Does the choice of balance-measure matter under Genetic Matching?

Adeola Oyenubi and Martin Wittenberg

ERSA working paper 819

May 2020

Does the choice of balance-measure matter under Genetic Matching?

Adeola Oyenubi[†] and Martin Wittenberg[‡]

May 6, 2020

Abstract

In applied studies, the influence of balance measures on the performance of matching estimators is often taken for granted. This paper considers the performance of different balance measures that have been used in the literature when balance is being optimized. We also propose the use of the entropy measure in assessing balance. To examine the effect of balance measures, we conduct a simulation study where we optimize balance using Genetic Algorithm (GenMatch).

We found that balance measures do influence matching estimates under the GenMatch algorithm. The bias and Root Mean Square Error (RMSE) of the estimated treatment effect vary with the choice of balance measure. In the artificial Data Generating Process (DGP) with one covariate considered in this study, the proposed entropy balance measure has the lowest RMSE.

The implication of these results is that sensitivity of matching estimates to the choice of balance measure should be given greater attention in empirical studies.

Keywords: Genetic matching, balance measures, information theory, entropy metric

JEL Classification: *I38, H53, C21, D13*

1 Introduction

Randomized Control Trials (RCTs) are popular because researchers hope to easily achieve balance using RCTs. This is because, with randomization, the treated and untreated units are drawn from the same population, at random.

*The authors are grateful to the Centre for High Performance computing, Rosebank, Cape Town, South Africa (https://www.chpc.ac.za/) for giving us access to their machine without which most of the simulations in this study would not have been possible. The authors are also grateful for the comments of the Editor and reviewers of Empirical Economics (where the full version of this paper is published), these comments have helped to shape the ideas in this paper.

[†]University of the Witwatersrand, South Africa

[‡]DataFirst and University of Cape Town, South Africa

This ensures that the treatment and control samples have identical distributions of covariates (or are balanced in expectation) in both observed and unobserved covariates. Therefore, under an RCT the control sample provides the appropriate counterfactual for the treated sample. Although we note that randomization may fail as a result of chance imbalance, and there are other methods that may help improve balance (see Lock, Morgan and Rubin (2012) for details).

However, when randomization is not possible, estimation is based on an observational study or quasi-experiment. The key challenge for observational studies, therefore, is to replicate the kind of result one would expect from a randomized experiment (as argued in Lalonde (1986)). Consequently, similar to the situation under a randomized experiment, balance is important under quasi-experiments. The term 'balance (under randomization)', as it is used here, is in terms of the distribution of covariates and not just in some moments, like Mean and Variance. Under the relevant assumption (i.e. Conditional independence and Common support assumptions (CIA and CSA)), when a control group that balances the distribution of covariates in the treatment group is used in evaluation, the treatment effect will be unbiased (i.e. replicate experimental result) and robust across econometric methods, as one would expect from randomized data.

To assess balance, a researcher has to rely on balance measures which quantify the level of balance or lack thereof. However, there are a number of balance measures that have been used in the literature. Each measure compares different parts of distributions to assess balance and more often than not, the ability of a balance measure to provide adequate information about balance in a particular application is taken as a given. Hence, the central question in this study is: do balance measures vary in their performance? In this study, we argue that the choice of balance measure does, indeed, matter under the matching algorithm (and the DGP) considered in this paper – and by extension any matching algorithm that seeks to optimize balance.

In this study, we focus on one approach: Genetic Matching (GenMatch) which is a generalization of two popular matching approaches – Mahalanobis distance and propensity score matching (Diamond and Sekhon, 2013). Although the idea in this paper can be extended to other matching and weighting methods, the choice of GenMatch in this study is because using one matching approach allows for focus on the main idea in the paper – the sensitivity of matching estimates to balance measures – without having to deal with nuances associated with other matching methods (see Lehrer and Kordas (2013))[1]. Beyond evaluating the sensitivity of matching estimates to balance measures, this paper also introduces a new (distributional) balance measure. We discuss the rationale behind this balance measure and compare its performance with other popular balance measures in our simulation study. The possible implication of our

[1] The traditional implementation of propensity score matching requires discussion of the propensity score specification (see Lehrer and Kordas (2013) and their discussion on estimation of propensity scores). This, in itself, may affect the results, apart from the impact of balance measures. Here we fix the matching method and check if the result is sensitive to the choice of balance measure.

findings for other matching methods are also discussed.

Our result builds on the result of Diamond and Sekhon (2013) who showed that GenMatch produces lower Bias and Root Mean Square Error (RMSE) when compared to other matching methods. We demonstrate that the performance of GenMatch itself varies, in terms of Bias and RMSE, with the choice of balance measure. Specifically, under the univariate stylised DGP considered in this study, the proposed entropy measure has higher precision than other measures[2].

The rest of the study is organized as follows. Section 2 first reviews the literature on Genetic Matching and balance measures, then introduces the entropy balance measure and, finally, presents the balance measures used in this study. Section 3 discusses the Simulation study, the results and the implication of our findings for other matching methods while Section 4 concludes.

2 Literature review and Motivation for the study

A number of matching methods has been proposed in the literature and their finite sample properties have been studied (see Busso et al., (2014), Zhao (2004) and Frolich (2004)). The main focus of previous studies was the relative performance of different matching and weighting methods. However, as noted by Zhao (2004) and Caliendo and Kopeinig (2008), there is no clear winner among different matching estimators. This is because the performance of different matching estimators depends largely on the data structure (Caliendo and Kopeinig, 2008).

Although the above-mentioned studies mostly focus on the performance of estimators that rely on the propensity score, a number of studies in the literature have sought to solve the problems associated with estimating the propensity scores using various propositions. These propositions include Covariate Balancing Propensity Scores (Imai and Ratkovic, 2014), Entropy balancing (Hainmueller, 2012) and Genetic Matching (Diamond and Sekhon, 2013). These methods seek to optimize balance in the covariates in various ways: directly without relying on the propensity score (under the CIA and CSA); or by estimating propensity scores that incorporate the balancing condition. Simulation results suggest that optimizing balance with these methods yield gains in MSE or/and Bias over propensity score matching (see Imai and Ratkovic (2014), Hainmueller (2012) and Diamond and Sekhon (2013)). GenMatch is one of these methods in that it seeks to optimize balance. It, however, adopts a different approach called Genetic Algorithm – a machine learning method for optimization – to optimize balance.

One important aspect of the matching literature that has largely been overlooked is the possibility that balance measures can affect the performance of matching estimators. This is especially important when genetic algorithm is

[2]Note that the Simulation design considered in this working paper is limited in many ways, the full analysis that is expected to be published soon contain more realistic simulation designs. Our result in the full analysis suggests that the performance of the proposed entropy measure do not persist in real data. Instead the standardized difference in means tent to out-perform other measures.

used to optimize balance. Genetic algorithms depend on user-specified fitness function to guide it through the optimization process (Mitchell, 1998; Carr, 2014). In the context of GenMatch, the fitness function for the optimization is defined by the balance measure. In theory, one should not expect different fitness functions (i.e. balance measures) to give similar results. This is because the performance of balance measures may vary based on the form of imbalance they are designed to capture (e.g., distributional balance measures will be sensitive to imbalance in the form of shape differences and these may be ignored by balance measures that focus on the first two moments (see Diamond and Sekhon (2013: pg. 934) and Huber (2009)).

We build on the results in Diamond and Sekhon (2013) by showing that GenMatch, like other genetic algorithms, is sensitive to the choice of fitness function. Since the fitness function is defined by the balance measure, variation in performance as a result of variation in balance measures should be expected. Kinnear (1994; pg. 9), explains that genetic algorithm will "ruthlessly exploit" all subtle defects in the fitness function and this is why it is important for the fitness function to concisely reflect the intention of the optimization process. Therefore, variation in the form of imbalance that is captured by different balance measures may result in differences in matching results. In contrast to existing studies, this study, hence, fixes the matching method and explore the variation in Bias and RMSE across balance measures.

2.1 Genetic Matching

GenMatch performs multivariate (or univariate) matching using an evolutionary search algorithm. GenMatch is a non-parametric approach and does not depend on the estimation of propensity scores. The method seeks to maximise covariate balance by finding covariate weights that optimize balance. This is achieved by optimising a user-specified fitness function, which is, in turn, a function of some balance measure. In general, the aim is to optimize balance as much as possible rather than using a stopping rule (i.e., critical value in a statistical test). Diamond and Sekhon (2013) argue that this method helps to address some of the limitations of popular matching procedures such as the Mahalanobis distance and propensity score matching. Genmatch can be thought of as a generalisation of the Mahalanobis metric to include an additional weight matrix:

$$d\left(w_i, w_j\right) = \left\{ \left(w_i - w_j\right)' \left(S^{-1/2}\right)' M \, S^{-\frac{1}{2}} \left(w_i - w_j\right) \right\}^{1/2} \tag{1}$$

where w_i is a vector of covariates for individual i, M is a $t \, x \, t$ positive definite weight matrix and $S^{\frac{1}{2}}$ is the Cholesky decomposition of the variance-covariance matrix of the covariates (X). The goal is to find the weight matrix M that achieves the best balance when the distance produced by $d\left(w_i, w_j\right)$ is used to match observations in the sample. GenMatch searches for the best balance possible by generating random solutions (i.e., it generates a number of random

weight matrices M[3]). These solutions are then used to estimate Equation (1), and for each solution, balance is checked in the matched sample produced by using the distance $d(w_i, w_j)$. A solution that arises from weight matrix M_i is preferred to another solution M_j, if M_i produces more balance in the matched sample according to the fitness function supplied by the user[4]. The default fitness function optimizes balance using lexical optimization. This approach sorts the balance statistics (from all covariates) from the most discrepant to the least. The algorithm then aims to optimize the first, second, third ..., n^{th} component. If multiple sets of weights (M) result in the same maximum discrepancy, the second-largest discrepancy is examined to choose the best weight. This process continues iteratively until all ties are broken (Sekhon, 2011). After assessing balance and ranking the solutions in the first population according to their fitness values, a new population of solutions is formed. This is done using genetic operations: mutation, crossover, and selection. These operators work on one or more current trial solutions to produce one or more trial solutions in the new population[5]. The new population is then assessed and ranked using their fitness values. This process continues until the balance can no longer be improved. We refer interested readers to Diamond and Sekhon (2013) for a detailed discussion on how GenMatch works.

2.2 Choice of balance measure

Ideally, imbalance should be thought of in terms of differences in the joint densities of covariates across treatment arms (Iacus et al., 2012). However, in applied studies, balance is often assessed in univariate densities of covariates. The practice of comparing univariate densities may be informed by the expectation that, if all the univariate densities are balanced, then the joint density will also be balanced.

In the applied literature, balance often refers to identical first moments of covariates in the two treatment arms. This is often accomplished by a t-test of difference in means. However, Imai et al. (2008) suggest that rather than limit the comparison to the first moment, one can compare higher-order moments of baseline covariates. The standard deviation of covariates can be compared (in addition to the mean) in assessing balance (as in the standardized difference in means). What this suggests is that, by comparing variance and means, one can obtain a broader description of balance, which is especially important for continuous covariates (Austin, 2009).

There are other proposals in the literature that go beyond the Mean and

[3] This number is called population size in Genetic Algorithm.

[4] This fitness function can be to minimize the mean of the balance statistics across all covariates or perform lexical optimization (see Diamond and Sekhon, 2013).

[5] Selection gives preference to improve the solution to make it into the next generation of solutions (or the offspring population). Crossover combines two or more current solutions to form a new solution (offspring in the new population). Mutation is used to encourage diversity amongst solutions. This is achieved by changing parts of a candidate solution in the current population randomly to produce new solutions. See Mabane and Sekhon (2011) for more details.

Variance in assessing balance. One can speculate that the distributional measures of balance may be more appropriate since they are more in line with what one would expect randomization to achieve (i.e., balance in distribution). Austin (2009) and Huber (2009), among others, suggest comparing quantiles of the covariate distributions to allow for a broader description of balance in continuous variables. Other proposals include side-by-side box plots (Hoaglin et al., 1983), empirical cumulative distribution functions (Casella and Berger, 2002; Austin, 2009), quantile-quantile plots (Imai et al., 2008; Ho et al., 2007), non-parametric density functions (Austin, 2009), the Kolmogorov Smirnov (KS) test/statistic (Belitser et al. (2011); Diamond and Sekhon (2013); Huber (2009) and the entropic distance metric (Oyenubi, 2018). What these measures have in common is that they can provide a broader description of balance relative to the first two moments. However, as noted by Oyenubi (2018), these distributional measures differ in their usability and performance. Apart from the KS statistic and the entropy measure, the other distributional balance measures involve the use of subjective assessment of balance (i.e., based on visual inspection), for example, see the use of kernel densities in Austin (2009). On the other hand, the KS statistic and the entropy measure provide a summary statistic that can be used to compare different levels of imbalance.

2.3 Entropic distance metric as a balance measure

One of the main contributions of this paper is to propose a new balance measure that can be used to assess imbalance for causal inference. The motivation behind this proposal is to have a balance measure that is sensitive to imbalance in all moments, or one that is sensitive to all forms of imbalance in the distributions being compared. To achieve this, we are interested in measures that quantify the overlap between distributions, or one that quantifies the "distance" between distributions.

The proposed Entropy measure builds on the idea of comparing non-parametric density functions (Austin, 2009). However, unlike the application in Austin (2009; pg. 3100; figure 4) where visual inspection was used, one can summarize the difference between the kernel densities of covariates using the entropy measure. This proposal (to use entropy measure as a balance metric) was first put forward in Oyenubi (2018). The main argument is that since entropy measure compares (entire) distributions, it provides a broader description of balance. This argument is similar to the one by Huber (2009) who proposed the use of non-parametric quantile regression, distribution-free Kolmogorov-Smirnov (KS) and Cramer-von-Mises-Smirnov (CMS) test statistics to check for differences in entire distribution. The author argues that restricting balance checks to the Mean is necessary but not sufficient.

Entropy measure is the normalization of the Bhattacharya-Matusita-Hellinger measure of distance between probability distributions. The measure is given by

$$\mathrm{S}_{\rho}=\frac{1}{2}\int_{-\infty}^{\infty}\left(f_1^{1/2}-f_0^{1/2}\right)^2\,\mathrm{dx}$$

for continuous covariates. For discrete covariates, we have

$$\mathrm{S}_\rho = \frac{1}{2} \sum \left(p_1^{1/2} - p_0^{1/2} \right)^2$$

where f_1 and f_0 represent the density of the two distributions being compared (treatment and control, in our case) and p_1 and p_0 represent the mass in the discrete case. The fact that this measure is defined for both discrete and continuous variables (see Maasoumi and Racine (2008)) means that it can be used to assess balance across different types of variables on the same scale. Therefore, unlike the default measure under GenMatch that combines p-values of t-test and KS tests, the entropy measure does not need to use the p-values to facilitate comparisons.

The metric entropy, S_ρ, is a function of the differences between the kernel density estimate across the support of the distributions being compared. This means that, apart from picking up imbalance in the conventional sense, it will also pick up cases where imbalance manifests as thin or no common support problem (see Lechner & Strittmatter (2019)). Such areas of thin or no support may increase biases and variances of estimators (e.g. Crump et al, 2009; Khan and Tamer, 2010). Furthermore, this is in contrast to a measure like the KS statistic that is based on the maximum distance between cumulative distribution functions. Certain kinds of imbalance, especially at the tails, may be ignored by the KS measure because it places all its weight on the largest difference between cumulative distributions (see Tan et al. (2003); Parizzi and Brcic (2011) and Kvam and Vidakovic (2007) for a similar argument).

The entropy measure is normalized such that its value ranges between 0 to 1. $\mathrm{S}_\rho = 0$ when the densities overlap completely (i.e., overlap in terms of support and the density on the support), while $\mathrm{S}_\rho = 1$ when the densities don't share the same support (i.e., the densities don't overlap)[6]. The greater the value of S_ρ the higher the level of imbalance.

2.4 Balance measures used in this study

To compare the performance of different balance measures under GenMatch, we use seven popular balance measures namely: a combination of the p-value of t-tests and KS tests (default balance measure under GenMatch); the mean; the p-value of the t-test; the standardized difference in means; the KS statistic; the p-value of the KS statistic; and the proposed entropy distance metric. These measures range from those that compare only the means of covariates, to those that compare both Mean and Variance, and those that compare distributions to assess balance. Another dimension is that the first measure combines two measures (p-value of t-tests and KS tests) to assess balance, while others optimize the p-value instead of the measure itself. The implementation of the Entropy

[6] We do not expect this extreme case in our context because of the common support assumption. However, there may be areas of thin/no support in finite samples which this measure will be useful picking up.

measure, and other balance measures used in our simulation, is presented in the appendix A.

3 Simulation study and results

In this section, the details of the simulation study is presented, and the results discussed.

3.1 Simulation Study

Monte Carlo studies are useful in examining the small sample properties of different matching estimators. We use a simulations design that have been used previously in the literature to assess if the performance of GenMatch varies with the choice of balance measure.

We use the simulation design of Frolich (2004) (also been used by Busso et al., (2014)). The author uses a stylised data generating process (DGP), which may be unrealistic in an empirical setting, to assess the performance of matching estimators. Frolich's (2004) simulation is limited by the fact that it is based on one covariate. However, the design is useful in that it allows for the manipulation of the shape of the covariate distribution across treatment arms. Such designs may be, however, harder to simulate in a multivariate setting. This stylised design has also been criticized by Huber et al, (2013) and Busso et al. (2014) for being unrealistic. Therefore the interpretation of our results is only valid in the context of this design.

Since the simulation design is not new, we relegate the presentation of the design to the appendix (see appendix B). Across all simulations, we estimate the average treatment effect on the treated (ATT). The treatment effect is homogeneous and is set equal to 0. The sample size is 300. The number of iterations is 500, and population size for each generation of GenMatch is 1000.

3.2 Results

The Bias and RMSE results are shown in Table 1. The Bias is presented in relative terms (i.e., Bias is presented as the percentage difference relative to the minimum Bias estimate). Bold zeros (**0**) indicate that the balance measure produces the minimum Bias estimate, while the actual minimum Bias is presented in the last row (i.e. the row labelled "**Min**").[7] The raw RMSEs are also presented.

Simulation 1 has six designs controlled by the parameters ($\alpha\beta$). As we move from (*0.15*, *0.17*) to (*0*, *3*) the difference in the shapes of the propensity scores

[7] i.e. $\%\,Bias = \left| \frac{\sum_{i=1}^{500} \frac{(\hat{\theta}-\theta)}{500}}{\theta} \right| * 100$ where θ is the minimum ATT and $\hat{\theta}$ is the estimate of θ in each iteration of the simulations. The RMS s given by $\sum_{i=1}^{500} \frac{(\hat{\theta}-\theta)^2}{500}$.

distributions becomes more extreme (see Table B1 and Figure B1-B6 in Appendix B for the initial balance under each design, the ratio of treatment to control observations and a sample of the shape of the propensity score distributions under each design). In other words, balance becomes worse as we move from (*0.15*, *0.17*) to (*0*, *3*).

The result presented in Table 1 is for the case where the outcome is linear and shows that as initial imbalance increase, the minimum Bias ("**Min**") increases and the variance in the performance of balance measures (in terms of Bias) generally reduce. This implies that the choice of balance measure becomes less important as initial imbalance gets worse under this DGP. The result also shows that there is considerable variation in Bias under different balance measures. While the variation in Bias does not follow any particular pattern in terms of a best-performing measure across designs, the RMSE results show that the entropy measure performs better than other measures under the DGP, irrespective of design.

Table B2 and B3 in Appendix B presents the result when the outcome is nonlinear. The results show a clear pattern: the default balance measure (a combination of p-values of KS statistic and t-test) has the lowest Bias when initial imbalance is low, and the entropy measure has the lowest Bias for higher levels of initial balance. The RMSE results show that the entropy measure dominates other measures across the designs. In all, under this DGP the default and the proposed entropy measure perform better than competing measures[8].

These results show that the performance of GenMatch varies under different balance measures. This is important for the application of this matching method. For example, Diamond and Sekhon (2013) show that GenMatch performs better than other matching estimators, however, our result shows that this performance is sensitive to the choice of balance measure. Specifically, under the stylized DGP considered here, the proposed entropy measure would have outperformed the default balance measure under GenMatch. We, however, note that this result cannot be generalized and is only valid under the stylized DGP considered here.

The full version of this paper (this is a working paper) presents the results when different balance measures are used under a more realistic DGP. The result shows that the superior performance of the entropy measure did not persist under realistic DGPs. Instead, the results suggest that the standardized difference in means is a robust balance measure across different DGPs.

4 Conclusion

Our results support the hypothesis that the performance of balance measures vary under GenMatch.

[8] We note that GenMatch is a pre-processing algorithm. Therefore the resulting estimate after matching with GenMatch may still be sensitive to the econometric method used to calculate the treatment effect from the matched data. For example we found the results with and without bias correction can be very different.

Giving the variation in our results, one recommendation that can be helpful is for researchers to incorporate balance measures in Monte Carlo experiments that are designed to pick the preferred estimator.

Balance is a general word that means different things depending on what the balance measure is designed to capture. Being mindful of this is useful for minimizing Bias and MSE for matching estimators.

References

[1] Austin, P. C., 2009.Balance diagnostics for comparing the distribution of baseline covariates between treatment groups in propensity-score matched samples. Statistics in Medicine, 28(25), 3083-3107.

[2] Belitser, S. V., Martens, E. P., Pestman, W. R., Groenwold, R. H., Boer, A., and Klungel, O. H., 2011. Measuring balance and model selection in propensity score methods. Pharmacoepidemiology and Drug Safety, 20(11), 1115-1129.

[3] Busso, M., DiNardo, J. and McCrary, J., 2014. New evidence on the finite sample properties of propensity score reweighting and matching estimators. Review of Economics and Statistics, 96(5), 885-897.

[4] Caliendo, M., and Kopeinig, S., 2008. Some practical guidance for the implementation of propensity score matching. Journal of Economic Surveys, 22(1), 31-72

[5] Carr, J., 2014. An introduction to genetic algorithms. Senior Project, 1, p.40.Casella, G., and Berger, R. L., 2002. Statistical inference (Vol. 2). Duxbury Pacific Grove, CA.Crump, R.K., Hotz, V.J., Imbens, G.W. and Mitnik, O.A., 2009. Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), pp.187-199.

[6] Dehejia, R.H. and Wahba, S., 1999. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. Journal of the American statistical Association, 94(448), pp.1053-1062.

[7] Diamond, A., and Sekhon, J. S., 2013. Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. Review of Economics and Statistics, 95(3), 932-945.

[8] Frölich, M., 2004. Finite-sample properties of propensity-score matching and weighting estimators. Review of Economics and Statistics, 86(1), 77-90.

[9] Hainmueller, J., 2012. Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. Political Analysis, 20(1), 25-46.

[10] Ho, D. E., Imai, K., King, G., and Stuart, E. A., 2007. Matching as Nonparametric Preprocessing for Reducing Model Dependence in Parametric Causal Inference. Political Analysis, 15, 199-236

[11] Hoaglin, D. C., Mosteller, F., and Tukey, J. W., 1983. Understanding robust and exploratory data analysis (Vol. 3). Wiley New York

[12] Huber, M., 2009. Testing for covariate balance using nonparametric quantile regression and resampling methods. Unpublished Working and Discussion Papers.

[13] Huber, M., Lechner, M. and Wunsch, C., 2013. The performance of estimators based on the propensity score. Journal of Econometrics, 175(1), pp.1-21.

[14] Iacus, S. M., King, G., Porro, G., and Katz, J. N., 2012. Causal inference without balance checking: Coarsened exact matching. Political Analysis, 1-24.

[15] Imai, K. and Ratkovic, M., 2014. Covariate balancing propensity score. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 76(1), 243-263.

[16] Imai, K., King, G., and Stuart, E. A., 2008. Misunderstandings between experimentalists and observationalists about causal inference. Journal of the royal statistical society: series A (statistics in society), 171(2), 481-502.

[17] Khan, S. and Tamer, E., 2010. Irregular identification, support conditions, and inverse weight estimation. Econometrica, 78(6), pp.2021-2042.

[18] Kinnear Jr, K.E., 1994. A perspective on the work in this book. Advances in Genetic Programming. Pg. 3-19.

King, G., Lucas, C. and Nielsen, R.A., 2017. The balance-sample size frontier in matching methods for causal inference. American Journal of Political Science, 61(2), pp.473-489.

[19] Kvam, P. H., and Vidakovic, B., 2007. Nonparametric statistics with applications to science and engineering (Vol. 653). John Wiley & Sons.

[20] LaLonde, R.J., 1986. Evaluating the econometric evaluations of training programs with experimental data. The American economic review, pp.604-620.

[21] Lee, B.K., Lessler, J. and Stuart, E.A., 2010. Improving propensity score weighting using machine learning. Statistics in medicine, 29(3), 337-346.

[22] Lehrer, S.F. and Kordas, G., 2013. Matching using semiparametric propensity scores. Empirical Economics, 44(1), 13-45.

[23] Lechner, M. and Strittmatter, A., 2019. Practical procedures to deal with common support problems in matching estimation. Econometric Reviews, pp.1-15

[24] Maasoumi, E., and Racine, J. S., 2008.A robust entropy-based test of asymmetry for discrete and continuous processes. Econometric Reviews, 28 (1-3), 246-261.

[25] Mitchell, M., 1998. An introduction to genetic algorithms. MIT press.

[26] Racine, J.S., 2012. Entropy-Based Inference using R and the np Package: A Primer. R package vignette, version 0.40-13, URL http://CRAN. R-project. org/package= np.

[27] Oyenubi, A., 2018. Quantifying balance for causal inference: An information-theoretic perspective (Doctoral dissertation, University of Cape Town).

[28] Parizzi, A., and Brcic, R., 2011. Adaptive InSAR stack multilooking exploiting amplitude statistics: A comparison between different techniques and practical results. IEEE Geoscience and Remote Sensing Letters, 8(3), 441-445.

[29] Sekhon, J. S., 2011. Multivariate and propensity score matching software with automated balance optimization: The matching package for R.

[30] Setoguchi, S., Schneeweiss, S., Brookhart, M.A., Glynn, R.J. and Cook, E.F., 2008. Evaluating uses

[31] of data mining techniques in propensity score estimation: a simulation study. Pharmacoepidemiology and drug safety, 17(6), 546-555.

[32] Tan, Y.P., Nagamani, J. and Lu, H., 2003. Modified Kolmogorov-Smirnov metric for shot boundary detection. Electronics Letters, 39(18), 1313-1315.

[33] Zhao, Z., 2004. Using matching to estimate treatment effects: Data requirements, matching metrics, and Monte Carlo evidence. Review of Economics and Statistics, 86(1), 91-107

Table 1: Simulation Results

Simulation 1												
	% Bias (relative to minimum bias)						RMSE					
	(0.15,0.7)	(0,1)	(0,1.5)	(0,2)	(0,2.5)	(0,3)	(0.15,0.7)	(0,1)	(0,1.5)	(0,2)	(0,2.5)	(0,3)
d	1520.36	191.28	1.96	0.37	0.68	0.41	3.33	7.09	18.68	17.14	17.36	14.16
m	15064.89	2.02	3.79	6.27	4.53	***0.00***	2.09	3.58	9.18	5.51	3.01	0.79
p	907548.68	2595.88	0.12	0.52	***0.00***	0.25	86.33	41.39	22.75	17.67	17.09	13.87
s	14990.22	***0.00***	3.02	4.88	3.50	0.38	2.40	3.21	10.25	7.05	5.79	3.87
KS	3980.80	70.83	10.23	8.84	2.66	0.50	2.06	2.67	5.56	3.28	3.01	2.83
KSp	14484.66	177.41	***0.00***	***0.00***	0.68	0.41	10.80	11.66	22.76	17.68	17.36	14.16
e	***0.00***	145.66	14.44	10.69	4.77	1.07	***0.00***	***0.00***	***0.00***	***0.00***	***0.00***	***0.00***
Min	$6x10^{-6}$	0.001	0.06	0.111	0.151	0.203	0.049	0.064	0.16	0.215	0.264	0.292

d the default measure (i.e. a combination of the p-values of t-test of mean difference and KS distance); **m** mean difference; **p** p-value of t-test of mean difference; **s** the standardized difference in means; **KS** the KS statistic; **KSp** p-value of the KS statistic; **e** entropy distance metric

Appendix A

A1.1 Estimating entropic distance by kernel techniques

In practice, implementing the entropy measure (S_ρ) to compare two distributions involves a two-step procedure. First, the densities to be compared, f_1 and f_0 , must be estimated, then the distance between the estimated densities is measured. Naturally, any error in estimating the densities will filter into the resulting distance measure. Following Granger, *et al.* (2004), Maasoumi & Racine (2008) and Maasoumi & Wang (2012), the kernel density estimates of f_1 and f_2 will be used so that

$$\widehat{S_\rho} = \frac{1}{2}\int_{-\infty}^{\infty}\left(\widehat{f_1}^{1/2} - \widehat{f_0}^{1/2}\right)^2 dx$$

where $\widehat{f_1}^{1/2}\ and\ \widehat{f_0}^{1/2}$ are kernel density estimates of f_1 and f_0 respectively (Note similar expression can be written for the discrete case). To do this, the choice of bandwidth and kernel becomes important in making sure that the distance measure in the second step is reliable.

The implementation of S_ρ in this study follows the implementation in Maasoumi & Wang (2012). Like these authors, we use the Gaussian kernel and a robust version of the "normal reference rule-of-thumb" bandwidth $\left(= 1.06 \min\left(\sigma, \frac{IQR}{1.349}\right) n^{-\frac{1}{5}}\right)$ where σ is the standard deviation and IQR is the interquartile range. We use the "npunitest" in R to implement S_ρ (Racine, 2012).

A1.2 Other balance measures

The formulae used to compute the other balance measures are presented below

- $mean = E(x_t) - E(x_c)$
- $p - value\ of\ t\ test$ makes use of the mean statistic, the p-value of the t-test is used to assess balance (this is implemented with the t-test command in R)

- $Standardized\ difference\ in\ means = \frac{E(x_t)-E(x_c)}{\sqrt{\frac{(n_t-1)\widehat{\sigma_t}^2+(n_c-1)\widehat{\sigma_c}^2}{n_t+n_c}}}$

  Where n_t and n_c represent the sample size for treatment and control observations and $\widehat{\sigma_t}^2$ and $\widehat{\sigma_c}^2$ is the sample variance for the treatment and control samples. This is implemented by the command "smd" in the R package MBESS.

- $KS\ statistic = sup_x|F(x_t) - G(x_c)|$

  Where $F(x_t)$ and $G(x_c)$ represent the cumulative distribution function of the treated and control samples. The command "ks.test" in R is used to implement the KS statistic

- $p - value\ of\ the\ KS\ statistic$ makes use of the KS statistic and p-value is computed by Monte Carlo Simulation (as implemented in the R command)

- $Default\ Measure\ under\ GenMatch$ combines the p-value of t-test and the KS test

Appendix B

B.1.1 Simulation I (Frolich (2004))

The design consist of two parts. The first part deals with the distribution of propensity scores across treatment arms while the second part deals with the specification of the conditional expectation of the outcome given the propensity scores.

The set up can be written as

$$Y_i(T) = m(Z_i) + \sigma\varepsilon_i \dots\dots (1)$$

$$T_i^* = \alpha + \beta Z_i - U_i \dots\dots (2)$$

where $Z_i = \Lambda(\sqrt{2} * X_i)$ is a function of a single standard normal covariate X_i, ε_i the error term is independent and identically distributed (i.i.d) uniform with mean 0 and variance 1. X_i and ε_i are independent. The error term U_i is i.i.d. uniform and is independent of ε_i and X_i. T_i^* is the

latent variable that determines treatment status. A unit is in the treatment group if $T_i^* > 0$. α and β are parameters that determine the shape of the propensity scores distributions.

$Y_i(0)$ and $Y_i(1)$ represent the counterfactual outcome under control and treatment respectively, lastly, we set $\sigma = 0.1$. By manipulating the values of α and β one can generate different designs for the propensity scores across treatment arms where the propensity score densities are denoted by $f_{p|T=0}$ and $f_{p|T=1}$ for the control and treatment distributions respectively. The true propensity score is given by

$$p(X_i) = \alpha + \beta\, \Lambda\left(\sqrt{2} * X_i\right) - U_i \dots\dots\dots (3)$$

Frolich (2004) consider a total of thirty DGPs i.e. six specifications of $Y_i(T)$ – the outcome equation and five designs of $f_{p|T=0}$ and $f_{p|T=1}$ – the propensity score designs (this is controlled by the parameters α and β). In this study, we use twelve DGPs. This consist of two specifications of $Y_i(T)$ (specification 1 and 6 in Frolich (2004)). These specifications are given by

$$Y_i(T) = \theta + 0.15 + 0.7Z_i \equiv m_1(Z_i) \dots\dots\dots (4)$$

$$Y_i(T) = \theta + 0.4 + 0.25\sin(8Z_i - 5) + 0.4\exp[-16(4Z_i - 2.5)^2] \equiv m_2(Z_i) \dots\dots (5)$$

The outcome surface $m_1(Z_i)$ is linear while $m_2(Z_i)$ is nonlinear. The motivation behind this choice is to check if the performance of the matching estimator varies with the outcome surface. We consider 6 propensity score designs corresponding to values (0.15, 0.17) (0,1) (0,1.5) (0,2) (0,2.5) (0,3) for (α, β) [1]. The support of the propensity score densities is given by $(\alpha, \alpha + \beta)$, therefore to make sure that the support is always in (0,1) we use the rescaled propensity score

[1] According to Frolich (2004) α shifts the average value of the propensity upwards so that the treated-control ratio increase while β controls the spread of the propensity score. However, we use a different approach, to simulate cases where there are common support (thin and no support problems) problems we found that increasing β has the same effect of increasing treated-control ratio while also introducing increasing common support problem.

$$\hat{p}(X) = \frac{(p(X_i) - \alpha)}{\beta} \ldots\ldots\ldots (6)$$

The first two designs correspond to designs 2 and 1 (respectively) in Frolich (2004). The other designs increase the value of β in steps of 0.5. Progressively increasing the value of β from 1 to 3 has two effects. First, it increases the control-treated ratio form 1:1 in the first two designs to 7:1 in the last design (when β =3). Second, it generates different shapes for $f_{p|T=0}$ and $f_{p|T=1}$ such that the imbalance in the propensity score density increases for all balance measures.

B1.2 Set-up of the Simulation

The important thing about the stylized simulation is that one can manipulate the parameters to generate different shapes for the treatment and control distributions. As a consequence, the ratio of treatment to control observations change from one design to another. Table B1 shows the average balance (before matching) under the different measures for all the designs and the average treatment to control ratio. These designs are controlled by setting parameters α and β which controls the shape of the distributions across treatment arms.

Figure B1: Design 1

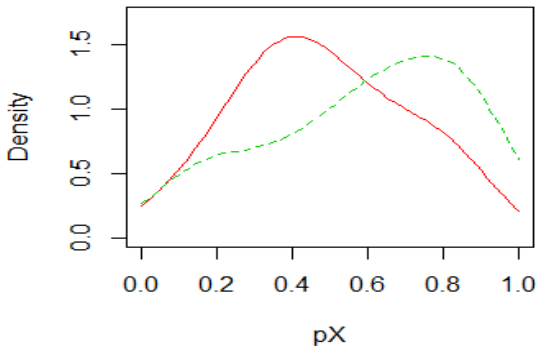


Figure B2: Design 2

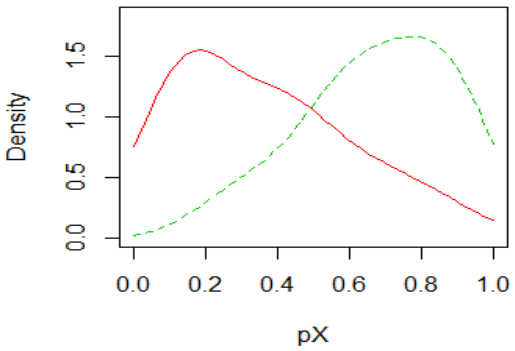


Figure B3: Design 3

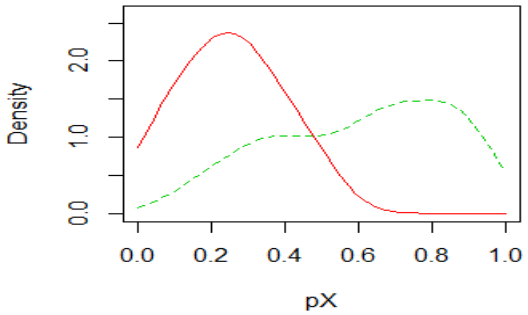


Figure B4: Design 4

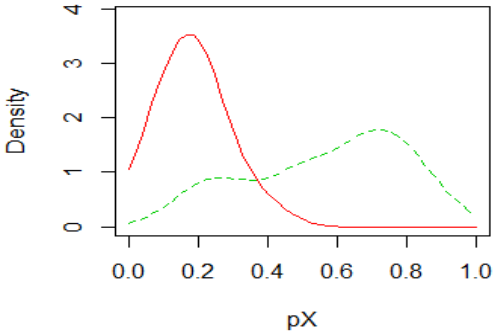


Figure B5: Design 5

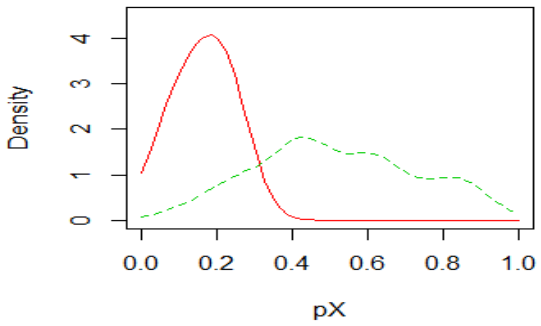
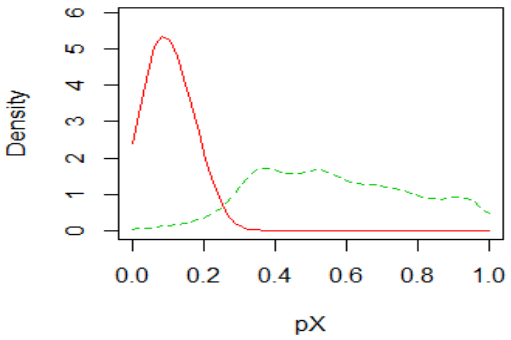


Figure B6: Design 6



*****pX is the propensity score. The red and green lines represent the density for the control and treatment group respectively.

It is clear from the results in table B1 that imbalance increase across the designs. Figure B1 to B6 show a sample of the kernel density estimates under the different designs. Notice that not only did the difference in density increase as one move from design1 (with parameter setting 0.15, 0.7) to design 6 (0,3) the thin / no support problem increase in the same direction.

We use GenMatch to optimize balance using the DGP described[2]. Recall that the focus here is on univariate matching such as one would have under PSM or when GenMatch reduces to PSM. The weight matrix M is no longer relevant since there is only one variable to balance. All that is important is how the different balance measures guide the optimization algorithm towards the optimal point.

As mentioned in the main text, we set the treatment effect (θ) equal to zero, the sample size is 300 (sample size in Frolich (2004) was 100[3]). We estimate the treatment effect for 500 samples. The population size for the GenMatch is set to 1000 (i.e. each generation of the genetic algorithm contain 1000 random solutions). The algorithm stops if there are no improvements in balance after two consecutive generations.

To make sure that the true effect estimate does not vary across designs we focus on the ATT. This means treated observation are never dropped in the process of matching but control observation can be dropped if dropping them improves balance. If the choice of balance measure does not matter the effect estimate under different balance measures should be very similar.

[2] The Stata code to generate the DGP is provided by Busso et al, (2014) and it is available on the publishers website. We merely re-write the code in R.
[3] Note that this choice did not affect our results

Table B2: Initial balance and the treated-control ratio

	Initial balance					
Design	**1**	**2**	**3**	**4**	**5**	**6**
(α, β)	**(15,0.7)**	**(0,1)**	**(0,1.5)**	**(0,2)**	**(0,2.5)**	**(0,3)**
Default	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Mean difference	0.192	0.275	0.359	0.389	0.327	0.419
p-value of t-test	0.007	0.000	0.000	0.000	0.000	0.000
Standardized difference	0.796	1.244	1.806	1.918	1.542	1.932
KS statistic	0.383	0.514	0.688	0.777	0.673	0.874
p-value of KS statistic	0.013	0.000	0.000	0.000	0.000	0.000
Entropy	0.079	0.163	0.364	0.473	0.435	0.593
Treatment: control ratio	1:1	1:1	2:1	3.5:1	5:1	7:1

*Default refers to the default balance measures under GenMatch i.e. a combination of the p-value of t-tests and the KS test.

Table B2: Bias Estimate (Nonlinear outcome equation)

	Bias					
Design	**1**	**2**	**3**	**4**	**5**	**6**
(α, β)	**(0.15,0.7)**	**(0,1)**	**(0,1.5)**	**(0,2)**	**(0,2.5)**	**(0,3)**
Default	***0.00***	***0.00***	3709.88	1.46	8.04	14.57
Mean difference	58.46	43.42	2942.43	1.32	3.96	1.99
p-value of t-test	1734.61	178.28	4313.87	1.66	7.36	13.34
Standardized difference	61.72	42.59	3164.11	1.35	5.24	6.41
KS statistic	28.32	34.55	1562.09	1.06	1.88	4.10
KS p-value	25.50	5.92	4324.99	1.52	8.04	14.57
entropy	12.98	21.28	***0.00***	***0.00***	***0.00***	***0.00***
min Value	0.001	0.004	0.001	0.261	0.204	0.139

Table B3: RMSE Estimate (Nonlinear outcome equation)

	RMSE					
Design	**1**	**2**	**3**	**4**	**5**	**6**
(α, β)	**(0.15,0.7)**	**(0,1)**	**(0,1.5)**	**(0,2)**	**(0,2.5)**	**(0,3)**
Default	3.06	6.41	21.61	10.57	18.56	23.42
Mean difference	1.95	3.36	11.94	4.14	4.42	2.36
p-value of t-test	33.04	20.34	26.14	10.94	18.24	22.77
Standardized difference	2.22	3.00	13.06	5.09	7.36	7.61
KS statistic	1.97	2.51	6.87	2.56	3.69	5.27
KS p-value	10.15	10.54	26.19	10.90	18.56	23.42
entropy	***0.00***	***0.00***	***0.00***	***0.00***	***0.00***	***0.00***
min Value	0.050	0.064	0.163	0.315	0.295	0.248