Comments from Paul Rosenbaum

Nice paper. It is a nice, useful paper.

Figure 1. Figure 1, page 15, makes a key point very nicely.

Page 19, Select covariates. An option you don't discuss is *how* to include a covariate in the matching. For example, with rare binary variables, one standard piece of advice is to include them in the propensity score, but not in the Mahalanobis distances. Also, if you exclude a covariate from the matching, you can check that the covariate is reasonably well balanced after matching before accepting the matching. I've actually discovered some important covariates this way, eg covariates (like measures of pain) that the docs said were unimportant. (Various similar, albeit fairly obvious, suggestions appear in chapter 3 of my book.)

Propensity score tautology. I agree with the point you make on page 20, "Fortunately, there is a way out ...". Perhaps the last sentence of the paragraph should say, "and when it doesn't work, keep working at it." Often, balance checks show imbalances, but often these can be fixed by adding quadratics, interactions, etc, providing the covariate distributions exhibit substantial overlap.

p 21, middle. "The collective wisdom ..." Two issues. (1) About multiple controls, a nice case study is Smith, H. (1997) Matching with multiple controls to estimate treatment effects in observational studies. *Sociological Methodology* **27**, 325-353. (2) There is quite a bit of theory and evidence showing that variable matching is vastly better than fixed ratio matching; see matching vs subclassification below. Perhaps that deserves mention here. It's not just variance; it's also big gains in bias reduction.

p 24, last paragraph. I agree with what you say here about uncertainty estimates.

p. 28, last paragraph. "every possible specifications" -> "every possible specification"

Matching vs subclassification; Importance of structure. On page 31, you say that matching did not improve things much but subclassifying did. That only makes sense if you restrict the meaning of matching quite a bit. Even if you don't want to do full matching, perhaps this section should be careful with the terminology It is possible to show that the best subclassification is always a full matching (Rosenbaum, P. R. 1991 A characterization of optimal designs for observational studies. *Journal of the Royal Statistical Society* **B53** 597-610.) Moreover, one can calculate the gains in bias reduction from using variable numbers of controls, and the gains are substantial (Ming, K. and Rosenbaum, P. R. 2000 Substantial gains in bias reduction from matching with a variable number of controls. *Biometrics*, 56, 118-124.) Anything you can achieve by subclassification you can achieve by variable/full matching, and the matching can use multivariate distances which stratification usually does not.

Combining quantitative and qualitative research. One issue that might interest certain social scientists is that matching facilitates coordination of qualitative and quantitative research (Rosenbaum, P. R. and Silber, J. H. 2001 Matching and thick description in an observational study of mortality after surgery. *Biostatistics*, **2**, 217-232.)

Optimal matching code. When computer scientists and operations researchers see statisticians do greedy matching, they giggle. To them, it would be like doing least squares regression with an algorithm that didn't minimize the sum of squares of the residuals, and instead just made a rough stab at it. With careful programming, optimal matching is not only better than nearest available matching (page 26), but it is also often faster. Ben Hansen has been talking about creating some R code. Some C and Fortran code for optimal nonbipartite matching (a very general form of matching) is available. (There are faster, but less general algorithms for treatment/control matching.) The C code can be downloaded free from http://elib.zib.de/pub/Packages/mathprog/matching/weighted/index.html and I've used it, and it works fine. There is a test problem you can download, then just imitate it. You can set the input up in Splus, write it out, run the C program, put the results back in Splus; there may be better ways. (The algorithm is described in Galil, Z. (1986) Efficient algorithms for finding maximum matching in graphs. *Computing Surveys*, 18, 23-38, but you don't have to read the paper to use the code.) More or less, you give it the distances, and it gives you the matches. Couple of tips: (i) Let d_{ij} be the distance between i and j. Set $d_{ij} = L$ where L is large if you want to forbid matching of i and j. With treatment/control matching (as distinct, say from dose matching), you would set $d_{ij} = L$ if i and j are both controls or both treated. (ii) Rather than minimizing the total distance, the program maximizes the total weight, w_{ij}. Define $w_{ij} = \left(\max_{m,n} d_{mn}\right) - d_{ij}$, which is typically $L - d_{ij}$. (iii) The weights w_{ij} need to be integers, so one multiplies by a constant to get a few digits of accuracy and rounds. Example: With 20 covariates, if you used the Mahalanobis distance within propensity score calipers, which Don & I found to be best in our 1985 Am Stat example, then set $d_{ij} = 20,000$ if i and j are both treated, or both control, or if their propensity scores differ by more than the caliper, and otherwise set $d_{ij} = 10 \times Mahalanobis$. The typical Mahalanobis distance between two observations with 20 covariates would be $2 \times 20 = 40$, so d_{ij} would be ten times bigger or typically 400, and L would be 50 times larger. Round the d_{ij} to the nearest integer, find $w_{ij} = L - d_{ij} = 20,000 - d_{ij}$, hand them to the C program, and it hands back the optimal match. You can also do it in Fortran (Derigs,

U. 1988 Solving non-bipartite matching problems via shortest path techniques. *Annals of Operations Research*, **13**, 225-261), but I think you have to ask Derigs for the code. Bo Lu used Derigs' code several times: Lu, B., Zanutto, E., Hornik, R., Rosenbaum, P. R. (2001). Matching with doses in an observational study of a media campaign against drug abuse. *Journal of the American Statistical Association* **96**, 1245-1253 and Lu, B. and Rosenbaum, P. R. (2004). Optimal pair matching with two control groups. *Journal of Computational and Graphical Statistics* **13**, 422-434. You can also do it in SAS: (1) Ming, K. and Rosenbaum, P. R. (2001) A note on optimal matching with variable controls using the assignment algorithm. *Journal of Computational and Graphical Statistics* 10, 455-463 and (2) Bergstralh, E. J., Kosanke, J. L., and Jacobsen, S. L. (1996) Software for optimal matching in observational studies. *Epidemiology*, **7**, 331-332. http://www.mayo.edu/hsr/sasmac.html. Very general Fortran and C code that is not easy to use is given by Bertsekas, D. P. (1991) *Linear Network Optimization.* Cambridge, MA: MIT Press. http://www.mit.edu:8001//people/ dimitrib/home.html.