Why Propensity Scores Should Not Be Used for Matching: Supplementary Appendix

Gary King* Richard Nielsen†

August 17, 2015

Abstract

This paper is the Supplementary Appendix to Gary King and Richard Nielsen, 2015, "Why Propensity Scores Should Not Be Used For Matching," copy at j.mp/psnot

*Institute for Quantitative Social Science, Harvard University, 1737 Cambridge Street, Cambridge MA 02138; GaryKing.org, king@harvard.edu, (617) 500-7570.

†Department of Political Science, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139; mit.edu/∼rnielsen, rnielsen@mit.edu, (857) 998-8039.

1 PSM Approximates Random Matching

In a simple simulation, we provide intuition for how relatively balanced data makes PSM, but not MDM or CEM, highly sensitive to trivial changes in the covariates, often producing nonsensical results approximating random matching. In the left panel of Figure 1, we generate data with 12 observations and two covariates, with one covariate plotted by the other. The data are well balanced between treated (black disks) and control (open circles) units. From these initial data, we generate 10 data sets, where we add to each of the 12 observations a small amount of random error drawn from a normal distribution with mean zero and variance 0.001. This error is so small relative to the scale of the covariates that the new points are visually indistinguishable from the original points (in fact, the graph plots all 10 sets of 12 points nearly on top of one another, but it only appears that one set is there). Next, we run CEM and MDM; in both cases, as we would expect, the treated units are matched to the nearest control in every one of the 10 data sets (as portrayed by the pair of points linked by curved solid lines).

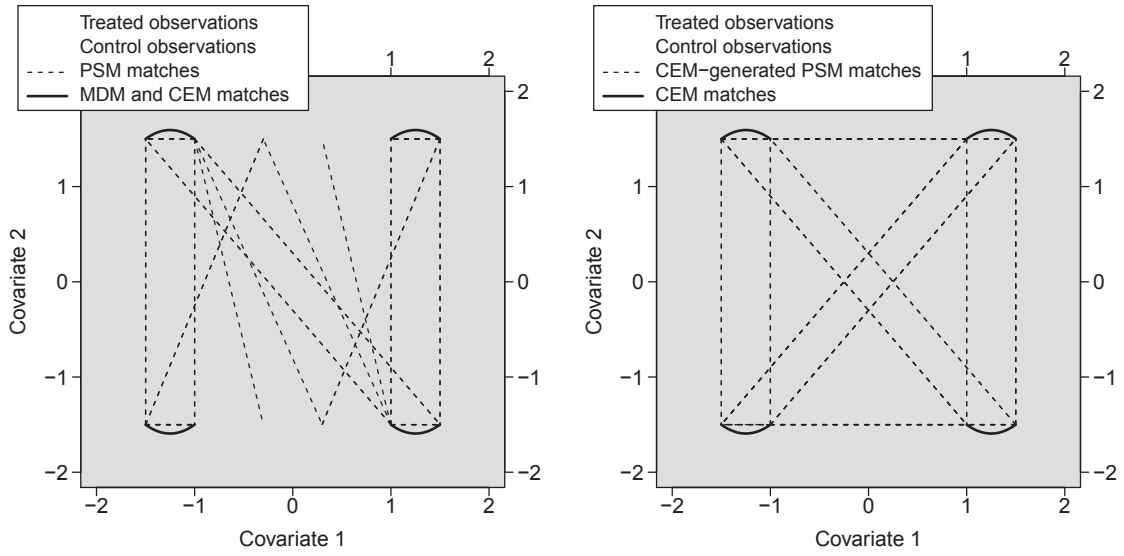

Figure 1: Ten data sets (differing from each other by imperceptibly small amounts of random error) with 4 treated units (black disks) and 8 control units (open circles). CEM and MDM match the closest control units to each treated (curved black lines). The two-step procedures match different control units for each data set, as can be seen for PSM (dashed lines, left panel) and PS-CEM (dashed lines, right panel). (The four open circles in the middle of the right panel are never matched; lines are passing through them on the way to show how other points are matched.)

However, when we run PSM on each of the 10 data sets generated for Figure 1, the four treated units are each matched to *different* control units (as portrayed by the maze of dashed lines connecting the black disks to different open circles). PSM is approximating random matching in this situation because it is unable to distinguish treated and control units; it is blind to the space of X that is not represented in $\hat{\pi}$.

Finally, we illustrate how the paradox results from PSM's two-step procedure. We do this by developing a (similarly unnecessary and ill-advised) two-step "propensity score CEM" (PS-CEM) algorithm: to do this, we use CEM to compute a nonparametric estimate of the propensity score (i.e., the proportion of treated units within each coarsened stratum; see Iacus, King and Porro 2011) and, second, without running CEM as usual, we match directly on the nonparametric estimate of the propensity score. The right panel in Figure 1 is constructed the same way as the left panel except that instead of the dashed lines representing propensity score matches, they represent PS-CEM matches. The result is almost as bad as PSM. The dashed lines in the right panel show how in the different (but highly similar) data sets, the two-step PS-CEM procedure matches control units (circles) close to and also distant from treated (closed disks) units. This suggests that ignoring X and only matching based on the scalar propensity score generates the PSM paradox.

2 The Paradox With Other Methods

In the first simulation, we contrive a data set where nature is malicious. We begin by generating 100 values of a single covariate X deterministically, in pairs along the number line as $X = 1, 2, 4, 5, 7, 8, \ldots, 145, 146, 148, 149$. We then assign observations with even values of X to receive treatment and those with odd values to receive control. If we stopped here, each treated unit would match best to the control observation 1 unit away and T would be independent of X in sample (and where both treated and control units of X have a mean of 75). Then to each value of X, we add a tiny amount of jitter drawn from a uniform on the interval $[-0.00001, 0.00001]$. This results in some pairs being slightly better matches than others, although solely due to random jitter. We then introduce confounding (which can be productively fixed via matching) by taking the three

treated units with the lowest values of X and reassigning them to control, and taking the three control units with the highest X values and assign them to treatment. For example, this creates a substantial difference between the mean value of X for the treated ($\approx$83) and control ($\approx$67). We generate the outcome variable as $Y = T + 0.01X + \epsilon$, where $\epsilon \sim N(0, 1)$.

The resulting data set has important levels of imbalance (and confounding) due to the units at the low and high values of X. The rest of the data will have matches that are effectively at random. The idea is that any method of matching will first prune the extreme (imbalanced) observations first for good reason and then start pruning at random.

We measure model dependence by first estimating the regression of Y on a constant, T, and elements of one of the subsets of $\{X, X^2, X^3, X^4, X^5\}$, and then repeat for all the other subsets. Then our measure is the range of estimates of the coefficient on T across all these regressions. Results appear in Figure 2, with model dependence plotted vertically and the number of treated units pruned by MDM horizontally.

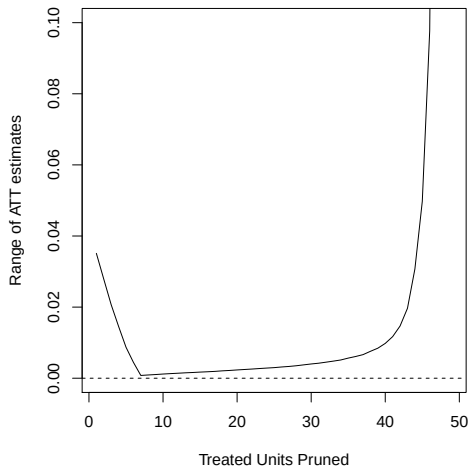

Figure 2: The Paradox with Mahalanobis Distance Matching

Thus, MDM first prunes the six extreme values of X which causes model dependence to drop. After that point, when all pairs differ by pure randomness, MDM continues to prune without accomplishing anything of value. Matching in this way does not overcome the fact that pruning itself increases imbalance, and so the overall imbalance line starts heading upward.

For a second illustration, we create a very small data set in high dimensional space so that points are so far spread out that few good matches are available. This is easy to see in MDM since Mahalanobis distances in this situation have the characteristic property of differing by tiny, essentially random amounts, only after many digits to the right of the decimal point. Thus, we generate a small data set, $n = 200$, with k covariates, for $k = 2, 3, 4, 5, 10$. For each k, we generate 100 data sets with covariates drawn from independent standard normals with means drawn from a uniform on the interval $[-10, 10]$. Then, for units designated as control, we add an independent draw for each covariate from a normal with mean zero and standard deviation 5.

We then define a set $\mathcal{M}$ of linear regression models that includes all possible specifications that include subsets of covariates, squared terms, and interactions, with squared terms and interactions included only if the main effects are included. We draw one model from $\mathcal{M}$ to define the true data generating process. We use this one true model to generate Y as a linear function of the treatment times its effect of 100, the covariates with coefficients drawn from a uniform distribution on the interval of [0,500], a constant term of 500, and a normal error term with mean 0 and standard deviation 500.

For each of the 100 data sets and each sample size, we run PSM and MDM, using all main effects only. To compute model dependence for a (matched) data set, we draw 1,000 models from $\mathcal{M}$, estimate the treatment effect for each as the coefficient on the treatment variable, and then compute the variance across these estimates. In order to have a comparable measure, the subset of 1,000 models is fixed across all runs (within a fixed k). We then average the standardized estimates of model dependence within each run, over the 100 runs, and plot scaled estimates.

Figure 3 gives our results, in parallel to previous figures, so that number of units pruned is on the horizontal axis and model dependence on the vertical axis. With PSM in red and MDM in blue, one panel appears for each k. Four patterns are apparent. First, PSM has higher levels of model dependence than MDM throughout all five graphs. Second, the advantage of MDM over PSM increases in all five graphs as more observations are pruned. Third, the PSM paradox is evident in all five graphs. And finally, a paradox,

with more units pruned leading to higher levels of imbalance, also affects MDM in 10 dimensional space in the last graph (and to some small extent right at the end of the some of the others).

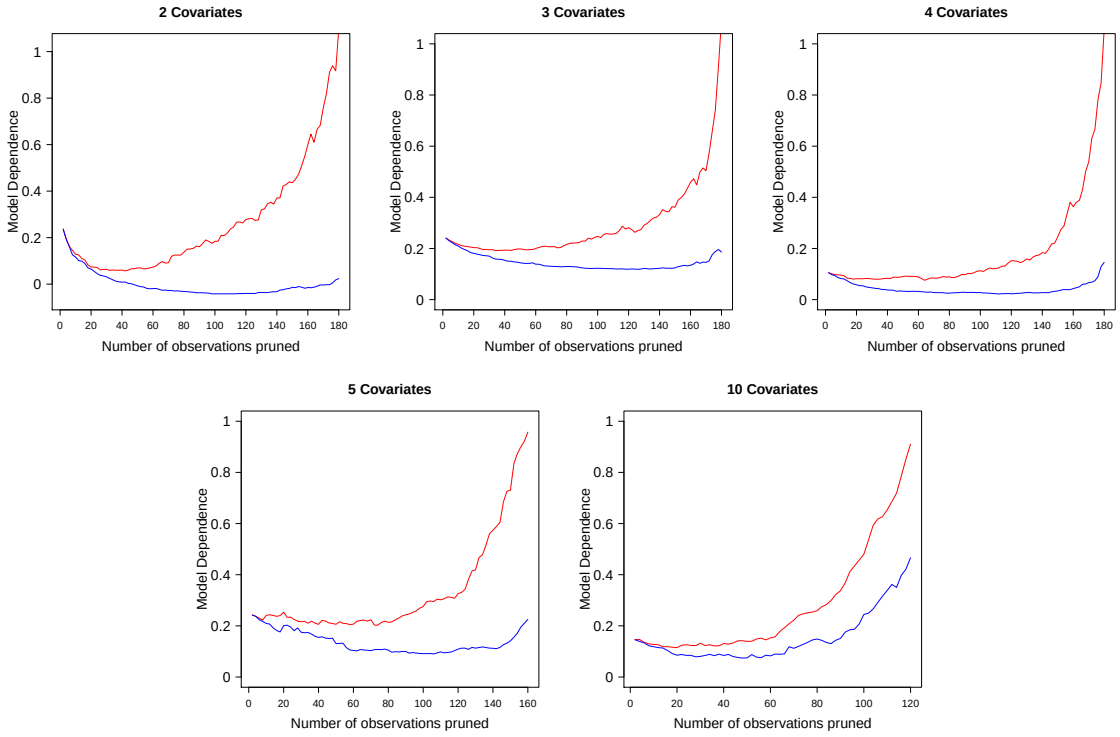

Figure 3: Model Dependence by Number of Covariates, with PSM in red and MDM in blue.

References

Iacus, Stefano M., Gary King and Giuseppe Porro. 2011. "Multivariate Matching Methods that are Monotonic Imbalance Bounding." *Journal of the American Statistical Association* 106:345–361. http://gking.harvard.edu/files/abs/cem-math-abs.shtml.