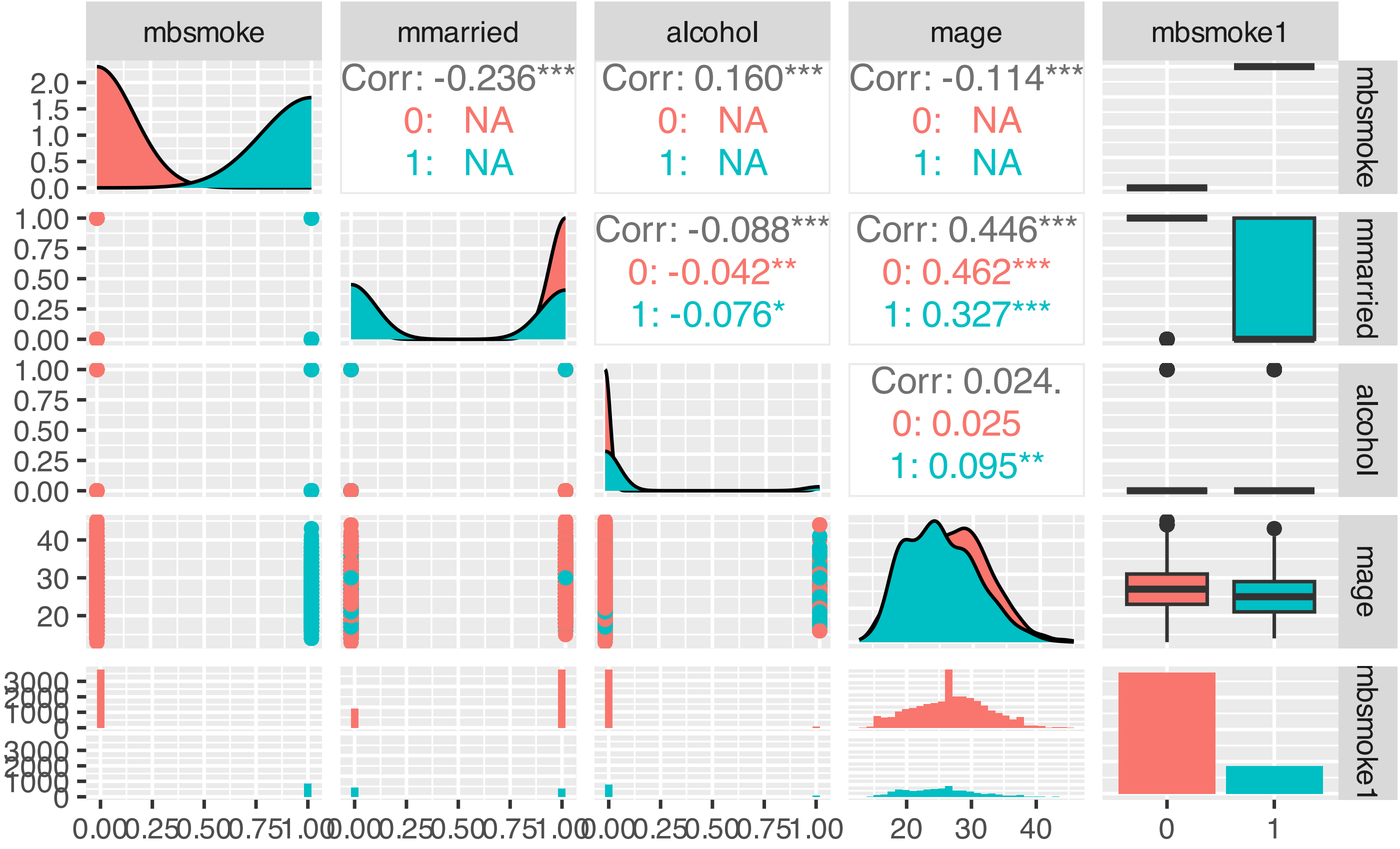
BMJ DOI:10.1136 bmj.l5657

Part 2 - Propensity score method (Overview)

Example

What is the effect of maternal smoking on infant health?

Data are from a subsample (*N* = 4642) of singleton births between 1989-1991. See Almond et al. 2005. "The Costs of Low Birth Weight" **Bring in the data**



Step 1 - Fit PS by logistic regression

- We regress the Treatment/Exposure/Intervention as a function of other covariates.

```
#d<- as.data.frame(read.csv("PSdata.csv"))
psmod <- glm( mbsmoke ~ mmarried + alcohol + mrace + fbaby +
            mage + I(mage^2) + medu + nprenatal , data = d ,family = binomial )
round(summ(psmod, confint = T)$"coeftable", 2)
```

```
##              Est.  2.5% 97.5% z val.    p
## (Intercept) -3.04 -4.66 -1.43  -3.70 0.00
## mmarried    -1.24 -1.43 -1.04 -12.36 0.00
## alcohol      1.57  1.20  1.93   8.47 0.00
## mrace        0.67  0.43  0.90   5.63 0.00
## fbaby       -0.41 -0.58 -0.23  -4.47 0.00
## mage         0.31  0.19  0.44   4.83 0.00
## I(mage^2)   -0.01 -0.01  0.00  -4.95 0.00
## medu        -0.14 -0.18 -0.11  -8.02 0.00
## nprenatal   -0.03 -0.05 -0.01  -2.70 0.01
```

This model SHOULD NOT be interpreted! but it can be made as complex as desirable

Step 2 - Estimate PS

- We use the `predict` function to estimate the predicted probability of X (i.e., Treatment, Exposure, or Intervention)

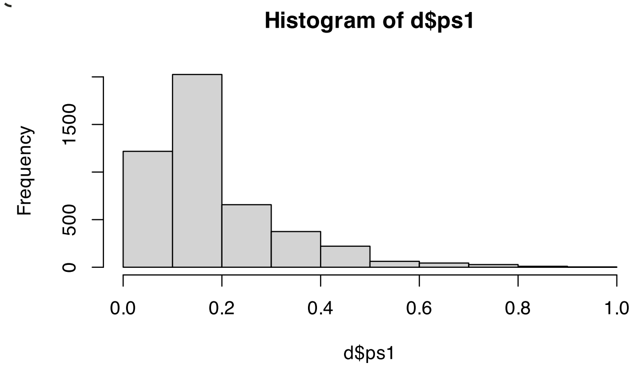
```
# add PS to DF
d <- d %>%
  mutate(ps1 = predict(psmod, type = "response"))
```

- Usually one would like to know/explore that distribution

```
round(summary(d$ps1), 2)
```

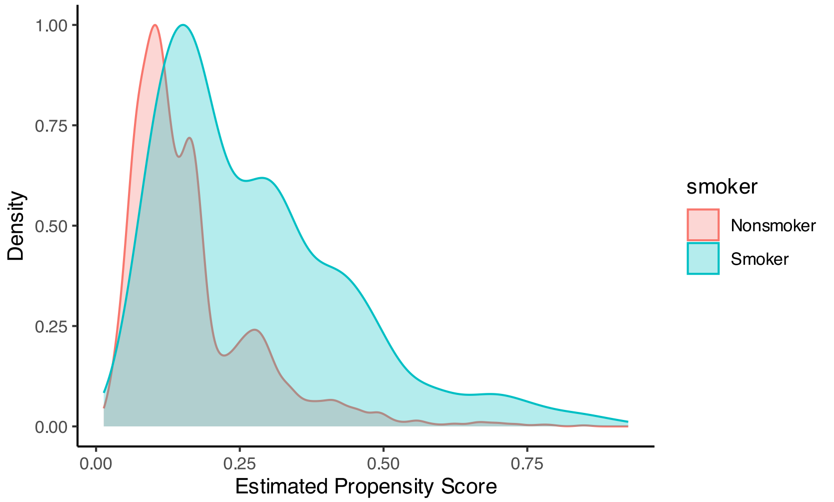
```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.01    0.10    0.15    0.19    0.24    0.93
```

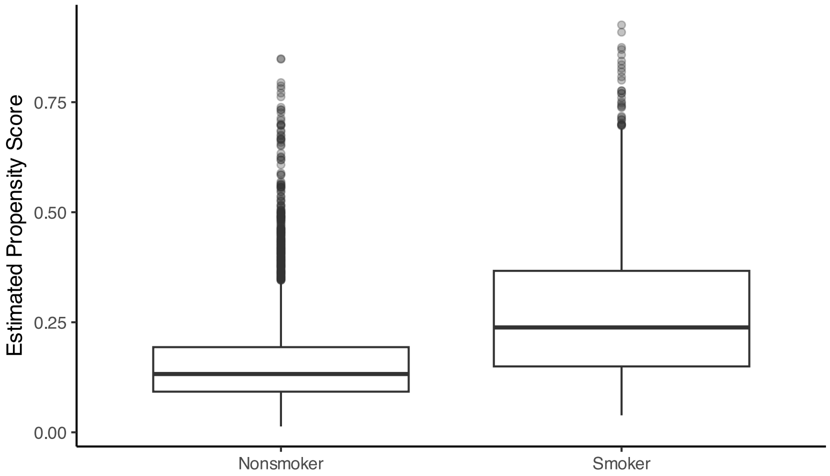
```
hist(d$ps1, breaks=10)
```



Step 2 - Examine Distribution of PS estimates

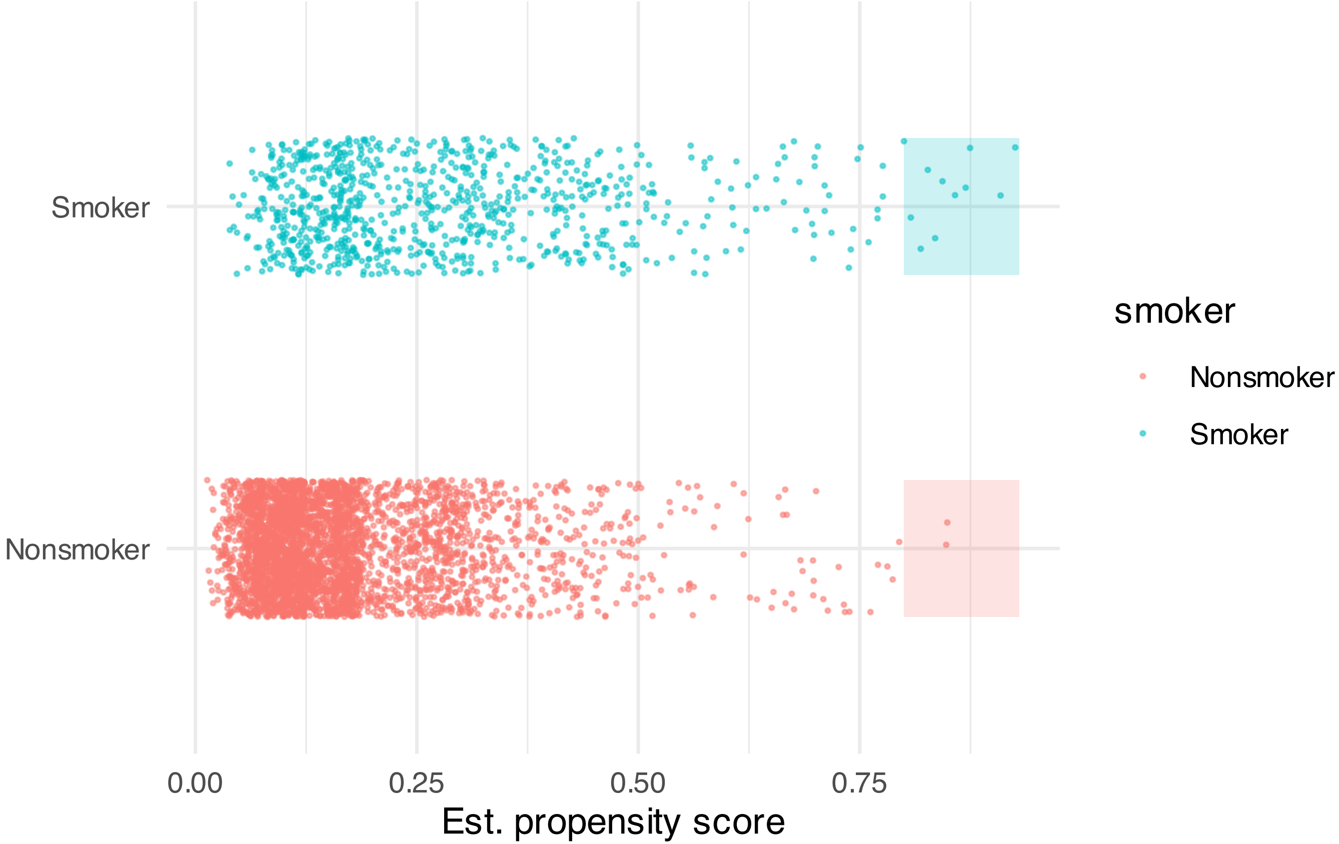
- Explore, visually the distribution of PS across levels of the Treatment/Exposure/Intervention





Step 2 - Examine Distribution of PS estimates

- Explore, visually the distribution of PS across levels of the Treatment/Exposure/Intervention



Model specification

- All the same issues that apply to logistic regression in ordinary circumstances apply here (e.g., goodness of fit, functional form).
- If the model is wrong, the propensity scores may also be wrong!
- The main assessment procedure in the traditional workflow is checking for **covariate balance**.

Step 3 - Covariate balance

- Propensity scores can be used to balance the treatment and control groups overall in three ways:
  - **Stratification** (create 5-10 subclassifications with similar p-scores, a.k.a., "subclassification" or "interval matching")
  - **Matching** (e.g., matching typically 1:1 treated cases to controls with same or very similar PS)
  - **Weighting** (e.g., applying inverse probability of treatment weights to the control cases to make their distribution look like the treatment group)
- Balance on the one-dimensional propensity score, however, does *not* guarantee that the treatment and control groups will be balanced on each of the individual component variables that were used to estimate the PS.

Example: similar scores, different profiles

```
##      smoker mmarried mage medu fbaby alcohol mrace nprenatal        ps1
## 1 Nonsmoker        1   36   16     0       0     0         8 0.03856625
## 2    Smoker        1   36   17     1       0     1        12 0.03858220
## 3 Nonsmoker        1   28   16     1       0     0        12 0.03882426
## 4    Smoker        1   31   15     1       0     0        13 0.03884993
## 5 Nonsmoker        1   34   17     1       0     1        16 0.04183232
## 6    Smoker        1   43   12     0       0     1        10 0.04190382
## 7 Nonsmoker        1   38   13     0       0     0        10 0.04298770
## 8    Smoker        1   32   14     1       0     0        12 0.04320067
```

This is OK as long as the differences between the treatment and control groups aren't systematic for any variable.

Will need formal assessments of balance.

Matching and covariate balance

- Propensity scores are a means to an end: making *all* the variables summarized in the score *S*, independent of *T*.

- Balance on the *propensity score* does not guarantee balance on all the individual covariates.

- Consider the following equation: $log(\frac{p}{1-p}) = -1 + .3x_1 + .3x_2$.

- If this is the propensity-score equation, x_1 and x_2 have the same effect on the propensity.

- Thus *any* combination of x_1 and x_2 with the same *sum* will produce the same propensity score, even if the cases are quite different.

Common support and overlap assumption

```
d %>%
  group_by(smoker) %>%
  summarize_at(vars(ps1), list(min=min,max=max))
```

```
## # A tibble: 2 × 3
##   smoker        min   max
##   <chr>       <dbl> <dbl>
## 1 Nonsmoker 0.0134 0.849
## 2 Smoker    0.0386 0.926
```

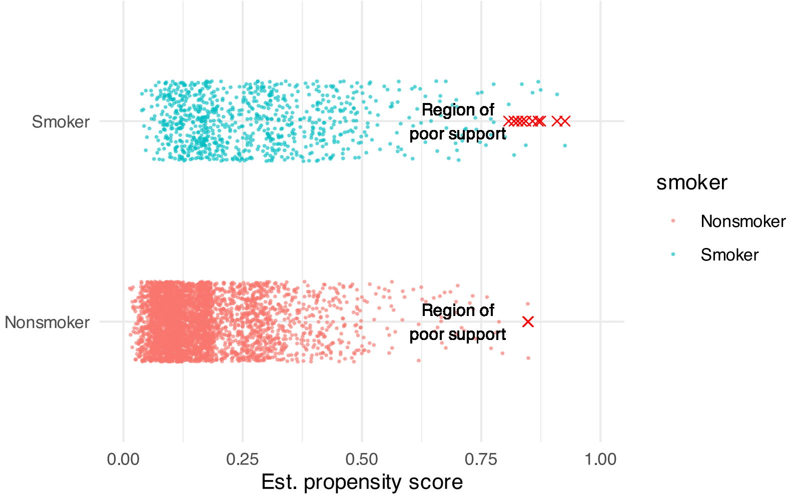
There are some smokers with higher propensities than any nonsmoker and some nonsmokers with lower propensities than any smoker

Need to decide if these violate the overlap (positivity) assumption or whether these non-overlapping regions are strictly due to sampling error

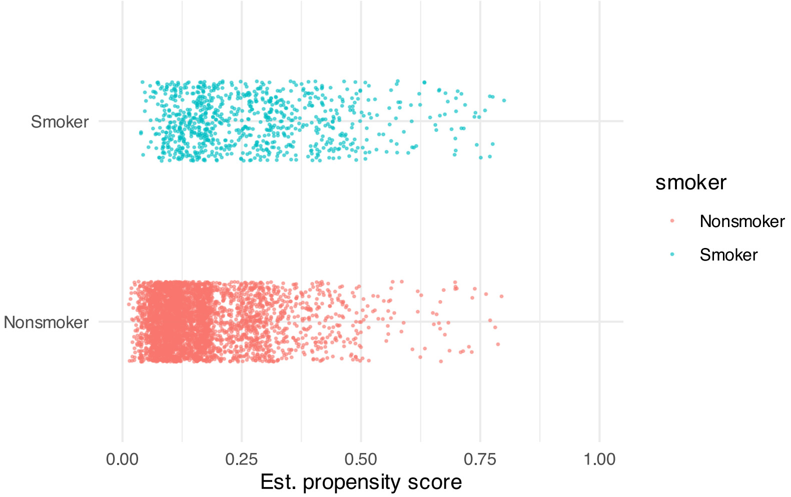
Graphs on slide 23 and 24 can be helpful.

Visualizing common support

Observing the entire Distribution (0, 1)



Restricting to region of common support (0, 0.8)



Assessing covariate balance

We use **standardized differences** to assess covariate balance or **SDM**

Presented as well as:

- For continuous variables: $d = |\frac{E(Z|X=1) - E(Z|X=0)}{\sqrt{\frac{\sigma^2_{X=1} + \sigma^2_{X=0}}{2}}}|$

- For dichotomous variables: $d = |\frac{P(Z|X=1) - P(Z|X=0)}{\sqrt{\frac{P(Z|X=1)[1-P(Z|X=1)] + P(Z|X=0)[1-P(Z|X=0)]}{2}}}|$

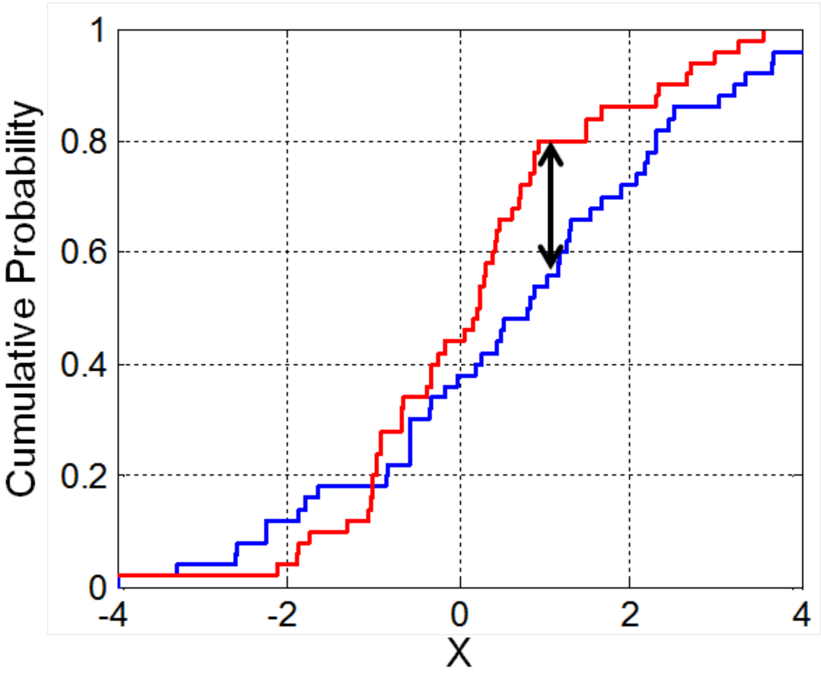
Variables with standardized differences of **less than 10%** are usually considered to be balanced

Assessing covariate balance

Covariate distributions should also be balanced beyond measures of central tendency

- Minimums, maximums, Q1, Q3, etc. should be similar

The **Kolmogorov-Smirnov distance** is the proportion of non-overlap between two distributions (or maximum distance between two cumulative distributions)



- Identical distributions have a K-S distance of **0**
- Non-overlapping distributions have a K-S distance of **1**
- A value of **less than 5%** indicates balance

Assessing covariate balance

Recall:

The **ratio of variance** can also be used to assess balance

- A ratio close to 1 indicates balance
- Can be calculated with `MatchBalance` function from the `Matching` package

Hypothesis testing to assess balance should be **avoided**

- Balance is a **property of the sample** and should not be inferred to a larger population

What if covariates aren't balanced?

Time to **re-specify your propensity score model** by:

- Adding **covariates**
- Adding **interaction terms**
- Adding **higher order terms** or **smoothers** (ex. splines)
- Using **machine learning algorithms**

A more complex PS model

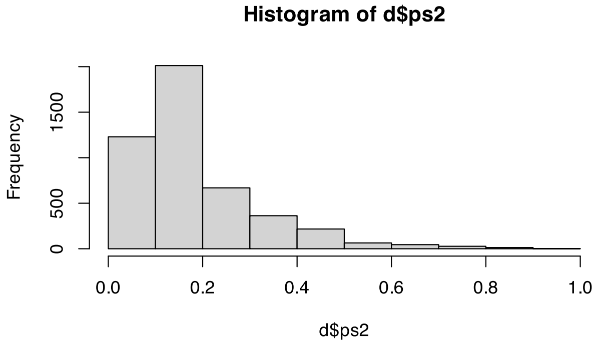
```
psmod2 <- glm(mbsmoke ~ mmarried + mrace + alcohol + fbaby + mage + medu + nprenatal +
                I(mage^2)  + I(nprenatal^2),
              data=d, family=binomial(link = "logit"))

d$ps2 <- predict(psmod2, type="response")
#d %>% group_by(smoker) %>% summarize_at(vars(ps2), list(min=min, max=max))
```

- Again one would like to know/explore that distribution

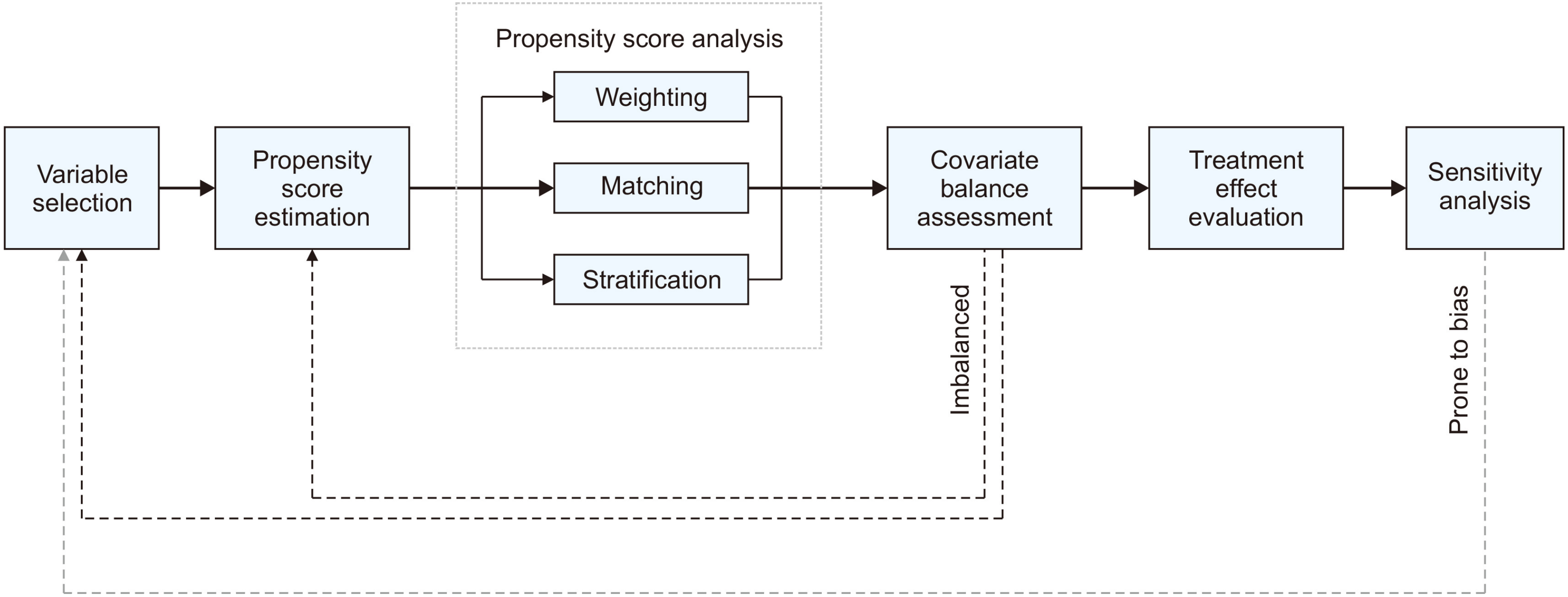
```
round(summary(d$ps2), 2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.01    0.10    0.15    0.19    0.24    0.93
```

```
hist(d$ps2, breaks=10)
```



PS (standard) workflow*


Propensity score analysis
Weighting
Matching
Stratification
Variable selection
Propensity score estimation
Covariate balance assessment
Treatment effect evaluation
Sensitivity analysis
Imbalanced
Prone to bias

*Step 1 and 2 can be iterative, and before the empirical application one should plan ahead.

Part 2.2 – Propensity score methods (Stratification)

What is propensity score stratification?

(a.k.a. subclassification or interval matching)

Is the procedure when the sample is divided into equal-sized bins of the propensity score.

PS stratification is also when we:

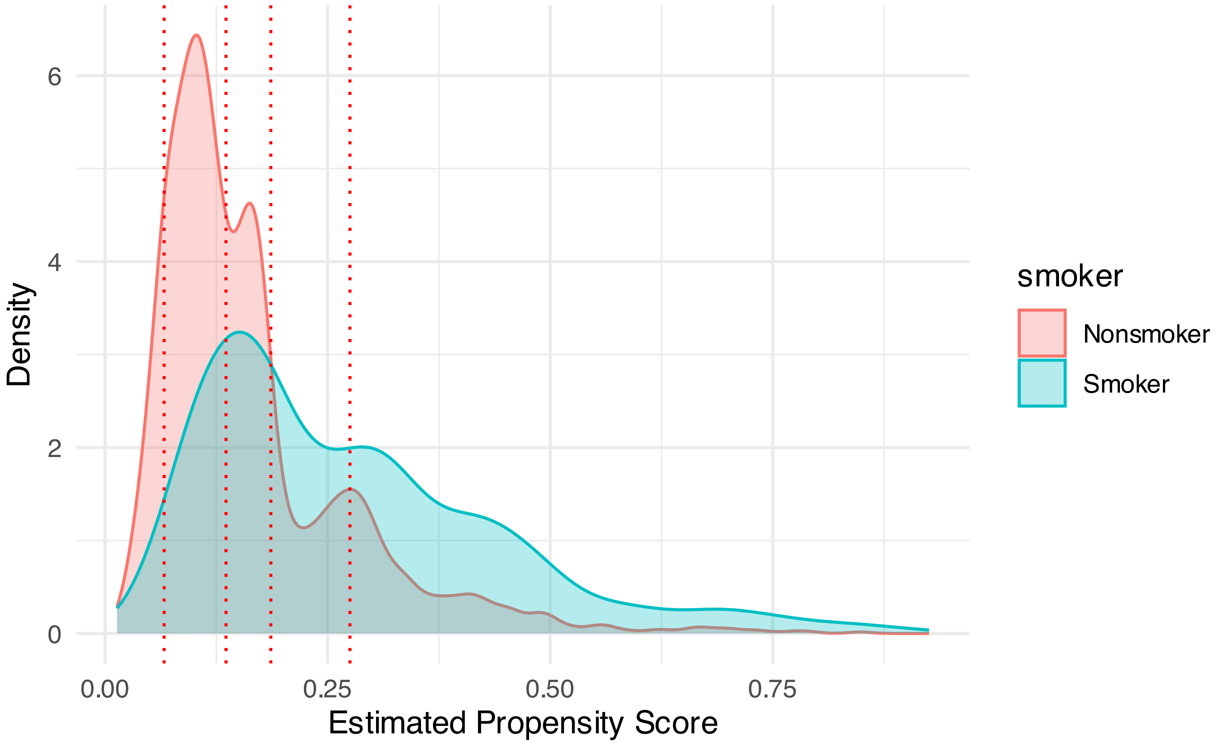
1. separate subjects into **mutually exclusive** equally-sized strata of the PS distribution (usually 5),
2. **estimate the effect of the treatment on the outcome** within each of the strata, and
3. **pool the estimates** to obtain a weighted average.

Stratification Strategies according to the Estimand

- To estimate the **ATE**, determine breakpoints using the PS distribution of the **entire population** and assign of each of the *k* strata a weight of $\frac{1}{k}$
- To estimate the **ATT**, either:
    - determine breakpoints using the PS distribution of the **treated group only** and assign each of the *k* strata a weight of $\frac{1}{k}$, or
    - determine breakpoints using the PS distribution of the **entire population** and assign each of the *k* strata a weight of the **proportion of treated subjects within each stratum.**

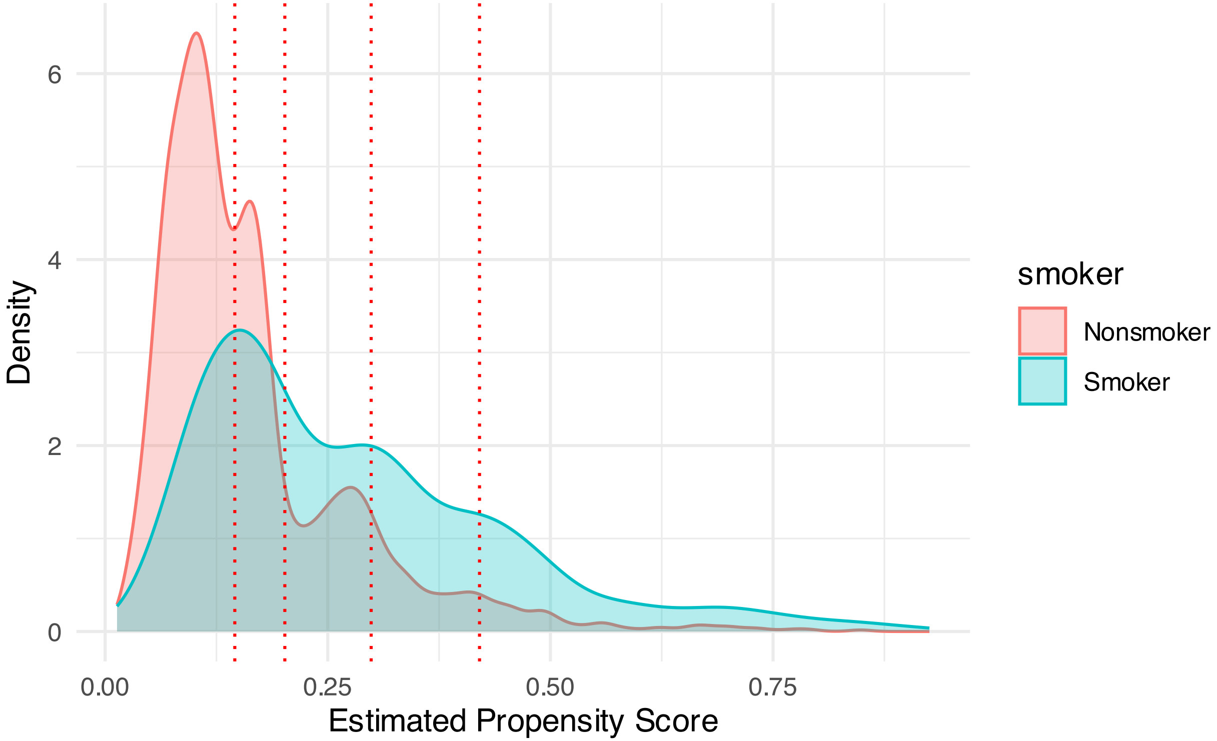
Stratification: ATE breakpoints

```r
strat.ATE <- MatchIt::matchit(mbsmoke ~ mmarried + mrace + alcohol + fbaby + mage + medu +
                       nprenatal + I(mage^2) + I(medu^2) + I(nprenatal^2), data=d,
                    method="subclass", subclass=5, estimand="ATE")
```



Stratification: ATT breakpoints

```
strat.ATT <- MatchIt::matchit(mbsmoke ~ mmarried + mrace + alcohol + fbaby + mage + medu + np
                      I(mage^2) + I(medu^2) + I(nprenatal^2), data=d,
                     method="subclass", subclass=5, estimand="ATT")
```



Stratification

Can be conceptualized as a **meta-analysis of quasi-randomized controlled trials**

- To assess covariate balance, compute standardized differences for each variable within each stratum and take the average across strata

It is one of the less commonly used strategies and often discouraged method

Part 2.3 – Propensity score methods (Matching)

Matching

Matching is most often used when the **ATT** is the estimand of interest.

To estimate the ATT, we take each treated individual and **match them on propensity score** to an untreated individual.

To estimate the ATE, we do the same, but additionally match **each untreated individual** to a treated individual on propensity score.

There are multiple matching options that we need to think about

Basic matching plan

Four primary steps:

1) Planning - type of effect (conditional vs marginal), target population, selecting covariates too balance

2) Matching - exact, nearest neighbor, full ...

3) Assessing - quality of matches

4) Estimating - the treatment effect and its uncertainty

Matching options

Distance metric: matching on PS vs. logit(PS)

Caliper: determines how far a neighbour may be

Replacement: whether one individual can serve as a match for only one individual or for multiple individuals

Matching order: we generally want to go from largest to smallest PS

Ratio: 1:1, 1:many, variable ratio matching

Algorithm: greedy (nearest neighbour) vs. optimal

Exact matching: whether we should match exactly on certain variables

Preparing to use `MatchIt`

Code will be easier to write and read if we pre-define formulae

1. A version that simply lists the covariates
2. A version that contains a more complex regression specification

```
# List version
trt_form1 <- "mbsmoke ~ mmarried + mrace + alcohol + fbaby +  mage + medu + nprenatal"

# Propensity score model version
trt_form2 <- "mbsmoke ~ mmarried + mrace + alcohol + fbaby +
  mage + medu + nprenatal + I(mage^2) + I(medu^2) + I(nprenatal^2)"
```

More complicated version of the regression specification with added squared terms to all of the continuous variables. Often a more reasonable default

MatchIt example

```
set.seed(704) # for reproducible results (tied p-scores)
m_out <- matchit(as.formula(trt_form2), data = d, # key function with many arguments
          distance = "glm",          # model-based
          link = "linear.logit",     # use log-odds directly
          m.order = "largest",       # match order matters; start w/ highest PS
          replace = FALSE )         # default estimand ="ATT", ATE, ATC also available
```

```
m_out
```

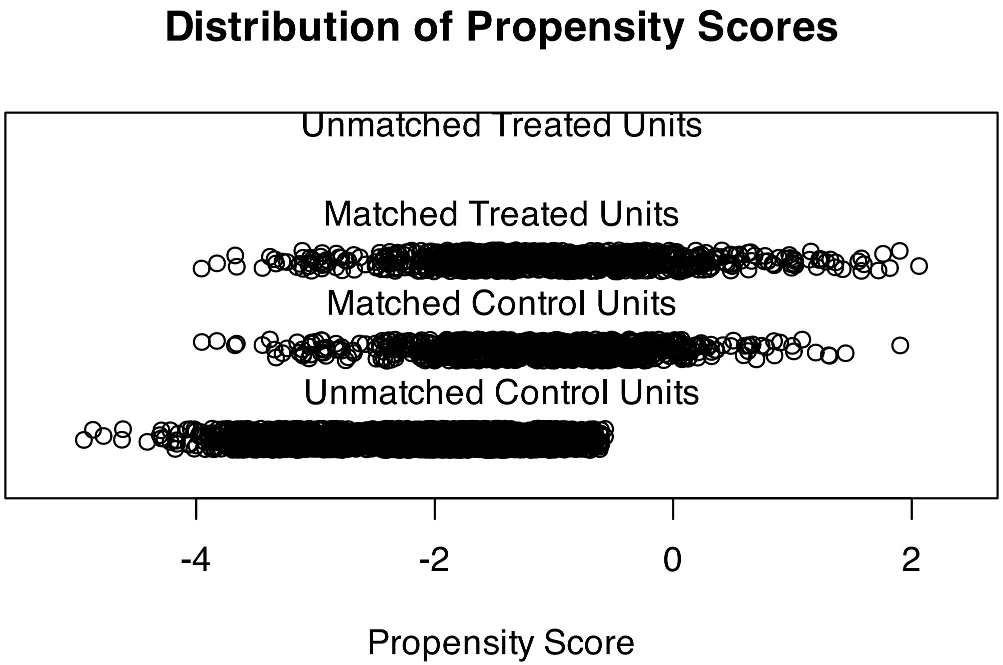
```
## A matchit object
##  - method: 1:1 nearest neighbor matching without replacement
##  - distance: Propensity score
##              - estimated with logistic regression and linearized
##  - number of obs.: 4642 (original), 1728 (matched)
##  - target estimand: ATT
##  - covariates: mmarried, mrace, alcohol, fbaby, mage, medu, nprenatal, I(mage^2), I(medu^2), I(np
```

```
summary(m_out)$nn
```

```
##                 Control Treated
## All (ESS)           3778     864
## All                 3778     864
## Matched (ESS)        864     864
## Matched              864     864
## Unmatched           2914       0
```

A simple plot from MatchIt

```
plot(m_out, type = "jitter", interactive = FALSE)
```



Assessing covariate balance

Standardized mean differences For ATE:

$$\text{bias} = \frac{\bar{x}_T - \bar{x}_C}{\sqrt{\frac{s_T^2 + s_C^2}{2}}}$$

Standardized mean differences For ATT:

$$\text{bias} = \frac{\bar{x}_T - \bar{x}_C}{s_T}$$

- This is an **effect size**: the difference in z-scores between the treatment and control groups.
- The convention is that this should be no greater than 0.1 for any Treatment & Control comparison for *any* variable.

- Some people use the pooled definition (on the left) regardless of the desired estimand as it is more "conservative."

- Can also use: **Kolmogorov-Smirnov distance (KSD)** for the proportion of non-overlap between two distributions (or maximum distance between two cumulative distributions).

Balance checking with `cobalt`

- `cobalt` is an *excellent* package to help check balance
- *CO*variate *BAL*ance *T*ables
- main functions are:
  - `bal.tab` (balance tables)
  - `bal.plot` (to compare single covariate distributions)
  - `love.plot` (graphical balance checking; same info as `bal.tab` in graph form)

Matching Methods for Confounder Adjustment: An Addition to the Epidemiologist's Toolbox, Epidemiologic Reviews, Volume 43, Issue 1, 2021, Pages 118–129

Setting `love.plot` options

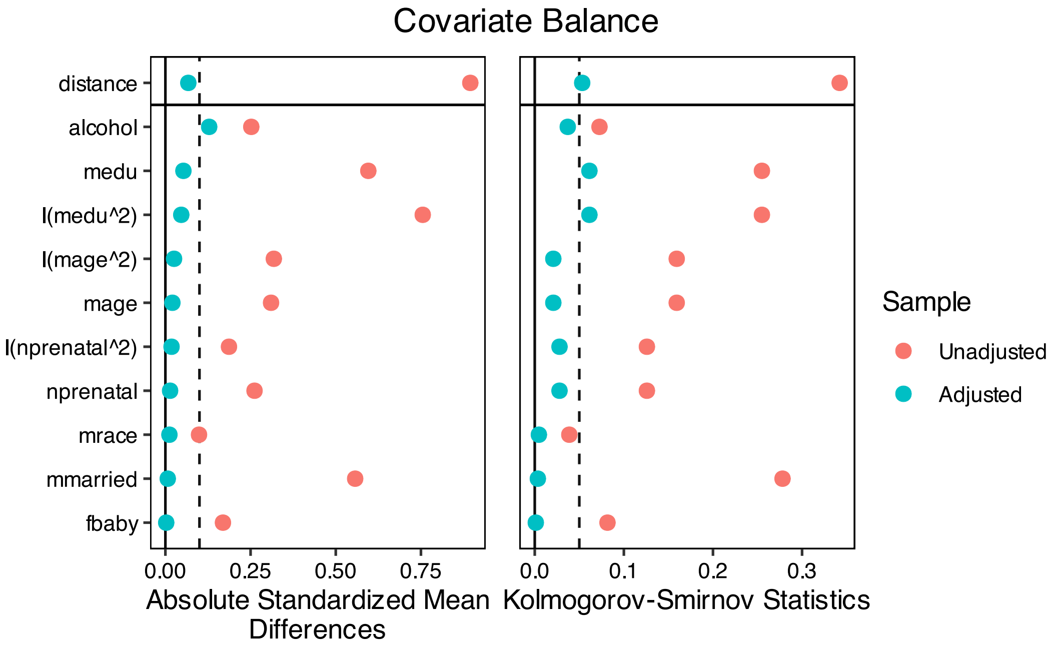
Can avoid typing by defining a custom function for `love.plot` function in `cobalt` package

```
love_plot <- function(x) {
  love.plot(x,
            binary = "std" ,            # use same formula for binary vars
            continuous = "std" ,        # standardize cont. variables
            abs = TRUE ,                # absolute value
            stats = c("m", "ks") ,      # std. bias and Kolmogorov-Smirnov
            s.d.denom = "treat",        # use for ATT
            line = FALSE ,              # to not connect with lines
            var.order = "adj" ,         # sort by adjusted order
            thresholds = c(.10, .05))   # rules of thumb
}
```

KS distance threshold set to .05 per `cobalt` documentation
Set 0.10 SD as mean difference

Check balance

```
love_plot(m_out)
```

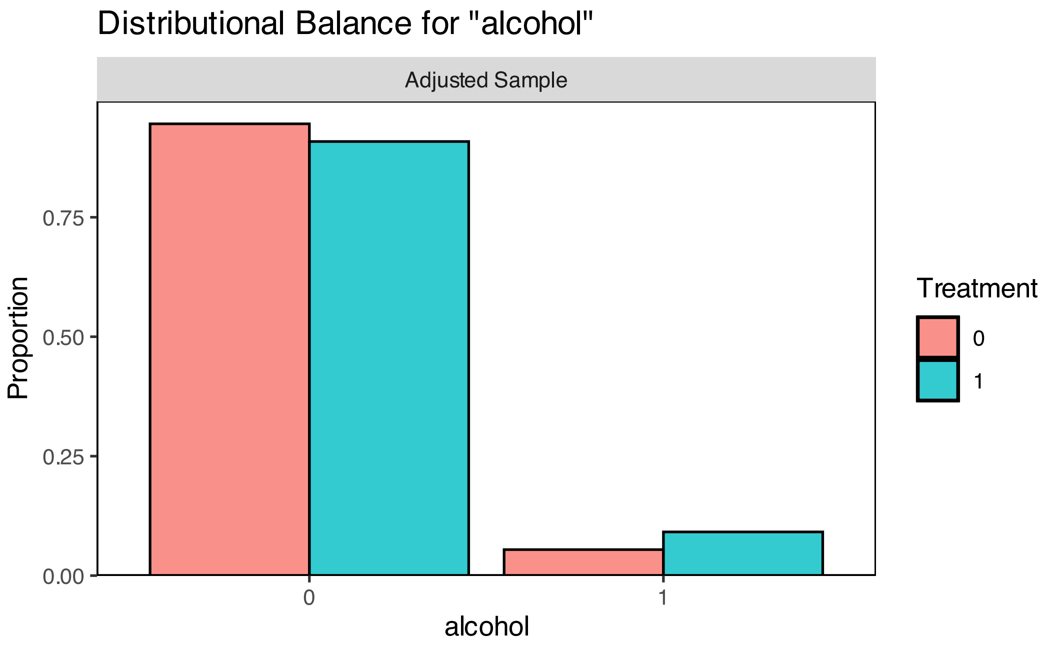


We're close here, but not quite balanced as well as we'd like.

Examine variables with issues

Alcohol

```
bal.plot(m_out, "alcohol")
```

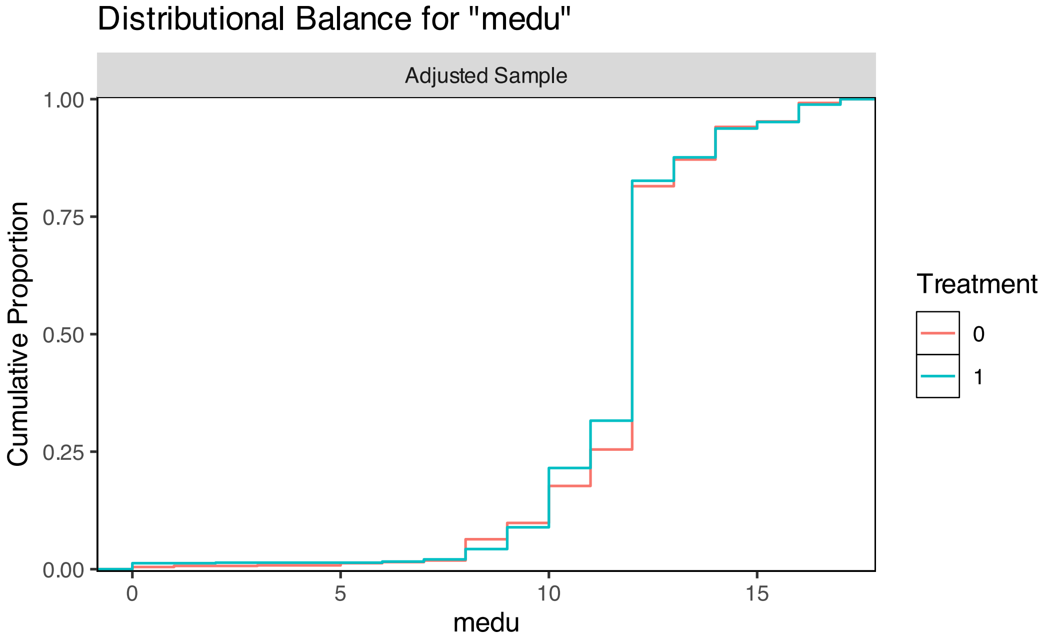


Looks like we didn't quite achieve good balance.

Examine variables with issues

Education

```
bal.plot(m_out, var.name = "medu", type = "ecdf") # cumulative dist. function
```



Smokers are slightly *more* educated than their matches.

Imbalance: reasons and possible solutions

Lack of common support

- discard treatment cases above maximum control propensity
- this will result in "feasible" estimates that may be biased by incomplete matching

Poorly specified propensity-score model

- add interaction terms in PSM
- add higher-order polynomials in PSM
- generally better to "overfit" since model is not really a "model" but a measure of similarity

"Inliers" (not enough available controls in a treatment-dense area of the PS distribution)

- use calipers (thus feasible estimates)
- use matching *with* replacement (not forcing as many bad matches)

Matching *with* replacement

```
set.seed(704) # for reproducible results (tied p-scores)
m_out2 <- matchit(as.formula(trt_form2), data = d, distance = "linear.logit", # use log-odds
          m.order = "largest", replace = TRUE )
m_out2; summary(m_out2)$nn
```

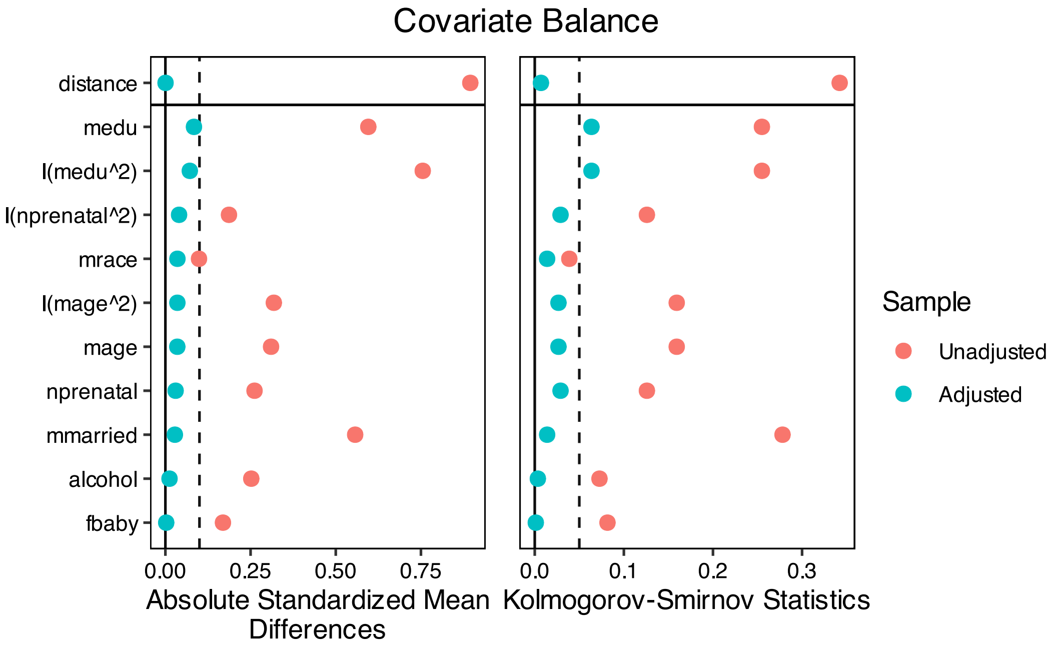
```
## A matchit object
##  - method: 1:1 nearest neighbor matching with replacement
##  - distance: Propensity score
##              - estimated with logistic regression
##  - number of obs.: 4642 (original), 1498 (matched)
##  - target estimand: ATT
##  - covariates: mmarried, mrace, alcohol, fbaby, mage, medu, nprenatal, I(mage^2), I(medu^2), I(np
```

```
##                 Control Treated
## All (ESS)     3778.0000     864
## All           3778.0000     864
## Matched (ESS)  470.0856     864
## Matched        634.0000     864
## Unmatched     3144.0000       0
## Discarded        0.0000       0
```

NOTE: Fewer control cases are matched here because some are used to match multiple treatment cases.

Checking balance

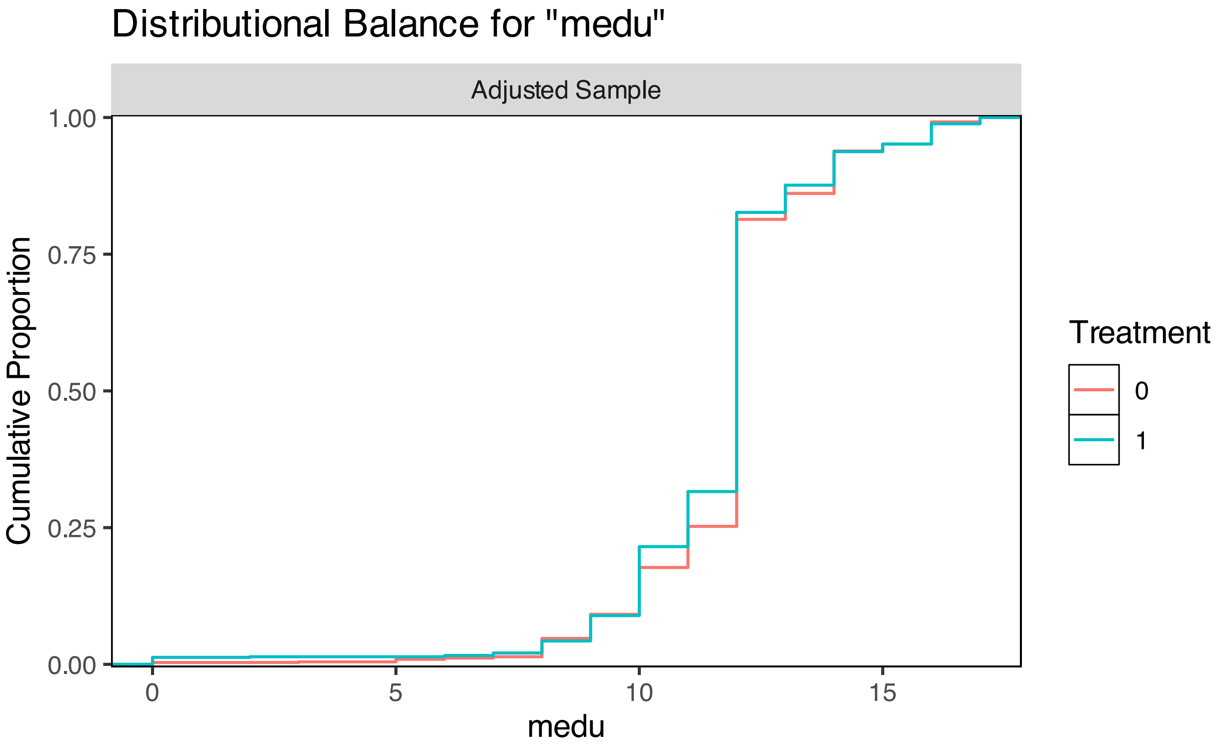
```
love_plot(m_out2)
```



Mean balance is improved, most noticeably on alcohol. The distribution of education is still a bit off, however.

Checking balance

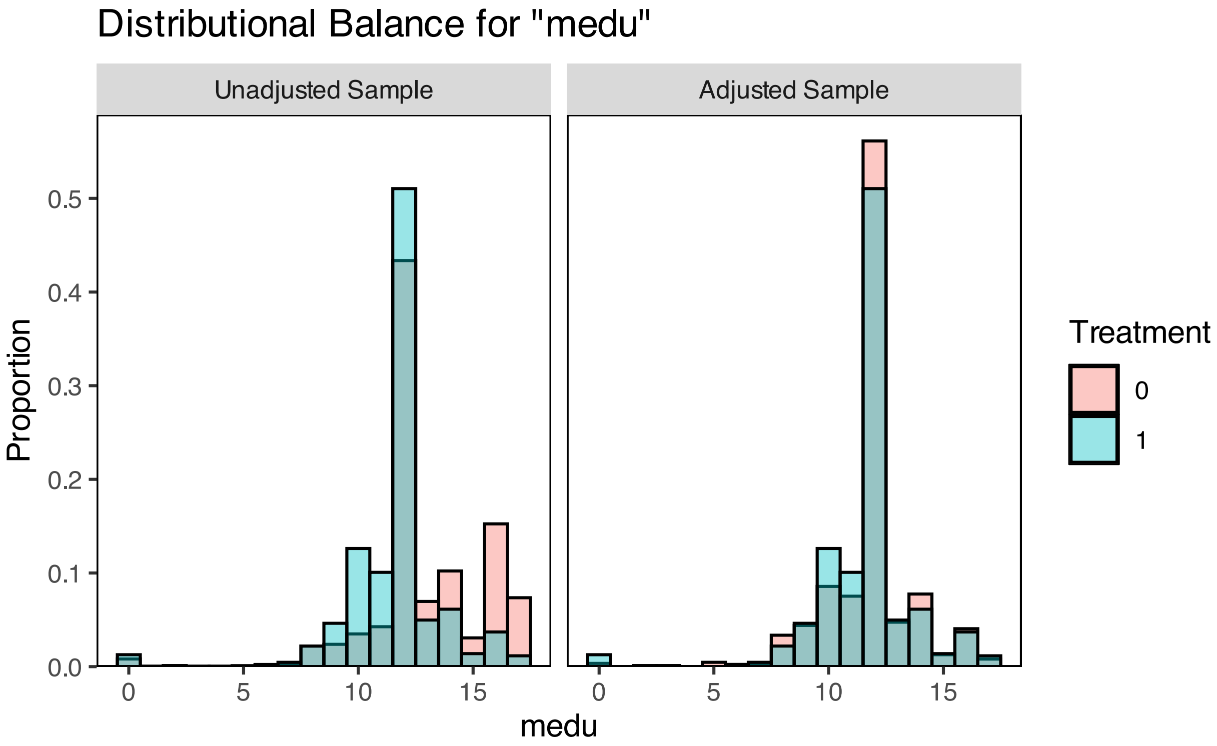
```
bal.plot(m_out2, "medu", type = c("ecdf"))
```



This is about the same as matching without replacement.

Checking balance

```
bal.plot(m_out2, "medu", type = c("hist"), bins = 18, which = "both")
```



The distribution isn't perfect here but it's much improved.

Calipers

- A "caliper" is maximum distance (in SDs of the propensity score) beyond which matches are not allowed. A typical caliper distance is .10.

- Using a caliper means discarding treatment cases without high-quality matches.

- Discarding treatment cases means getting **"feasible"** ATT estimates that may be biased as they don't generalize to the whole population from which the sample was drawn.

Calipers

```
m_cal <- matchit(as.formula(trt_form2), data = d,
                 caliper = .1, replace = FALSE)
summary(m_cal)$nn
```

```
##               Control Treated
## All (ESS)        3778     864
## All              3778     864
## Matched (ESS)     828     828
## Matched           828     828
## Unmatched        2950      36
## Discarded           0       0
```

Here we'd lose 36 treated cases, moving us from a SATT estimate to an FSATT (fine stratification-ATT) estimate.

- There are better alternatives than calipers, but they *are* useful for diagnostic purposes
- If you're dropping a lot of cases, the data need a much closer look.

Getting the matched dataset

Let's assume we're OK with our slightly imperfect match without replacement.
Use `match.data` from `MatchIt` package to get the dataset for analysis.

```
m_data <- match.data(m_out)
nrow(m_data)                    # confirm that you have 2x treatment cases
```

```
## [1] 1728
```

In the `m_data` object we just created, there are twice the number of treated cases, exactly as we'd expect with a 1:1 match without replacement.

Next step: working with the matched data

Now we just do a simple regression of the outcome (`zweight`) on the treatment indicator (`mbsmoke`). This will give us the ATT.

```
lm( zweight ~ mbsmoke , data = m_data ) %>% tidy()
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -0.000809    0.0338   -0.0239 9.81e- 1
## 2 mbsmoke     -0.386       0.0478   -8.08   1.20e-15
```

```
ATT <- mean(m_data$zweight[m_data$mbsmoke==1]) -mean(m_data$zweight[m_data$mbsmoke==0])
```

- The ATT estimate here is **-0.386**
- The SE estimate does *not* take into account that that PS was *estimated* prior to matching
- Since the match is 1:1, no weights are needed to unconfound the summary score S and T.

Next step: working with the matched data

Here is what the process would look like using the matching results *with* replacement.

```
m_data2 <- match.data(m_out2)
lm( zweight ~ mbsmoke , data = m_data2, weights = weights ) %>% tidy()
```

```
## # A tibble: 2 × 5
##   term         estimate std.error statistic  p.value
##   <chr>           <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -0.000757    0.0391   -0.0193 9.85e- 1
## 2 mbsmoke     -0.386       0.0515   -7.50   1.11e-13
```

```
ATT <- mean(m_data2$zweight[m_data2$mbsmoke==1]) -mean(m_data2$zweight[m_data2$mbsmoke==0])
```

Here we *do* need weights because some controls need to be counted multiple times.

The estimate of the ATT is slightly larger, **-0.399** and the **SE** is also slightly larger since replacements are being used.

Bootstrapping standard errors

Unlike a standard full-sample regression, there are multiple sources of uncertainty in matched ATT estimates:

1. Sampling error in treatment and outcome (the "normal" kind)
2. Sampling error in PS model covariates
3. Arbitrary choice among tied matches

If we take the `lm`-provided SE seriously, we are ignoring most of this.

So we need a better approach, such as bootstrapping

PS matching summary

1) Special case of matching, the distance metric is the (estimated) propensity score

2) 1-to-n closest neighbor matching is common when the control group is large compared to treatment group

3) **Pros**:

- Robust, matched pairs (within pair analysis), balance distributions in directions orthogonal to estimated PS, immensely popular, vast literature.

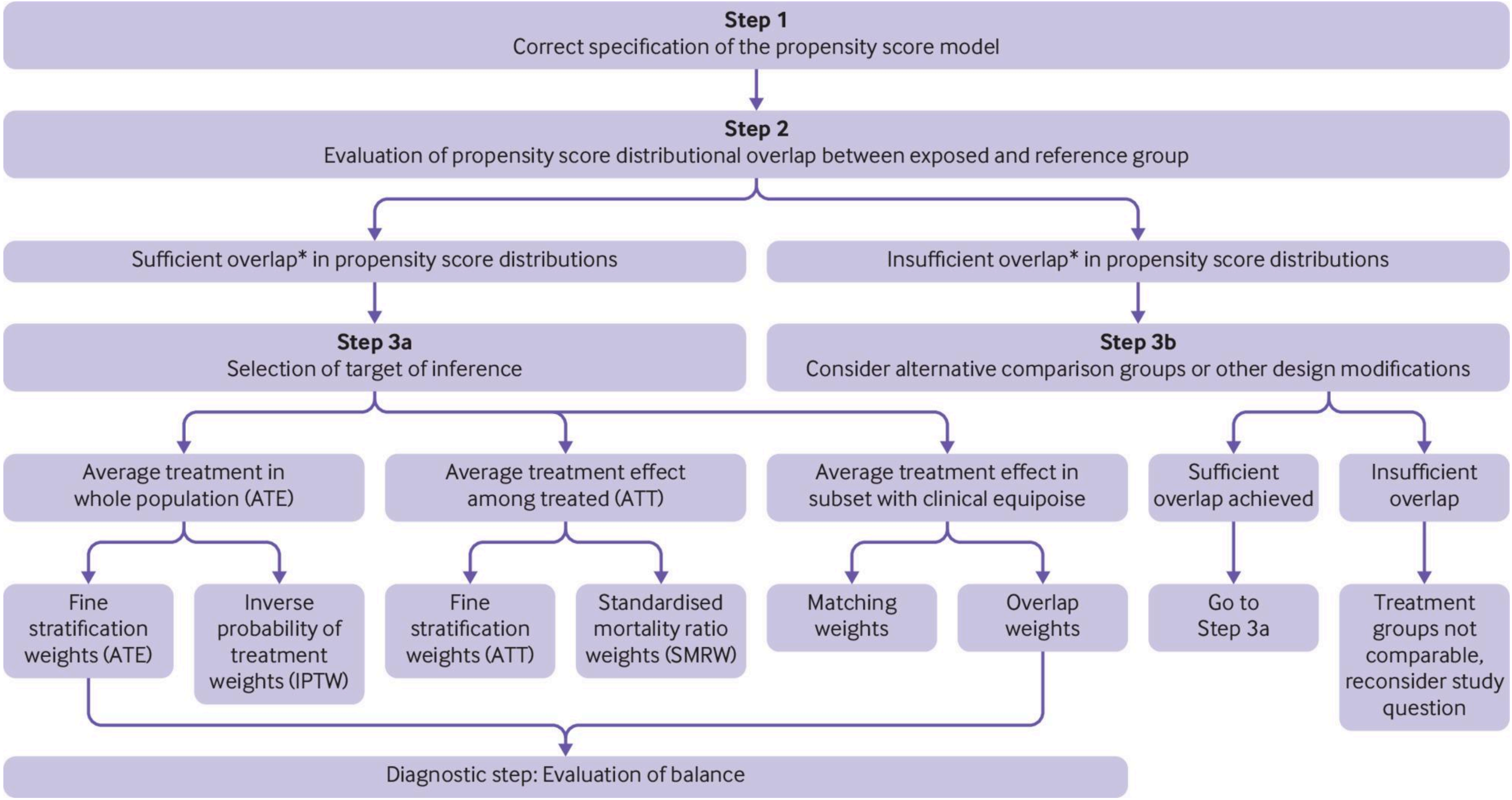
--

4) **Cons**:

- Sometimes dimension reduction via the propensity score may be too drastic, recent methods advocate matching on the multivariate covariates directly, of fine stratification (PS used to create stratum, then weighting according to stratum size)

Part 2.4 Propensity score (weighting)

Still more options???



Step 1
Correct specification of the propensity score model

Step 2
Evaluation of propensity score distributional overlap between exposed and reference group

Sufficient overlap* in propensity score distributions

Insufficient overlap* in propensity score distributions

Step 3a
Selection of target of inference

Step 3b
Consider alternative comparison groups or other design modifications

Average treatment in whole population (ATE)

Average treatment effect among treated (ATT)

Average treatment effect in subset with clinical equipoise

Sufficient overlap achieved

Insufficient overlap

Fine stratification weights (ATE)

Inverse probability of treatment weights (IPTW)

Fine stratification weights (ATT)

Standardised mortality ratio weights (SMRW)

Matching weights

Overlap weights

Go to Step 3a

Treatment groups not comparable, reconsider study question

Diagnostic step: Evaluation of balance

Desai R J, Franklin J M. Alternative approaches for confounding adjustment in observational studies using weighting based on the propensity score: a primer for practitioners BMJ 2019; 367 :l5657 doi:10.1136/bmj.l5657

What is weighting?

- An approach that skips the subclassification or matching step and uses **inverse probability of treatment weights** (IPTW) based on each case's propensity score directly.

- The idea is to apply a weight to each case that makes the treatment and the control *groups* balanced on the propensity score and therefore (hopefully) on the individual covariates.

IPTW is like survey weighting

Sex	P(population)	P(sample)	formula	weight
Male	.50	.40	.50/.40	1.25
Female	.50	.60	.50/.60	.80

- In surveys, more weight to groups that don't show up as much as they should in the sample

- The logic with IPTW is the same – in an experiment the treatment and control groups "should" be balanced in their propensity to receive treatment and on any other factors as well.

Recall Possible treatment effects

1. Average treatment effect (**ATE**) for all participants (quantity estimated from a RCT)
2. Average treatment effect for those who received the treatment (**ATT**)
3. Average treatment effect among controls (**ATC**)
4. Average treatment effect among the evenly matchable (**ATM**), nearly equivalent to cohort formed by one-to-one pair matching
5. Average treatment effect among the overlap population (**ATO**), estimates the treatment effect among those likely to have received either treatment or control

There are PS weights for each different treatment effects

The PS for participant i is defined here as e_i and the treatment assignment is T_i, where T=1 indicates the participant received the treatment and T=0 indicates they received the control.

$$w_{ATE} = \frac{T_i}{e_i} + \frac{1 - T_i}{1 - e_i}$$

$$w_{ATT} = \frac{e_i T_i}{e_i} + \frac{e_i(1 - T_i)}{1 - e_i}$$

$$w_{ATC} = \frac{(1 - e_i)T_i}{e_i} + \frac{(1 - e_i)(1 - T_i)}{1 - e_i}$$

$$w_{ATM} = \frac{\min\{e_i, 1 - e_i\}}{T_i e_i + (1 - T_i)(1 - e_i)}$$

$$w_{AT0} = (1 - e_i)T_i + e_i(1 - T_i)$$

Coding the weights can be done manually BUT, it is not necessary as the `WeightIt` package offers all these estimands

Example - Weighting for ATT

- We can transform the propensity score into a weight so that the control group has the same average propensity to smoke as the treatment group.
- This works by giving more weight to high-propensity controls and less weight to low-propensity controls (like survey weights).
- Treatment cases all get a weight of 1 because the treatment cases already look like themselves!

$$w_i = T_i + (1 - T_i)\frac{P(T = 1|\mathbf{X}_i)}{1 - P(T = 1|\mathbf{X}_i)}$$

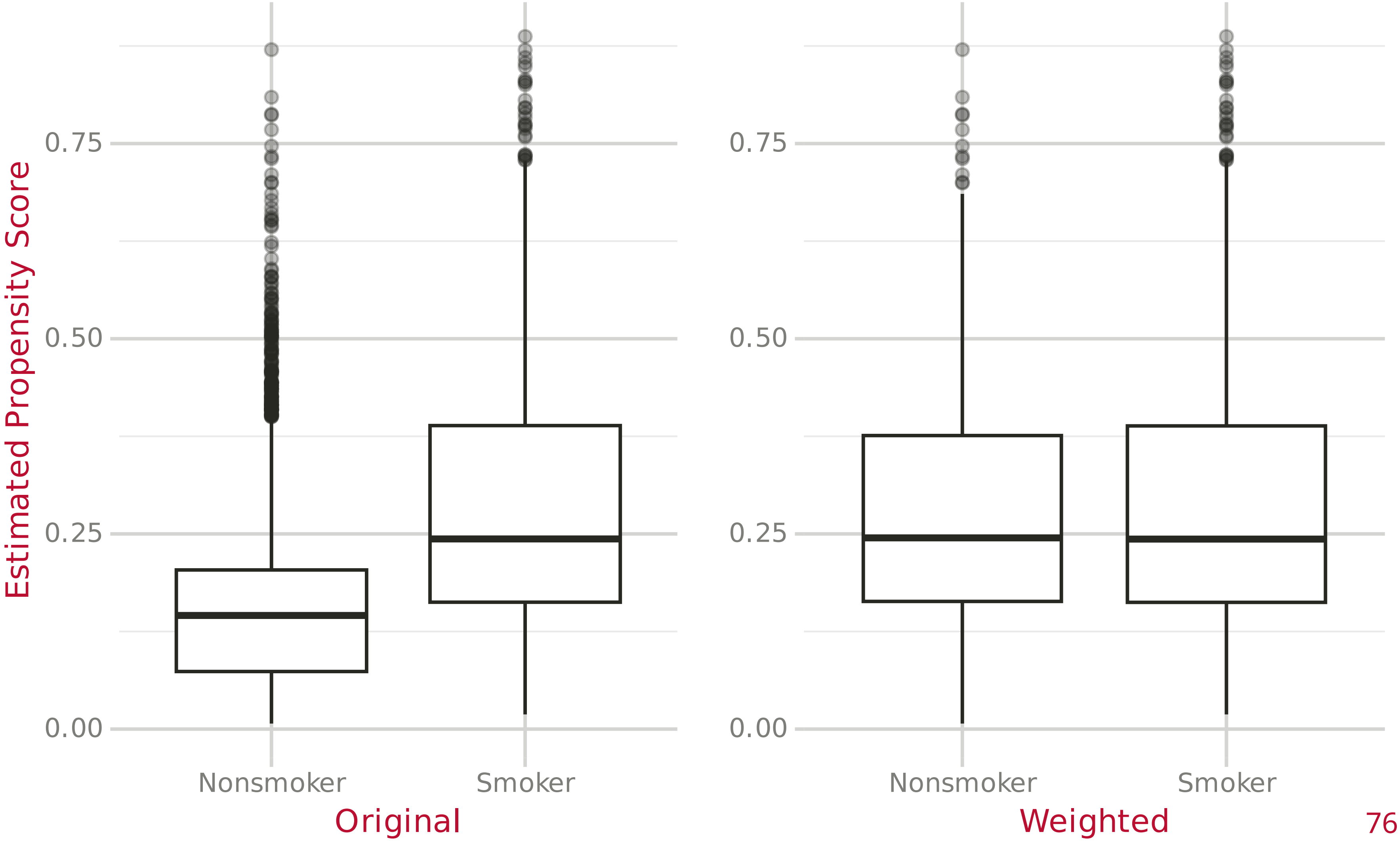
Creating the weights manually

Creating weights that balance the treatment and control groups on the propensity score is relatively simple.

```
psmod <- glm(as.formula(trt_form2), data = d, family = binomial())

d <- d %>%
  mutate(pscore = predict(psmod, type = "response") ,
         attwt = if_else(mbsmoke==1, 1 , pscore/(1-pscore) ),
         atewt = if_else(mbsmoke==1, 1/pscore , 1/(1-pscore) ))
```

- **The ATT weight makes the *controls* look like the treatment**
- **The ATE weight makes *both groups* look like the total sample**

ATT weights improve balance



Use `WeightIt` instead of manual calculation

- `WeightIt` is an excellent package (same author as `MatchIt` package)
- Just like `MatchIt` brings many capabilities under one package, `WeightIt` serves as a wrapper on the myriad weighting algorithms out there, giving them a unified syntax.

```r
W1 <- weightit(as.formula(trt_form2) ,
               data = d ,
               method = "ps" ,     # propensity score weighting
               s.weights = NULL,   # placeholder for sampling weights
               estimand = "ATT")   # ATT estimand (not ATE)
```

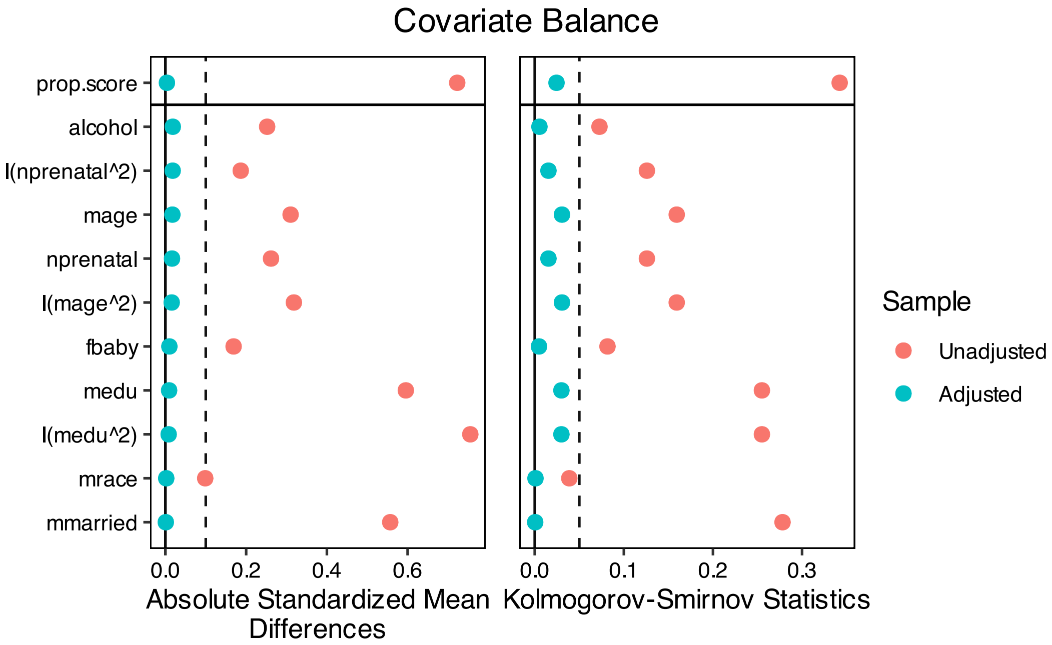
Results

```
summary(W1)
```

```
##                    Summary of weights
## 
## - Weight ranges:
## 
##            Min                                  Max
## treated 1.0000     ||                        1.0000
## control 0.0071 |---------------------------| 6.7093
## 
## - Units with the 5 most extreme weights by group:
##                                               
##               47     43     25     20     11
##  treated       1      1      1      1      1
##             1594   1084   4342   4408   3074
##  control 3.3045 3.6843 3.7092 4.2442 6.7093
## 
## - Weight statistics:
## 
##         Coef of Var   MAD Entropy # Zeros
## treated       0.000 0.000   0.000       0
## control       1.234 0.664   0.422       0
## 
## - Effective Sample Sizes:
## 
##            Control Treated
## Unweighted 3778.       864
```

Check balance

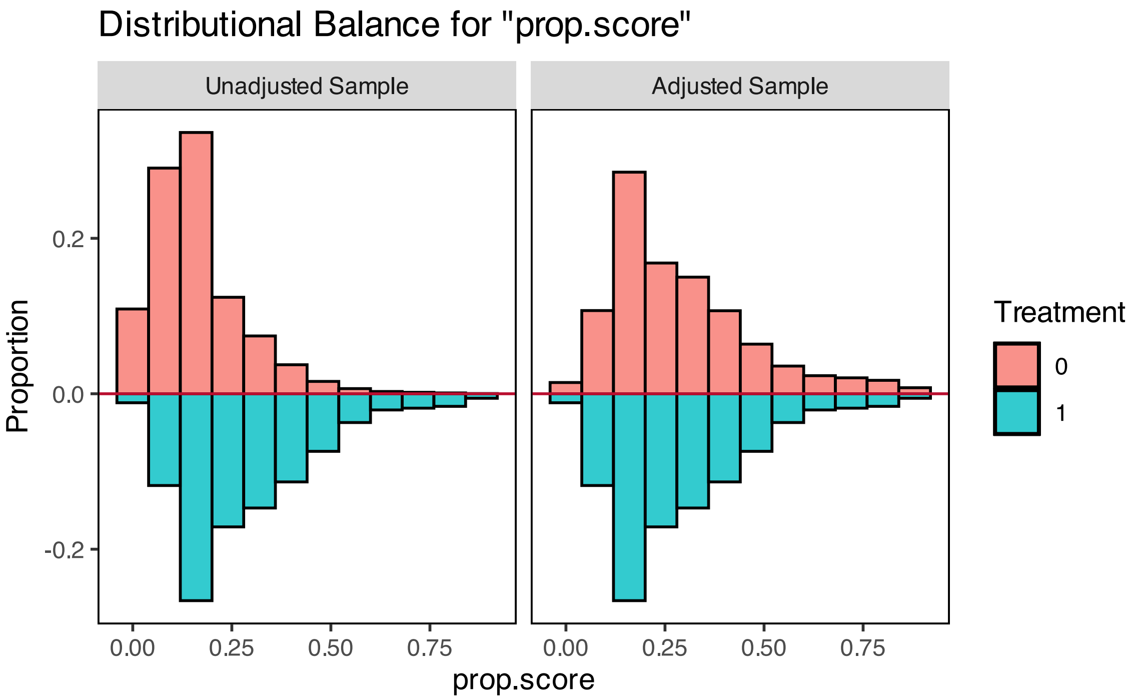
```
love_plot(W1)
```



Looks pretty good. Note: if we hadn't had `I(medu^2)` in the model, we would *not* have had acceptable balance.

Assessing overlap

```
bal.plot(W1, var.name = "prop.score", which = "both", type = "histogram",
         mirror = TRUE)
```



Estimate ATT

```
lm(zweight ~ mbsmoke, data = d, weights = W1$weights ) %>% tidy()
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -0.00508    0.0208    -0.244 8.07e- 1
## 2 mbsmoke     -0.382      0.0295   -13.0   8.39e-38
```

The ATT estimate is thus **-.382**. (ATE from matching was -0.386)
Again might want to do some bootstrapping to get a better estimate of the SE

IPTW (summary)

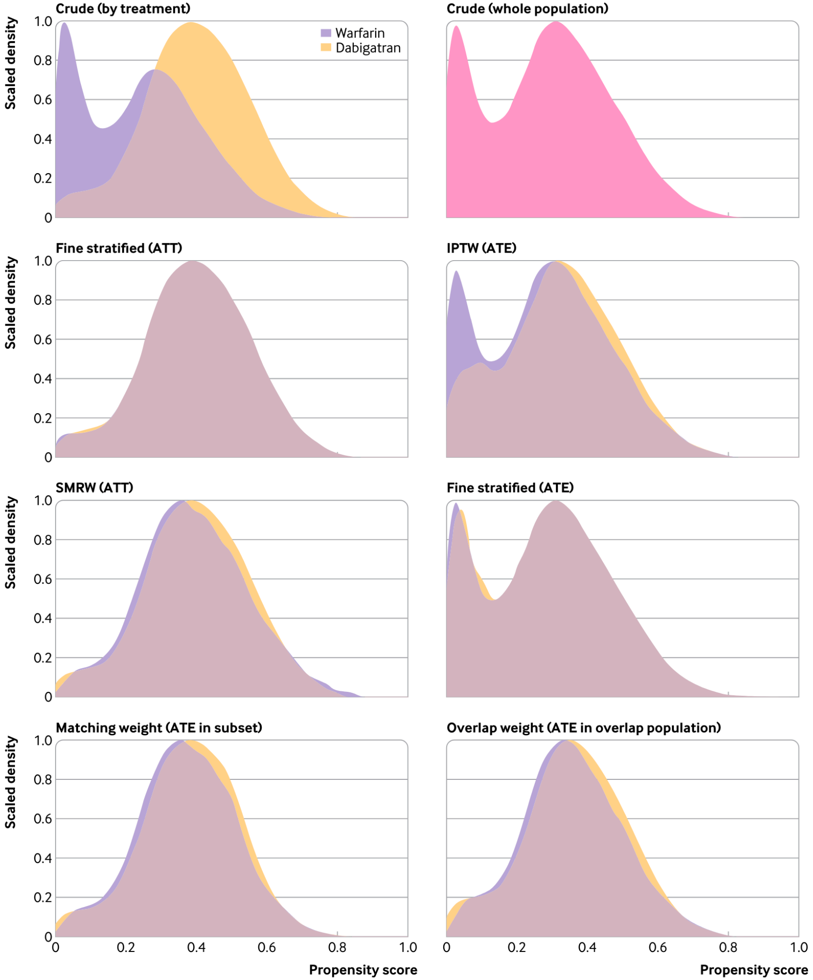
Advantages

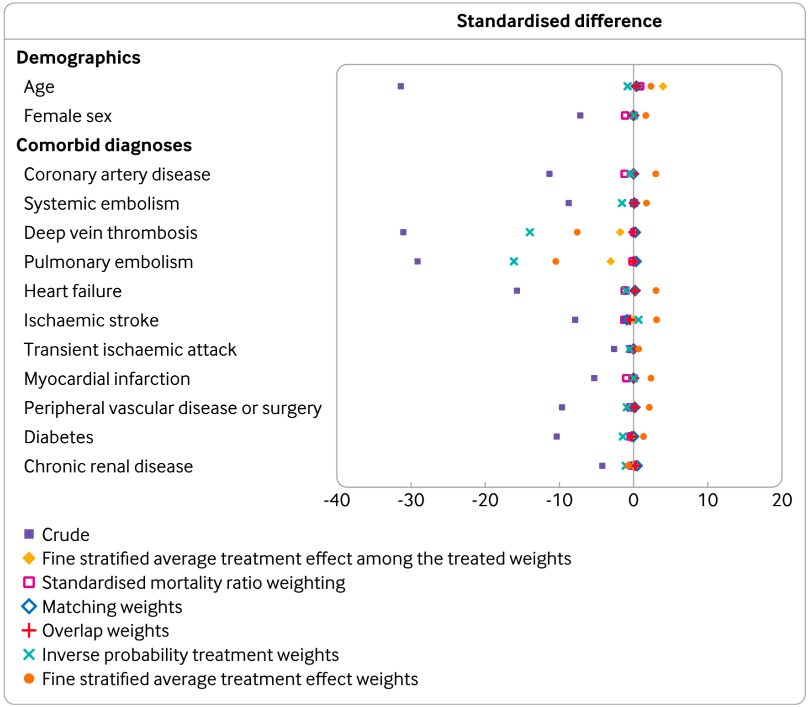
- Simple, with theoretical foundation
- Explicit target population
- Global balance
- Extends to more complex settings

Disadvantages

- More sensitive to model misspecification
- PS near 0 or 1 yield extreme weights
- ATE might not always be the sensible estimand

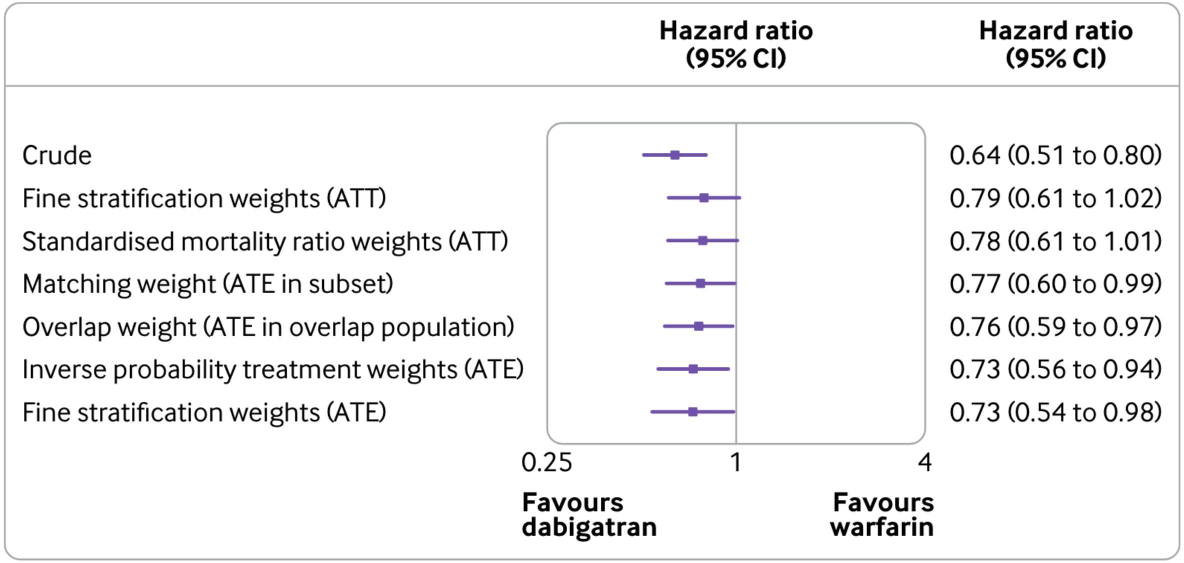
Example of new weighting schemes





Example of new weighting schemes

Results



	Hazard ratio (95% CI)
Crude	0.64 (0.51 to 0.80)
Fine stratification weights (ATT)	0.79 (0.61 to 1.02)
Standardised mortality ratio weights (ATT)	0.78 (0.61 to 1.01)
Matching weight (ATE in subset)	0.77 (0.60 to 0.99)
Overlap weight (ATE in overlap population)	0.76 (0.59 to 0.97)
Inverse probability treatment weights (ATE)	0.73 (0.56 to 0.94)
Fine stratification weights (ATE)	0.73 (0.54 to 0.98)

BMJ DOI:10.1136 bmj.l5657

QUESTIONS?

COMMENTS?

RECOMMENDATIONS?

Extra resources

- Propensity Score Analysis: A Primer and Tutorial by Noah Greifer Here
- Choosing the Causal Estimand for Propensity Score Analysis of Observational Studies (https://doi.org/10.48550/arXiv.2106.10577)
- Griffin BA, Schuler MS, Cefalu M, et al. A Tutorial for Propensity Score Weighting for Moderation Analysis With Categorical Variables: An Application Examining Smoking Disparities Among Sexual Minority Adults. Med Care. 2023;61(12):836-845. doi:10.1097/MLR.0000000000001922
- Quantitude Podcast: S3E27: Propensity Scores -- I Meant To Do That! link here
- Austin, P.C. (2008), A critical appraisal of propensity-score matching in the medical literature between 1996 and 2003. Statist. Med., 27: 2037-2049. https://doi.org/10.1002/sim.3150
- Austin, P.C. (2007), The performance of different propensity score methods for estimating marginal odds ratios. Statist. Med., 26: 3078-3094. https://doi-org.proxy3.library.mcgill.ca/10.1002/sim.2781
- Austin, P.C. (2009), Balance diagnostics for comparing the distribution of baseline covariates between treatment groups in propensity-score matched samples. Statist. Med., 28: 3083-3107. https://doi.org/10.1002/sim.3697

Extra resources

- Wan F. Matched or unmatched analyses with propensity-score-matched data?. Stat Med. 2019;38(2):289-300. doi:10.1002/sim.7976
- Til Stürmer, Kenneth J. Rothman, Jerry Avorn, Robert J. Glynn, Treatment Effects in the Presence of Unmeasured Confounding: Dealing With Observations in the Tails of the Propensity Score Distribution—A Simulation Study, American Journal of Epidemiology, Volume 172, Issue 7, 1 October 2010, Pages 843–854, https://doi.org/10.1093/aje/kwq198
- Kosuke Imai, Marc Ratkovic, Covariate Balancing Propensity Score, Journal of the Royal Statistical Society Series B: Statistical Methodology, Volume 76, Issue 1, January 2014, Pages 243–263, https://doi.org/10.1111/rssb.12027
- Paul R. Rosenbaum & Donald B. Rubin (1984) Reducing Bias in Observational Studies Using Subclassification on the Propensity Score, Journal of the American Statistical Association, 79:387, 516-524, DOI: 10.1080/01621459.1984.10478078
- Schneeweiss, Sebastian; Rassen, Jeremy A.; Glynn, Robert J.; Avorn, Jerry; Mogun, Helen; Brookhart, M Alan. High-dimensional Propensity Score Adjustment in Studies of Treatment Effects Using Health Care Claims Data. Epidemiology 20(4):p 512-522, July 2009. | DOI: 10.1097/EDE.0b013e3181a663cc

Step 2 - Codes to Examine Distribution of PS estimates

- Explore, visually the distribution of PS across levels of the Treatment/Exposure/Intervention

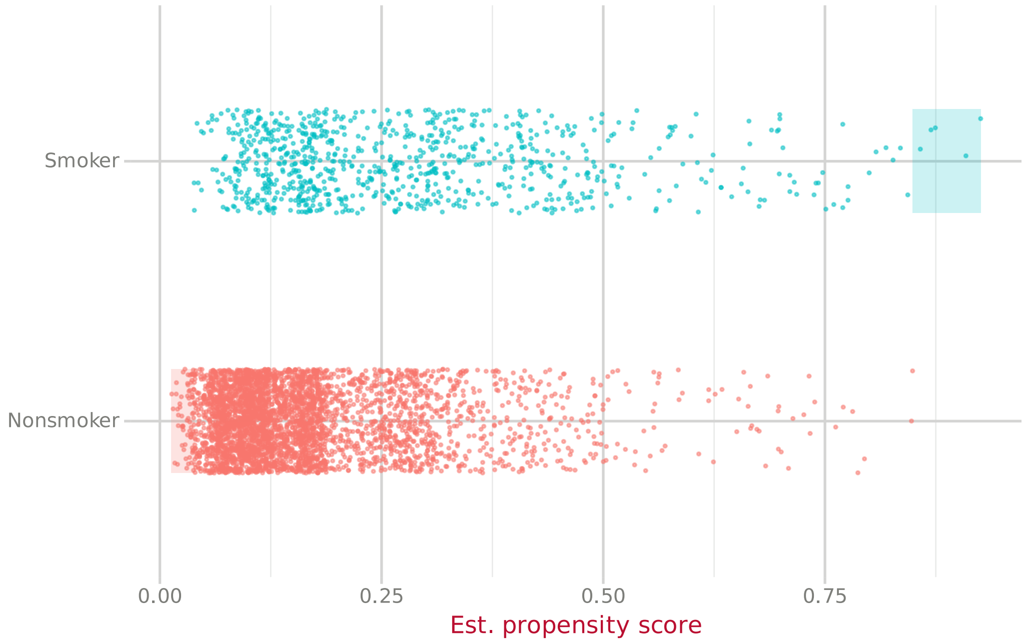
```r
# density plot
gg <- ggplot(d, aes( x = ps1 , fill = smoker, color = smoker, after_stat(scaled) )) +
  geom_density(alpha = .3) +
  labs(x = "Estimated Propensity Score" ,
       y = "Density") +
  theme(legend.position = "top") +
  theme(legend.title = element_blank()) +
  theme_xaringan(background_color = "#FFFFFF",
                 text_font_size = 10,
                 title_font_size = 12)
#gg
```

Step 2 - Codes to Examine Distribution of PS estimates

```
# boxplot
gg1 <- ggplot(d, aes( y = ps1 , x = smoker )) +
  geom_boxplot(outlier.alpha = .3) +
  labs(y = "Estimated Propensity Score" ,
       x = "") +
  theme(legend.position = "top") +
  theme(legend.title = element_blank()) +
  theme_xaringan(background_color = "#FFFFFF",
                 text_font_size = 10,
                 title_font_size = 12)
#gg1
```

Step 2 - Codes to Examine Distribution of PS estimates

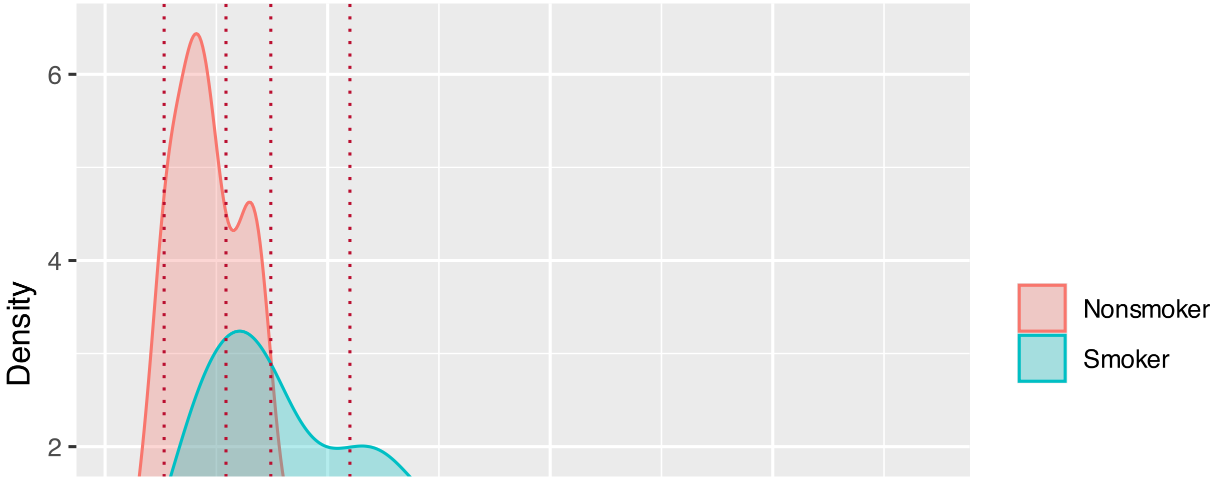
```
gg2 <- ggplot( d , aes( x = smoker , y = ps1 , color = smoker )) + geom_jitter(alpha = .5, s
  theme(legend.position = "none") + coord_flip() + labs(x = "", y = "Est. propensity score")
  annotate(geom="rect", ymin = .013, ymax = .039, xmin = .8, xmax = 1.2, fill = "#F8766D", a
  annotate(geom="rect", ymin = .849, ymax = .926, xmin = 1.8, xmax = 2.2, fill = "#00BFC4",
  theme_xaringan(background_color = "#FFFFFF", text_font_size = 10, title_font_size = 12)
gg2
```



Stratification: ATE breakpoints

```
strat.ATE <- matchit(mbsmoke ~ mmarried + mrace + alcohol + fbaby + mage + medu + nprenatal
                        I(mage^2) + I(medu^2) + I(nprenatal^2), data=d,
                      method="subclass", subclass=5, estimand="ATE")
```

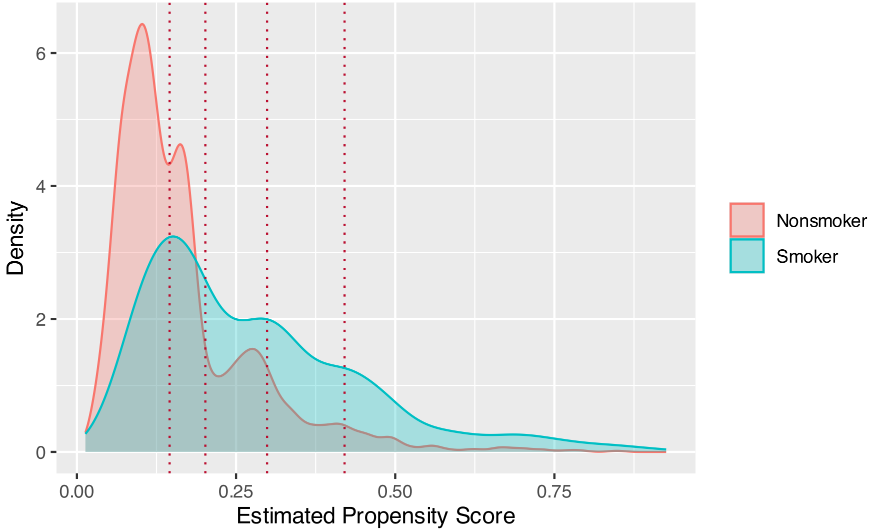
```
#ATE density plot
ggplot(d, aes(x=ps1, fill=smoker, color=smoker)) + geom_density(alpha=0.3) +
  labs(x = "Estimated Propensity Score", y = "Density") +
  theme(legend.title = element_blank()) +
  geom_vline(aes(xintercept=strat.ATE$q.cut[2]), linetype = 3) +
  geom_vline(aes(xintercept=strat.ATE$q.cut[3]), linetype=3) +
  geom_vline(aes(xintercept=strat.ATE$q.cut[4]), linetype=3) +
  geom_vline(aes(xintercept=strat.ATE$q.cut[5]), linetype=3)
```



Stratification: ATT breakpoints

```
strat.ATT <- matchit(mbsmoke ~ mmarried + mrace + alcohol + fbaby + mage + medu + nprenatal +
                       I(mage^2) + I(medu^2) + I(nprenatal^2), data=d,
                     method="subclass", subclass=5, estimand="ATT")
```

```
#ATT density plot
ggplot(d, aes(x=ps1, fill=smoker, color=smoker)) + geom_density(alpha=0.3) +
  labs(x = "Estimated Propensity Score", y = "Density") + theme(legend.title = element_blank
  geom_vline(aes(xintercept=strat.ATT$q.cut[3]), linetype=3) +
  geom_vline(aes(xintercept=strat.ATT$q.cut[4]), linetype=3) +
  geom_vline(aes(xintercept=strat.ATT$q.cut[5]), linetype=3)
```



Coding the weights manually (TL;NR)

```r
# add weights to hypothetical data frame
dat <- dat %>%
  mutate(
    w_ate = (treatment / propensity_score) +
      ((1 - treatment) / (1 - propensity_score)),
    w_att = ((propensity_score * treatment) / propensity_score) +
      ((propensity_score * (1 - treatment)) / (1 - propensity_score)),
    w_atc = (((1 - propensity_score) * treatment) / propensity_score) +
      (((1 - propensity_score) * (1 - treatment)) / (1 - propensity_score)),
    w_atm = pmin(propensity_score, 1 - propensity_score) /
      (treatment * propensity_score + (1 - treatment) * (1 - propensity_score)),
    w_ato = (1 - propensity_score) * treatment +
      propensity_score * (1 - treatment)
  )
```

This is not necessary as the **WeightIt** package offers all these estimands

Interpreting the summary form WeightIt

This `summary = summary(W1)` will give you a sense of the distribution of weights.

The lowest weight is 0.007. These are non-smokers that are so unlike the smokers that they don't tell us much about the counterfactual world in which they smoke

The highest weight is 0.887. These are non-smokers that are quite like the smokers so they tell us much about the counterfactual world in which they smoke

The **effective sample size (ESS)** gives a sense of how much useful counterfactual information we have about the control cases.

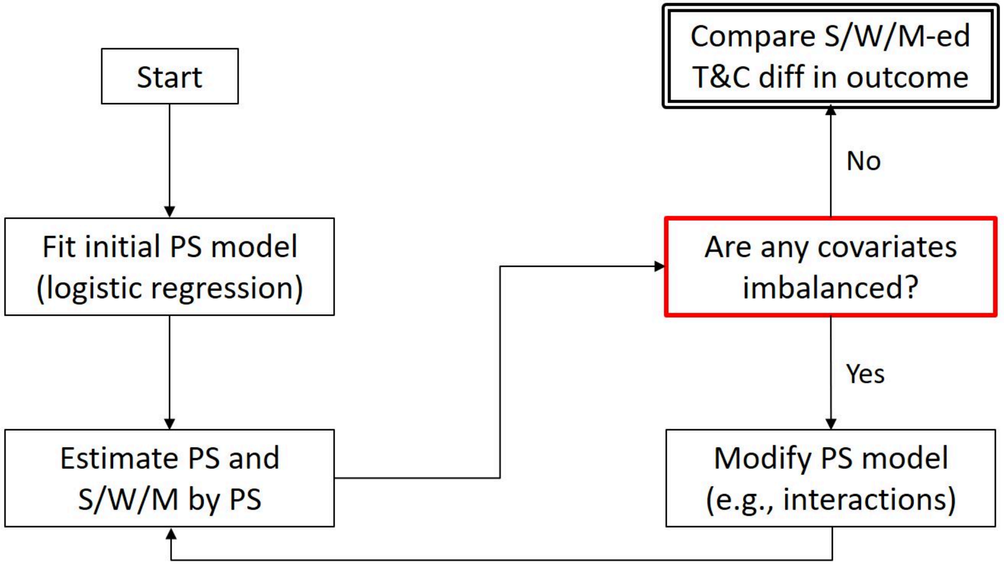
It's calculated as:

$$\Sigma(w)^2/\Sigma(w^2)$$

Weights with low variability are desirable because they improve the precision of the estimator Variability by i) ratio of the largest weight to the smallest in each group, ii) coefficient of variation (standard deviation divided by the mean) of the weights in each group, and iii) effective sample size computed from the weights.

Part 3. – Propensity score (miscellaneous topics)

The problem with the traditional workflow


Start
Fit initial PS model (logistic regression)
Estimate PS and S/W/M by PS
Are any covariates imbalanced?
No
Compare S/W/M-ed T&C diff in outcome
Yes
Modify PS model (e.g., interactions)

The "fit logistic regression" approach relies too much on human balance checking and refitting.
People aren't so good at this.
Modern techniques attempt to take humans out of the process.

Approaches to PS beyond logoistic regression

Covariate balancing propensity scores (CBPS)

- `CBPS` package (integrated in `WeightIt` package), a newer technique that can jointly minimize covariate imbalance while maximizing prediction of treatment selection, e.g. `CBPS::CBPS` or with `weightit( ... , method = "CBPS")` set `estimand` to `ATT` or `ATE`)

Mahalanobis distance matching

- match in multidimensional "covariate space" rather than on a unidimensional PS (available in `matchit`)

Entropy balancing

- find a *single weight* that can be applied to the control cases that will balance *all* the covariates (available in `matchit`)

SuperLearner

- a collection of machine learning and ensembling algorithms implemented in `WeightIt`

The "new approach"

- The traditional approach was simply to stratify, weight, or match, and then compare treatment and control using t-tests (or another difference in means test). That’s mostly what we've done so far.

- These days, many researchers regard matching or weighting as "preprocessing" that makes the parametric regression model you would have estimated anyway less dependent on modeling choices.

- We talked about how matching and weighting unconfound the relationship between T and X by breaking the "backdoor" connection at S. We also talked about how regression unconfounds the relationship by controlling for X. So why not do both at the same time? These are called "doubly robust" estimators of TEs because they give you two chances to get it right.

Basic workflow

1. **Balance treatment and controls** using stratifying, weighting, or matching for the covariates.

2. **Estimate the parametric regression** on the matched or weighted sample. Control for any covariates you think are pre-treatment confounders. *It's OK if they were also used in Step 1!*

3. **Interpret** $\hat{\beta}$ as either (F)SATT or (F)SATE, depending on what approach you used in Step 1.

4. **Resist the temptation** to interpret the coefficients on the control variables. They are *only there* to unconfound the relationship between T and Y. Table 2 fallacy

That's really all there is to it!

- You can extend this to *any* parametric model you would ordinarily use.

- For example, you can use this with logistic regression, count regressions, survival models, etc.

- You can think of stratification, matching, or weight as "getting the data ready" (or **preprocessing**) to estimate the model you would have run anyway.

Multinomial treatments

- Not all treatments are binary. Sometimes there are multiple (3+) treatment arms that we want to compare.

- These types of comparisons are most readily accomplished with *weighting.* The idea is to generate weights that make all the treatment groups have the same distribution of all the covariates. This breaks the association between treatment assignment and all selection variables.

- Because the target distribution is (almost always) the sample average, we are computing an ATE.

- This can also be handled by **weightIt**

Continuous treatments

- Treatments don't always need to be categorical. Sometimes we have a "dose-response" style of treatment that is actually continuous.

- Again, we proceed with ATE weighting because there is no clear "treated" vs. "untreated" distinction.

- The target here is to create weights that make the *Pearson's correlation* between the *treatment* and the *covariates* as close to zero as possible.

- This will break the backdoor path between the treatment and the selection variables.

- This can also be handled by **weightIt**