amor : An R Package to Assess Model Robustness in the Computation of Treatment Effects

Franco D'Angelo
Minerva Schools, Keck Graduate Institute
Advisor: Alexis Diamond

Abstract

Causal inference is a major concern in the statistical literature and many methods have been proposed as means of measuring the effect of a treatment in experimental and observational settings. However, there is no consensus on best practices regarding which to use and, most importantly, how to present the results. In experimental settings, the computation of the average impact is commonly achieved by a simple difference in means of the outcome variable for the treatment and the control groups. This is possible because units are assigned randomly into each group and there is no systematic baseline differences between the two. Observational settings, on the other hand, present a different set of challenges for assessing average treatment effects. Since the researcher often has no information on how the observed data was generated, analyzing observational data usually requires the additional effort of inferring the underlying functional form of the data. This is the case whenever the researcher applies regression adjustment (either as the sole analytical approach or as one component of a multi-part analysis). One of the biggest challenges is that parametric models are typically employed to adjust for baseline differences in confounding variables, and these may be heavily influenced by the model of choice. Following literature around the concern of model dependence in causal inferences, this paper builds on Gary King's proposal of using the effects of multiple permutations of the model. The paper extends the work and makes this approach available via an easy-to-use R package. Presenting the results obtained in the estimation of the causal effect based on multiple random models provides both more robustness to the author's claim, and more information to the reader. Additionally, the proposed approach aligns as a complementary tool to assess the degree to which using *matching* as a pre-processing method helps to make more accurate estimations. The goal of this paper is to invite the scientific community to adopt Gary King's approach as the standard way of presenting causal results whenever regression adjustment is implemented. Examples are provided using applications to real data-sets presented in previous literature.

Keywords: ATE, model dependence, imbalance, model robustness, bias, simulations, R.

1. Introduction

This paper focuses on how to present causal effects when using parametric regression-adjustment on binary program settings. Binary settings are special for understanding the impact of doing one thing over another. In other words, binary settings provide a 'playground' for asking 'what if?' questions. These kind of settings are very common in medical science where experimental clinical trials are performed to understand the effect of medical interventions. Social scientists, on the other hand, ask empirical questions using observational data to try to understand the effects of different measures.

Science strives to understand the underlying cause-and-effect mechanisms that describe the complex systems we live in. 'Correlation does not imply causation' is a standard motto in the statistical literature. Understanding the underlying causal models takes more than distinguishing causation from association. These mechanisms are often difficult to infer because in most non-natural science settings (and in many natural settings, also) there is no way to know the correct or most correct parametric model (if one even exists). In other words, there are often multiple variables that may be affecting a given outcome at any given time, and the functional form is all too often unknowable and consequential for the causal estimate. Hence, the results of many causal models are model dependent (Ho, Imai, King Stuart, 2007; King Zeng, 2007).

In statistical analysis, a standard way of thinking about causes and effects is through the Rubin Causal Model (RCM). The model is named after Donald Rubin (Holland, 1986) and builds on the potential outcomes framework introduced in (Neyman, 1923). The potential outcomes framework consist of pairing two conditional counterfactuals to each unit. The condition is on the occurrence of a prior event[1]. Each unit has a potential outcome for treatment, that is, the outcome of a given unit if exposed to treatment. And, each unit has a potential outcome for control.

Researchers can only observe one potential outcome for each unit. It is possible only to see what happens to a unit if it was assigned to treatment or to control. For instance, if someone you know suffers from a headache and you provide an aspirin, two hours later you will be able to ask how she feels, but you will never know how she would feel if you had not given her the pill. In this way, causal inference plays a major role in our decision-making.

Probability theory has helped researchers and scientists understand the outcomes of events given an *a priori* mathematical procedure since the sixteenth century. Statistical learning, on the other hand, has dealt with the inverse problem of inferring the properties of the underlying mathematical structure given such experiments, plus additional assumptions. Causal reasoning, the subject of interest of this article, juggles both these fields of study and deals with even more assumptions in a quest to infer the underlying causal structure from empirical implications, may they be observational or include data under interventions or distribution changes (Peters, Janzing & Scholkopf, 2017).

If understanding the underlying causal structure of the cause-effect mechanism is the main goal of causal reasoning, measuring its effect follows closely. But using statistical methods of causal inference requires assumptions, and in observational settings, the assumptions tend to be consequential for the results obtained. Randomized experiments are often regarded as the holy grail of statistics because of the number of aspects that are accounted for by randomization

[1] Where the prior event is the assignment to the treatment group.

and, therefore, do not require assumptions. Observational studies present a different set of challenges that require to make suppositions. In observational studies researchers usually do not know what how the data was generated nor have full information on pre-treatment characteristics of the observations. This often leads them to make parametric assumptions about the data-generating process. This scenario is far from ideal since getting to know the right data-generation model is almost impossible and estimating the correct effect depends on getting it right.

In statistical literature, authors typically respond to this by providing 3-10 permutations of the parametric model used to estimate the causal impact. This approach reflects the widely-acknowledged reality that a single model of the data-generating process is not enough to provide an accurate estimate – for which if one is not enough, nor are ten. But most importantly, accepting a single model is not enough exposes the fact that arbitrarily picking the model (or a handful of permutations) may be subject to biases and researcher discretion (King & Nielsen, 2019). The scientific community builds over previous works and presenting results in the most accurate and robust way is key in ensuring future research is based on solid ground.

Gary King addresses this issue and works on providing different solutions: he advocates for the use of matching as a pre-processing method (see Ho et al, 2007), he developed several methods to assess the degree of model dependence and even provides a practical solution to increase robustness and mitigate model dependence bias by illustrating causal effect estimates across multiple permutations (King & Zeng, 2007; Diamond, 2008; and Iacus, King & Porro, 2011). This paper advocates for all three. Matching as a pre-processing method, assessing the degree of model dependence through analysis like the *convex hull* and weighted distances of extrapolations from the data (King & Zeng, 2007) and presenting the results in a more robust and unbias form.

To that end, this paper presents an **R** library that makes it easy to implement and share results from Gary King's proposed approach to assessing model dependence when regression-adjustment is used for causal inference.

2. Literature Review

The literature on causal inference builds on the potential outcomes framework notation introduced by Rubin (1974). Notwithstanding, over the last twenty years, this literature has seen significant advances in terms of improving the robustness of causal estimates and leveraging new technologies to conduct causal analysis; following the rise of data science and machine learning (Peters, Janzing & Scholkopf, 2017).

Following the collection of the data, researchers typically try different assumptions, statistical models, variations, use pre-processing methods and perform several runs of analysis. Each of these changes will produce a different estimation of the causal effect they are trying to measure. Therefore, most of these estimates have some degree of *model dependence* (Ho, Imai, King & Stuart, 2007). Applying statistical methods does not solve this issue. Statistical models cannot be used as black-boxes, and the researcher should have considerable understanding of how a given model works and assumptions statistical models make; as well as deep understanding of the data in use (McElreath, 2016).

Providing additional information on results produced from parametric models is highly signif-

icant. In the first place, researchers have to convince readers that their results are accurate. Hence, readers should be able to assess the robustness of the claim beyond an arbitrarily picked (or potentially cherry-picked) statistical model. Also, additional information allows to assess the reduction of model dependence and evaluate the work of pre-processing methods such as *matching*.

With a view to address these issues, Ho et al. (2007) present *matching* as a non parametric pre-processing method and "define model dependence at point x as the difference, or distance, between the predicted outcome values from any two plausible alternative models ... . By 'plausible' alternative models, we mean models that fit the data reasonably well and, in particular, they fit about equally well around either the 'center' of the data (such as a multivariate mean or median) or the center of a sufficiently large cluster of data nearest the counterfactual x of interest." In addition to making the point on the importance of using *matching* as a method for reducing model dependence and improving estimation of causal effects, the article also provides a set of visual tools that are worth exploring. In particular, this paper focuses on two: Fig.1 – discussed in section 3 – and Fig.4, which provides a visual representation of the causal effect estimate using 63 possible permutations, as well as a comparison to how performing matches helps to narrow the spectrum to a more accurate estimation. This approach is also used in other works such as (Diamond, 2008; and Iacus, King & Porro, 2011) and is central to this paper.

3. Model Dependence as a Threat for Causal Inference

In binary program settings, researchers have to deal with unobserved potential outcomes (or *counterfactuals*). This is the main challenge of causality. It is not possible for researchers to observe the potential outcomes for both the treatment and the control groups, for any given unit *i (i = 1,..., n)*. In other words, it is not possible to observe what "would have happened" to i if it had been exposed to the treatment and, at the same time, observe what "would have happened" if it had not been exposed to it. Potential outcomes will be denoted as $Yi(Ti))$, for $Ti \in \{ 0, 1 \}$, where: 0 refers to the control group and 1 to the set of observations exposed to treatment. Since only one of $Y(0)$ or $Y(1)$ can be observed, the scalar outcome variable for each unit i is: $Yi = TiYi(1) + (1 - Ti)Yi(0)$; making the treatment effect equal to:

$$TEi = Yi(1) - Yi(0) \tag{1}$$

In a randomized experiment, observations are drawn from the same population and are assigned to treatment or control groups randomly. This means that the differences, at baseline, between these groups are due to chance and they are assumed to have no advantage over each other (Rosenbaum, 2005). Hence, all units have the same probability of being assigned to either treatment or control (Dehejia & Wahba, 1999).

$$\{ Yi(1), Yi(0) \perp Ti \} \tag{2}$$

In contrast, when dealing with observational settings, the process by which the data was generated is usually unknown or out of the researcher's control. In observational settings, that is, without randomization to account for baseline differences, the independence breaks

$\{ Yi(1), Yi(0) \not\perp Ti \}$ and the researcher has to make a series of assumptions to continue with the analysis.

In the first place, in order to be able to estimate the underlying data-generation process, it is necessary to assume that any given observation may have been assigned to the control group. This is called the *overlap assumption*: $0 < Pr(Ti = 0|Xi) < 1$ (Imbens, 2004). Second, the *unconfoundedness assumption* (or selection on observable covariates) accounts for the differences in a set of observable covariates, allowing researchers to remove biases from the comparison between the groups. Therefore, conditional on a set of pretreatment covariates Xi, it is possible to assume:

$$\{Yi(1), Yi(0) \perp Ti \}|Xi \tag{3}$$

Essentially, this assumption states that all variables that affect Yi and may interfere with the effect of the treatment variable are being accounted for, conditional on Xi. Making the treatment variable responsible for any difference in the the groups' outcomes. Finally, this assumption is reinforced by the *stable unit treatment value assumption* (SUTVA) which accounts for *spillover effects* between observations. Strictly speaking, SUTVA assumes the potential outcome for any given unit i is fixed, meaning $Yi(Ti)$ is not subject to influences from other observed units $j \neq i$ (King and Nielsen, 2019).

In rare occasions, researchers may be able to estimate the unobserved counterfactual by performing exact matching. This method pairs treatment and control observations based on key variables of interests and, once every treatment observations has its match, prunes the remaining control observations. Exact matching provides an unbiased estimator and reduces the computation of the treatment effect back to the difference in the outcome variable for the treatment and control groups seen in (1).

However, exact matching is nearly impossible when dealing with most of real data sets, forcing researchers to use other methods when exact matches are not found for all treatment units. In these cases, it is necessary to use a statistical model that would fit the space between each treated unit and the closest counterfactual point, represented by the closest control unit (King & Nielsen, 2019).

A visual representation of the consequences of having to make so many assumptions in terms of trying to understand the underlying functional relationship between the treatment Ti and the set of predictors Xi is offered in (Ho, Imai, King, and Stuart, 2007) (see Figure 1). In Fig.1, the authors show in a very intuitively way, how two different models may estimate differ so much as to estimate both a positive and negative effect, using the same data. The panel on the left analyses the average treatment effect for two models. The first one, a standard linear regression: $y \sim t + x$, where t is the treatment indicator variable and x a variable of interest. The average treatment effect for this model is depicted by the difference between the two parallel lines. In this case, the black line representing the estimate of the model on the treated observations (T) is greater on Y than the grey line representing the estimate of the model on the control units (C), making the ATE positive. However, the authors illustrate the issue of *model dependence* by showing a contrary effect if the quadratic model: $y \sim t + x + x^2$ is used. It is possible to see the negative ATE by observing the grey dashed line over the black dashed parabola. The panel on the right addresses the advantages of *matching* as a way of adjusting the data and reducing model dependence (Ho, Imai, King, and Stuart, 2007).

The package **amor** accounts for this issue by enabling the user to include permutations of the

model encompassing interaction terms. These interactions range from single one-way terms, up to 3-way interactions (single, quadratic and cubic terms).

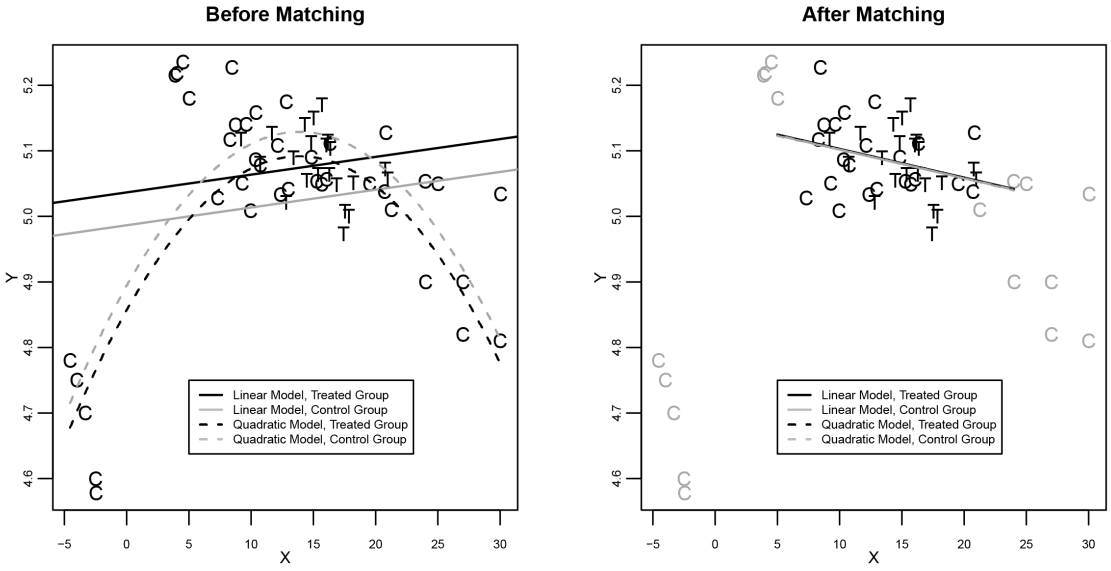

Figure 1: Replication of Figure 1 from (Ho, Imai, King, and Stuart, 2007, p.210). Data gathered from: *"Replication data for: Matching as Nonparametric Pre-processing for Reducing Model Dependence in Parametric Causal Inference"*. Retrieved from: https://doi.org/10.7910/DVN/RWUY8G. Harvard Dataverse.

4. The R Package: amor

Despite all the efforts for measuring causality and getting insights on causal reasoning, there is no consensus or best practices regarding how to present such results. What is worse, the lack of consensus undermines the robustness of the claims. In general, average treatment effects are presented using an arbitrarily picked statistical model and, at best, a few comparison results with similar models. As seen before, this standard practice allows for biases and researcher discretion in the results, which attempt towards the robustness of the claims.

The package **amor** proposes a set of tools to better assess model robustness. The main goal is to help researchers to provide unbiased results of the effect of a given experiment. Following Gary King's approach, this paper proposes to present the results through a density plot (or data frame of estimated effects) based on multiple, randomly generated models. Instead of researchers providing the results based on a single (or a small selection) of arbitrarily picked models.

The first set of tools of the package **amor** is composed by the "...md" functions: `mlmd()`, `olsmd()` and `plotmd()`. These functions present an easy-to-use method to follow Gary King's multiple-permutations-treatment-effects approach. The functions work by simulating a user-defined number of random models, estimating the effect for each model, storing each value and returning them as a data frame; which can be turned into a density plot using `plotmd()`.

The package offers functions to simulate, compute and plot average treatment effects for both ordinary least squares and maximum likelihood scenarios (`olsmd()` and `mlmd()` respectively).

The package **amor** limits its application to binary settings and needs a treatment variable specifying those units exposed to regime 1 (treatment) or 0 (control). Following most of the literature, **amor** assumes that the outcome of one unit is independent of other units receiving treatment or not. In other words, this package is restricted to settings where the Stable-Unit-Treatment-Value-Assumption (SUTVA) is met (Imbens & Rubin, 2015).

Package **amor** uses dependencies from the **MASS** package and is available from Github at `https://github.com/pampakid/amor` while it goes through the CRAN and JSS submissions processes (see Appendix 8.1 for information on installing a package from Github).

4.1. General Usage

The functions: `olsmd()` and `mlmd()` estimate the treatment effects using ordinary least squares and maximum likelihood respectively. Both functions work by iterating over `n.sims`. For each iteration, a random formula is generated by selecting a random number of terms from the `predictors` vector. These terms may include one-way, two-way and three-way interactions depending on the specified `level`. All models include the variable `treat`. Once the formula is generated, it becomes the input for the model – ols or ml depending on the function. Then, the function runs the model and stores the coefficient for the treatment variable (`treat`), before passing on the next iteration. Finally, each function returns a data frame containing the estimated effect and the formula of the permutation of the model used in each iteration.

The two functions have an optional argument called `matched.data` that allows the user to compare the estimated effects of two data sets (original and pre-processed) in order to assess if matching had a possitive impact in reducing model dependence.

Ordinary Least Squares

```
olsmd(data, dep, treat, predictors, level, n.sims, matched.data = NULL)
```

Argument	Description
data	data set collected by the researcher to conduct the study.
dep	the name of the dependent variable by which the researcher aims to measure the treatment, as named in the data set.
treat	the name of the treatment variable by which the researcher intervened the study, as named in the data set and with values of 1 if the unit was exposed to treatment and 0 otherwise.
predictors	vector of variables the user wants to include in the simulations
level	the degree of interactions the user wants to include in the simulations (= 1, 2, 3).
n.sims	the number of models the user wants to simulate.
matched.data	(optional) extra data set of pre-processed data to compare to the original one.

Table 1: Overview arguments for `olsmd()`

Maximum Likelihood

```
mlmd(data, dep, treat, predictors, level, n.sims, method = "odds", matched.data
= NULL)
```

Argument	Description
data	data set collected by the researcher to conduct the study.
dep	the name of the dependent variable by which the researcher aims to measure the treatment, as named in the data set.
treat	the name of the treatment variable by which the researcher intervened the study, as named in the data set and with values of 1 if the unit was exposed to treatment and 0 otherwise.
predictors	vector of variables the user wants to include in the simulations
level	the degree of interactions the user wants to include in the simulations (= 1, 2, 3).
n.sims	the number of models the user wants to simulate.
method	by default, results are presented as predictions of log-odds ="odds". For results on the scale of the predictors ="coeffs".
matched.data	(optional) extra data set of pre-processed data to compare to the original one.

Table 2: Overview arguments for `mlmd()`

Data Visualization

```
plotmd(data)
```

The data visualization function takes the data frame outputted by either `olsmd()` or `mlmd()` and produces a visual representation of the estimated effects. The function automatically infers whether the data frame was generated using a single data set or is comparing two data sets. In the latter case, the density plot will have two labelled curves representing the original and matched data sets.

4.2. Random Model Generation

As mentioned before, every iteration begins with the generation of the random model that is going to be used to estimate the average treatment effect. The first step in generating the formula for each model is to sample a random number from the length of the `predictors` vector. This sample will randomly define the number of terms that the iteration's formula will have. Once the number of terms is defined, the function adds the `treat` variable as the first term of the formula, and fills the rest with randomly sampled interactions (depending on the selected `level`) from the `predictors` vector. Before adding each interaction to the formula, the function will check that the interaction is not repeated. In order to avoid interactions being treated as special formula operators, and have them work as they would outside a formula, the function defines the variables using the `I(...)` notation. As defined by the `level`, each interaction term can be: a one-way interaction (standard single variable terms), a two-way interaction (includes interactions between two variables – or a square), and a three-way interaction (which includes interactions of up to three variables – or a cube).

4.3. Calculating the coefficients for treatment

Once the formula is generated, it becomes the input for `lm()` or `glm()` depending on whether the user is using `olsmd()` or `mlmd()`. In the case of simple one-way interactions, the computation of the treatment coefficients is achieved through the standard coefficients summary provided by the `lm()` function. In contrast, the computation of the treatment coefficients for level = 2, 3 is not possible through simple coefficient summary because the treatment effect is no longer isolated to the coefficient on the `treat` variable, but is now dependent upon that coefficient and also the coefficients on the interaction terms.

This issue is resolved by duplicating the `data`. One of the copies is coded as *treatment* (`treat` variable = 1) and the other as *control* (`treat` variable = 0). Doing so allows to estimate the treatment effect as the difference of predicted estimates for each data set. Since the only difference between both data sets is the `treat` variable, the product of all terms that do not include the `treat` variable will cancel each other, making the difference equal to the sum of the coefficients of the terms including the `treat` variable.

```
d.treated = data
d.control = data
d.treated[treat] = rep(1, length(data[treat]))
d.control[treat] = rep(0, length(data[treat]))
```

Listing 1: Duplication of the data, for the treated and the control.

Using the `predict()` function, the formula generates predictions given the model predictors, on the new data sets. The only difference between these data sets is the `treat` variable. Therefore, the difference in the means of the predicted values for each data set (d.treated - d.control) equals the estimated treatment effect of the given iteration.

```
m = glm(formula = f, family = 'binomial', data = data)
predicted.treat = predict(m, newdata = d.treated)
predicted.control = predict(m, newdata = d.control)
Estimated.ATE = mean(predicted.treat - predicted.control)
```

Listing 2: Representation of the estimation of the treatment using `glm()`. Where `f` is the randomly generated formula. It is important to note that the model `m` is generated using the original data set, while the predictions are generated on the data duplicates (given the original predictors).

5. Examples

The paper presents two examples applying the functions to real data-analyses. The first example builds upon (Dehejia & Wahba, 1999) and uses the Lalonde dataset. The second example builds upon (King & Zeng, 2007) and uses the UN Peacekeeping data set. Both examples include the use of `plotmd()` for visual representations.

5.1. Estimating the treatment effect of the National Supported Work (NSW) Demonstration (Lalonde, 1986; Dehejia and Wahba, 1999)

The research conducted by (Dehejia & Wahba, 1999) builds on (Lalonde, 1986). In order to

evaluate how non-experimental studies were able to compare with experimental ones, Lalonde used experimental data on the impact of the National Supported Work Demonstration labour program (Lalonde, 1986) as well as different observational control groups drawn from the Current Population Server (CPS) and the Panel Study of Income Dynamics (PSID). Both Lalonde and Dehejia & Wahba data sets are widely used in the causal inference literature.

Lalonde uses several econometric methods to estimate the effect of the program and proves much of these procedures are subject to biases caused by errors in the specifications the researcher defines. For their part, Dehejia & Wahba propose to use Propensity Score Matching (PSM) and demostrate that following this method a researcher may estimate the experimental benchmark up to a decent degree. The authors estimate a treatment effect ranging between \$1,473 - \$1,774, compared to the experimental benchmark of \$1,794 (Dehejia & Wahba, 1999).

The authors used stratification and matching on the propensity and leveraged the use of multiple control groups as an additional way of evaluating the estimates. For these reasons, the estimations are conditional on the estimated propensity score. In terms of the models used in the study, the authors and where conducted using different statistical models, including seven OLS and three maximum likelihood models. Naturally, models differ in their specifications (Dehejia & Wahba, 1999).

It is important to note that the NSW (Dehejia-Wanha Sample) limits Lalonde's data set to those observations with information regarding 1974 earnings and offers the same distribution of pre-intervention variables for both treatment and control groups; a key property of experimental settings (Dehejia & Wahba, 1999).

The following figures were generated using two data sets[2]: the one the authors used (NSW_DW) and a combination of the treated units from NSW_DW and the CPS observational control units.

Figure 2 replicates the findings of the authors using NSW_DW data. The figure compares the average treatment effect (ATE) calculated over the full data set, against an ATT calculation of a genetically matched data set. Both densities are aligned with the authors' claim in terms of the estimated range of effects, but it also provides additional information. In the first place, the figure provides a visual representation of the threat of model dependence by looking at the range of possible values. Second, the comparison of the two data sets allows the researcher to see how matching helps making the inference more accurate. In this case, matching appears to provide a range of estimates much closer to the experimental benchmark. Finally, analyzing the density probabilities adds a complementary layer of valuable information to have a more accurate estimation of the possible values.

In the case of the combined data set between the NSW_DW treatment units and the CPS observational control units there is such variability that a comparison in the same plot was not possible. Figure 3 shows an `olsmd()` analysis on the raw data, while Figure 4 does the same on a pre-processed set. The pre-processing, as in Figure 2, was done via genetic matching.

[2]Data sets retrieved from: https://users.nber.org/~rdehejia/nswdata.html

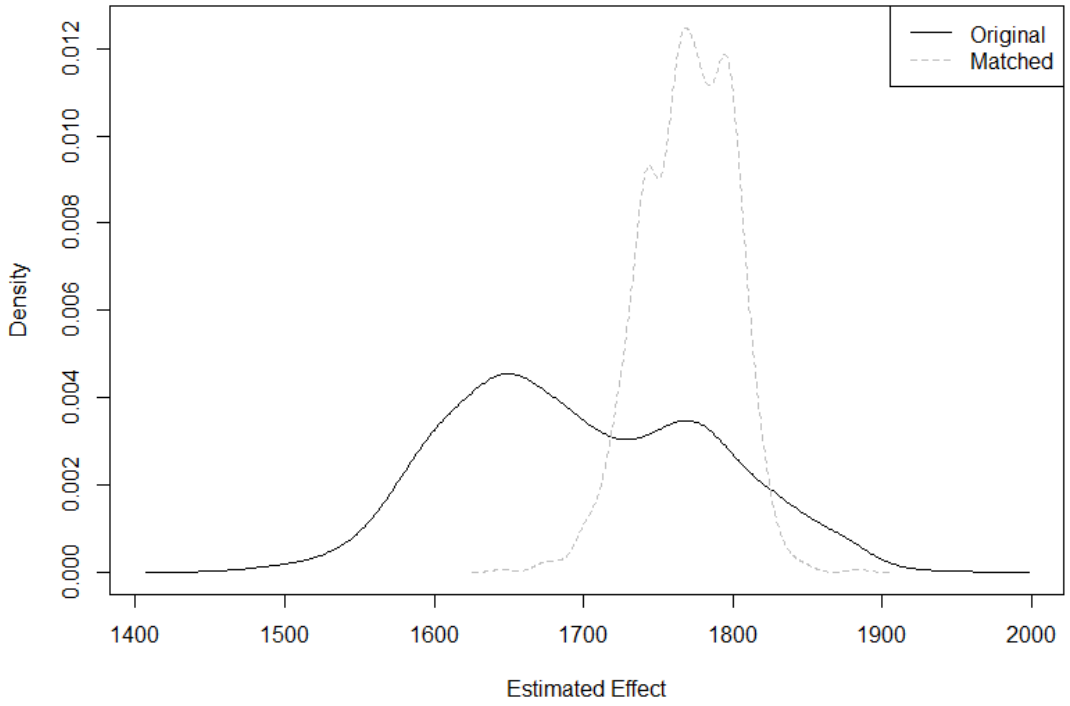

Figure 2: Estimations of 1,000 random models. The figure was generated using `olsmd()` with two-way interactions and a `matched.data` comparison by genetic matching. The pre-processing was performed using **GenMatch** to calculate the ATT with a population size of 40. The data was matched over the predictor variables, achieving an after-matching minimum p-value of 0.32.

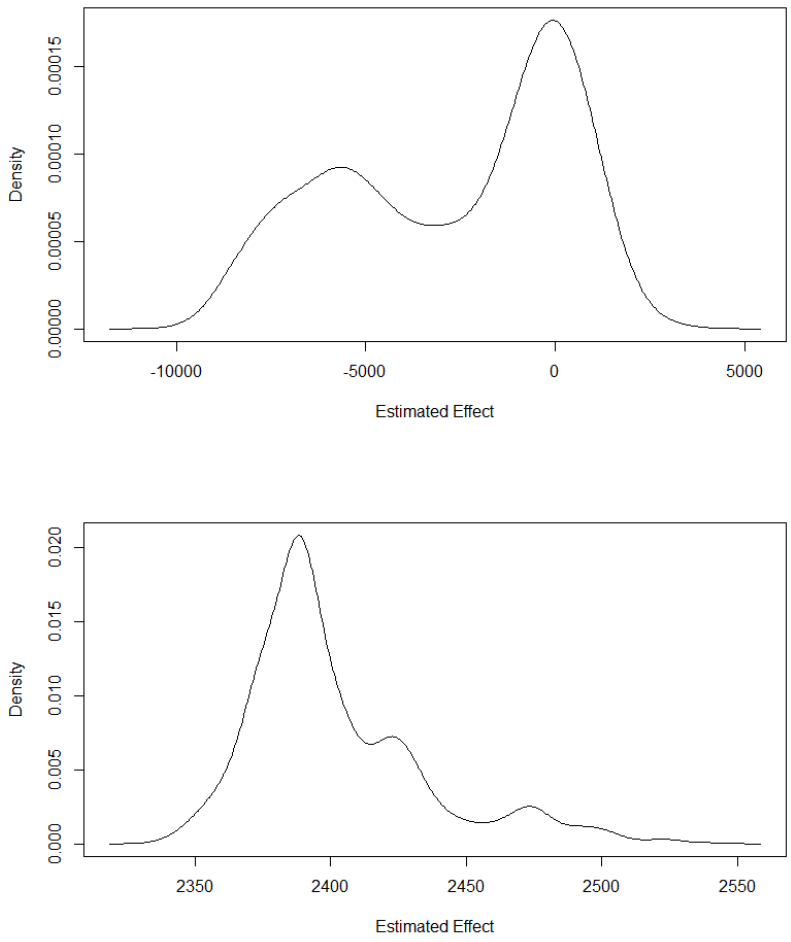

Figure 3: Estimations effects of 1,000 random models the combined CPS-NSW_DW data set. The figure was generated using `olsmd()` with two-way interactions on the original data set (top) and over a pre-processed version of the combined CPS-NSW_DW data set (bottom). The pre-processing was performed using **GenMatch** over the predictor variables, achieving an after-matching minimum p.value of 0.19. To note the possibility of visually analyzing the high variability of the observational data, compared to the experimental one. Warning: the scale of the x-axis differs significantly between the plots.

5.2. Inference-based peace building strategies (King & Zeng, 2007)

In 2007, Gary King and Langche Zeng restated the concern around model dependence in counterfactual inference and provided a set of tools for assessing its degree. In (King & Zeng, 2007), the authors replicated the inferences on UN Peacekeeping Operations made by (Doyle & Sambanis, 2000) . Among these tools, it is important to highlight: stating the difference between *intrapolation* and *extrapolation*, the introduction of an easy-to-use software for analyzing the whether data points fell inside or outside of the *convex hull* and the concept of weighting the distance between a counterfactual and the data to add a layer of depth to the

dichotomous convex hull analysis (see King & Zeng, 2007). Building on (King & Zeng, 2007), this next example complements their analysis by applying `mlmd()` to the results obtained by the authors.

Far from discussing (Doyle & Sambanis, 2000) conclusions, King and Zeng's focus was to provide methods of assessing model dependence. They concluded the data provided by Doyle and Sambanis presented several challenge regarding the capacity of drawing robust causal inferences from it. On the one hand, the authors pointed there was no supporting evidence for preferring the original model [3] to an alternative. For instance, adding an interaction between the duration of the civil war (Wardur) and UN operation (UNOP4) to A8. The authors made their case with mere intuition pointing that the duration of the operation, could be a variable of interest. The authors also presented a comparison table of the coefficients, standard errors and p.values for both models (see King & Zeng, 2007, Table.2).

Additionally, the authors performed their proposed methods for evaluating the data used by (Doyle & Sambanis, 2000)[4]. First they checked the number of counterfactuals that fell within the convex hull, and found none did. This exposed the data required information for extrapolation, which often lead to more model-dependent inferences (King & Zeng, 2007). To make things worse, the authors found few counterfactuals near the data, making the risk of extrapolation even greater (King & Zeng, 2007).

Following the authors' goal of using this data set as a case for applying easy-to-use methods to assess model dependence, Figure 5 runs `mlmd()` on both the original data set and a genetically matched comparison. The matched estimates narrow the probability of peace building success two years after the war, with a threshold of 3 (PBS2S3)[5] at almost 0.6. Nonetheless, model dependence is still high, as can seen in the wide range of estimates.

Once more, the visual representation provides a quick analysis of the model dependence by looking at the range of possible estimates, and provides a visual assessment of how matching helps to make more accurate inferences.

[3] Doyle and Sambanis used a single model numbered A8 to conclude that UN interventions increase the odds of peace building success by 23 times, compared to no UN operations. A8 is composed of ten single terms and a two-way interaction ($Factnum^2$) (see Doyle & Sambanis, 2000).

[4] Data set containing 124 post-WWII civil wars.

[5] see Data Set Notes for Doyle and Sambanis *International Peacebuilding: A Theoretical and Quantitative Analysis* (November 6, 2000).

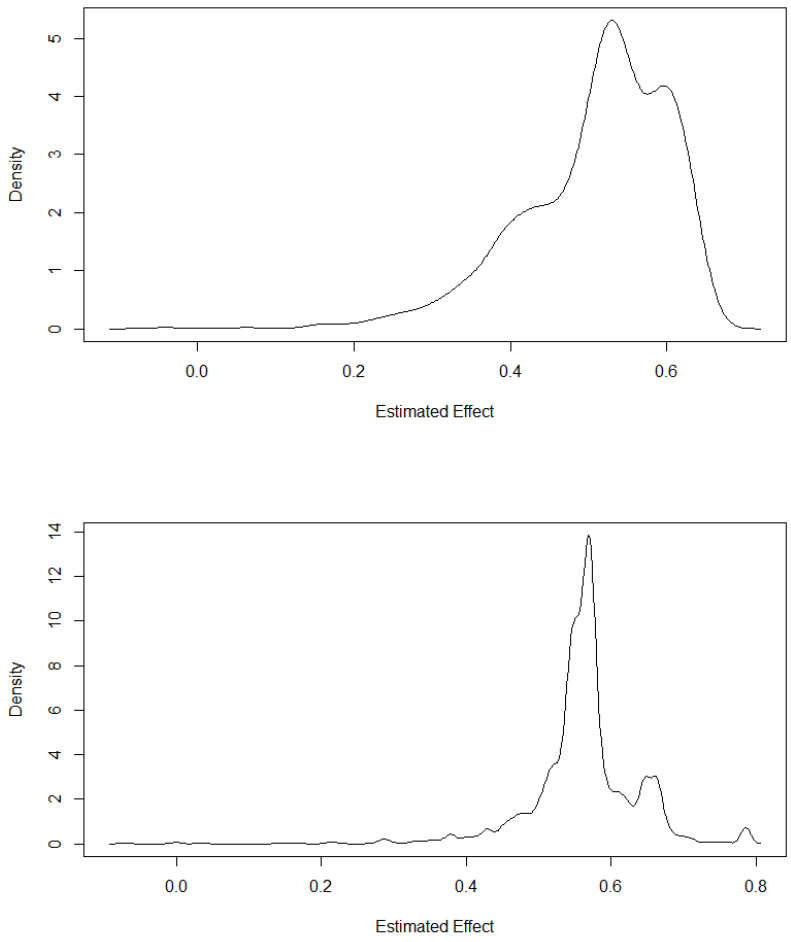

Figure 4: Estimations of 2,000 random models. The figure was generated using `mlmd()` with three-way interactions over the original peace building data set (top) and a pre-processed genetically matched, ATT version of it (bottom). The pre-processing was performed using **GenMatch** over the predictor variables, achieving an after-matching minimum p.value of 0.2.

6. Conclusion

Causal questions require causal inference, and parametric regression modeling is often part of the process. Responsibility in the presentation of the findings is paramount to the robustness of the claims and further development of knowledge. This paper aims to standardize Gary King's approach, as it appears that King and his co-authors were the first to introduce the multiple-model-permutations to the scientific community. King is also responsible in part for the popularization of matching as a pre-processing method. Both these methods have been published and acknowledged by the scientific community, with several works detailing their benefits in reducing model dependence and helping to make more accurate causal inferences; and continuing their development (e.g. Diamond & Sekhon, 2013).

The package **amor** extends the work on the previous methods through the inclusion of user-defined interaction terms and the possibility of comparing the original data set to the pre-processed one using the same randomly generated models on each. Moreover, the package amor makes these methods available for researchers to easily use them as a standard way of analyzing results and presenting their findings. The use of the functions presented in this paper should become a standard practice for presenting results of works involving parametric modeling for causal inference as long as the parametric form is unknown to the researchers and whenever doing matching as pre-processing.

Further Research

In a similar way to how this paper builds on previous methods, further research may bridge this set of functions to other methods such as how the reduction of model dependence through random-model-permutations works with different resampling techniques for *model assessment* and the computation of the MSE.

7. References

(Neyman, 1923) Neyman, J. (1923). *Master's Thesis.* Excerpts reprinted in English, Statistical Science, Vol. 5. p. 463 - 472.

(Rubin, 1974) Rubin, D. (1974). *Estimating Causal Effects of Treatments in Randomized and Non-randomized Studies.* Journal Of Educational Psychology, 66(5), 688-701. doi: 10.1037/h0037350

(Holland, 1986) Holland, P. (1986). *Statistics and Causal Inference.* Journal Of The American Statistical Association, 81(396), 945. doi: 10.2307/2289064.

(Dehejia & Wahba, 1999) Dehejia, R., & Wahba, S. (1999). *Causal Effects in Nonexperimental Studies: Reevaluating the Evaluation of Training Programs.* Journal Of The American Statistical Association, 94 (448), 1053-1062.

(Doyle & Sambanis, 2000) Doyle, M., & Sambanis, N. (2000). *Data Set Notes for International Peacebuilding: A Theoretical and Quantitative Analysis.* American Political Science Review.

(Ho, Imai, King & Stuart, 2007) Ho, D., Imai, K., King, G., & Stuart, E. (2007). *Matching as Nonparametric Preprocessing for Reducing Model Dependence in Parametric Causal Inference.* Political Analysis, 15(3), 199-236. doi: 10.1093/pan/mpl013

(Imbens & Wooldridge, 2007) Imbens, G.W., Wooldridge, J.M. (2007). *Estimation of Average Treatment Effects Under Unconfoundedness.* Lecture Notes: What's New in Econometrics. NBER.

(King & Zeng, 2007) King, G., & Zeng, L. (2007). *When Can History Be Our Guide? The Pitfalls of Counterfactual Inference.* International Studies Quarterly, 51(1), 183-210.

(Diamond, 2008) Diamond, A. (2008). *Essays on Causal Inference in Observational Studies.* Thesis presented to The Doctoral Program in Political Economy and Government. Cambridge, MA. Harvard University.

(Iacus, King & Porro, 2011) Iacus, S., King, G., & Porro, G. (2011). *Multivariate Matching Methods That Are Monotonic Imbalance Bounding.* Journal Of The American Statistical Association, 106(493), 345-361.

(Diamond & Sekhon, 2013) Diamond, A., & Sekhon, J. (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. doi 10.1162/rest_a_00318

(James et al., 2013) James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning : with applications in R.* New York. Springer.

(Imbens & Rubin, 2015) Imbens, G.W., & Rubin, D.B. (2015). *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction..* Cambridge University Press. 461-476. doi:10.1017/cbo9781139025751.021

(McElreath, 2016) McElreath, R. (2016). *Statistical rethinking A Bayesian Course with Examples in R and Stan.* Boca Raton CRC Press, Taylor & Francis Group.

(Peters, Janzing & Scholkopf, 2017) Peters, J. Janzing, D. & Scholkopf, B. (2017). *Elements of Causal Inference Foundations and Learning Algorithms.* Cambridge, MA. MIT Press. Adaptive Computation and Machine Learning Series.

(Pearl & Mackenzie, 2018) Pearl, J., & Mackenzie, D. (2018). *The Book of Why The New Science of Cause and Effect.* Penguin Random House.

(King & Nielsen, 2019) King, G., & Nielsen, R. (2019). *Why Propensity Scores Should Not Be Used for Matching.* Political Analysis, 1-20. doi:10.1017/pan.2019.11

8. Appendix

8.1. Installing AMOR from Github

```
# First install devtools to use install_github() function
install.packages("devtools")
library(devtools)
install_github("pampakid/amor")
library(amor)
```

8.2. Replication Code Figure 2:

```
library(foreign)
library(Matching)
library(amor)

foo <- read.dta("nsw_dw.dta")

attach(foo)
X <- cbind(age, education, black, hispanic, married, nodegree, re74, re75)
detach(foo)

# GENETIC MATCHING
genout <- GenMatch(X = X, Tr = foo$treat, estimand ='ATT', pop.size = 40)
mout.gen <- Match(X = X, Tr = foo$treat, Y = foo$re78, Weight.matrix=genout)
mb.gen <- MatchBalance(treat ~ age + education + black + hispanic + married +
    nodegree + re74 + re75, data=foo, match.out = mout.gen, nboots=1000)
foo.gen <- foo[c(mout.gen$index.treated, mout.gen$index.control),]

# RUNNING AMOR
p <-
    c("treat","age","education","black","hispanic","married","nodegree","re74","re75")
t <- olsmd(foo, "re78", "treat", p, 2, 1000, matched.data = foo.gen)
plotmd(t)
```

8.3. Replication Code Figure 3

```
library(foreign)

DW_data <- read.dta("nsw_dw.dta")
cps_controls <- read.dta("cps_controls.dta")
cps_controls_new_names <- cps_controls
names(cps_controls_new_names) <- names(DW_data)
DW_data_nocontrols <- DW_data[-which(DW_data$treat == 0),]
DW_treated_data_with_CPS <- rbind(DW_data_nocontrols, cps_controls_new_names)
foo = DW_treated_data_with_CPS

attach(foo)
X <- cbind(age, education, black, hispanic, married, nodegree, re74, re75)
detach(foo)

# GENETIC MATCHING
library(Matching)
genout <- GenMatch(X = X, Tr = foo$treat, estimand ='ATT', pop.size = 70)
mout.gen <- Match(X = X, Tr = foo$treat, Y = foo$re78, Weight.matrix=genout)
mb.gen <- MatchBalance(treat ~ age + education + black + hispanic + married +
    nodegree + re74 + re75, data=foo, match.out = mout.gen, nboots=1000)
foo.gen = foo[c(mout.gen$index.treated, mout.gen$index.control),]

# RUN AMOR
library(amor)
p
    =c("treat","age","education","black","hispanic","married","nodegree","re74","re75")
m = olsmd(foo, "re78", "treat", p, 2, 1000)
o = olsmd(foo_gen, "re78", "treat", p, 2, 1000)
# PLOT
par(mfrow = c(2, 1))
plotmd(o)
plotmd(m)
```

8.4. Replication Code Figure 4

```
# DATA PREPROCESSING
foo <- read.csv("https://tinyurl.com/y2zt2hyc")
foo <- foo[, c(6:8, 11:16, 99, 50, 114, 49, 63, 136, 109, 126, 48, 160, 142, 10)]
foo <- foo[c(-19, -47), ]
which(is.na(foo) == TRUE)
head(foo)

library(Matching)

attach(foo)
X <- cbind(wartype, logcost, wardur, factnum, factnum2, trnsfcap, treaty, develop,
    exp, decade)
detach(foo)

# GENETIC MATCHING
genout <- GenMatch(X = X, Tr = foo$untype4, estimand = "ATT", pop.size = 20)
matchout.gen <- Match(X = X, Tr = foo$untype4, Y = foo$pbs2s3, Weight.matrix=genout)
mb.gen <- MatchBalance(untype4 ~ wartype + logcost + wardur + factnum + factnum2 +
    trnsfcap + treaty + develop + exp + decade, data=foo, match.out = matchout.gen,
    nboots=1000)
foo.gen = foo[c(matchout.gen$index.treated, matchout.gen$index.control),]

# RUN AMOR
library(amor)
p = c('untype4', 'wartype', 'logcost', 'wardur', 'factnum', 'factnum2', 'trnsfcap',
    'treaty', 'develop', 'exp', 'decade')
o = mlmd(foo, "pbs2s3", "untype4", p, 3, 2000, method = "odds")
m = mlmd(foo.gen, "pbs2s3", "untype4", p, 3, 000, method = "odds")
# PLOT
par(mfrow = c(2, 1))
plotmd(o)
plotmd(m)
```

8.5. Application of Program Learning Outcomes

Introduction

This work is rooted in the âĂIJthinking criticallyâĂİ HCs that we studied in the program. The main goal of the R library introduced in the paper is to help researchers reduce their biases[6] and standardize the way of analyzing and presenting the results of the treatment effect estimations[7] in binary settings[8]. To accomplish this, the work builds on Gary King's multiple-model-permutations approach.

While doing the program, I became particularly interested in systematic ways of measuring the effect of experiments. My goal is to add value to a business. Some of my best qualities are ideation and strategy. However, I always felt coming up with new ideas was not enough. I always felt that I needed to have feedback on how my ideas impacted the business and the users. I found the answer to my concern in causal inference. Causal questions are at the centre of innovation, and making the right decisions depends upon the accurate measurement of the estimated effects.

Formal Analyses and Information-Based Decisions are two courses loaded with causal analysis, and it was during a class from CS112 with Prof. Diamond that the first idea of furthering the development of Gary King's approach came up. The initial idea was only to make this approach available through an R package. However, as we moved onto other topics such as pre-processing (and matching in particular), it became clear that extending the work to include permutations of the model encompassing interaction terms, adding the possibility to compare the inputted data to a matched version of it -using the same random models- and easily visualizing the results could be an added values.

Problem-Solving HCs

After reviewing the existing literature, I was able to discover and delve deeper into the problem of model dependence, a threat to causal inference. At this moment I leveraged the knowledge acquired in Empirical Analysis courses, on analyzing problems. Particularly, in understanding the right problem and performing gap analysis to develop a roadmap of potential solutions.

As mentioned before, the first step was to define the problem: model dependence, and considering Gary King's approach as a potential solution. Although it is an issue acknowledged by the scientific community[9], designing how I would fill that gap was still challenging. I decided to narrow down on the problem as defined in Ho et al. (2007), who present matching as a non-parametric pre-processing method and "define model dependence at point x as the difference, or distance, between the predicted outcome values from any two plausible alternative modelsâĂę. By 'plausible' alternative models, we mean models that fit the data reasonably well and, in particular, they fit about equally well around either the 'center' of the data (such as a multivariate mean or median) or the center of a sufficiently large cluster of data nearest the counterfactual x of interest.". With the problem clearly stated I focused on evaluating whether and where I needed to close the gap, and where I could build on existing tools. The end analysis was that I needed to come up with a solution that could generate multiple model

[6] #biasidentification
[7] #estimation
[8] #biasmitigation
[9] #sourcequality

permutations based on a set of predictors and the inputted data, estimate the effect of each model and provide a density plot, following Gary King. Naturally, in addition to using problem analysis on the literature review and overall structure of the paper, this set of HCs were also important in regards to coding the package. And the code can be analyzed by looking at those three sections as well[10].

In order to create the random models, I made three functions[11], one for each level of interactions (one-way, two-way three-way). After careful consideration, I decided I would take advantage of the `formula =` parameter from `lm()` and `glm()`, which takes an input of type character, to generate the model formulas doing string manipulation. Each function would be used based on the user's choice of level, and produce formulas[12] encompassing interactions up to the named level (= 1, 2, 3). I used existing tools including `lm()` and `glm()` to run the models, and `predict()` to generate the probabilities that would allow me to calculate the effect. Finally, I resolved the issue of data visualization of the outcomes by developing a function that automatically recognizes whether the inputted data frame comes from a single analysis, or it refers to a comparison of two data sets. This function allows the user to plot either the treatment effect analysis of a single data set or a comparison of two, in one line of code.

It is important to note that either from in-class learning or from motivation to keep learning with other resources, I gradually acquired the technical skills needed to perform all these tasks from the moment I started doing the program. I am even venturing as a data analyst at Nielsen for over a year now.

Complex Systems HCs

One of the points that drew me towards causal inference is that it goes far beyond statistics. As I mentioned before, causal questions are at the centre of innovation, but they also have a major role in people's day to day lives decision making[13]. Analyzing people's behavior was one of my passions before the program, and the debates we had in Complex Systems helped me find a way in which all of my interests interact.

This paper addresses directly the mitigation of researcher bias and ethics so as to avoid the scientific community to build upon previous literature that may have dubious claims[14]. As we learnt in Complex Systems, bias identification and mitigation are hard to achieve. But a big part of my motivation to tackle this topic lays in the fact that there are multiple levels of analysis by which accurately estimating effects may have a profound impact on people.

Presentation HCs

Professionalism[15] was my low point throughout all the program. My background was not in academia and I struggled to make scholarly works. That is why I wanted to focus on the HCs regarding presentation in my final work. Taking this into consideration, I approached my thesis with the goal of making a piece worth publishing[16]. I followed established guidelines for

[10] #breakitdown
[11] #algorithms
[12] #simulation
[13] #expectedutililty
[14] #ethicalconflicts
[15] #professionalism
[16] #organization

developing and presenting statistical software. I tried to situate the work within the existing literature and respecting the target audience[17].

Conclusion

The learning outcomes I take from doing this work are varied. On the one hand, it is an incentive for me to be able to connect all my interests, and I feel this is a narrow work, with wide applications. Second, there are certain milestones I accomplished while doing this job, which I did not think I could do. Two years ago it was unthinkable for me to write a piece of code, much less create an **R** library. I thought I had no place in academia after letting many opportunities go by during my undergrad degree; when I did not worry the way I could have about grades. Writing this paper proved to me that I could still do it: I could find a place where I could add value. Finally, writing this paper tested -yet again in the last two years- my emotional intelligence in ways I was not used too, and I gather the experience as an additional learning outcome. And, there are many more HCs I could have included. I believe all five courses helped me grow in ways I cannot yet explain.

Regarding this paper, I believe its added value is found in having a clear objective: extending previous work in a very narrow and easy-to-use manner and accomplishes so by interacting with the existing literature and encouraging further use and development.

[17] #audience