Optmatch: Flexible, Optimal Matching for Observational Studies

Ben B. Hansen

Observational studies compare subjects who received a specified treatment to others who did not, without controlling assignment to treatment and comparison groups. When the groups differ at baseline in ways that are relevant to the outcome, the study has to adjust for the differences. An old and particularly direct method of making these adjustments is to match treated subjects to controls who are similar in terms of their pretreatment characteristics, then conduct an outcome analysis conditioning upon the matched sets. Adjustments of this type enjoy properties of robustness (Rubin, 1979) and transparency not shared with purely model-based adjustments, such as covariance adjustment without matching or stratification; and with the introduction of propensity scores to matching (Rosenbaum and Rubin, 1985), the approach was shown to be more broadly applicable than was previously thought. Arguably, the reach of techniques based on matching now exceeds that of purely model-based adjustment (Hansen, 2004).

To achieve these benefits, matched adjustment requires the analyst to articulate a distinction between desirable and undesirable potential matches, and then to match treated and control subjects in such a way as to favor the more desirable pairings. Propensity scoring fits under the first of these tasks, as do the construction of Mahalanobis matching metrics (Rosenbaum and Rubin, 1985), prognostic scoring (Hansen, 2006b), and the distance metric optimization of Diamond and Sekhon (2006). The second task, matching itself, is less statistical in nature, but doing it well can substantially improve the power and robustness of matched inference (Hansen and Klopfer, 2006; Hansen, 2004). The main purpose of **optmatch** is to relieve the analyst of responsibility for this important, if potentially tedious, undertaking, freeing attention for other aspects of the analysis. Given discrepancies between each treatment and control subject that might potentially be matched, **optmatch** places them into non-overlapping matched sets, in the process solving the discrete optimization problems needed to make sums of matched discrepancies as small as possible; after this, the analysis can proceed using permutation inference (Rosenbaum, 2002; Hothorn et al., 2006; Bowers and Hansen, 2006), conditional inference (Breslow and Day, 1980; Cox and Snell, 1989; Hansen, 2004; Lumley and Therneau, 2006), approximately conditional inference (Pierce and Peters, 1992; Brazzale, 2005; Brazzale et al., 2006), or multilevel models (Smith, 1997; Raudenbush and Bryk, 2002; Gelman and Hill, 2006).

Optimal matching of two groups

To illustrate the meaning of optimal matching, consider Cox and Snell's (1981, p.81) study of costs of nuclear power. Of 26 light water reactor plants constructed in the U.S. between 1967 and 1972, seven had been built on the site of existing plants. The problem is to estimate the cost benefit (or penalty) of building on an existing site as opposed to a new one. A matched analysis seeks to adjust for background characteristics determinative of cost, such as the date of construction and the capacity of the plant, by linking similar refurbished and new plants: plants of about the same capacity and constructed at about the same time, for example. To highlight the analogy with intervention studies, I refer to existing-site plants as "treatments" and new-site plants as "controls."

Consider the problem of arranging the plants in disjoint triples, each containing one treatment and two controls, placing each treatment and 14 of the 19 controls into some matched triple or another. A straightforward way to create such a match is to move down the list of treatments, pairing each to the two most similar controls that have not yet been matched; this is *nearest-available matching*. Figure 1 shows the 26 plants, their capacities and dates of construction, and a 1 : 2 matching constructed in this way. First A was matched to I and J, then B to L and N, and so forth. This example is discussed by Rosenbaum (2002, ch.10).

Existing site

	date	capacity
A	2.3	660
B	3.0	660
C	3.4	420
D	3.4	130
E	3.9	650
F	5.9	430
G	5.1	420

New site

	date	capacity
H	3.6	290
I	2.3	660
J	3.0	660
K	2.9	110
L	3.2	420
M	3.4	60
N	3.3	390
O	3.6	160
P	3.8	390
Q	3.4	130
R	3.9	650
S	3.9	450
T	3.4	380
U	4.5	440
V	4.2	690
W	3.8	510
X	4.7	390
Y	5.4	140
Z	6.1	730

"date" is date of construction, in years after 1965; "capacity" is net capacity of the power plant, in MWe above 400.

Figure 1: 1:2 matching by a nearest-available algorithm.

How might this process be improved? To complete step i, the nearest-available algorithm requires

a ranking of potential matches for treatment unit i, an ordering of available controls accounting for their differences with plant i in generating capacity and in year of construction. Typically controls j are ordered in terms of a numeric discrepancy, $d[i, j]$, from i; Figure 1's match follows Rosenbaum (2002, ch.10) in using the sum of rank differences on the two covariates (after restricting to a subset of the plants, `pt!=1`):

```
> data("nuclear", package="boot")
> attach(nuclear[nuclear$pt!=1,])
> drk <- rank(date)
> d <- outer(drk[pr==1], drk[pr!=1], "-")
> d <- abs(d)
> crk <- rank(cap)
> d <- d +
    abs(outer(crk[pr==1], crk[pr!=1], "-"))
```

(where `pr==1` indicates the treatment group). The d that results from these operations is shown (after rounding) in Figure 3. Having calculated this d, one can pose the task of matching as a discrete optimization problem: find the match $M = \{(i, j)\}$ minimizing $\sum_M d(i, j)$ among all sets of pairs (i, j) in which each treatment i appears twice and each control j appears at most once.

Optimal matching refers to algorithms guaranteed to find matches attaining this minimum, or falling within a specified tolerance of it, given a $n_t \times n_c$ discrepancy matrix M. Optimal matching's performance advantage over heuristic, non-optimal algorithms can be striking. For example, in the problem of Figure 1 optimal matching reduces nearest-available's sum of discrepancies by 23%. This optimal solution, found by **optmatch**'s `pairmatch` function, is shown in Figure 2.

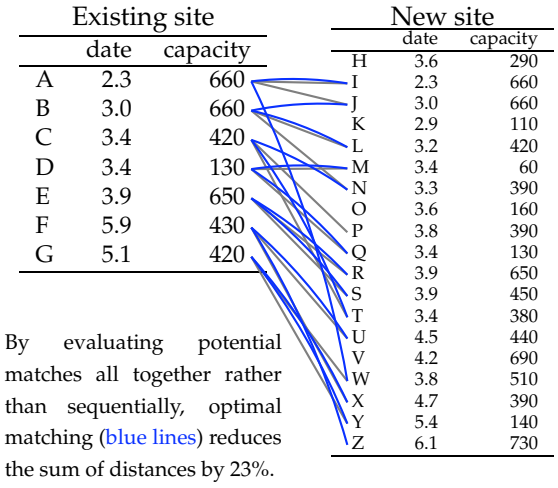

Figure 2: Optimal vs. greedy 1:2 matching.

An optimal match is optimal relative to given structural requirements, here that all treatments and a corresponding number of controls be arranged in 1 : 2 matched sets, and a given distance, here d. It is best-possible for the purposes of the analyst only insofar as the given distance and structural stipulations best represent the analyst's goals; in this sense "optimal" in "optimal matching" is analogous to the "maximum" of "maximum likelihood," which is never better than the chosen likelihood model it maximizes.

For example, in the problem just discussed the structural stipulation that the matched sets all be 1:2 triples may be poorly tailored to the goal of reducing baseline differences between the groups. It is appropriate if these differences stem entirely from a small minority of controls being too unlike any treatment subjects to bear comparison to them, since it does exclude 5/19 of potential controls from the final match; but differences on a larger number of controls, even small differences, require the techniques to be described under Generalizations of pair matching, below, which may also give similar or better bias reduction without excluding as many control observations. (See also Discussion, below.)

Growing your own discrepancy matrix

Figures 1 and 2 both illustrate *multivariate distance matching*, aligning units so as to minimize a sum of rank discrepancies. **Optmatch** is entirely flexible about the form of the distance on which matches are to be made. To propensity-score match nuclear plants, for example, one would prepare a propensity distance using

```
> pscr <- glm(pr ~ . -(pr+cost),
    family = binomial,
    data = nuclear)$linear.predictors
> PR <- nuclear$pr==1
> pdist <- outer(pscr[PR], pscr[PR], "-")

> pscr.v <- (var(pscr[PR])*(sum(PR)-1)+
    var(pscr[!PR])*(sum(!PR)-1))/
    (length(PR)-2)
> pdist <- abs(pdist)/sqrt(pscr.v)
```

or, more simply and reliably,

```
> pmodel <- glm(pr ~ . -(pr+cost),
    family = binomial, data = nuclear)
> pdist <- pscore.dist(pmodel)
```

Then `pdist` is passed to `pairmatch` or `fullmatch` as its first argument. Other discrepancies on which one might match include Mahalanobis distances (which can be produced using `mahal.dist`) and combinations of Mahalanobis and propensity-based distances (Rosenbaum and Rubin, 1985; Gu and Rosenbaum, 1993; Rubin and Thomas, 2000). Many special requirements, such as that matches be made only within given subclasses, or that specific matches be avoided, are also introduced through the discrepancy matrix.

First consider the case the matches are to be made within subclasses only. In the nuclear dataset, plants

with and without partial turnkey (pt) guarantees should be compared separately, since the meaning of the outcome variable, cost, changes with pt. Figure 2 shows only the pt!=1 plants, and its match is generated with the command pairmatch(d,controls=2), where d is the matrix in Figure 3. To match also partial turnkey plants to each other, one gathers into a list, dl, both d and a distance matrix dpt comparing new- and existing-site partial turnkey plants, then feeds dl to pairmatch as its first argument. For propensity or Mahalanobis matching, pscore.dist or mahal.dist would do this if given the formula pr~pt as their optional arguments 'structure.fmla'.

More generally, the helper function makedist (which is called by pscore.dist and mahal.dist) eases the application of a (user-defined) discrepancy matrix-producing function to each of a sequence of strata in order to generate a list of distance matrices. For separate summed-rank distances by pt subclass, one would write a function that extracts portions of relevant variables from a given data frame, looking to a treatment-group variable to decide what portions to take, as in

```
> capdatediffs <- function(trt, dat) {
    crk <- rank(dat[names(trt),"cap"])
    names(crk) <- names(trt)
    dmt <- outer(crk[trt], crk[!trt],"-")
    dmt <- abs(dmt)

    drk <- rank(dat[names(trt),"date"])
    dmt <- dmt +
    abs(outer(drk[trt], drk[!trt],"-"))
    dmt
    }
```

Then one would use makedist to apply the function separately within levels of pr:

```
> dl <- makedist(pr ~ pt, nuclear,
    capdatediffs)
```

The result of this is a list of two distance matrices, both submatrices of d created above, one comparing pt!=1 treatments and controls and a smaller one for pt==1 plants.

In larger problems, matching can be substantially faster if preceded by a division of the sample into subclasses; see Under the hood, below. The use of pscore.dist, mahal.dist, and makedist carry another advantage, that the lists of distances they generate carry metadata to prevent fullmatch or pairmatch from getting confused about the order of observations in the data frame from which the distances were generated.

Another common aim is to forbid unwanted matches. With **optmatch**, this is done by placing NA's, NaN's or Inf's at the relevant places in a distance matrix. Consider matching nuclear plants within *calipers* of three years on date of construction. Pairings of plants that would violate this requirement are indicated in red in Figure 3. To enforce the caliper, one could generate a matrix of discrepancies dy on year of construction, then replace the distance matrix of Figure 3, d, with d/(dy<=3); this new matrix has an Inf at each entry in Figure 3 currently shown in red, and otherwise is the same as in Figure 3.

Operations of these types, division and logical comparison, are compatible with subclassification prior to matching, despite the fact that the operations seem to require matrices while subclassification demands lists of matrices. Assuming one has defined a function datediffs as

```
> datediffs <- function(trt,data){
    sclr <- data[names(trt), 'date']
    names(sclr) <- names(trt)
    abs(outer(sclr[trt], sclr[!trt], '-'))
    }
```

then the command

```
> dly <- makedist(pr ~ pt, nuclear,
    datediffs)
```

tabulates absolute differences on date of construction, separately for pr==1 and pr!=1 strata. With **optmatch**, the expression dly<=3 returns a list of indicators of whether potential matches were built within three years of one another. Furthermore, dl/(dly<=3) is a list imposing the three-year caliper upon distances coded in *dl*. To pair match on propensity scores, but with a 3-year date-of-construction caliper, one would use pairmatch(dl/(dly<=3)).

Generalizations of pair matching

Matching with a varying number of controls

In Figures 1 and 2, both non-optimal and optimal matches insist on precisely two controls per treatment. If one's aim is to match as closely as possible, this is a limitation. To optimally match 14 of the 19 controls, as Figure 2 does, but without requiring that they always be matched two-to-one to treatments, one would use the command fullmatch, with options 'min.controls=1' and 'omit.fraction=5/19'. The flexibility this adds improves matching even more than the switch from greedy to optimal matching did; while optimal pair matching reduced greedy pair matching's net discrepancy from 82 to 63, optimal matching with a varying number of controls brings it to 44, just over half its original value.

If *mvnc* is the match created in this way, then the structure of *mvnc* is returned by

```
> stratumStructure(mvnc)
stratum treatment:control ratios
1:1 1:2 1:3 1:5
  4   1   1   1
```

Existing	New sites																		
	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	28	0	3	22	14	30	17	28	26	28	20	22	23	26	21	18	34	40	28
B	24	3	0	22	10	27	14	26	24	24	16	19	20	23	18	16	31	37	25
C	10	18	14	18	4	12	6	11	9	10	14	12	6	14	22	10	16	22	28
D	7	28	24	8	14	2	10	6	12	0	24	22	4	24	32	20	18	16	38
E	17	20	16	32	18	26	20	18	12	24	0	2	20	6	8	4	14	20	14
F	20	31	28	35	20	29	22	20	14	26	12	9	22	5	15	12	9	11	12
G	14	32	29	30	18	24	17	16	10	22	12	10	17	6	16	14	4	8	17

Figure 3: Rank discrepancies of new- and existing-site nuclear plants without partial turnkey guarantees. New- and existing-site plants which differ by more than 3 years in date of construction are indicated in red.

This means it consists of four matched pairs and a matched triple, quadruple, and sextuple, all containing precisely one treatment. Ming and Rosenbaum (2000) discuss matching with a varying number of controls, implementing it in their example with a somewhat different algorithm.

Full matching

A propensity score is the conditional probability, $p(\mathbf{x})$, of falling in the treatment group given covariates $\mathbf{x}$ (or an increasing transformation of it, such as its logit). Because subjects with large propensity scores more frequently fall in the treatment group, and subjects with low propensity scores are more frequently controls, propensity score matching is fundamentally at odds with matching treatments and controls in fixed ratios, such as $1:1$ or $1:k$. These ratios must be allowed to adapt, so that 1:1 matches can be made where $p(\mathbf{x})/(1-p(\mathbf{x})) \approx 1$ while $1:k$ matches are made where $p(\mathbf{x})/(1-p(\mathbf{x})) \approx 1/k$; otherwise either some subjects will have to go unmatched or some subjects are bound to be poorly matched on their propensity scores. Matching with multiple controls addresses part of this problem, but only *full matching* (Rosenbaum, 1991) addresses it in its entirety, by also permitting l:1 matches, for when $p(\mathbf{x})/(1-p(\mathbf{x})) \approx l \geq 2$.

In general, full matching is useful when there are some regions of covariate space in which controls outnumber treatments but others in which treatments outnumber controls. This pattern emerges most clearly when matching on a propensity score, but they influence the quality of matches even without propensity scores. The rank discrepancies of new- and existing-site `pt==1` plants, shown in Figure 4, show it; earlier dates of construction, and smaller capacities, are more common among controls (d and e) than treatments (b only), and this is reflected in Figure 4's sums of discrepancies on rank. As a consequence, full matching achieves a net rank discrepancy (3) that is half of the minimum possible (6) with techniques that don't permit both 1:2 and 2:1 matched sets.

Existing	New sites		
	d	e	f
a	6	6	[0]
b	[0]	[3]	6
c	6	6	[0]

Figure 4: Rank discrepancies of new- and existing-site nuclear plants with partial turnkey guarantees. Boxes indicate the optimal full matching of these plants.

Gu and Rosenbaum (1993) compare full and other forms of matching in an extensive simulation study, while Hansen (2004) and Hansen and Klopfer (2006) present applications. This literature emphasizes the importance of using *structural restrictions*, upper limits on K in $K:1$ matched sets and on L in $1:L$ matched sets, when full matching, in order to control the variance of matched estimation. With `fullmatch`, an upper limit K:1 on treatment:control ratios is conveyed using '`min.controls=1/K`', while a lower limit of $1:L$ on the treatment : control ratio would be given with '`max.controls=L`'. Hansen (2004) and Hansen and Klopfer (2006) give strategies to optimize these tuning parameters. In the context of a specific application, Hansen (2004) finds $(\text{min.controls}, \text{max.controls}) = (1/2, 2) \cdot (1-\hat{p})/\hat{p}$ to work best, where $\hat{p}$ represent the proportion treated in the stratum being matched. In an unrelated application, Stuart and Green (2006) find these values to work well; they may be a good starting point for general use.

A somewhat related technique is matching "with replacement," in which overlap between matched sets is permitted in the interests of achieving closer matches. Because of the overlap, methods appropriate to stratified data are not generally appropriate for samples matched with replacement. The technique forces one to resort to specialized techniques, such as those of Abadie and Imbens (2006). On the other hand, with-replacement matching would appear to offer the possibility of closer matches, since its pairing of one treatment unit in no way limits its pairing of the next treatment unit.

However, it is a surprising, and evidently

little-known, fact that with-replacement matching achieves no closer matches than full matching, a without-replacement matching technique. As discussed by Rosenbaum (1991) and (more explicitly) by Hansen and Klopfer (2006), given any criterion for a potential pairing of subjects to be acceptable, full matching matches *all* subjects with at least one suitable match in their comparison group, matching them *only* to acceptable counterparts. So one might insist, in particular, that each treatment unit be matched only to one of its nearest neighbors; by sharing controls among treated units where necessary, omitting controls who are not the nearest neighbor of some treatment, and matching to multiple controls where that can be done, full matching finds a way to honor this requirement. Since the matched sets produced by full matching never overlap, it has the advantage over with-replacement matching of combining with any method of estimation appropriate to finely stratified data.

Under the hood

Hansen and Klopfer (2006) describe the network-flows algorithm on which **optmatch** relies in some detail, establishing its optimality for full matching and matching with a fixed or varying number of controls. They also give upper bounds for the time complexity of the algorithm: roughly, $O(n^3 \log(nC))$, where n is the size of the sample and C is the quotient of largest discrepancy in the distance matrix and the matching tolerance. This is comparable to the time complexity of squaring a $n \times n$ matrix. More precisely, the algorithm requires $O(nn_t n_c \log(nC))$ floating-point operations, where n_t and n_c are the sizes of the treatment and control groups.

These bounds have two practical consequences for **optmatch**. First, computational costs grow steeply with the size of the discrepancy matrix. Just as squaring two $(n/2) \times (n/2)$ submatrices of an $n \times n$ matrix is about four times faster than squaring the full $n \times n$ matrix, matching is made much faster by subdividing large matching problems into smaller ones. For this reason `makedist` is written so as to facilitate subclassification prior to matching, the effect of which is to split larger matching problems into a sequence of smaller ones. Second, the C-factor contributes secondarily to computational cost; its contribution is reduced by increasing the value of the '`tol`' argument to `fullmatch` or `pairmatch`.

Discussion

When and how matching reduces systematic differences between groups

Matching can address bias in observational studies in either of two ways. In *matched sampling*, it is used to select a subset of control subjects most like treatments, with the remainder of subjects excluded from analysis; in *matched adjustment*, it is used to force treatment control comparisons to be based on individualized comparisons made within matched sets, which will have been so engineered that matched counterparts are more like one another than are treatments and controls on the whole. Matched sampling is typically followed by matched adjustment, but matched adjustment can be useful even when not preceded by matched sampling.

Because it excludes some five control subjects, the match depicted in Figures 1 and 2 might be used in a context of matched sampling, although it differs in important respects from typical matched samples. In archetypal cases, matched sampling is used when for cost reasons the number of controls to be followed up for outcome data has to be reduced anyway, not when outcome data is already available for the entire sample already; and in archetypal cases, the reservoir of potential controls is many times larger than the size of the desired control group. See, *e.g.*, Althauser and Rubin (1970) or Rosenbaum and Rubin (1985). The matches in Figures 1 and 2 are typical of matched sampling in matching a fixed number of controls to each treatment subject. When bias can be addressed by being very selective in the choice of controls, flexibility in the structure of matched sets becomes less important.

When there is no additional data to be collected, there may be little use for matched sampling per se, while matched adjustment may still be attractive. In these cases, it is important to recognize that matching, even optimal matching, does not in itself reduce systematic differences between treatment and control groups unless it is specifically given the flexibility to do so. Suppose, for instance, that adjustment for the variable `t2` is needed in the comparison of new- and existing site-plants. This variable, which represents the time between the issue of an operating permit and a construction permit, differs markedly in its distribution among "treatments" and "controls," as seen in Figure 5: treatments have systematically larger values of it, although the two distributions well overlap. When two groups compare in this way, no fixed-ratio matching of them can reduce their overall discrepancy. Some of the observations will have to be set aside — or, better yet, one could match the two groups in varying ratios, using matching with multiple controls or full matching. These techniques have surprising power to reconcile differences between treatments and controls while setting aside few or even no subjects because they lack suitable counterparts; the reader is referred to Hansen (2004, § 2) for general discussion and a case study.

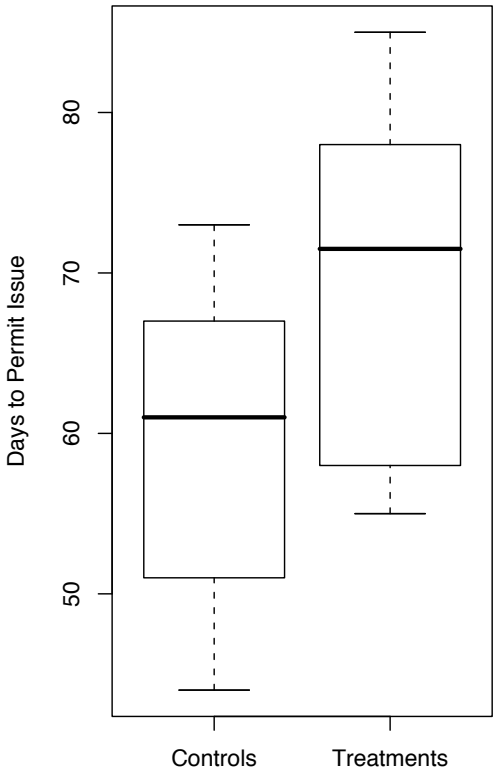

Figure 5: New and existing sites' differences on the variable `t2`. To reduce these differences, one has either to drop observations or to use flexible matching techniques.

In practice, one should look critically at an optimal match before moving ahead with it toward outcome analysis, refining its generating distance and structural requirements as appropriate, just as a careful analyst deploys various diagnostics in the process of developing and refining a likelihood-based analysis. Diagnostics for matching are discussed in various methodological papers, many of them recent (Rosenbaum and Rubin, 1985; Rubin, 2001; Lee, 2006; Hansen, 2006a; Sekhon, 2007).

optmatch output

Matched pairs are often analyzed by methods particular to that structure, for example the paired t-test. However, matching with multiple controls and full matching require methods that treat the matched sets as strata. With these uses in mind, matching functions in **optmatch** give factor objects as their output, creating unique identifiers for each matched set and tagging them as such in the factor. (Strictly speaking, the value of a call to `fullmatch` or `pairmatch` is of the class `c("optmatch", "factor")`, but it is safe to treat it as a factor.) If one in fact has produced a pair match, then one can recover the paired differences using the `split` command:

```
> pm <- pairmatch(dl)
> attach(nuclear)
> unlist(split(cost[PR],pm[PR])) -
    unlist(split(cost[!PR],pm[!PR]))
```

— the result of which is the vector of differences

```
  0.1    0.2    0.3  ...   1.2   1.3
-9.77 -10.09 184.75 ... -17.77 -4.52
```

For matched comparisons after full matching or matching with a varying number of controls, one uses such commands as

```
> fm <- fullmatch(dl)
> tapply(cost[PR], fm[PR], mean) -
    tapply(cost[!PR], fm[!PR], mean)
```

to return differences of treatment and control means by matched set. The sizes of the matched sets, in terms of treatment units, controls, or both, can be tabulated by

```
> tapply(PR, fm, sum)
> tapply(!PR, fm, sum)
> tapply(fm,fm,length)
```

respectively. Unmatched units are automatically dropped, and `split` and `tapply` return matched-set specific results in a common ordering (that of the levels of the match object, *e.g.* `pm` or `fm`.)

Summary

Optmatch offers a comprehensive implementation of matching of two groups, such as treatments and controls or cases and controls, including optimal pair matching, optimal matching with k controls, optimal matching with a varying number of controls, and full matching, with and without structural restrictions.

Bibliography

A. Abadie and G. W. Imbens. Large sample properties of matching estimators for average treatment effects. *Econometrica*, 74(1):235–267, 2006.

R. Althauser and D. Rubin. The computerized construction of a matched sample. *American Journal of Sociology*, 76(2):325–346, 1970.

J. Bowers and B. B. Hansen. Attributing effects to a cluster randomized get-out-the-vote campaign. Technical Report 448, Statistics Department, University of Michigan, October 2006. URL `http://www-personal.umich.edu/%7Ejwbowers/PAPERS/bowershansen2006-10TechReport.pdf`.

A. R. Brazzale. hoa: An R package bundle for higher order likelihood inference. *Rnews*, 5/1 May 2005: 20–27, 2005. URL `ftp://cran.r-project.org/doc/Rnews/Rnews_2005-1.pdf`. ISSN 609-3631.

A. R. Brazzale, A. C. Davison, and N. Reid. *Applied Asymptotics*. Cambridge University Press, 2006.

N. E. Breslow and N. E. Day. *Statistical Methods in Cancer Research (Vol. 1) — The Analysis of Case-control Studies*. Number 32 in IARC Scientific Publications. International Agency for Research on Cancer, 1980.

D. Cox and E. Snell. *Applied Statistics*. Chapman and Hall, 1981.

D. R. Cox and E. J. Snell. *Analysis of Binary Data*. Chapman & Hall Ltd, 1989.

A. Diamond and J. S. Sekhon. Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. Technical report, Travers Department of Political Science, University of California, Berkeley, 2006. version 1.2.

A. Gelman and J. Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press, 2006.

X. Gu and P. R. Rosenbaum. Comparison of multivariate matching methods: Structures, distances, and algorithms. *Journal of Computational and Graphical Statistics*, 2(4):405–420, 1993.

B. B. Hansen. Appraising covariate balance after assignment to treatment by groups. Technical Report 436, University of Michigan, Statistics Department, 2006a.

B. B. Hansen. Full matching in an observational study of coaching for the SAT. *Journal of the American Statistical Association*, 99(467):609–618, September 2004.

B. B. Hansen. Bias reduction in observational studies via prognosis scores. Technical Report 441, University of Michigan, Statistics Department, April 2006b. URL `http://www.stat.lsa.umich.edu/%7Ebbh/rspaper2006-06.pdf`.

B. B. Hansen and S. O. Klopfer. Optimal full matching and related designs via network flows. *Journal of Computational and Graphical Statistics*, 15(3):609–627, 2006. URL `http://www.stat.lsa.umich.edu/%7Ebbh/hansenKlopfer2006.pdf`.

T. Hothorn, K. Hornik, M. van de Wiel, and A. Zeileis. A lego system for conditional inference. *The American Statistician*, 60(3):257–263, 2006.

W.-S. Lee. Propensity score matching and variations on the balancing test, 2006. URL `http://papers.ssrn.com/sol3/papers.cfm?abstract_id=936782#`.

T. Lumley and T. Therneau. **survival***: Survival analysis, including penalised likelihood*, 2006. R package version 2.26.

K. Ming and P. R. Rosenbaum. Substantial gains in bias reduction from matching with a variable number of controls. *Biometrics*, 56:118–124, 2000.

D. Pierce and D. Peters. Practical use of higher order asymptotics for multiparameter exponential families. *Journal of the Royal Statistical Society. Series B (Methodological)*, 54(3):701–737, 1992.

S. W. Raudenbush and A. S. Bryk. *Hierarchical Linear Models: Applications and Data Analysis Methods*. Sage Publications Inc, 2002.

P. R. Rosenbaum. A characterization of optimal designs for observational studies. *Journal of the Royal Statistical Society*, 53:597– 610, 1991.

P. R. Rosenbaum. *Observational Studies*. Springer-Verlag, second edition, 2002.

P. R. Rosenbaum and D. B. Rubin. Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. *The American Statistician*, 39:33–38, 1985.

D. B. Rubin. Using propensity scores to help design observational studies: application to the tobacco litigation. *Health Services and Outcomes Research Methodology*, 2(3):169–188, 2001.

D. B. Rubin. Using multivariate matched sampling and regression adjustment to control bias in observational studies. *Journal of the American Statistical Association*, 74:318–328, 1979.

D. B. Rubin and N. Thomas. Combining propensity score matching with additional adjustments for prognostic covariates. *Journal of the American Statistical Association*, 95(450):573–585, 2000.

J. S. Sekhon. Alternative balance metrics for bias reduction in matching methods for causal inference. Survey Research Center, University of California, Berkeley, 2007. URL `http://sekhon.berkeley.edu/papers/SekhonBalanceMetrics.pdf`.

H. Smith. Matching with multiple controls to estimate treatment effects in observational studies. *Sociological Methodology*, 27:325–353, 1997.

E. A. Stuart and K. M. Green. Using full matching to estimate causal effects in non-experimental studies: Examining the relationship between adolescent marijuana use and adult outcomes. Technical report, Johns Hopkins University, 2006.

Ben B. Hansen
Department of Statistics
University of Michigan
`bbh@umich.edu`