Propensity score method: a non-parametric technique to reduce model dependence

Zhongheng Zhang

Department of Emergency Medicine, Sir Run-Run Shaw Hospital, Zhejiang University, School of Medicine, Hangzhou 310016, China
Correspondence to: Zhongheng Zhang, MMed. 351#, Mingyue Road, Jinhua 321000, China. Email: zh_zhang1984@hotmail.com.

Author's introduction: Dr. Zhongheng Zhang is a fellow physician working at Sir Run Run Shaw Hospital. He graduated from School of Medicine, Zhejiang University in 2009, receiving Master Degree. His major research interests include hemodynamic monitoring in sepsis and septic shock, delirium, and outcome study for critically ill patients. He is experienced in data management and statistical analysis by using R and STATA, big data exploration, systematic review and meta-analysis. He has published more than 50 academic papers (science citation indexed) that have been cited for over 700 times. He has been appointed as reviewer for 10 journals, including *Journal of Cardiovascular Medicine*, *Hemodialysis International*, *Journal of Translational Medicine*, *Critical Care*, *International Journal of Clinical Practice*, *Journal of Critical Care*.

Zhongheng Zhang, MMed.

Abstract: Propensity score analysis (PSA) is a powerful technique that it balances pretreatment covariates, making the causal effect inference from observational data as reliable as possible. The use of PSA in medical literature has increased exponentially in recent years, and the trend continue to rise. The article introduces rationales behind PSA, followed by illustrating how to perform PSA in R with *MatchIt* package. There are a variety of methods available for PS matching such as nearest neighbors, full matching, exact matching and genetic matching. The task can be easily done by simply assigning a string value to the method argument in the matchit() function. The generic summary() and plot() functions can be applied to an object of class *matchit* to check covariate balance after matching. Furthermore, there is a useful package *PSAgraphics* that contains several graphical functions to check covariate balance between treatment groups across strata. If covariate balance is not achieved, one can modify model specifications or use other techniques such as random forest and recursive partitioning to better represent the underlying structure between pretreatment covariates and treatment assignment. The process can be repeated until the desirable covariate balance is achieved.

Keywords: Propensity score; observational study; logistic regression

Submitted Jun 05, 2016. Accepted for publication Jul 02, 2016.
doi: 10.21037/atm.2016.08.57
View this article at: http://dx.doi.org/10.21037/atm.2016.08.57

Introduction

In clinical researches, an important task is to estimate the causal effect of an intervention on patient-important outcomes. Causal effect can be estimated using randomized controlled trial (RCT), in which both measured and unmeasured confounding factors are balanced in both treatment and control arms. While RCT is the gold standard to assess biological efficacy of an intervention, this design is sometimes not feasible due to ethical problem and/or lack of funding (1,2). In contrast, observational studies utilizing electronic medical records are cheap and easy to access (3). Furthermore, such observational studies are conducted in real world setting and can evaluate the clinical effectiveness of an intervention. There are situations in which an intervention shows biological efficacy in RCTs, but loses its clinical effectiveness in real world setting (4). Therefore, observational studies are widely used in clinical researches in spite of their numerous inherent shortcomings. The major limitation in using observation data to estimate causal effect is the confounding factors. Traditionally, these confounding factors can be adjusted with multivariate models (5-7). However, the distribution of confounding factors may be different between intervention and control groups, and model extrapolation can be wrong (8). Furthermore, regression models have specific assumptions and specifications such as linearity, normal distribution of error term and interaction (9). Frequently, these assumptions are arbitrarily made without empirical evidence. As a result, the causal effect estimated with regression models can vary substantially depending on different specifications and assumptions of the model. This is termed model dependence in the literature (8).

Propensity score method is employed to solve the problem of imbalance in baseline characteristics between intervention and control groups. Initially, the treatment status is used as dependent variable, and regressed on pretreatment covariates with logistic regression model. Propensity score, or the probability of assigning to the treatment, can be calculated with the fitted model. Then propensity score is used for subsequent causal effect inference. Propensity score is considered as nonparametric although parametric regression model is used to estimate propensity score. Other advanced models such as random forest, naïve Bayes and repeated partitioning can be used to estimate propensity score. Propensity matching or stratification is nonparametric. The two-step procedure in causal effect estimation is considered doubly robust by Ho and coworkers in that if either propensity score matching or parametric model is correct, the causal estimates can be consistent (8). The article will show how to perform propensity score analysis (PSA) with R packages (10). Readers may consult other references for detailed mathematical descriptions of PSA (11-13).

Working example

To illustrate PSA using R, I create a dataset including continuous (x.cont) and categorical (x.cat) pretreatment covariates. The functional form between treatment (treat) and covariates includes high order terms and interaction, reflecting the complexity in real world setting. Furthermore, the outcome (y) is regressed on treatment and pretreatment covariates. A total of 1,000 subjects are created.

```
> set.seed(888)
> x.cat <- rep(0:1,c(200,800))
> x.cont <- rnorm(1000)
> lp <- -3 + 2*x.cat*x.cont+5*x.cont^2+3*x.cont-4*x.cat
> link_lp = exp(lp)/(1 + exp(lp))
> treat <- (runif (1000) < link_lp)
> lp.y<-2 + 3*x.cont+2*x.cat+4*treat
> link_y<-exp(lp.y)/(1 + exp(lp.y))
> y<- (runif (1000) < link_y)
> data<-data.frame(treat,x.cat,x.cont,y)
```

PSA with MatchIt package

The *MatchIt* package contains useful functions to perform PSA. The first step is to install the package and load it to the workspace. The package *rgenoud* should also be installed if you want to perform genetic matching.

```
> install.packages("rgenoud")
> install.packages("MatchIt")
```

```
> library(rgenoud)
> library(MatchIt)
> m.out <- matchit(treat ~ x.cat+x.cont, method =
"nearest",discard="both",data = data)
```

The main function matchit() performs PSA. The first argument passes a formula to the function, which defines how the pretreatment covariates influence the treatment assignment. In research practice, investigators usually specify a main effect model without interactions and high-order terms. The *method* argument passes string values including "nearest" (nearest neighbor matching), "exact" (exact matching), "full" (full matching), "genetic" (genetic matching), "optimal" (optimal matching), and "subclass" (subclassification). The default option is the nearest neighbor matching. The discard argument specifies whether to discard observations fall outside of the common support, not allowing them to be used in matching process. The "both" value dictates to discard observations that are outside of the common support in both treatment and control groups.

Balance can be checked with summary() function.

```
> summary(m.out)

Call:
matchit(formula = treat ~ x.cat + x.cont, data = data, method = "nearest",
  discard = "both")

Summary of balance for all data:
          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean eQQ Max
distance         0.5137        0.2175     0.1418    0.2963  0.3284   0.3019  0.4488
x.cat            0.6440        0.8698     0.3368   -0.2257  0.0000   0.2233  1.0000
x.cont           0.6084       -0.2543     0.5965    0.8627  1.2080   1.1241  1.9759

Summary of balance for matched data:
          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean eQQ Max
distance         0.4727        0.3643     0.1459    0.1084  0.1276   0.1084  0.1879
x.cat            0.7342        0.6261     0.4849    0.1081  0.0000   0.1081  1.0000
x.cont           0.6391        0.1160     0.6099    0.5231  0.7374   0.7336  1.1489

Percent Balance Improvement:
          Mean Diff. eQQ Med eQQ Mean eQQ Max
distance      63.4146  61.1436  64.0798  58.1412
x.cat         52.1097   0.0000  51.5864   0.0000
x.cont        39.3634  38.9554  34.7366  41.8539

Sample sizes:
              Control Treated
All               691     309
Matched           222     222
Unmatched         469       0
Discarded           0      87
```

The output of generic function summary() is important in assessing balance after PSA. The "Means Treated" and "Means Control" columns show the weighted means for the treated and control groups. "SD Control" is the standard deviation for the control group. "Mean Diff" is the mean difference between control and treated groups. The last three columns show the median, mean and maximum distance between empirical quantile functions of the treated and control groups. A value greater than 0 indicates deviations between the two groups in some part of quantile distributions. The last table shows the number of observations that have been matched or discarded. These statistics can be visualized with generic plot() function.

```
> plot(m.out,type="jitter")
> plot(m.out,type="hist")
> plot(m.out)
```

Figure 1 is a jittered plot showing matched and unmatched observations, as well as their distribution on

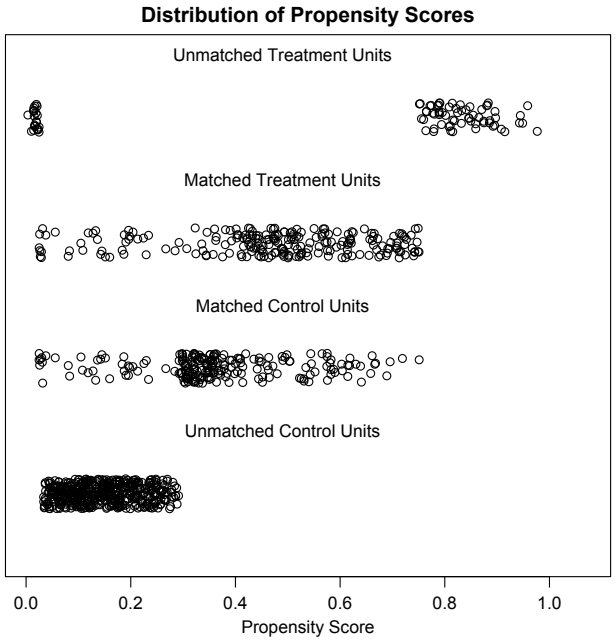

Figure 1 Jittered plot showing matched and unmatched observations, as well as their distribution on propensity score values. It appears that many observations with high propensity scores in the treated group and many with low propensity scores in the control group are excluded.

propensity score values. It appears that many observations with high PS in the treated group and many with low PS in the control group are excluded. *Figure 2* are histograms showing the density of PS distribution in the treated and control groups before and after matching. Before matching (raw) treated groups have significantly higher PS than the control group. After matching the density distributions of the two groups become somewhat similar. Quantile-quantile (QQ) plot compares the probability distributions of a given covariate for the treated and control groups by plotting their quantiles against each other. The points on the QQ plot will lie on the y=x line if two distributions are similar. The results show that although the points are not located on the y=x line exactly after matching, it is much improved as compared to law data (*Figure 3*).

Checking balance with PSAgraphics package

The *PSAgraphics* package provides several functions for evaluating balance of covariates in each stratum (14). So in this section, I first create strata containing treated and control groups, trying to balance covariates between both groups.

```
> m.out.strata<-matchit(treat ~ x.cat+x.cont, method =
"subclass",discard="both",data = data)
```

In the above example, subclassification is used to form strata in which the distribution of covariates in treated and control groups are as similar as possible.

```
> m.data.strata<-match.data(m.out.strata)
> str(m.data.strata)
'data.frame':        913 obs. of 7 variables:
 $ treat : logi TRUE TRUE TRUE FALSE TRUE FALSE ...
 $ x.cat : int 0 0 0 0 0 0 0 0 0 0 ...
 $ x.cont : num -1.951 -1.544 0.73 -0.278 -1.656 ...
 $ y   : logi TRUE TRUE TRUE TRUE TRUE FALSE ...
 $ distance: num 0.105 0.16 0.742 0.464 0.143 ...
 $ weights : num 1 1 1 8.82 1 ...
 $ subclass: num 1 1 5 3 1 3 1 5 5 4 ...
```

Matched dataset can be stored to an object using match.data() function. In addition to original variables, three variables *distance*, *weights* and *subclass* were added. The subclass variable denoted the stratum to which an observation belongs. Next I will install the *PSAgraphics* package and proceed to examine covariate balance within each stratum.

```
> install.packages("PSAgraphics")
> library(rpart)
> library(PSAgraphics)
> box.psa(m.data$x.cont, m.data$treat,
m.data$subclass, xlab = "Strata",ylab = "Covariate:
x.cont", balance = TRUE)
Press <enter> for bar chart...
```

The box.psa() function depicts a pair of side-by-side boxplots for each stratum to compare the difference between treated and control groups on a given covariate. If balance=TRUE, it calls bal.ms.psa() function to draw a histogram of a permutation distribution and reference statistic to assess balance across strata (*Figure 4*). Balance statistic is defined as:

$$\hat{\delta}_{\alpha} = \sum_{k=1}^{k} |\hat{\mu}_{0k} - \mu_{1k}|, \quad [1]$$

where $\hat{\delta}_{\alpha}$ is the balance statistic, the subscript α is to denote a particular subclassification scheme, K is the total number of stratum, and $\hat{\mu}_{0k}$ and $\hat{\mu}_{1k}$ are mean values of the control and treated group within stratum k. Note that smaller value of the balance statistic indicates a better balance on that given covariate. The histogram is drawn by randomly assigning observations to strata and in our example it generated 1,000

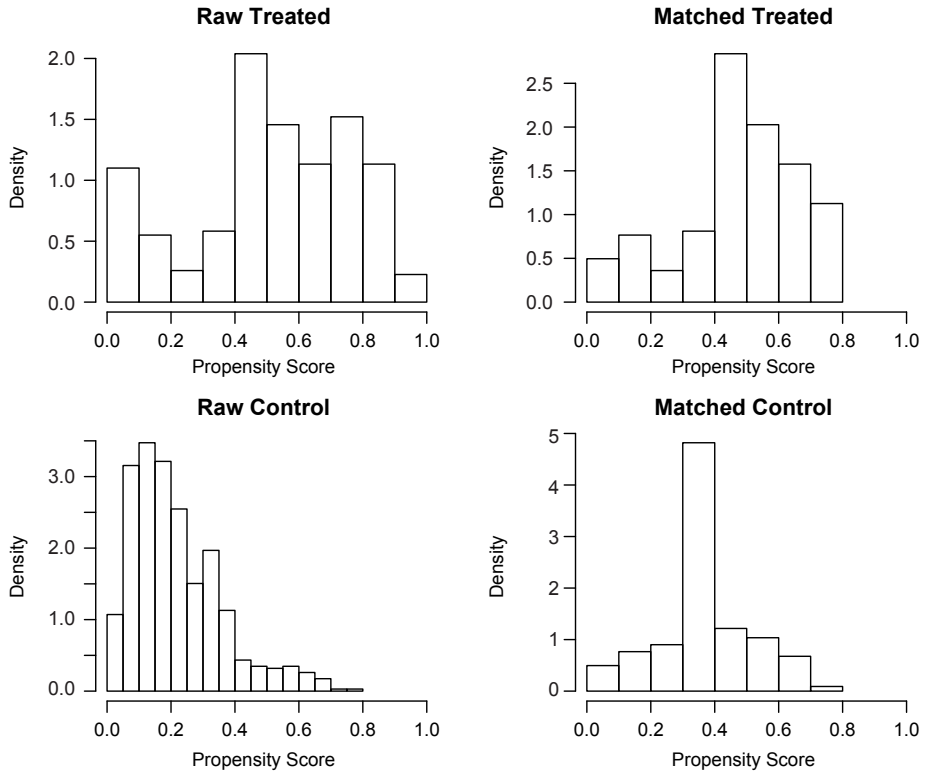

Figure 2 Histograms showing the density of propensity score distribution in the treated and control groups before and after matching.

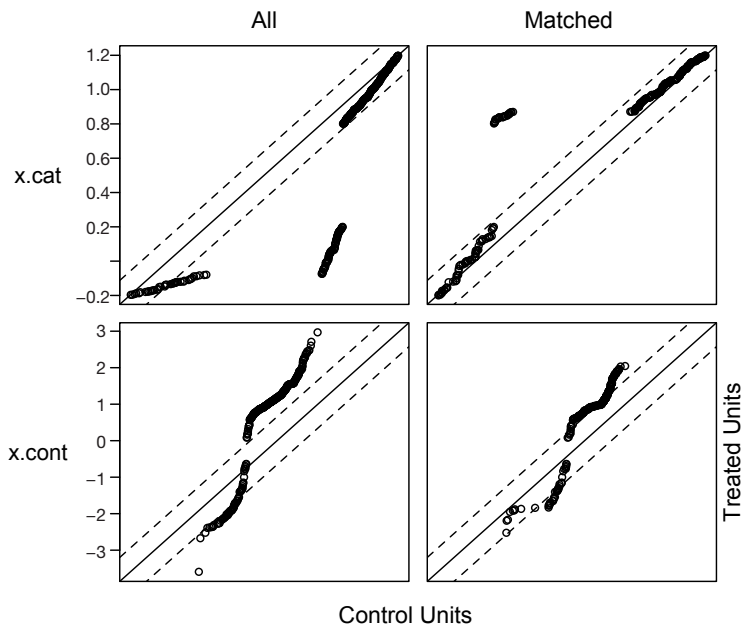

Figure 3 Quantile-quantile (QQ) plot compares the probability distributions of the treated and control groups on a given covariate by plotting their quantiles against each other. The results show that although the points are not located on the y=x line exactly after matching, it is much improved as compared to law data.

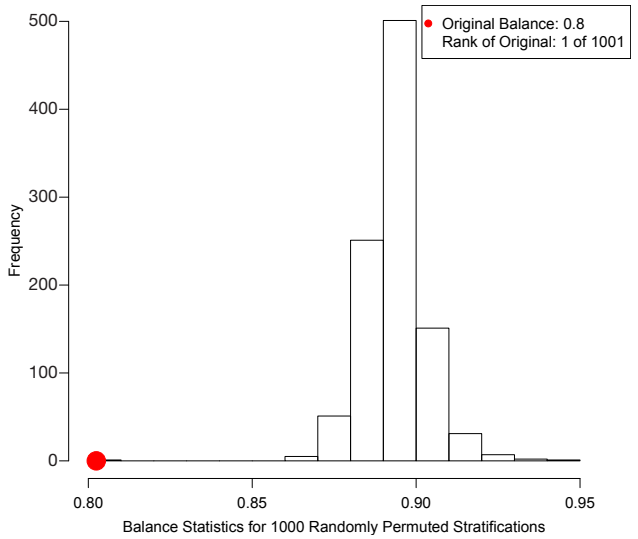

Figure 4 Histogram of a permutation distribution and reference statistic to assess balance across strata. The balance statistic locates at the left end of the mass of permutation distribution, indicating a good balance.

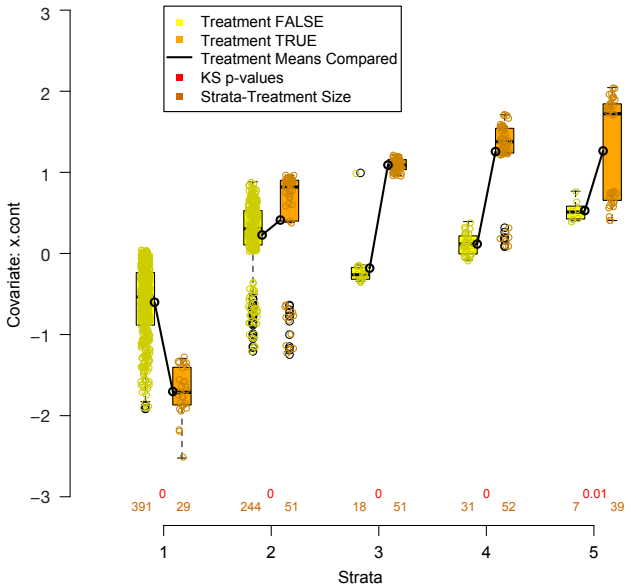

Figure 5 Side-by-side boxplots, 5 strata, for covariate x.cont produced by box.pdf.

balance statistics. By comparing our original balance ($\hat{\delta}_\alpha$) to the mass of permutation distribution, we conclude that the subclassification method has balanced the covariate x.cont as much as possible (e.g., it is the smallest among all randomly generated balance statistics).

By pressing enter as indicated by output message, there pops up a series of boxplots comparing the difference on *x.cont* between treated and control groups within each stratum. Means of the two groups are connected by a heavy solid line, with the slope of the line indicate size of the difference. *Figure 5* shows that the balance of *x.cont* within each stratum is unsatisfactory and the distribution changes moderately across strata. The sizes (number of observations) of groups are printed below corresponding boxplots. The "balance=TRUE" argument adds Kologmorov-Smirnov p-values to the graph for the test of equivalence of control/treatment distributions within each stratum.

PSAgraphics package has special function cat.psa() for balance check of categorical variable. The argument of the function is similar to box.psa() function as described above.

The function produces side-by-side barplots comparing proportion of cases in each category (*Figure 6*). The sizes of treatment groups within strata are printed at the top of the bars. It is noted that the subclassification method generates a poorly matched strata. Along with the barplot, cat.psa() function also produces a cross-tabulation between treatment and categorical covariate across strata.

It may be interesting to compare outcomes between treatment groups across strata. The following function circ.psa() is designed for this purpose. While the R code to draw circles is simple, the key lessons are how to interpret the output plot.

```
> circ.psa(m.data$y,m.data$treat,m.data$subclass, revc
= TRUE,xlab = "Treatment", ylab = "Control")
$summary.strata
```

	n.FALSE	n.TRUE	means.FALSE	means.TRUE
1	391	29	0.8286445	0.7931034
2	244	51	0.9180328	1.0000000
3	18	51	0.8333333	1.0000000

```
> cat.psa(m.data$x.cat,m.data$treat, m.data$subclass, xlab = "Strata",ylab = "Proportion for 'x.cat'",barnames =
c("Control", "Treatment"), rtmar = 2)
$`treatment:stratum.proportions`
```

	FALSE:1	TRUE:1	FALSE:2	TRUE:2	FALSE:3	TRUE:3	FALSE:4	TRUE:4	FALSE:5	TRUE:5
0	0	0.724	0.143	0.235	0.944	0	1	0.135	1	0.487
1	1	0.276	0.857	0.765	0.056	1	0	0.865	0	0.513

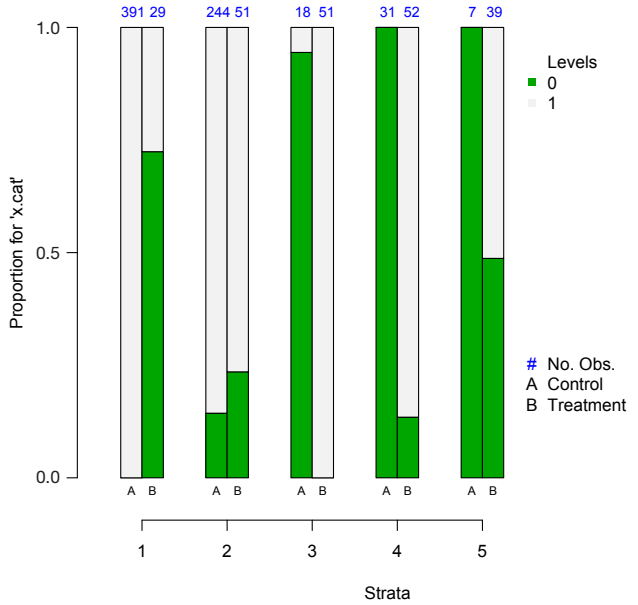

Figure 6 Side-by-side barplots comparing proportion of cases in each category for variable x.cat.

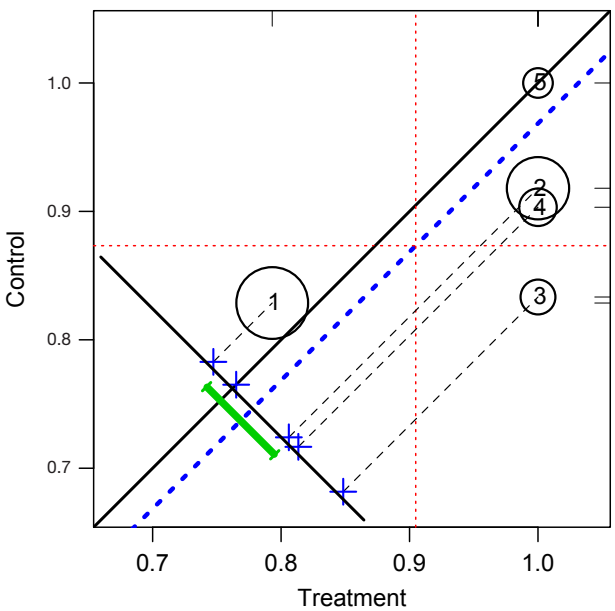

Figure 7 Propensity score analysis assessment plot of outcome variable y, 5 strata, constructed using circ.psa.

```
4       31      52      0.9032258       1.0000000
5       7       39      1.0000000       1.0000000

$wtd.Mn.TRUE
[1] 0.9048231
$wtd.Mn.FALSE
[1] 0.8732947
$ATE
[1] -0.03152831
$se.wtd
[1] 0.02654633
$approx.t
[1] -1.187671
$df
[1] 903
$CI.95
[1] -0.08362798 0.02057137
```

In *Figure* 7, stratum is represented by a circle with the circle size proportional to the number of observations in each stratum. The number within each circle is the stratum number. Because the outcome variable y is binary denoted by 0 and 1, the mean of y in each treatment group is equal to the proportion of outcome events. The center of circle projecting to the x-axis corresponds the outcome means for the treated group, and that projecting to the y-axis is the outcome means for the control group. Circles below the solid identity line (y=x) are those with treatment effect larger than control group, and vise versa. The dashed blue line parallel to the identity line is the mean difference of the outcome. The cross symbols represent the distribution of strata difference. The horizontal and vertical dashed lines represent the (weighted) means for the treated and control groups respectively. Rug plots on vertical and horizontal sides of the graph show marginal distributions of control and treatment outcome measures. Along with the graph, circ.psa() output a summary statistics for the means on outcome variables in treatment groups across strata. The weighted means for each treatment group are given under objects "wtd.Mn.TRUE" and "wtd. Mn.FALSE". The following objects "ATE", "se.wtd", "approx.t", "df" and "CI.95" respectively represent average treatment effect, weighted standard error (15), approximate t statistics, degree of freedom and 95% confidence interval for the direct adjustment estimator (shown as the heavy green line in *Figure* 7).

Since the above methods indicate that the balance was not achieved by the matching method, one needs to modify model specifications in calculating logistic-regression based propensity scores. Higher order terms and interactions can be added as follows.

```
> m.out.right <- matchit(treat ~ x.cat*x.cont+x.cont^2,
method = "nearest",discard="both",data = data)
```

The summary output of the object *m.out.right* is omitted to save space. The percent balance improvement of distance is approximately 80%, which is significantly greater than 60% in the original matching. In practice, the matching process can be repeated until the model with best balance statistics is obtained.

Effect estimation after PSA

Any parametric analyses can be performed on the matched dataset obtained with PSA. The procedures are the same to that would have been used without PSA. This is left to readers for practices. Also they can consult the excellent tutorial written by Ho and colleagues on how to perform analysis after matching using *Zelig* package (16).

Acknowledgements

None.

Footnote

Conflicts of Interest: The author has no conflicts of interest to declare.

References

1. Albert RK. "Lies, damned lies ..." and observational studies in comparative effectiveness research. Am J Respir Crit Care Med 2013;187:1173-7.
2. Zhang Z. Big data and clinical research: perspective from a clinician. J Thorac Dis 2014;6:1659-64.
3. Zhang Z. Big data and clinical research: focusing on the area of critical care medicine in mainland China. Quant Imaging Med Surg 2014;4:426-9.
4. Singal AG, Higgins PD, Waljee AK. A primer on effectiveness and efficacy trials. Clin Transl Gastroenterol 2014;5:e45.
5. Yeh RW, Mauri L. Choosing methods to minimize confounding in observational studies: do the ends justify the means? Circ Cardiovasc Qual Outcomes 2011;4:581-3.
6. Quartey G, Feudjo-Tepie M, Wang J, et al. Opportunities for minimization of confounding in observational research. Pharm Stat 2011;10:539-47.
7. Pfeiffer RM, Riedl R. On the use and misuse of scalar scores of confounders in design and analysis of observational studies. Stat Med 2015;34:2618-35.
8. Ho DE, Imai K, King G, et al. Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. Political Analysis 2007;15:199-236.
9. Zhang Z. Multivariable fractional polynomial method for regression model. Ann Transl Med 2016;4:174.
10. Keller B, Tipton E. Propensity score analysis in R: a software review. Journal of Educational and Behavioral Statistics 2016;41:326-48.
11. Patorno E, Grotta A, Bellocco R, et al. Propensity score methodology for confounding control in health care utilization databases. Epidemiology, Biostatistics and Public Health 2013;10:e8940,1-16.
12. Li M. Using the propensity score method to estimate causal effects: a review and practical guide. Organizational Research Methods 2013;16:188-226.
13. Little RJ, Rubin DB. Causal effects in clinical and epidemiological studies via potential outcomes: concepts and analytical approaches. Annu Rev Public Health 2000;21:121-45.
14. Helmreich JE, Pruzek RM. PSAgraphics: an RPackage to support propensity score analysis. Journal of Statistical Software 2009;29:1-23.
15. Conniffe D. Evaluating state programmes: "Natural experiments" and propensity scores. Economic and Social Review 2000;31:283-308.
16. Ho D, Imai K, King G, et al. MatchIt: nonparametric preprocessing for parametric causal inference. Journal of Statistical Software 2011;42:1-28.

Cite this article as: Zhang Z. Propensity score method: a non-parametric technique to reduce model dependence. Ann Transl Med 2017;5(1):7. doi: 10.21037/atm.2016.08.57

