Improving Computational Efficiency of Treatment and Control Match Generation in *optmatch*

Adam Rauh Data Science concentrator, University of Michigan**, Ben Hansen**, Associate Professor of Statistics, University of Michigan

Background/Context

This project builds improvements for an existing R package, *optmatch*, which optimally matches members of treatment and control groups in observational studies by transforming the statistical problem into a minimum cost flow problem, then handing this problem off to an appropriate solver (Hansen & Klopfer, 2006). This methodology is useful for studying differences in outcomes between treatment and control groups if salient differences at baseline can be captured in a measure of discrepancy between subjects, for instance absolute differences on a propensity score; the optimal matching routine arranges the subjects into specified structures, for example non-overlapping pairs, that jointly minimize the aggregated discrepancies between matched treatment and control subjects among all possible arrangements of those subjects into structures of the specified type. Matching-based analyses are useful for educational research purposes, as studies have found that they are successful in producing effect/impact estimates comparable to those from randomized trials (Fortson, Verbitsky-Savitz, Kopa, & Gleason, 2012). Optimal matching is a particularly sound choice because the generated matches have the lowest possible average intra-set dissimilarity, a property not shared by other commonly chosen methods (Pimentel, 2016). The software changes made are specifically intended to improve computation time, facilitating its use by data analysts.

Objective

The primary goal of this project is to improve overall computational efficiency of the package without compromising performance in other areas by means of providing well-informed “warm starts” for the solver. With this addition, analysts will be able to find matches more quickly, facilitating parameter tweaking and experimentation with the data set to try out different combinations of constraints and matching distances. There are a number of methods and workarounds intended to simplify matching problems and improve efficiency, such as restricting the number of matched comparisons per treatment and possibly parallelization (Kilcioglu & Zubizaretta, 2016). Additionally, it is common to create subclasses in the data set prior to matching when using optimal matching procedures in order to reduce computation times. This would entail splitting the data with respect to race, gender, or other convenient grouping variables in order to transform one larger matching problem into a sequence of smaller, more easily computable ones.

A secondary aim of this project is to make it possible to combine optimal solutions of these subproblems into a collection of starting values for the computation parameters of the larger matching problem, without the prior sub-classification. This would have the effect of reducing the computational time needed to solve the larger matching problem and thus enabling a statistician to more easily try out and compare additional potential matches.

Implementation/Intervention

The aforementioned "warm start" values could be included as part of the end user's input into the program. Alternatively, when a user is generating matches for multiple similar matching problems, such as in the situations described above, one could tell the program to calculate these values for a particular problem based on the solutions to a previously generated match. These values would be saved from previous calls to the solver and appropriately passed along to facilitate other matches. This requires changes in FORTRAN code to properly extract certain values from the solver, and modifications to R code to store, modify, and utilize such information. Our research implements and studies the effects of this "intervention" upon a working R package.

Generating these inputs is somewhat analogous to providing "starting values" as an initial guess for the solver. However, these values are not given in the form of the end results that the user sees after generating matches. In optimization terms, the solver finds a solution to the primal minimum cost flow problem indirectly by solving an associated dual problem, which is related, but transformed from the original (Bertsekas, 1998). (The optmatch package in effect performs two transformations: first, the statistical matching problem provided by the user is transformed into a primal minimum cost flow problem; second, the embedded FORTRAN code translates that problem to a corresponding dual problem.) The "warm start" values, thus, must be provided in the form of an initial guess to a solution to this secondary dual problem in the form of an array of "node prices" (Bertsekas 1998, p. 21), one for each treatment and control group member.

Data and Results

The results of the changes to the *optmatch* system will be evaluated by looking at computation times on matches generated from data from an observational study examining the impact of commercial test preparation for the SAT on students (Hansen, 2004). The original investigation made use of a full matching approach, so improvements with respect to computational costs can be observed as a logical extension. Computation times for generating matches for questions from this study (and similarly structured problems) will first be found without calculating or utilizing any well-informed initial starting values. These will be compared to computation times with the new changes in the system. Additional opportunities for previously computationally prohibitive matches could be examined as well.

While utilizing this data set as the primary example, the described modifications will be tested with other data as well. This should help to further understand the extent and impact of these improvements in a wider variety of problems and contexts.

References

Bertsekas, D. P. (1998). Network optimization: continuous and discrete methods. Belmont (Mass.): Athena scientific.

Fortson, K., Verbitsky-Savitz, N., Kopa, E., & Gleason, P. (2012). Using an Experimental Evaluation of Charter Schools to Test Whether Nonexperimental Comparison Group Methods Can Replicate Experimental Impact Estimates. NCEE 2012-4019. *National Center for Education Evaluation and Regional Assistance*.

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

Hansen, B.B., & Klopfer, S.O. (2006). Optimal full matching and related designs via network flows. Journal of computational and Graphical Statistics, 15(3), 609-627.

Kilcioglu, C., & Zubizarreta, J. R. (2016). Maximizing the information content of a balanced matched sample in a study of the economic performance of green buildings. The Annals of Applied Statistics, 10(4), 1997-2020.

Pimentel, S. D. Large, Sparse Optimal Matching with R package rcbalance.